

LAPORAN PROYEK
SISTEM MANAJEMEN INFORMASI BERBASIS WEB PADA
PERUSAHAAN PENGECORAN LOGAM



Disusun Oleh:

Aisyah Dwi Puspita	2200018104
Felia Nadia Fikarda	2200018159
Naufal Muzakki	2200018160
Gia Sulipri	2200018171
Annisa Aulia	2200018184

PROGRAM STUDI S1 INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS AHMAD DAHLAN

2025

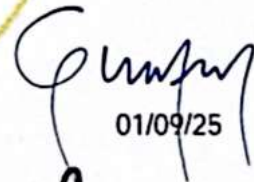
LEMBAR PENGESAHAN

**SISTEM MANAJEMEN INFORMASI BERBASIS WEB PADA PERUSAHAAN
PENGECORAN LOGAM**

Dipersiapkan dan disusun oleh:

Aisyah Dwi Puspita	2200018104
Felia Nadia Fikarda	2200018159
Naufal Muzakki	2200018160
Gia Sulipri	2200018171
Annisa Aulia	2200018184

Pembimbing : Guntur Maulana Zamroni, B.Sc., M.Kom.
NIPM : 19840309 201810 111 1205917


01/09/25

Penguji : Ir. Nur Rochmah Dyah Puji Astuti, S.T., M.Kom.
NIP : 197608192005012001


02/09/25

Yogyakarta, 02 September 2025
Kepala Program Studi S1 Informatika


Dr. Murinto, S.S., M.Kom.
19730710 200409 111 0951298

KATA PENGANTAR

Puji syukur kami panjatkan ke hadirat Tuhan Yang Maha Esa, yang atas rahmat dan karunia-Nya, kami dapat menyelesaikan Laporan Akhir Proyek Pengembangan Sistem Informasi CV Kembar Jaya Abadi ini dengan baik dan tepat waktu. Laporan ini disusun untuk memenuhi tugas akhir mata kuliah Manajemen Proyek dan Teknologi Informasi (MPTI) di lingkungan Universitas Ahmad Dahlan. Laporan ini menguraikan seluruh proses manajemen proyek, mulai dari tahap perencanaan, perancangan, implementasi, hingga evaluasi sistem informasi yang mencakup situs web publik dan dashboard admin untuk studi kasus CV Kembar Jaya Abadi. Penyelesaian laporan ini tidak terlepas dari bimbingan, arahan, dan dukungan dari berbagai pihak. Oleh karena itu, kami ingin mengucapkan terima kasih yang tulus kepada Bapak Guntur Maulana Zamroni, B.Sc., M.Kom., selaku dosen pengampu mata kuliah MPTI yang telah memberikan bimbingan dan masukan yang sangat berharga. Ucapan terima kasih juga kami sampaikan kepada seluruh rekan-rekan kelompok yang telah bekerja sama dengan solid dan penuh dedikasi selama pengerjaan proyek ini. Kami menyadari bahwa laporan ini masih memiliki banyak kekurangan dan jauh dari kata sempurna. Oleh karena itu, kritik dan saran yang membangun dari segenap pembaca sangat kami harapkan untuk perbaikan di masa mendatang. Akhir kata, semoga laporan ini dapat memberikan manfaat, baik secara teoretis maupun praktis, serta menambah wawasan bagi kita semua.

DAFTAR ISI

LEMBAR PENGESAHAN.....	i
KATA PENGANTAR.....	ii
DAFTAR ISI.....	iii
DAFTAR GAMBAR.....	v
DAFTAR TABEL.....	vii
DAFTAR KODE PROGRAM.....	viii
BAB I PENDAHULUAN	1
1.1. Latar Belakang Masalah	1
1.2. Project Charter	2
BAB II PERENCANAAN PROYEK	4
2.1. Analisis Kelayakan	4
2.2. <i>Work Breakdown Structure</i>	6
2.3. Kebutuhan Sumber Daya.....	6
2.4. Rencana Jadwal Pelaksanaan Proyek.....	9
2.5. Rencana Nilai Proyek.....	9
BAB III PELAKSANAAN PROYEK	13
3.1. Realisasi Jadwal Pelaksanaan	13
3.2. Realisasi Hasil Pekerjaan	13
3.3. Penjamin Kualitas Proyek.....	114
3.4. Keberlanjutan Proyek.....	117
BAB IV KESIMPULAN DAN SARAN	118
4.1. Kesimpulan.....	118
4.2. Saran.....	118
LAMPIRAN	120
Lampiran 1. Proposal Proyek.....	120
Lampiran 2. MOU	130
Lampiran 3. LogBook Kelompok.....	135
Lampiran 4. LogBook Individu	138
Lampiran 5. Dokumentasi	143
Lampiran 6. Bukti Serah Terima	145
Lampiran 7. Bukti Pembayaran	146

Lampiran 8. Source Code	148
Lampiran 9. Link Video Profile	192
Lampiran 10. Poster	193
Lampiran 11. Slide Presentasi	194

DAFTAR GAMBAR

Gambar 2.1 Work Breakdown Structure.....	6
Gambar 3.1 Desain Tampilan Login Admin	14
Gambar 3.2 Desain Tampilan Dashboard Admin	15
Gambar 3.3 Desain Tampilan List Gambar Homepage Admin.....	16
Gambar 3.4 Desain Tampilan List Gambar About Us Admin	17
Gambar 3.5 Desain Tampilan List Gambar Casting Admin.....	18
Gambar 3.6 Desain Tampilan List Gambar Turning Admin	19
Gambar 3.7 Desain Tampilan List Gambar Drilling Admin	20
Gambar 3.8 Desain Tampilan List Gambar Galeri Admin.....	21
Gambar 3.9 Desain Tampilan Tambah Konten About Us Admin	22
Gambar 3.10 Desain Tampilan Tambah Layanan Casting Admin	22
Gambar 3.11 Desain Tampilan Tambah Produk Galeri Admin.....	23
Gambar 3.12 Desain Tampilan Pesanan Admin	24
Gambar 3.13 UseCase	25
Gambar 3.14 Activity Diagram Login Admin	26
Gambar 3.15 Activity Diagram Edit Konten Admin.....	27
Gambar 3.16 Activity Diagram Unduh Konten Pengguna	28
Gambar 3.17 Sequence Login Admin	29
Gambar 3.18 Sequence Pemesanan	30
Gambar 3.19 Sequence Diagram Mengunduh Konten	31
Gambar 3.20 Sequence Edit Konten Admin.....	32
Gambar 3. 21 BPMN Sebelum Memiliki System	33
Gambar 3. 22 BPMN Sesudah Memiliki System.....	33
Gambar 3.23 Desain Tampilan Beranda.....	34
Gambar 3.24 Desain Tampilan About Us	35
Gambar 3.25 Desain Tampilan Casting	36
Gambar 3.26 Desain Tampilan Turning.....	37
Gambar 3.27 Desain Tampilan Drilling.....	38
Gambar 3.28 Desain Tampilan Galeri.....	39
Gambar 3.29 Desain Tampilan Order.....	40
Gambar 3.30 Halaman Beranda	41

Gambar 3.31 Halaman About Us.....	42
Gambar 3.32 Halaman Casting.....	43
Gambar 3.33 Halaman Turning	44
Gambar 3.34 Halaman Drilling	45
Gambar 3.35 Halaman Galeri	45
Gambar 3.36 Halaman Order	46
Gambar 3.37 Halaman Home Admin	51
Gambar 3.38 Halaman About Us Admin	55
Gambar 3.39 Halaman Casting Admin	60
Gambar 3.40 Halaman Drilling Admin.....	68
Gambar 3.41 Halaman Ubah Password	76
Gambar 3.42 Halaman Ubah Username dan Password Admin	80
Gambar 3.43 Halaman Manajemen List Gambar Data Galeri.....	90
Gambar 3.44 Halaman List Gambar Turning Admin	92
Gambar 3. 45 Halaman Statistik Data Pesanan Admin	102
Gambar 3. 46 Halaman Daftar Pesanan Admin	104
Gambar 3.47 Halaman Kelola Pesanan Dashboard Admin	106
Gambar 3.48 Halaman Kelola Daftar Pesanan Admin	114
Gambar 3.49 Tampilan Cetak Pdf Laporan	114
Gambar 3.50 Daftar Responden.....	116
Gambar 3.51 Hasil Perhitungan	117

DAFTAR TABEL

Tabel 2.1 Rencana Jadwal Pelaksanaan Proyek	9
Tabel 2.2 Biaya Pemasukan.....	10
Tabel 2.3 Gaji.....	11
Tabel 2.4 Biaya Lain-lain.....	11
Tabel 2.5 Rekapitulasi Dana	12
Tabel 3. 1 Realisasi Jadwal Pelaksanaan	13
Tabel 3. 2 Hasil Pengujian Blackbox Testing.....	115

DAFTAR KODE PROGRAM

Kode Program 3.1 Pengaturan Database	50
Kode Program 3.2 Halaman Home Admin	52
Kode Program 3.3 Halaman Edit Konten Home Admin	55
Kode Program 3.4 Halaman About Us Admin	58
Kode Program 3.5 Halaman Edit About Us Admin	59
Kode Program 3.6 Halaman Casting Admin	62
Kode Program 3.7 Halaman Tambah Konten Casting Admin.....	64
Kode Program 3.8 Halaman Edit Konten Casting Admin	68
Kode Program 3. 9 Halaman Drilling Admin.....	70
Kode Program 3.10 Halaman Tambah Konten Drilling Admin	73
Kode Program 3.11 Halaman Edit Konten Drilling Admin	75
Kode Program 3.12 Halaman Forgot Password	79
Kode Program 3.13 Ubah Password.....	83
Kode Program 3.14 Update Username	86
Kode Program 3.15 Halaman Pengaturan Profile	89
Kode Program 3. 16 Manajemen Data Galeri	92
Kode Program 3.17 Tampilan Turning Admin	95
Kode Program 3.18 Tampilan Tambah Produk Turning Admin.....	97
Kode Program 3.19 Tampilan Edit Turning Admin.....	101
Kode Program 3.20 Statistik Data Pelanggan.....	103
Kode Program 3. 21 Daftar Pesanan Pelanggan.....	106
Kode Program 3. 22 Filter Search Bar Dashboard Admin	113

BAB I

PENDAHULUAN

1.1. Latar Belakang Masalah

CV Kembar Jaya Abadi merupakan perusahaan manufaktur dan konstruksi yang telah memiliki reputasi solid di industrinya. Dengan portofolio lebih dari 100 proyek dan keahlian teknis dalam pengecoran logam, pembubutan, dan pengeboran presisi tinggi, perusahaan ini telah membuktikan kemampuannya dalam menyediakan produk dan layanan berkualitas.

Namun, di tengah persaingan industri yang semakin kompetitif di era digital, kapabilitas operasional yang kuat saja tidak lagi cukup. Tantangan utama yang dihadapi CV Kembar Jaya Abadi adalah ketiadaan sebuah etalase digital yang profesional dalam bentuk situs web. Tanpa kehadiran online yang terstruktur, perusahaan mengalami kesulitan dalam menjangkau pasar yang lebih luas dan menampilkan rekam jejak serta skala kemampuannya secara maksimal kepada calon klien.

Sebuah situs web resmi akan menjadi pusat informasi dan kepercayaan bagi calon klien industri. Di platform tersebut, CV Kembar Jaya Abadi dapat menyajikan detail layanan, spesifikasi material, dan galeri proyek-proyek yang pernah dikerjakan dengan jelas dan mudah diakses. Bagi klien di sektor B2B (Business-to-Business), ketersediaan informasi teknis yang terperinci dan bukti kerja yang nyata adalah faktor krusial dalam mengambil keputusan.

Selain sebagai sarana pemasaran, proses internal perusahaan juga membutuhkan modernisasi. Alur kerja pemesanan yang bergantung pada metode manual seperti telepon atau email tidak efisien, sulit dilacak, dan rentan terhadap miskomunikasi spesifikasi teknis. Kehadiran sistem manajemen pesanan yang terintegrasi akan menyederhanakan proses ini, mulai dari penerimaan permintaan hingga pemantauan status produksi.

Oleh karena itu, proyek pengembangan situs web dan sistem admin ini menjadi langkah strategis yang sangat penting. Proyek ini tidak hanya akan membangun citra profesional dan memperluas jangkauan pasar CV Kembar Jaya Abadi, tetapi juga akan mengoptimalkan efisiensi operasional internal. Ini adalah fondasi digital yang diperlukan agar perusahaan dapat terus bertumbuh dan bersaing secara efektif di masa depan.

1.2. Project Charter

1. Tujuan

Tujuan dari proyek pengembangan sistem informasi CV Kembar Jaya Abadi adalah:

- a. Merancang dan membangun situs web korporat yang profesional, informatif, dan representatif untuk menampilkan kapabilitas CV Kembar Jaya Abadi di bidang manufaktur dan konstruksi.
- b. Mengembangkan Dashboard Admin terpusat yang memungkinkan pengelolaan konten situs web dan alur kerja pemesanan secara efisien, mulai dari penerimaan permintaan hingga penyelesaian produksi.
- c. Menyediakan platform digital yang terstruktur bagi calon klien untuk menemukan informasi layanan, melihat portofolio proyek, dan mengajukan permintaan penawaran melalui formulir pemesanan yang terintegrasi.

2. Ruang Lingkup

Ruang lingkup proyek pengembangan sistem informasi CV Kembar Jaya Abadi meliputi:

- a. Pengembangan sistem dengan dua antarmuka utama: situs web publik (Front-End) dan Dashboard Admin (Back-End).
- b. Implementasi sistem login yang aman untuk admin.
- c. Pengembangan Dashboard Admin dengan fitur-fitur berikut:
 1. Ringkasan status pesanan secara keseluruhan.
 2. Manajemen pesanan (melihat detail, mengubah status, memfilter, dan mencari pesanan).
 3. Sistem Manajemen Konten (CMS) untuk halaman *Homepage*, *About Us*, *Castings*, *Turning*, *Drilling*, dan *Galeri*.
- d. Pengembangan situs web publik dengan fitur-fitur berikut:
 4. Halaman profil perusahaan, layanan, galeri proyek, dan kontak.
 5. Formulir pemesanan online dengan kemampuan mengunggah file gambar teknik.
- e. Integrasi sistem di mana data dari formulir pemesanan di situs web publik secara otomatis masuk dan dapat dikelola melalui Dashboard Admin.

3. StakeHolder

Stakeholder dalam proyek ini meliputi:

- a. Pemilik/Manajemen CV Kembar Jaya Abadi: Sebagai sponsor utama proyek, klien, dan pengambil keputusan akhir.

- b. Manajer Proyek: Bertanggung jawab atas perencanaan, pelaksanaan, dan penyelesaian proyek sesuai jadwal dan anggaran.
- c. Tim Pengembang Sistem:
 - 6. Front-End Developer: Bertanggung jawab mengimplementasikan desain antarmuka menjadi situs web publik yang interaktif dan responsif.
 - 7. UI/UX Designer: Merancang alur, tampilan antarmuka, dan pengalaman pengguna untuk situs web publik dan Dashboard Admin agar mudah digunakan.
 - 8. Backend Developer: Bertanggung jawab menangani logika sisi server, database, keamanan, dan fungsionalitas Dashboard Admin.
- d. Admin CV Kembar Jaya Abadi: Pengguna utama Dashboard Admin yang bertanggung jawab mengelola pesanan dan memperbarui konten situs web.
- e. Klien/Pelanggan CV Kembar Jaya Abadi (Pengguna Akhir): Mengakses situs web publik untuk mencari informasi, melihat galeri, dan mengirimkan permintaan pemesanan.

BAB II

PERENCANAAN PROYEK

2.1. Analisis Kelayakan

Dengan analisis SWOT (Strengths, Weaknesses, Opportunities, Threats) pada proyek pengembangan sistem informasi ini, maka didapatkan:

1. Strengths (Kekuatan)

a. Peningkatan Citra dan Kredibilitas Profesional.

Situs web yang modern dan terstruktur akan secara signifikan meningkatkan citra CV Kembar Jaya Abadi sebagai perusahaan manufaktur dan konstruksi yang profesional dan dapat dipercaya.

b. Sentralisasi Informasi dan Portofolio.

Menjadi pusat informasi tunggal yang menampilkan detail layanan, spesifikasi teknis, dan galeri proyek secara komprehensif, memudahkan klien untuk mengevaluasi kapabilitas perusahaan.

c. Efisiensi Proses Bisnis.

Dashboard Admin menyederhanakan dan menstandarisasi alur kerja pemesanan, mulai dari penerimaan hingga pelacakan status produksi, mengurangi risiko miskomunikasi dan meningkatkan efisiensi operasional.

2. Weaknesses (Kelemahan)

a. Biaya Investasi dan Pemeliharaan.

Proyek memerlukan investasi awal untuk pengembangan, dan akan ada biaya berkelanjutan untuk pemeliharaan server, domain, serta pembaruan keamanan sistem.

b. Kebutuhan Pelatihan Internal.

Staf admin dari CV Kembar Jaya Abadi memerlukan pelatihan untuk dapat mengoperasikan Dashboard Admin dan mengelola konten serta pesanan secara efektif.

c. Ketergantungan pada Teknologi.

Operasional pemesanan menjadi bergantung pada ketersediaan sistem. Gangguan pada server atau koneksi internet dapat menghambat alur kerja.

3. Opportunities (Peluang)

a. Ekspansi Jangkauan Pasar.

Menjangkau segmen klien yang lebih luas di luar wilayah geografis saat ini melalui visibilitas online, membuka peluang untuk proyek berskala nasional.

b. Pemasaran Digital yang Terarah.

Memungkinkan implementasi strategi pemasaran digital seperti SEO (Search Engine Optimization) dan iklan online untuk menargetkan industri spesifik yang membutuhkan jasa manufaktur dan konstruksi.

c. Analisis Data untuk Pengambilan Keputusan.

Data pesanan yang terkumpul di dalam sistem dapat dianalisis untuk mengidentifikasi layanan atau jenis material yang paling diminati, membantu dalam pengambilan keputusan bisnis strategis.

4. Threats (Ancaman)

a. Ancaman Keamanan Siber.

Sistem online rentan terhadap ancaman seperti peretasan dan pencurian data pesanan atau informasi klien, yang dapat merusak reputasi perusahaan.

b. Persaingan Digital yang Ketat.

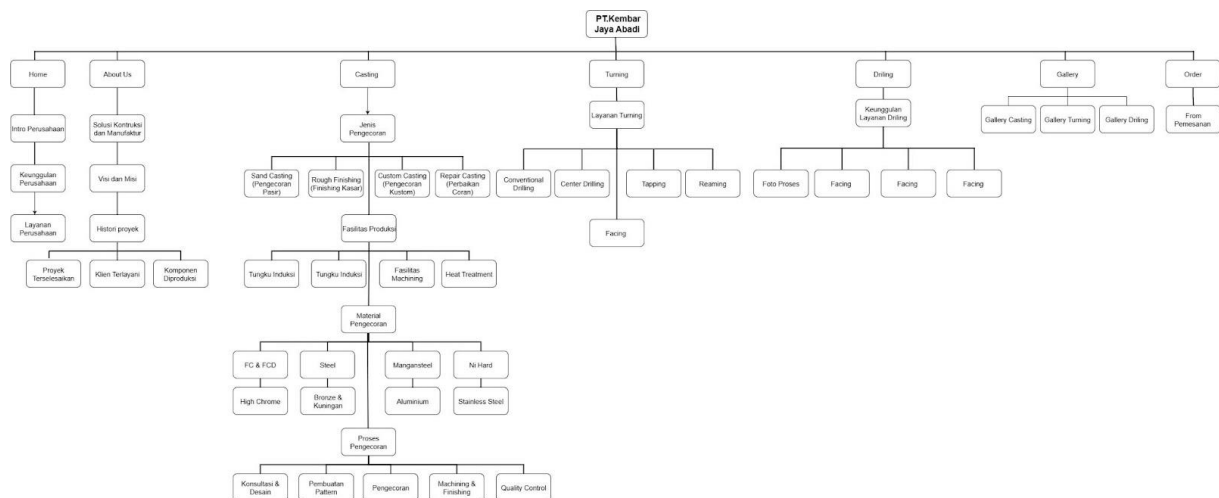
Kompetitor di industri sejenis mungkin sudah memiliki kehadiran digital yang matang, sehingga memerlukan upaya ekstra dalam pemasaran digital untuk dapat bersaing secara efektif.

c. Perubahan Teknologi yang Cepat.

Teknologi web berkembang dengan cepat. Sistem yang dikembangkan saat ini mungkin memerlukan pembaruan atau perombakan dalam beberapa tahun ke depan agar tetap relevan dan aman.

2.2. Work Breakdown Structure

Gambar 2.1 merupakan WBS CV Kembar Jaya Abadi, fungsinya adalah untuk memetakan seluruh struktur kegiatan perusahaan mulai dari informasi umum, layanan utama seperti casting, turning, dan drilling, hingga pendukung seperti galeri dan pemesanan. WBS ini membantu membagi proses bisnis menjadi bagian-bagian terperinci agar memudahkan pengelolaan setiap fungsi, pelacakan tahapan produksi, alur layanan, serta dokumentasi proyek dan material yang digunakan. Dengan WBS ini, perusahaan dapat mengelola aktivitas secara sistematis, memastikan setiap proses terpantau, dan mendukung efisiensi operasional secara keseluruhan.



Gambar 2.1 Work Breakdown Structure

2.3. Kebutuhan Sumber Daya

1. Sumber Daya Manusia

Proyek ini dikerjakan oleh 4 orang dengan pembagian jobdesk sebagai berikut:

a. Naufal Muzakki, NIM 2200018160

(*Project Manager*, Desain Tampilan, Testing) yang meliputi:

1. Mencari dan mengidentifikasi klien
2. Menjaga hubungan baik dengan klien agar klien selalu merasa puas dan terlibat dalam proyek
3. Penentuan Proyek dan Penyusunan rencana kerja yang mencakup jadwal, anggaran dan sumber daya
4. Mengatur tim dan mengalokasikan tugas sesuai keahliannya.

5. Memantau kinerja proyek untuk memastikan bahwa proyek tetap sesuai dengan jadwal dan pada jalur yang benar
 6. Memastikan semua fitur, tombol, dan alur sistem (baik di website publik maupun di dashboard admin) berfungsi sesuai dengan yang diharapkan sebelum diserahkan kepada klien.
 7. Pembuatan Tampilan Halaman Login Admin, Dashboard Admin, List Gambar Homepage, List Gambar About Us, List Gambar Casting, List Gambar Turning, List Gambar Drilling, List Gambar Galeri, Tambah About Us, Tambah Casting, Tambah Turning, Tambah Drilling, Tambah Galeri, Pesanan.
- b. Gia Sulipri, NIM 2200018171
- (Analisis dan Desain Tampilan) yang meliputi:
1. Sistem Analisis
 - a. Mengidentifikasi kebutuhan sistem dengan menggali informasi dari pengguna melalui observasi dan diskusi bersama stakeholder CV Kembar Jaya Abadi.
 - b. Membuat Use Case Diagram untuk memetakan interaksi antara aktor (pengguna dan admin) dengan sistem.
 - c. Menyusun Sequence Diagram untuk menggambarkan alur komunikasi antar objek dalam sistem pada CV Kembar Jaya Abadi
 - d. Merancang Business Process Diagram untuk menunjukkan alur proses bisnis layanan
 - e. Membuat Activity Diagram untuk memvisualisasikan alur logis kegiatan pengguna dan admin dari awal hingga akhir proses pada sistem.
 2. UI/UX Designer
 - a. Membuat wireframe dan prototype tampilan website menggunakan Figma, yang mencakup halaman Home, About Us, Layanan, Gallery, dan Order.
 - b. Mengimplementasikan prinsip User Experience (UX) seperti kemudahan navigasi, konsistensi visual, dan desain responsif untuk meningkatkan kenyamanan pengguna saat mengakses website.
- c. Felia Nadia Fikarda, NIM 2200018159
- (Frontend Developer dan Backend Developer) yang meliputi:
1. Frontend Developer
 - a. Mengembangkan tampilan antarmuka situs web.
 - b. Membuat halaman Home, About Us, Casting, Turning, Drilling, Gallery, dan Order untuk website profil CV Kembar Jaya Abadi menggunakan HTML, CSS, dan JavaScript.

- c. Mewujudkan desain UI/UX menjadi halaman web yang responsif dan interaktif.
- 2. Backend Developer
 - a. Menghubungkan data antara website publik dan dashboard admin.
 - b. Membuat halaman dan logika sistem untuk mengelola konten Home, About Us, Casting, dan Drilling di dashboard admin.
 - c. Mengembangkan back-end menggunakan PHP dan SQL untuk menyimpan data pemesanan dan konten ke dalam database.
 - d. Menyediakan sistem yang memungkinkan admin memperbarui konten secara dinamis melalui dashboard.
- d. Aisyah Dwi Puspita, NIM 2200018104
(Backend Developer) yang meliputi:
 - 1. Pembuatan Halaman Galeri Admin
 - 2. Pembuatan Halaman Turning Admin
- e. Annisa Aulia, NIM 2200018184
(Backend Developer) yang meliputi:
 - 1. Membuat halaman antarmuka dashboard admin yang menampilkan statistik pesanan pelanggan.
 - 2. Menghubungkan data pesanan pelanggan yang diambil melalui form pesanan pada website agar terhubung ke dashboard pesanan admin.
 - 3. Membuat halaman antarmuka admin untuk mengelola pesanan pelanggan.
 - 4. Menyediakan fitur pada dashboard admin yang memungkinkan admin mengelola pesanan secara mudah.
- 2. Sumber Daya Fisik
 - a. Hardware
 - 1. MSI Modern 14
 - 2. Acer National Geographic
 - 3. HP 240 G7
 - 4. Acer Nitro 5
 - 5. Asus Vivobook 14
 - b. Software
 - 1. Figma
 - 2. Vs Code

3. Draw.io
4. XAMPP PHP Myadmin
5. Chrome Browser, Safari Web Browser, Microsoft Edge Browser

2.4. Rencana Jadwal Pelaksanaan Proyek

Berdasarkan Tabel 2.1, dapat dilihat bahwa proyek dilaksanakan mulai bulan Maret hingga Juli dengan beberapa tahapan kegiatan. Kegiatan dimulai dengan mendefinisikan kebutuhan pengguna dan kebutuhan sistem pada bulan Maret hingga awal April. Selanjutnya dilakukan perancangan database dan pembuatan prototype UI yang berlangsung hingga akhir April. Implementasi program dilakukan pada bulan Mei sampai pertengahan Juli, kemudian dilanjutkan dengan tahap uji coba sistem dan penyerahan sistem pada bulan Juli. Setelah sistem diserahkan, tahap terakhir adalah maintenance sistem yang direncanakan pada akhir Juli.

Tabel 2.1 Rencana Jadwal Pelaksanaan Proyek

Kegiatan	Bulan																			
	Maret				April				Mei				Juni				Juli			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Mendefinisikan Kebutuhan Pengguna																				
Mendefinisikan Kebutuhan Sistem																				
Desain Database																				
Membuat Prototype UI																				
Implementasi Program																				
Uji Coba Sistem																				
Penyerahan Sistem																				
Maintenance Sistem																				

2.5. Rencana Nilai Proyek

1. Waktu Pengerjaan :
Mulai Bulan Maret 2025 hingga Bulan Agustus 2025
2. Fitur Proyek :
 - a. Login admin
 - b. Mengedit konten
 - c. Melihat konten

- d. Mengunduh konten
- e. Memberikan respon
- f. Pemesanan

3. Biaya Proyek :

a. Pemasukan

Berdasarkan Tabel 2.2, sumber pemasukan proyek berasal dari pembayaran klien yang terbagi menjadi dua tahap. Tahap pertama yaitu pembayaran DP sistem sebesar Rp 1.000.000, kemudian tahap kedua berupa pelunasan sistem sebesar Rp 500.000. Dengan demikian, total pemasukan yang diperoleh dari proyek ini adalah sebesar Rp 1.500.000.

Tabel 2.2 Biaya Pemasukan

Pemasukan	harga	Jumlah
Pembayaran DP Sistem dari Klien	Rp 1.000.000	Rp 1.000.000
Pelunasan Sistem dari Klien	Rp 500.000	Rp 500.000
Total Pemasukan		Rp 1.500.000

b. Pengeluaran

i. Gaji

Berdasarkan Tabel 2.3, dapat dilihat rincian biaya pengeluaran untuk setiap anggota tim sesuai dengan posisi dan tugas yang dikerjakan. Naufal Muzakki selaku Project Manajer & Desain bertanggung jawab pada manajemen proyek, pembuatan desain tampilan, serta pengujian sistem. Gia Sulipri sebagai analis & desainer mengerjakan berbagai kebutuhan seperti pembuatan diagram (Use Case, Activity, dan Sequence), serta penyusunan functional dan non-functional requirements. Felia Nadia Fikarda bertugas sebagai Front-End dan Back-End Developer dengan fokus pada pengembangan antarmuka pengguna dan koneksi database. Aisyah Dwi Puspita dan Annisa Aulia masing-masing berperan sebagai Back-End Developer dengan tanggung jawab pada pengembangan modul CMS dan manajemen sistem. Dari keseluruhan rincian gaji tersebut, total pengeluaran yang dialokasikan untuk pembayaran tim proyek adalah sebesar Rp 1.200.000.

Tabel 2.3 Gaji

Posisi	Kebutuhan	Jumlah	Harga	Sub Total
Naufal Muzakki (Project Manajer & Desain)	Manajemen Proyek	1	Rp 100.000	Rp 100.000
	Pembuatan Desain Tampilan	10	Rp 15.000	Rp 150.000
	Pengujian Sistem (Backbox Testing)	1	Rp 45.000	Rp 45.000
Gia Sulipri (Analisis & Desain)	pembuatan UseCase Diagram	1	Rp 45.000	Rp 45.000
	Pembuatan Activity Diagram	1	Rp 45.000	Rp 45.000
	Pembuatan Functional Requirements	1	Rp 45.000	Rp 45.000
	Pembuatan Non-Functional Requirements	1	Rp 45.000	Rp 45.000
	Pembuatan Sequence Diagram	1	Rp 45.000	Rp 45.000
	Pembuatan Desain Tampilan	7	Rp 15.000	Rp 105.000
	Pengembangan Front-End	1	Rp 125.000	Rp 125.000
Felia Nadi Fikarda (Front-End Developer & Back-End Developer)	Pengembangan Back-End (CMS Inti & Koneksi Database)	1	Rp 150.000	Rp 150.000
	Pengembangan Back-End (Modul CMS Galeri & Turning)	1	Rp 150.000	Rp 150.000
Aisyah Dwi Puspita (Back-End Developer)	Pengembangan Back-End (Modul Manajemen Pesanan & Dashboard)	1	Rp 150.000	Rp 150.000
Annisa Aulia (Back-End Developer)				
Total Pengeluaran				Rp 1.200.000

ii. Lain-lain

Berdasarkan Tabel 2.4, terdapat pengeluaran tambahan yang dialokasikan untuk kebutuhan domain dan hosting. Biaya yang dikeluarkan untuk satu paket domain dan hosting adalah sebesar Rp 300.000.

Tabel 2.4 Biaya Lain-lain

Kebutuhan	Jumlah	Harga	SubTotal
Doman dan Hosting	1	Rp 300.000	Rp 300.000

c. Rekapitulasi Dana

Berdasarkan Tabel 2.5, dapat dilihat perbandingan antara pemasukan dan pengeluaran proyek. Pemasukan diperoleh dari pembayaran klien, yaitu pembayaran DP sebesar Rp 1.000.000 dan pelunasan sistem sebesar Rp 500.000, sehingga total pemasukan mencapai Rp 1.500.000. Sementara itu, total pengeluaran proyek terdiri dari gaji tim sebesar Rp 1.200.000 dan biaya lain-lain sebesar Rp 300.000, sehingga total pengeluaran juga

mencapai Rp 1.500.000. Dengan demikian, sisa dana dari proyek ini adalah Rp 0 atau tidak terdapat kelebihan dana.

Tabel 2.5 Rekapitulasi Dana

Dana	Jumlah (Rp)
1. Pemasukan	
a. Pembayaran Dp	Rp 1.000.000
b. Pelunasan Sistem dari Klien	Rp 500.000
Total Pemasukan	Rp 1.500.000
2. Pengeluaran	
a. Total Gaji	Rp 1.200.000
b. Lain-lain	Rp 300.000
Total Pengeluaran	Rp 1.500.000
3. Sisa Dana	
a. Total Sisa Dana	Rp -

BAB III

PELAKSANAAN PROYEK

3.1. Realisasi Jadwal Pelaksanaan

Alur pelaksanaan proyek dijadwalkan secara sistematis selama enam bulan, diawali pada bulan Maret dengan tahap analisis yang mencakup pendefinisian kebutuhan pengguna (minggu 1-4) dan pendefinisian kebutuhan sistem (Maret minggu 2 - April minggu 1). Memasuki bulan April, proyek beralih ke tahap perancangan yang berjalan secara paralel. Kegiatan desain database dilaksanakan dari minggu pertama April hingga minggu ketiga Mei. Bersamaan dengan itu, proses pembuatan prototype UI dan implementasi program juga dimulai pada minggu pertama April, yang masing-masing dijadwalkan selesai pada minggu ketiga Juni dan minggu pertama Juli. Fase akhir proyek berfokus pada pengujian dan serah terima. Uji coba sistem akan dilaksanakan pada minggu kedua Juli, diikuti dengan penyerahan sistem secara resmi kepada klien pada minggu keempat Juli. Sebagai penutup, kegiatan maintenance sistem akan dilakukan sepanjang bulan Agustus untuk memastikan stabilitas dan fungsionalitas sistem pasca-implementasi.

Tabel 3. 1 Realisasi Jadwal Pelaksanaan

Kegiatan	Bulan																			
	Maret				April				Mei				Juni				Juli			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Mendefinisikan Kebutuhan Pengguna	■	■	■	■																
Mendefinisikan Kebutuhan Sistem		■	■	■																
Desain Database			■	■	■	■	■	■												
Membuat Prototype UI				■	■	■	■	■	■	■	■	■								
Implementasi Program									■	■	■	■	■	■	■	■	■	■		
Uji Coba Sistem																		■		
Penyerahan Sistem																			■	
Maintenance Sistem																			■	■

3.2. Realisasi Hasil Pekerjaan

1. Naufal Muzakki

(Project manajer dan desain tampilan)

a. Pembuatan Jadwal Project

Gambar di bawah merupakan diagram Gantt proyek telah disusun untuk memvisualisasikan jadwal pelaksanaan, merinci setiap tahapan kegiatan dari Maret hingga Agustus. Grafik ini

menjadi panduan visual yang efektif untuk memantau durasi, urutan, dan progres keseluruhan proyek.

Kegiatan	Bulan																			
	Maret				April				Mei				Juni				Juli			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Mendefinisikan Kebutuhan Pengguna																				
Mendefinisikan Kebutuhan Sistem																				
Desain Database																				
Membuat Prototype UI																				
Implementasi Program																				
Uji Coba Sistem																				
Penyerahan Sistem																				
Maintenance Sistem																				

b. Pembuatan Desain Tampilan Login Admin

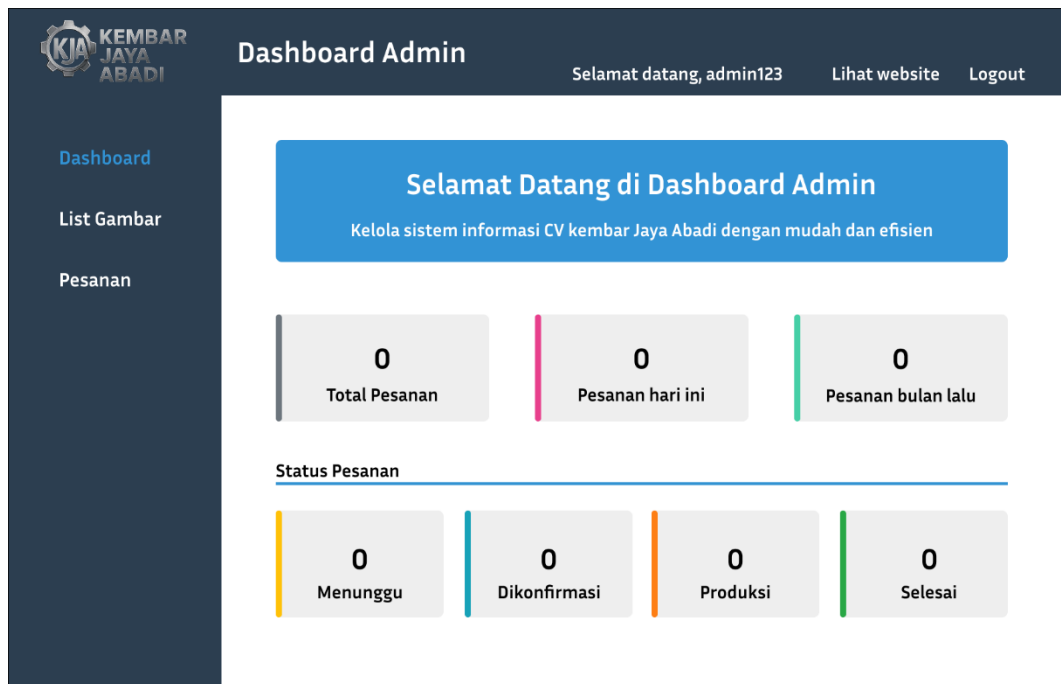
Gambar di bawah merupakan halaman login Admin Panel CV Kembar Jaya Abadi telah diselesaikan dengan mengedepankan konsep minimalis dan profesional untuk memastikan proses autentikasi yang mudah dan intuitif. Dengan layout terpusat dan hierarki visual yang jelas, halaman ini dirancang untuk memberikan pengalaman login yang efisien bagi administrator.

Gambar 3.1 Desain Tampilan Login Admin

c. Pembuatan Desain Tampilan Dashboard Admin

Gambar di bawah ini merupakan desain UI Dashboard Admin telah dirampungkan dengan tata letak tiga kolom yang intuitif untuk menyajikan data secara efisien melalui kartu statistik dan

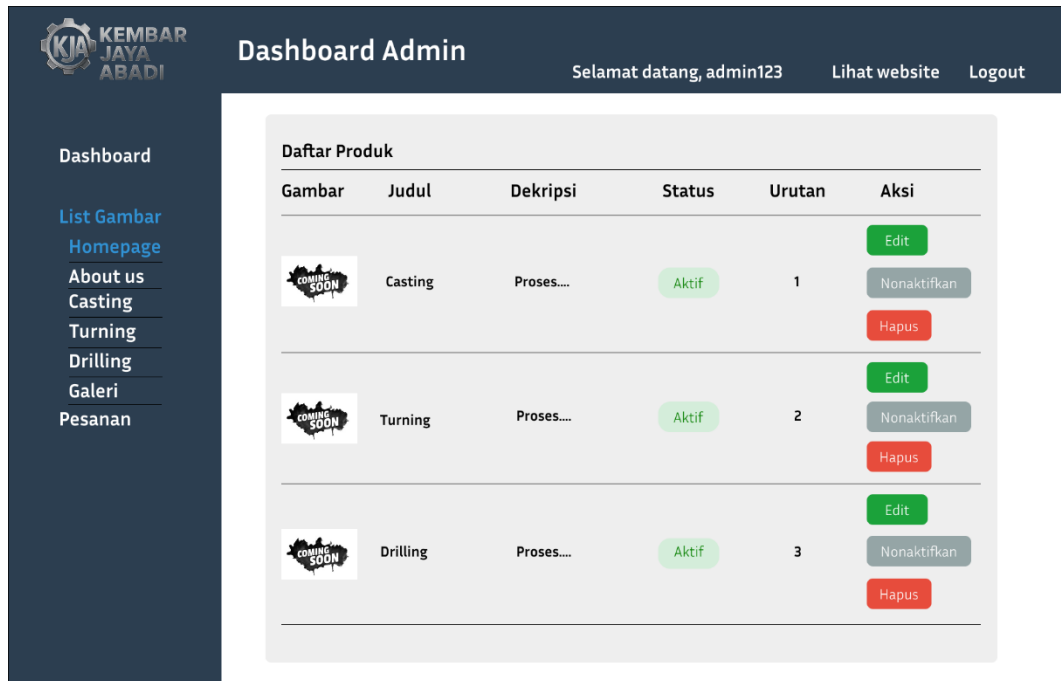
panel status berkode warna. Tampilan ini dirancang agar administrator dapat dengan cepat memantau ringkasan dan status pesanan secara visual dalam satu halaman utama.



Gambar 3.2 Desain Tampilan Dashboard Admin

d. Pembuatan Desain Tampilan List Gambar Homepage

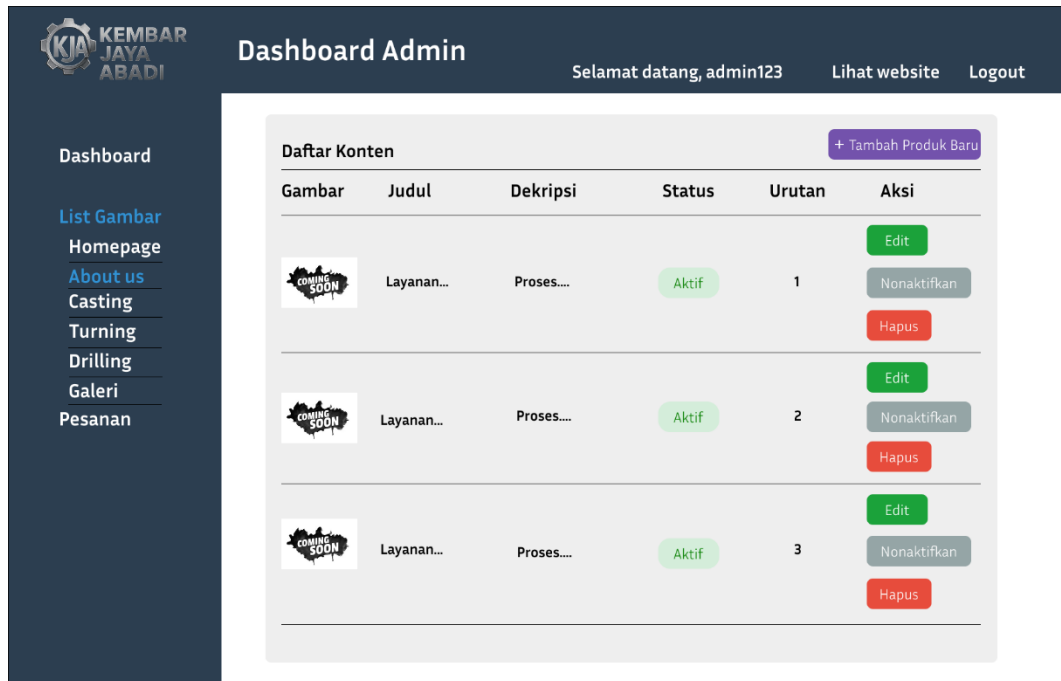
Gambar di bawah merupakan desain halaman "List Gambar Homepage" telah diselesaikan, menyajikan data dalam format tabel yang terorganisir untuk mempermudah pengelolaan konten produk atau layanan. Fungsionalitas utamanya berpusat pada kolom "Aksi" yang berisi tombol-tombol berkode warna (Edit, Nonaktifkan, Hapus), memastikan administrator dapat melakukan pengelolaan data secara cepat, intuitif, dan aman.



Gambar 3.3 Desain Tampilan List Gambar Homepage Admin

e. Pembuatan Desain Tampilan List Gambar About Us

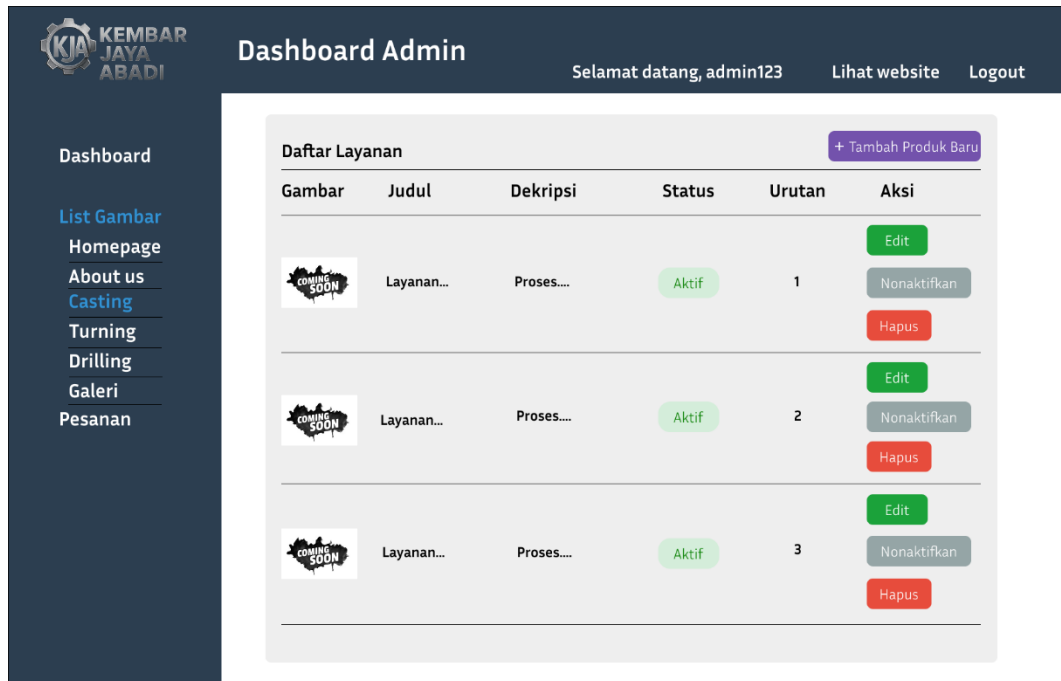
Gambar di bawah merupakan desain halaman "List Gambar About Us" kini telah disempurnakan dengan penambahan tombol utama "+ Tambah Produk Baru" di atas tabel untuk melengkapi fungsionalitas CRUD secara menyeluruh. Penambahan strategis ini, dikombinasikan dengan tabel data yang sudah ada, menciptakan antarmuka yang komprehensif untuk menambah dan mengelola seluruh konten situs web secara efisien.



Gambar 3.4 Desain Tampilan List Gambar About Us Admin

f. Pembuatan Desain Tampilan List Gambar Casting

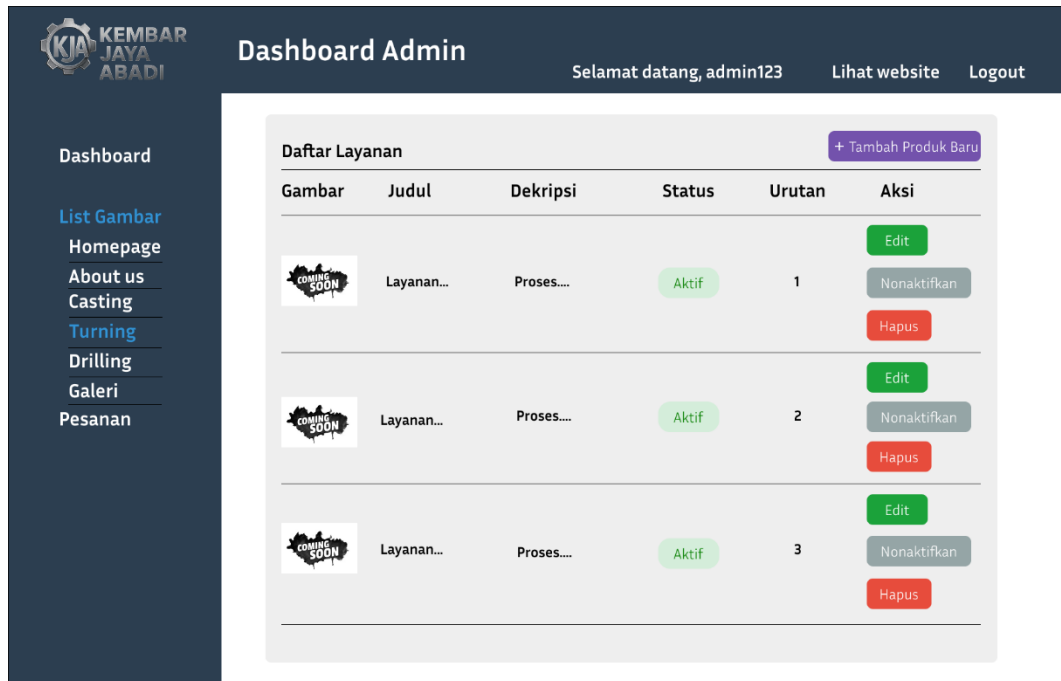
Gambar di bawah merupakan desain halaman "List Gambar Casting" kini telah disempurnakan dengan penambahan tombol utama "+ Tambah Produk Baru" di atas tabel untuk melengkapi fungsionalitas CRUD secara menyeluruh. Hal ini membuktikan konsistensi fungsionalitas dan template pada berbagai bagian manajemen, sehingga memberikan pengalaman pengguna yang efisien dan prediktif.



Gambar 3.5 Desain Tampilan List Gambar Casting Admin

g. Pembuatan Desain Tampilan List Gambar Turning

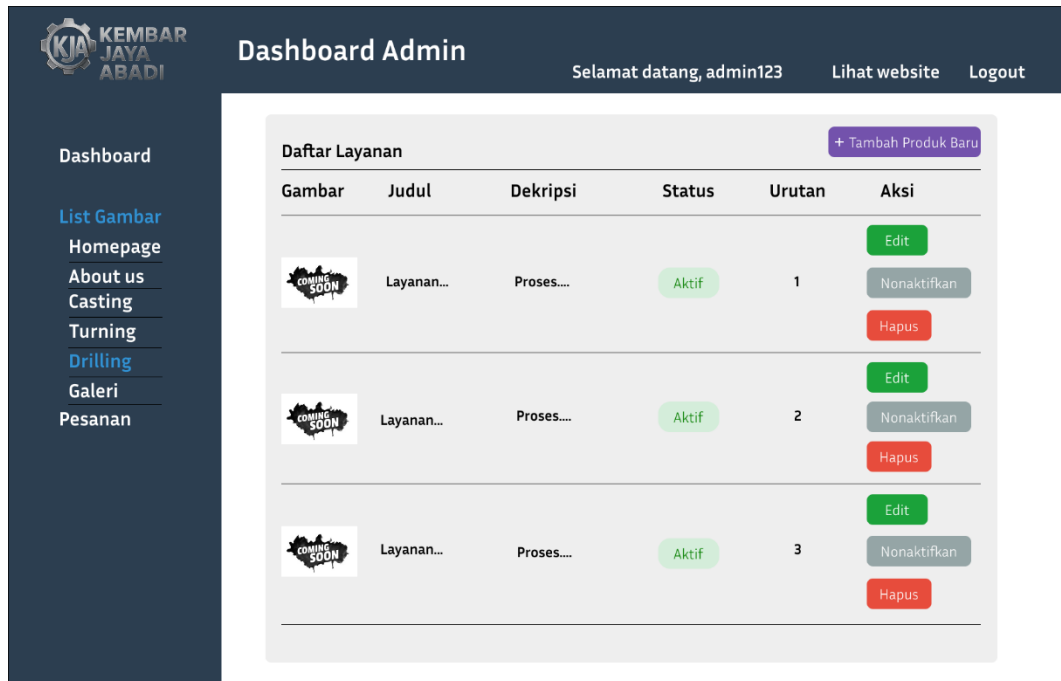
Gambar di bawah merupakan desain halaman "List Gambar Turning" kini telah disempurnakan dengan penambahan tombol utama "+ Tambah Produk Baru" di atas tabel untuk melengkapi fungsionalitas CRUD secara menyeluruh. Konsistensi ini memastikan pengalaman pengguna yang prediktif, memungkinkan administrator untuk mengelola berbagai layanan secara efisien tanpa perlu beradaptasi dengan antarmuka yang berbeda.



Gambar 3.6 Desain Tampilan List Gambar Turning Admin

h. Pembuatan Desain Tampilan List Gambar Drilling

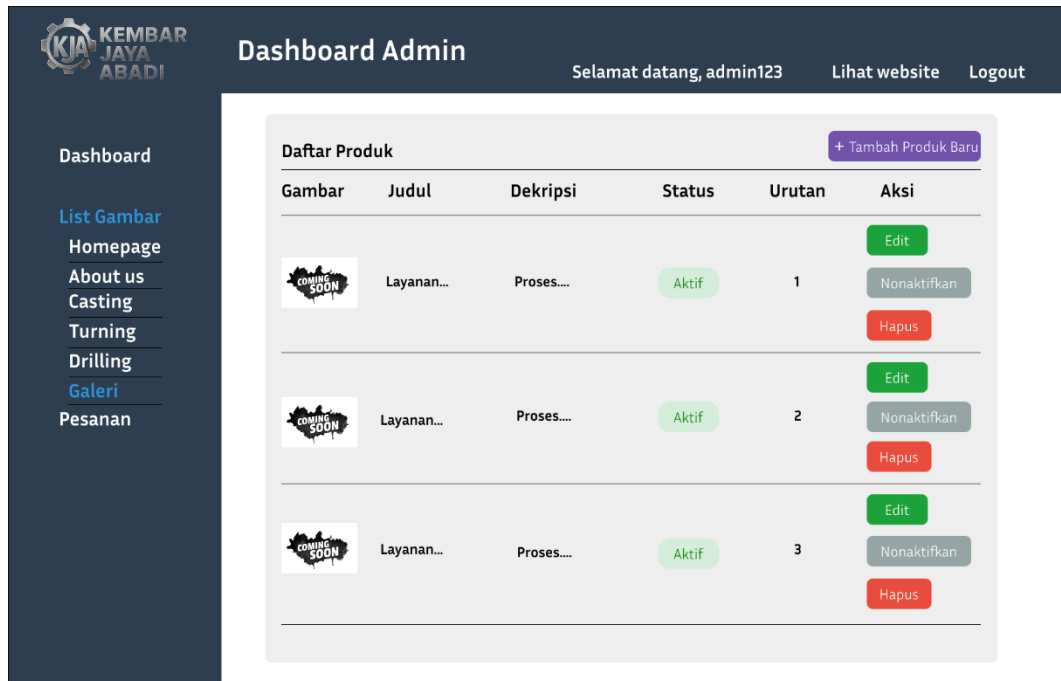
Gambar di bawah merupakan desain halaman "List Gambar Drilling" kini telah disempurnakan dengan penambahan tombol utama "+ Tambah Produk Baru" di atas tabel untuk melengkapi fungsionalitas CRUD secara menyeluruh. Pendekatan ini membuktikan keberhasilan desain modular yang menciptakan pengalaman pengguna yang mulus dan efisien di seluruh fitur manajemen layanan.



Gambar 3.7 Desain Tampilan List Gambar Drilling Admin

i. Pembuatan Desain Tampilan List Gambar Galeri

Gambar di bawah merupakan desain halaman "List Gambar Drilling" kini telah disempurnakan dengan penambahan tombol utama "+ Tambah Produk Baru" di atas tabel untuk melengkapi fungsionalitas CRUD secara menyeluruh. Fleksibilitas ini menunjukkan bahwa kerangka antarmuka yang sama dapat secara efektif digunakan untuk mengelola berbagai kategori konten yang berbeda, memastikan pengalaman pengguna yang seragam dan efisien.



Gambar 3.8 Desain Tampilan List Gambar Galeri Admin

j. Pembuatan Desain Tampilan Tambah Layanan, Produk, dan Konten

Gambar di bawah merupakan desain formulir "Tambah Konten" telah dirampungkan dengan tata letak yang bersih dan terfokus, menyediakan kolom input yang terstruktur secara logis untuk memudahkan entri data. Fitur kunci seperti teks bantuan pada unggahan file dan hierarki tombol aksi (Simpan/Batal) yang jelas diimplementasikan untuk menciptakan proses penambahan konten yang intuitif dan minim kesalahan.

Tambah Konten About Us

Informasi Konten
Judul Konten

Deskripsi

Gambar Produk
Format : JPG, PNG, GIF, WEBP (Maks. 5MB)
 Tidak ada file yang di pilih
Urutan Tampil

Gambar 3.9 Desain Tampilan Tambah Konten About Us Admin

Tambah Layanan Casting

Informasi Layanan
Judul Produk

Deskripsi

Gambar Produk
Format : JPG, PNG, GIF, WEBP (Maks. 5MB)
 Tidak ada file yang di pilih
Urutan Tampil

Gambar 3.10 Desain Tampilan Tambah Layanan Casting Admin

Tambah Produk Galeri

Informasi Produk

Judul Produk

Deskripsi

Gambar Produk

Format : JPG, PNG, GIF, WEBP (Maks. 5MB)

Pilih file Tidak ada file yang di pilih

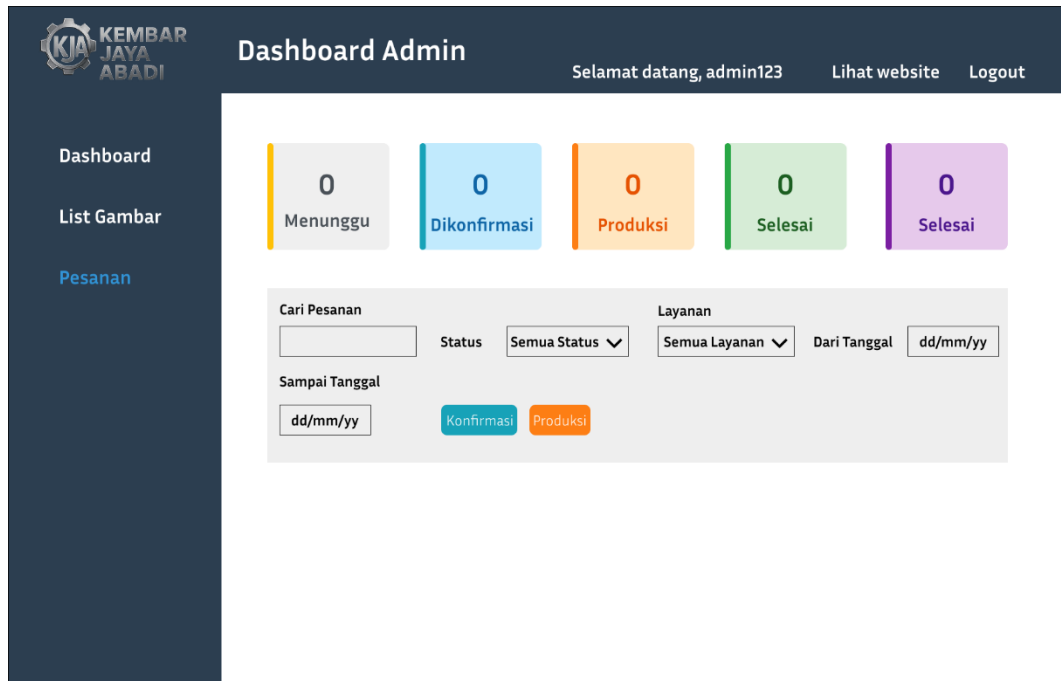
Urutan Tampil

Batal Simpan

Gambar 3.11 Desain Tampilan Tambah Produk Galeri Admin

k. Pembuatan Desain Tampilan Pesanan

Gambar di bawah merupakan desain halaman "Pesanan" menggabungkan kartu statistik berkode warna untuk ringkasan alur kerja secara visual dan sebuah panel filter yang komprehensif di bawahnya. Kombinasi ini memberikan perangkat yang kuat bagi administrator untuk memonitor gambaran umum pesanan secara sekilas sekaligus melakukan pencarian data yang spesifik dan mendetail.



Gambar 3.12 Desain Tampilan Pesanan Admin

2. Gia Sulipri

(Analisis dan desain tampilan)

a. Pembuatan FR

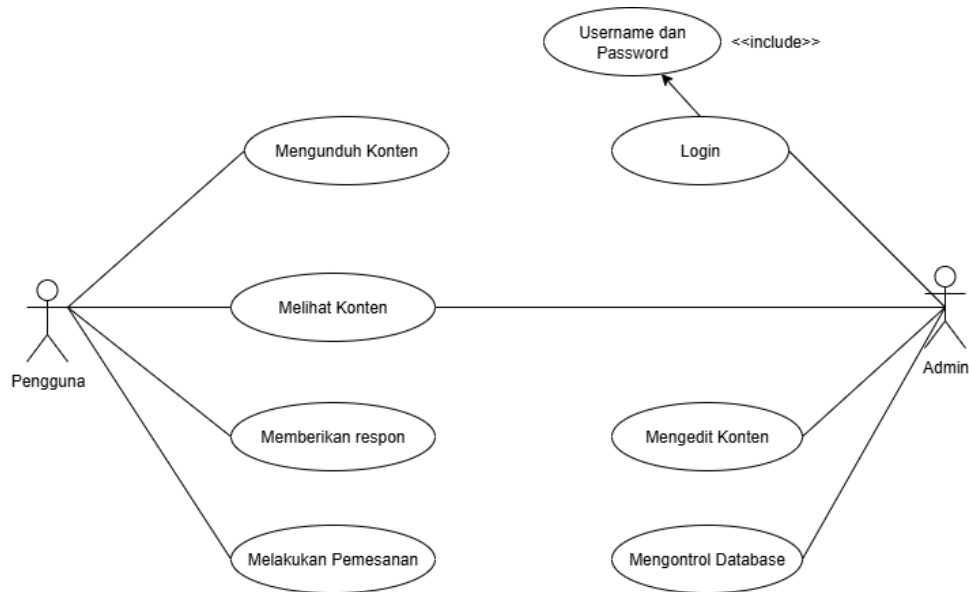
1. Menampilkan Halaman Informasi Perusahaan
2. Form Pemesanan Layanan
3. Penyimpanan dan Pengelolaan Data Pemesanan
4. Halaman Login Admin
5. Pengelolaan Data Pemesanan oleh Admin
6. Pengelolaan Galeri Gambar oleh Admin
7. Logout Admin
8. Validasi Form dan Keamanan Input

b. Pembuatan NFR

1. Ketersediaan Sistem (Availability)
2. Keamanan Sistem (Security)
3. Kinerja Sistem (Performance)
4. Kemudahan Penggunaan (Usability)
5. Kompatibilitas Perangkat dan Browser (Compatibility)
6. Responsif (Responsive Design)

c. UseCase

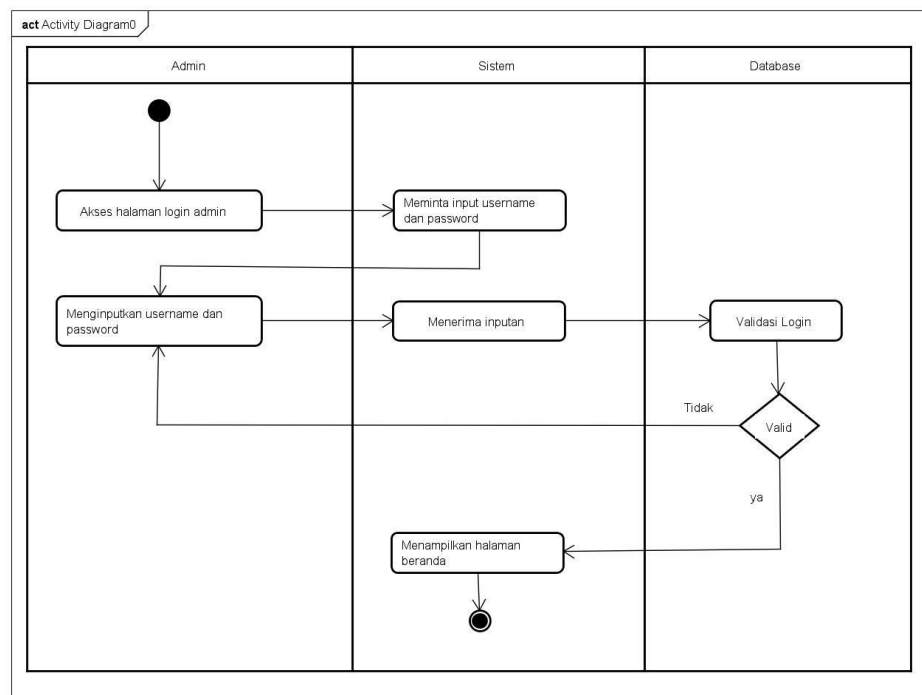
Use Case Diagram pada Gambar 3.13 menggambarkan interaksi antara Pengguna dan Admin dengan sistem. Pengguna dapat melihat konten, mengunduh konten, memberikan respon, serta melakukan pemesanan. Sementara itu, Admin harus login menggunakan username dan password sebelum bisa melihat konten, mengelola sistem. Setelah login, Admin dapat mengedit konten dan mengontrol database.



Gambar 3.13 UseCase

d. Activity Diagram Login Admin

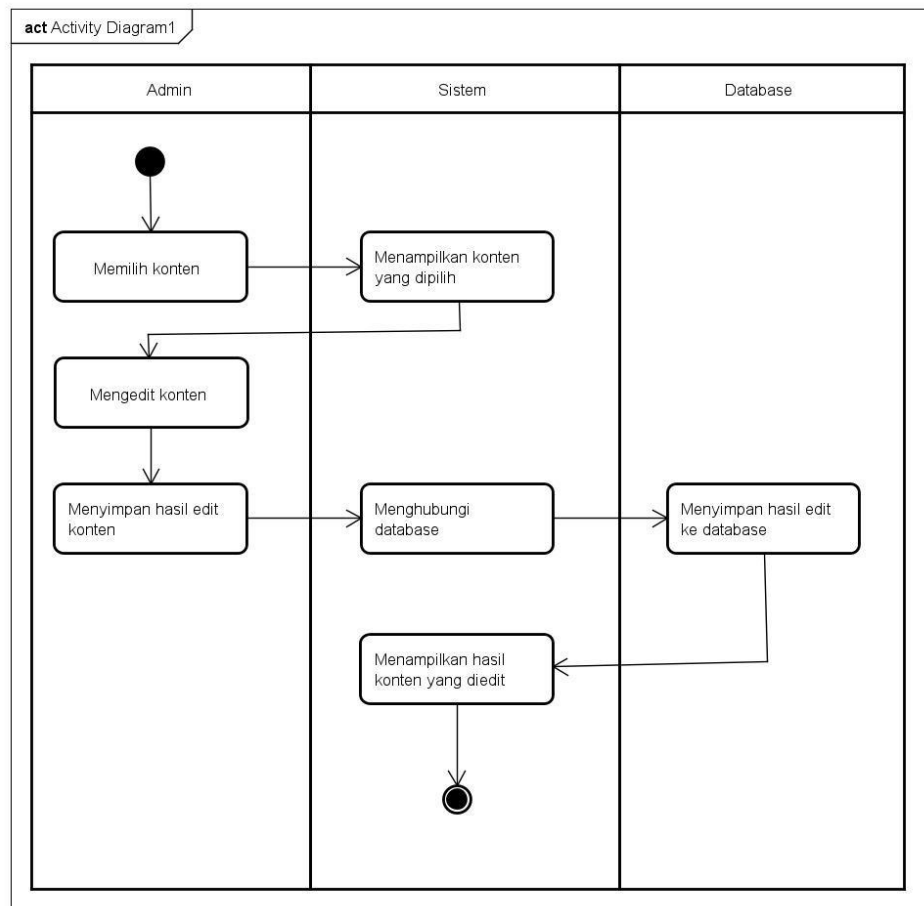
Activity Diagram pada Gambar 3.14 menjelaskan alur login admin. Proses dimulai ketika admin memasukkan username dan password. Sistem kemudian memverifikasi data tersebut ke database. Jika data valid, admin berhasil masuk ke dashboard, sedangkan jika tidak valid, sistem menampilkan pesan gagal login dan meminta admin mengulangi proses login.



Gambar 3.14 Activity Diagram Login Admin

e. Activity Diagram Edit Konten Admin

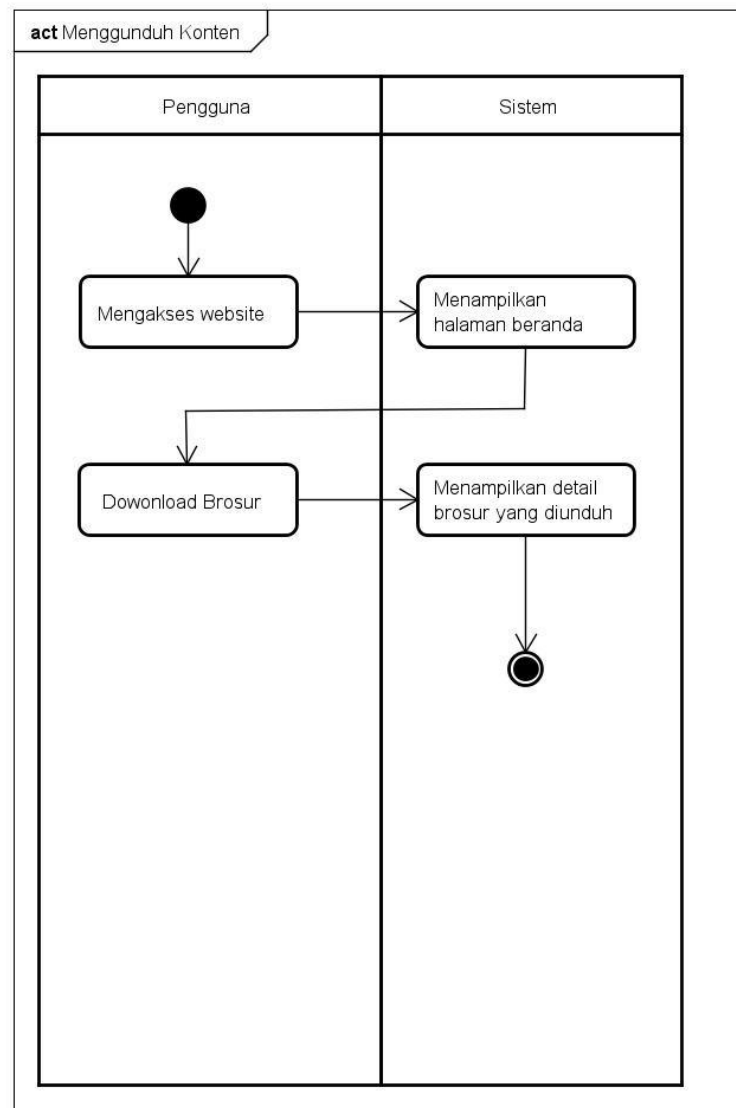
Activity Diagram pada Gambar 3.15 menggambarkan alur ketika admin melakukan pengelolaan konten. Admin memilih menu konten, lalu sistem menampilkan data yang ada. Setelah itu admin dapat mengedit konten sesuai kebutuhan, kemudian menyimpan hasil perubahan. Sistem akan memperbarui database dan menampilkan konten yang sudah diperbarui.



Gambar 3.15 Activity Diagram Edit Konten Admin

f. Activity Diagram Unduh Konten Pengguna

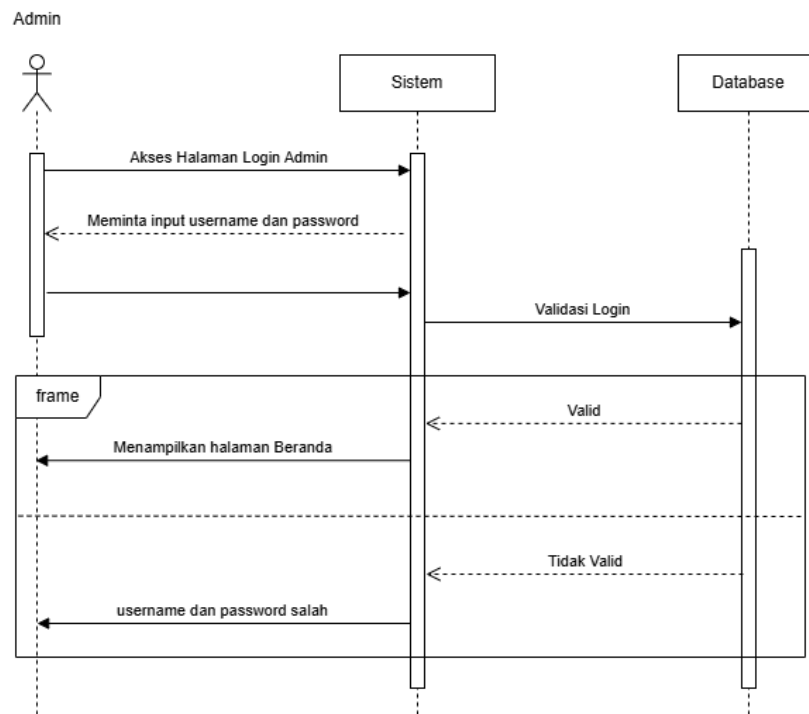
Activity Diagram pada Gambar 3.16 memperlihatkan proses pengguna saat mengunduh konten. Pengguna memilih file yang ingin diunduh, lalu sistem memproses permintaan tersebut. Setelah itu, sistem menyediakan file dan pengguna dapat menyimpan hasil unduhan ke perangkatnya.



Gambar 3.16 Activity Diagram Unduh Konten Pengguna

g. Squence Login Admin

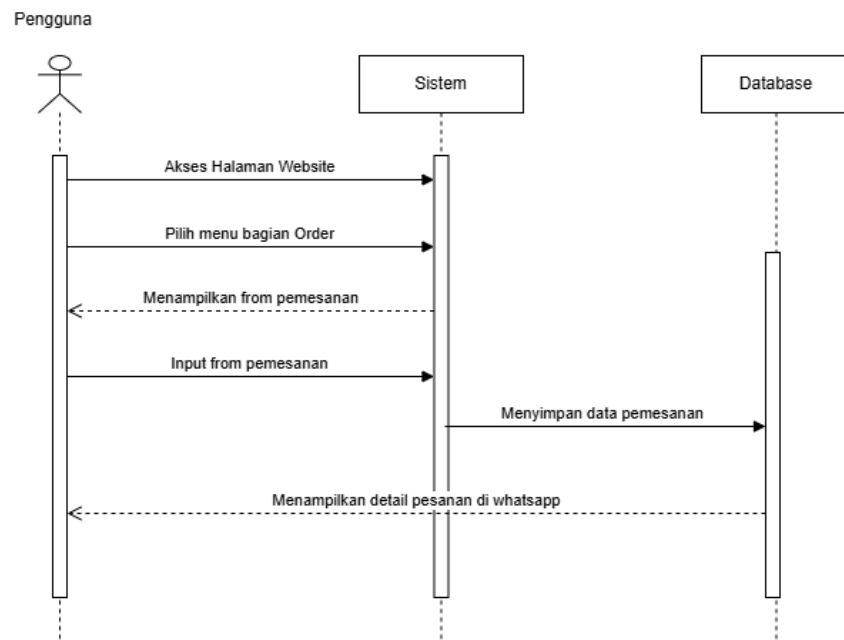
Sequence Diagram pada Gambar 3.17 menunjukkan proses login admin. Admin memasukkan username dan password melalui antarmuka login, kemudian sistem mengirim data ke database untuk diverifikasi. Jika data sesuai, sistem mengembalikan status berhasil login dan admin diarahkan ke dashboard. Jika tidak sesuai, sistem menampilkan pesan gagal login.



Gambar 3.17 Sequence Login Admin

h. Squence Pemesanan

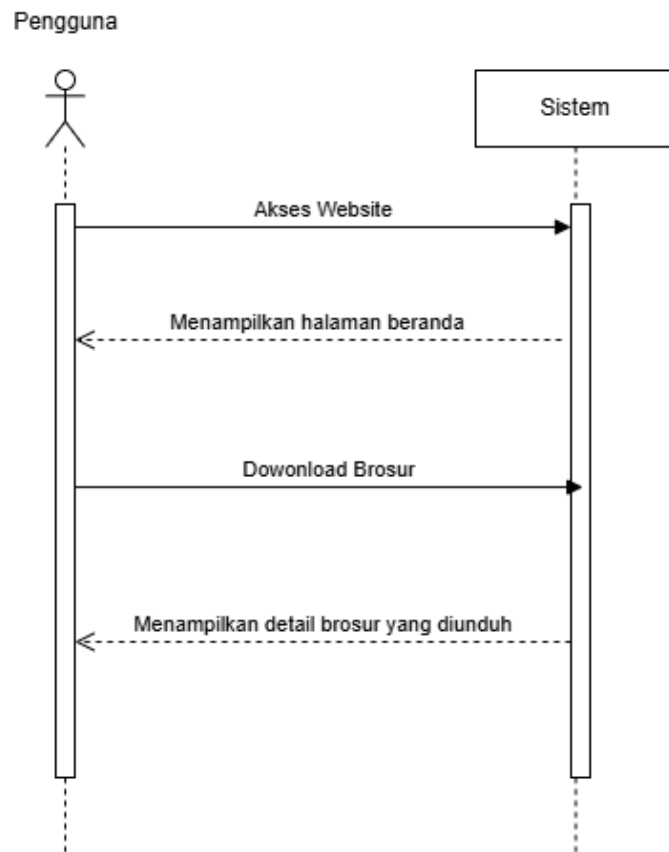
Sequence Diagram pada Gambar 3.18 menggambarkan proses pemesanan layanan. Pengguna mengisi form pemesanan, kemudian sistem memvalidasi data dan menyimpannya ke database. Setelah tersimpan, sistem memberikan konfirmasi bahwa pemesanan berhasil, dan data pesanan dapat dilihat oleh admin melalui dashboard.



Gambar 3.18 Sequence Pemesanan

i. Sequence Diagram Mengunduh konten

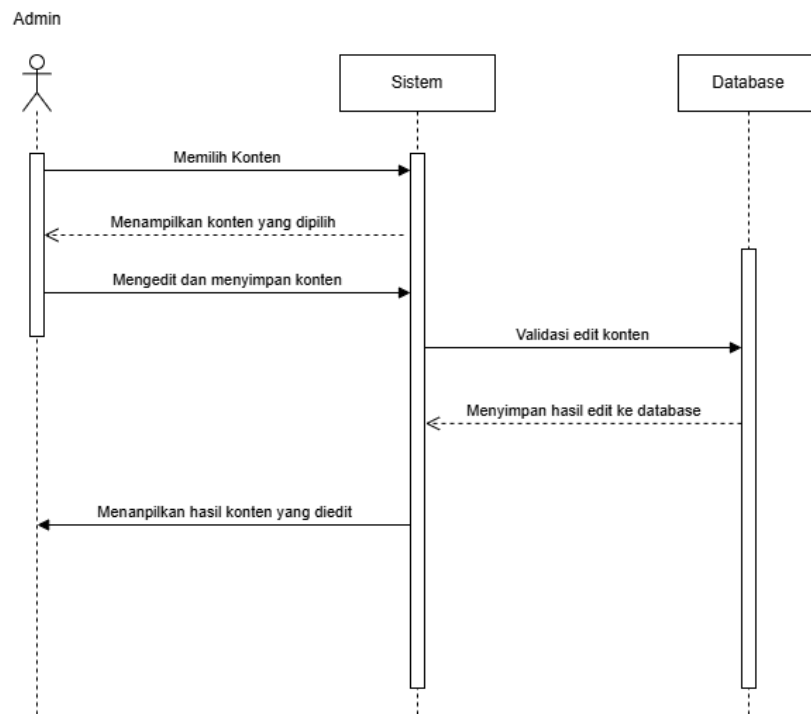
Sequence Diagram pada Gambar 3.19 menjelaskan proses ketika pengguna mengunduh konten. Pengguna memilih file yang ingin diunduh, lalu sistem memproses permintaan tersebut dengan mengambil data dari database. Setelah file tersedia, sistem mengirimkannya kembali kepada pengguna untuk diunduh.



Gambar 3.19 Sequence Diagram Mengunduh Konten

j. Squence Edit konten Admin

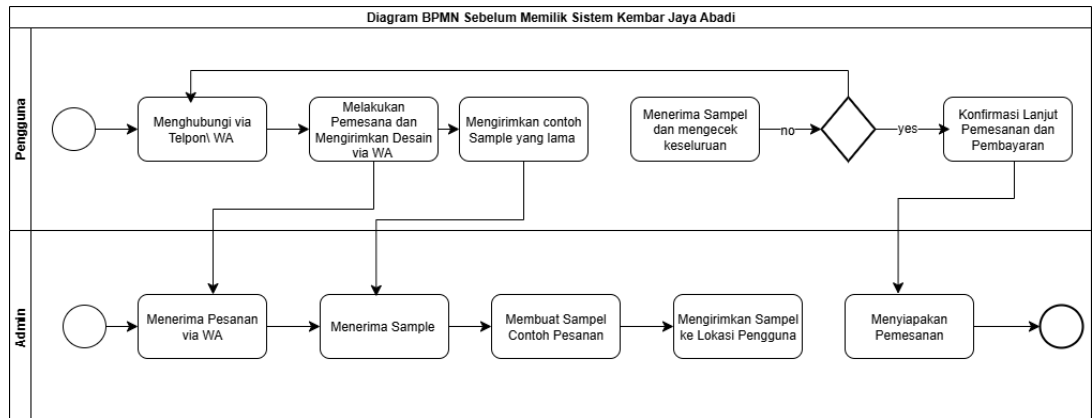
Sequence Diagram pada Gambar 3.20 menjelaskan alur proses ketika admin mengedit konten. Pertama, admin memilih konten yang ingin diperbarui, kemudian sistem menampilkan data konten tersebut. Selanjutnya admin melakukan perubahan dan menyimpan hasil edit. Sistem memvalidasi perubahan yang dilakukan lalu mengirimkan perintah ke database untuk menyimpan hasil edit konten. Setelah data berhasil diperbarui, sistem menampilkan kembali konten terbaru yang sudah diedit agar admin dapat melihat hasil perubahannya.



Gambar 3.20 Sequence Edit Konten Admin

k. BPMN sebelum memiliki system

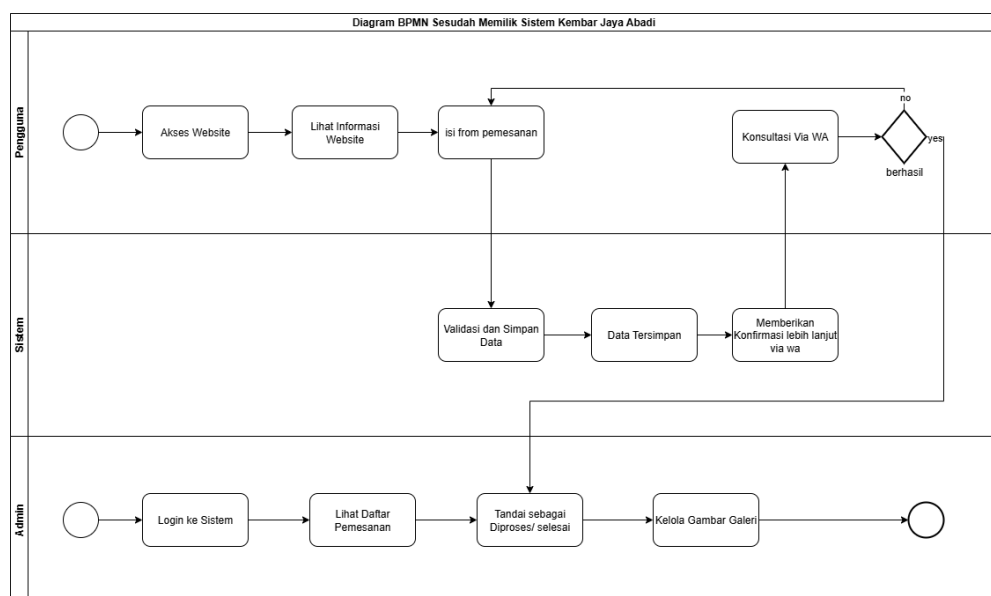
BPMN pada Gambar 3.21 memperlihatkan alur proses bisnis pemesanan sebelum adanya sistem informasi. Proses dimulai ketika pengguna menghubungi admin melalui telepon atau WhatsApp untuk melakukan pemesanan dan mengirimkan desain pesanan. Admin kemudian menerima pesanan tersebut dan membuat contoh sampel. Sampel yang sudah jadi dikirimkan kembali ke pengguna. Setelah pengguna menerima sampel, mereka memeriksa kesesuaian dengan desain yang dikirim. Jika sesuai, pengguna mengonfirmasi untuk melanjutkan pesanan dan melakukan pembayaran. Selanjutnya admin menyelesaikan pesanan hingga proses akhir. Alur ini menggambarkan bahwa proses masih manual, membutuhkan komunikasi langsung, dan memakan waktu lebih lama.



Gambar 3. 21 BPMN Sebelum Memiliki System

I. BPMN sesudah memiliki system

BPMN pada Gambar 3.22 memperlihatkan alur proses bisnis setelah perusahaan memiliki sistem informasi. Proses dimulai ketika pengguna mengakses website, melihat informasi yang tersedia, lalu mengisi form pemesanan secara online. Data yang dikirim pengguna divalidasi dan otomatis tersimpan oleh sistem. Admin yang sudah login dapat melihat daftar pemesanan melalui dashboard, kemudian menandai status pesanan sebagai diproses atau selesai. Jika diperlukan, pengguna masih bisa melakukan konsultasi lebih lanjut melalui WhatsApp. Setelah proses pesanan berhasil, sistem menampilkan hasil akhir berupa gambar atau informasi pesanan yang sudah jadi. Alur ini jauh lebih efisien dibandingkan proses manual, karena pemesanan dan pencatatan data dilakukan secara otomatis dan terdokumentasi dengan baik.



Gambar 3. 22 BPMN Sesudah Memiliki System

m. Desain Tampilan Beranda

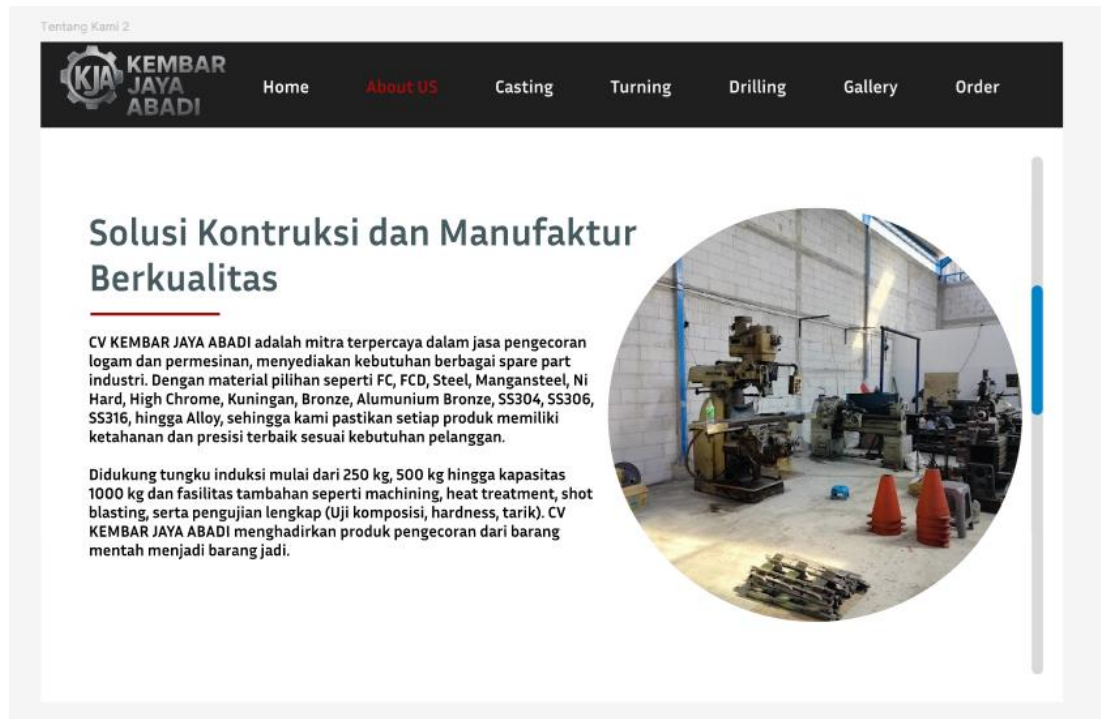
Pada Gambar 3.23 Pemilihan warna dominan merah dan hitam pada website perusahaan logam seperti CV Kembar Jaya Abadi sangat tepat secara psikologis dan fungsional dalam konteks UI/UX. Warna hitam merepresentasikan kekuatan, kestabilan, dan profesionalisme, yang selaras dengan karakter industri logam yang kokoh dan presisi. Sementara itu, warna merah memberikan kesan energi, keberanian, dan urgensi, menciptakan daya tarik visual yang kuat



Gambar 3.23 Desain Tampilan Beranda

n. Desain Tampilan About Us

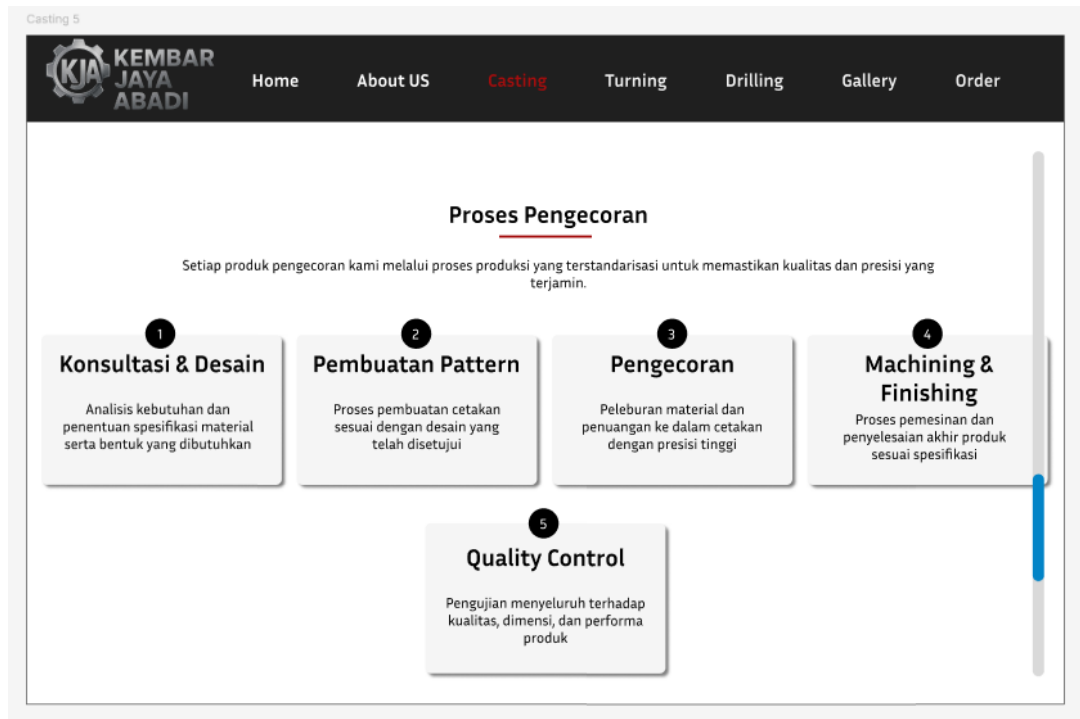
Pada Gambar 3.24 penggunaan gambar asli pada halaman "About Us" memperkuat kredibilitas visual dan membantu membangun trust kepada pengunjung website. Didukung dengan layout seimbang, pemilihan warna yang seimbang, dan konten yang padat namun relevan, halaman ini telah efektif untuk perusahaan industri seperti CV Kembar Jaya Abadi.



Gambar 3.24 Desain Tampilan About Us

o. Desain Tampilan Casting

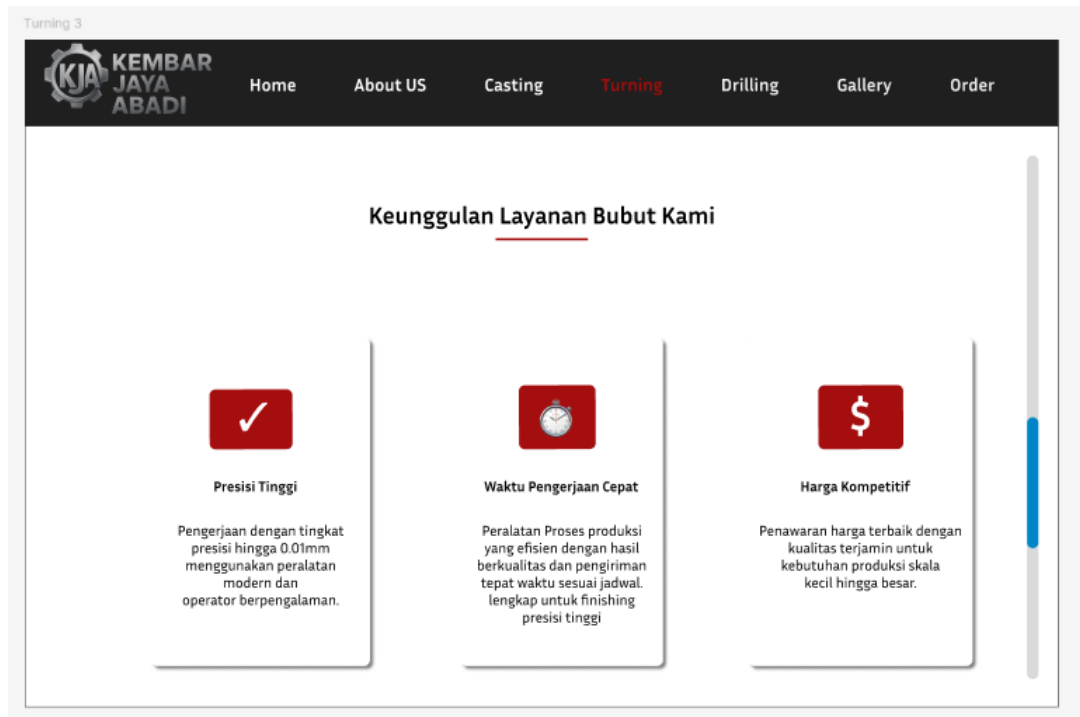
Pada Gambar 3.25 halaman "Casting" sudah sesuai prinsip UI/UX dengan alur visual yang jelas dan berurutan, memudahkan pengguna memahami langkah produksi. Penggunaan nomor, layout horizontal, dan card-style mendukung keterbacaan dan struktur informasi. Warna dan tipografi konsisten, memperkuat identitas visual dan fokus pengguna.



Gambar 3.25 Desain Tampilan Casting

p. Desain Tampilan Turning

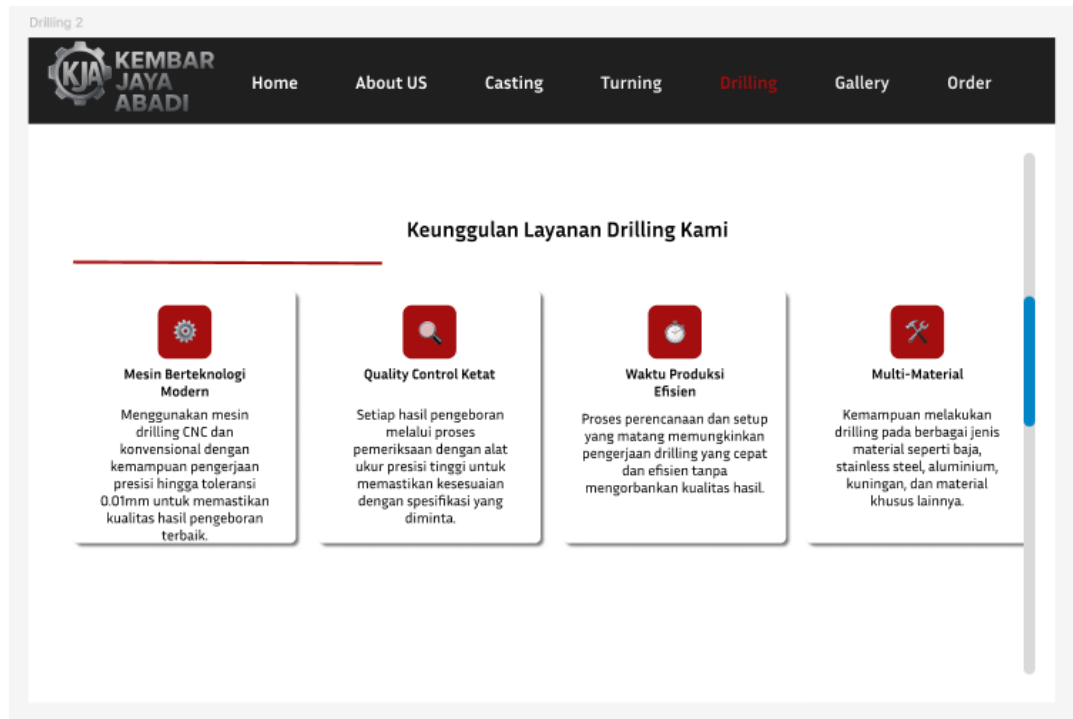
Pada Gambar 3.26 bagian "Turning" ini sudah sesuai prinsip UI/UX dengan penyajian informasi keunggulan layanan yang ringkas dan terstruktur. Penggunaan ikon di atas judul memperkuat pemahaman visual (visual cue), mendukung prinsip recognition over recall. Layout tiga kolom seimbang memudahkan pengguna memindai konten secara cepat (scannability) dan mendukung gestalt principle of proximity. Warna merah sebagai aksen konsisten dengan branding, sekaligus menarik perhatian pada poin penting. Desain ini efektif dalam menyampaikan nilai layanan secara jelas dan menarik.



Gambar 3.26 Desain Tampilan Turning

q. Desain Tampilan Drilling

Pada Gambar 3.27 bagian "Drilling" telah menerapkan prinsip UI/UX yang baik dengan tata letak empat kolom informasi yang sejajar dan seimbang, memudahkan pengguna untuk melakukan scanning cepat terhadap keunggulan layanan. Ikon berwarna merah di setiap card memperkuat pemahaman visual (prinsip recognition over recall) dan menjaga konsistensi branding. Penempatan judul, ikon, dan deskripsi dibuat hirarkis, mengikuti prinsip visual hierarchy untuk meningkatkan keterbacaan. Warna merah sebagai aksen digunakan secara



Gambar 3.27 Desain Tampilan Drilling

r. Desain Tampilan Galeri

Pada Gambar 3.28 bagian "Gallery" pada website ini telah mengikuti prinsip UI/UX yang baik, terutama dari segi navigasi visual dan kategorisasi konten. Penggunaan filter kategori (Semua, Casting, Turning, Drilling) memudahkan pengguna menyaring konten sesuai minat, sesuai prinsip user control and freedom. Penempatan teks "Coming Soon" pada kartu gambar menjaga ekspektasi pengguna dan mendukung prinsip transparansi informasi, agar tidak menimbulkan kebingungan. Layout grid yang rapi dengan spacing seragam memudahkan scannability dan memberikan kesan bersih. Warna aksen merah pada tombol "Semua" menjaga konsistensi branding sekaligus memberi penanda status aktif pada filter.



Gambar 3.28 Desain Tampilan Galeri

s. Desain Tampilan Order

Pada Gambar 3.29 Bagian Order pada website ini memiliki tampilan yang sederhana dan terstruktur dengan baik, menampilkan form isian untuk Email, Nama, dan No. Telepon secara vertikal agar mudah diisi oleh pengguna. Desainnya menggunakan prinsip UI/UX yang jelas dengan penandaan wajib isi pada email, konsistensi ukuran kolom, serta tata letak yang responsif. Hal ini mendukung pengalaman pengguna yang efisien dan mudah dipahami, dengan penambahan tombol kirim di sebelah kanan yang memudahkan pengguna.

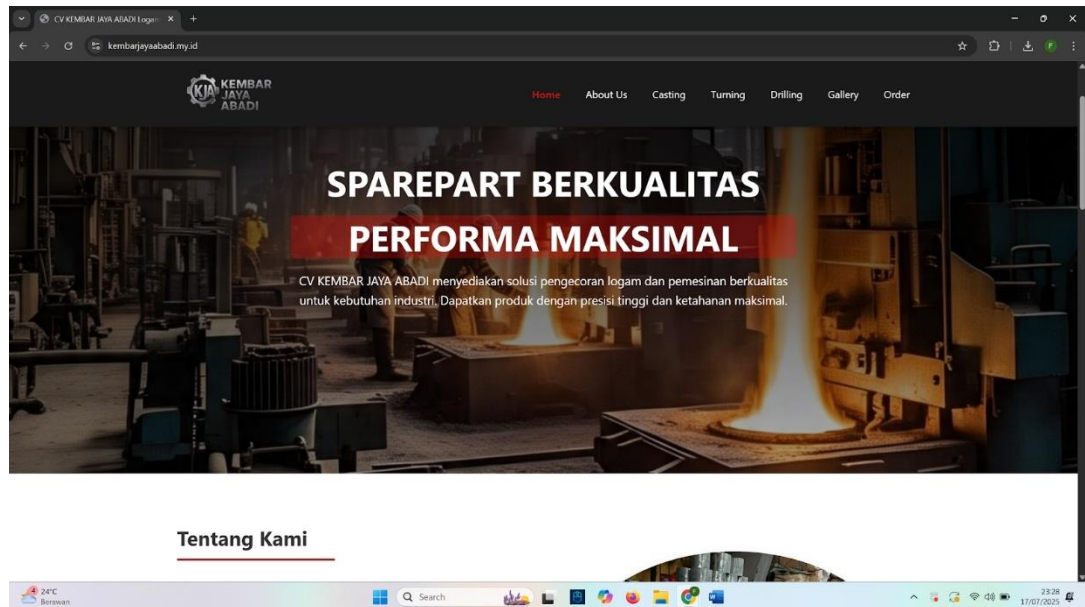
Gambar 3.29 Desain Tampilan Order

3. Felia Nadia Fikarda

(Frontend Developer dan Backend Developer)

a. Tampilan Home

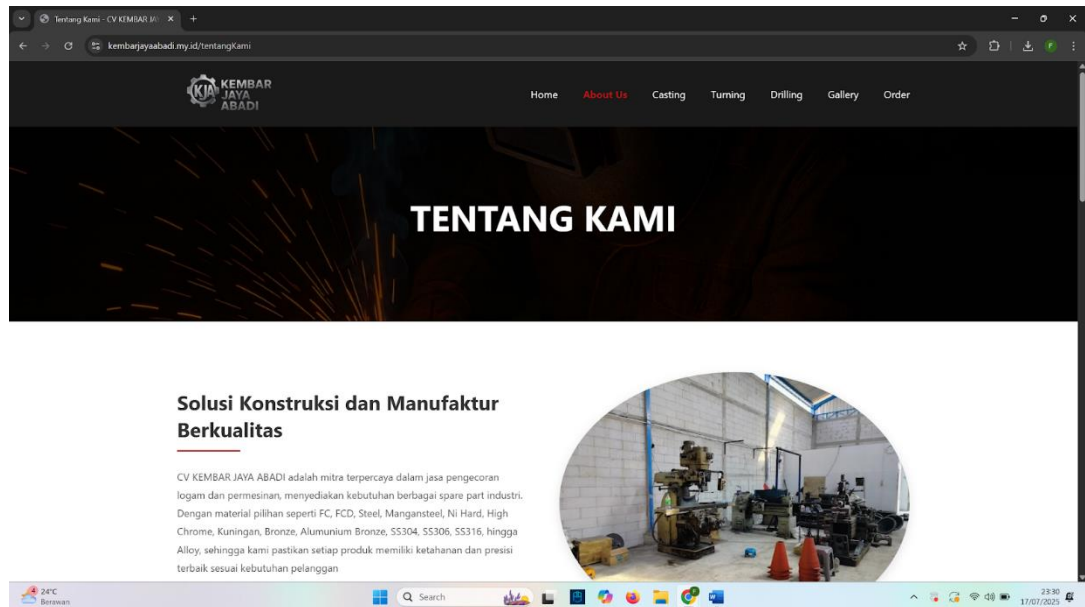
Pada Gambar 3.30 merupakan halaman utama yang dilihat pengunjung saat membuka *website* CV Kembar Jaya Abadi. Halaman ini dibuat untuk menunjukkan identitas perusahaan sebagai penyedia layanan pengecoran, pembubutan dan pengeboran logam. Di bagian atas halaman, terdapat menu navigasi seperti *About Us*, *Casting*, *Turning*, *Drilling*, *Gallery*, dan *Order* yang memudahkan pengguna untuk menjelajahi informasi yang tersedia. Selanjutnya, terdapat bagian hero banner dengan latar belakang foto proses pengecoran logam serta teks utama “SPAREPART BERKUALITAS, PERFORMA MAKSIMAL” yang disorot dengan desain kontras agar menarik perhatian. Di bawahnya terdapat deskripsi singkat mengenai layanan utama perusahaan, yaitu menyediakan solusi pengecoran, pembubutan dan pengeboran logam berkualitas tinggi untuk kebutuhan industri.



Gambar 3.30 Halaman Beranda

b. Tampilan About Us

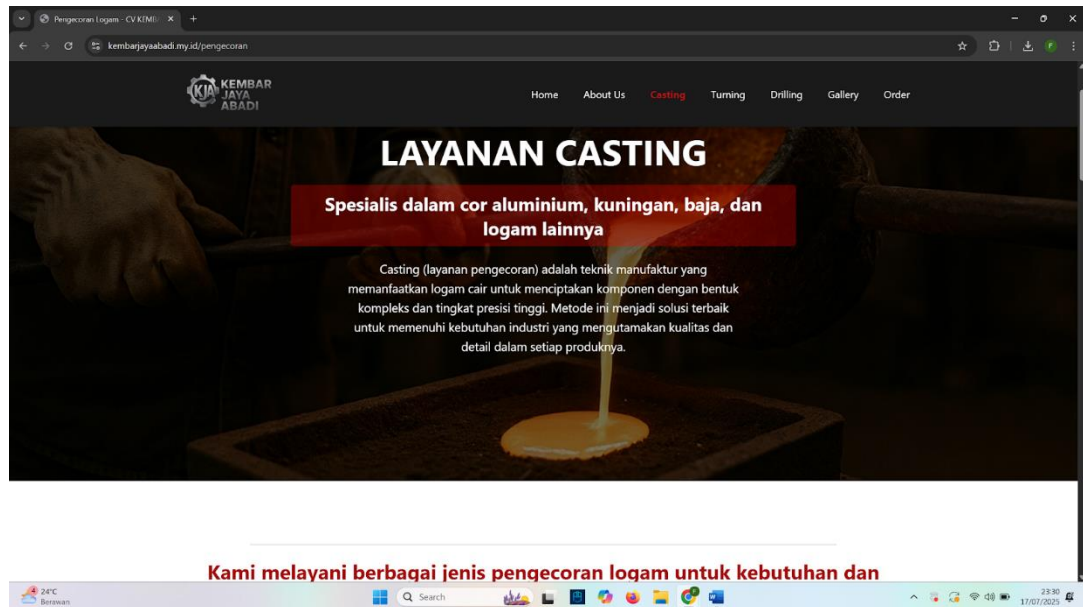
Gambar 3.31 merupakan halaman About Us berisi informasi lengkap mengenai profil CV Kembar Jaya Abadi. Di bagian awal, dijelaskan bahwa perusahaan ini bergerak di bidang pengecoran dan permesinan logam, dengan dukungan berbagai material berkualitas seperti FC, FCD, Steel, Stainless, dan lainnya. Terdapat juga foto lingkungan kerja sebagai bukti nyata kegiatan produksi yang dilakukan. Kemudian, bagian Layanan Kami menjelaskan bahwa perusahaan menyediakan jasa konstruksi dan manufaktur untuk memenuhi kebutuhan industri dan pembangunan di wilayah Jawa Tengah. Dengan dukungan tenaga ahli dan teknologi modern, CV Kembar Jaya Abadi memastikan setiap proyek selesai tepat waktu dan sesuai standar. Pada bagian Visi dan Misi, tuliskan tujuan utama perusahaan, yaitu memberikan hasil terbaik untuk pelanggan, menjaga kualitas layanan, serta tetap peduli terhadap lingkungan melalui proses produksi yang efisien dan ramah lingkungan. Terakhir, ditampilkan data pencapaian seperti jumlah proyek yang telah diselesaikan, klien yang sudah dilayani, dan komponen yang berhasil diproduksi. Angka-angka ini menjadi bukti bahwa perusahaan memiliki pengalaman dan kepercayaan dari banyak pihak.



Gambar 3.31 Halaman About Us

c. Tampilan Casting

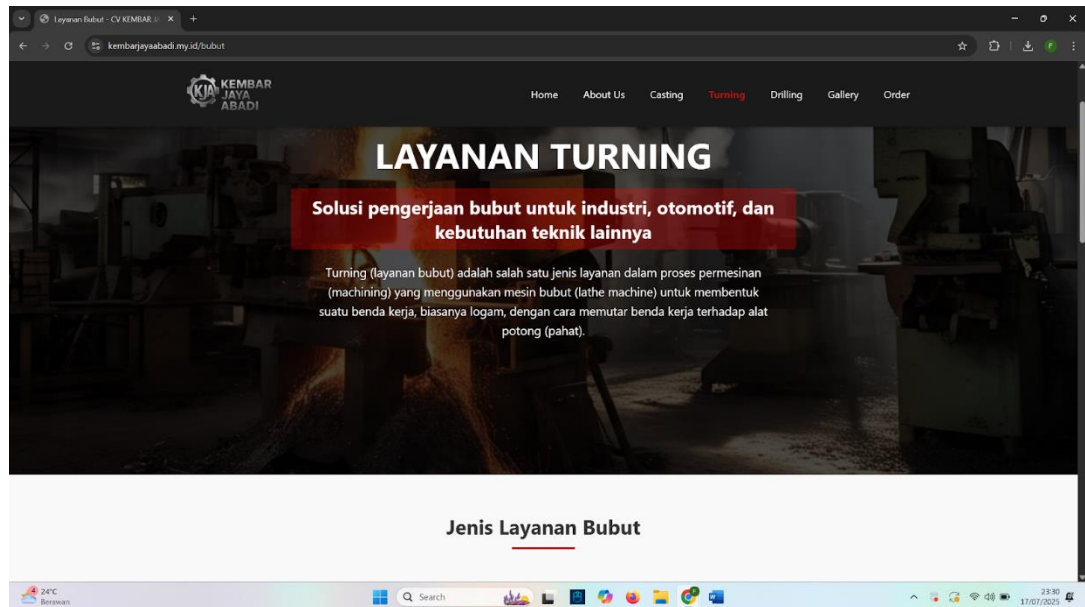
Gambar 3.32 merupakan halaman casting yang pada awal halaman ditampilkan latar visual aktivitas pengecoran logam beserta teks sorotan “Spesialis dalam cor aluminium, kuningan, baja, dan logam lainnya”. Hal ini mempertegas bahwa perusahaan memiliki fokus dalam produksi berbagai jenis logam untuk keperluan industri. Layanan pengecoran yang ditawarkan mencakup beberapa metode seperti Sand Casting (Pengecoran Pasir), Rough Finishing (Finishing Kasar), Custom Casting (Pengecoran Kustom), dan Repair Casting (Perbaikan Coran). Selanjutnya, terdapat informasi mengenai fasilitas pendukung yang digunakan untuk memastikan hasil produksi maksimal, yaitu tungku induksi, fasilitas machining, heat treatment, pengujian material. Selanjutnya terdapat informasi jenis-jenis material yang disediakan, antara lain FC & FCD, Steel, Mangan Steel, Ni Hard, High Chrome, Bronze & Kuningan, Aluminium, hingga Stainless Steel. Masing-masing material dijelaskan singkat karakteristik utamanya agar pengunjung dapat memahami keunggulan dan penggunaannya sesuai kebutuhan industri.



Gambar 3.32 Halaman Casting

d. Tampilan *Turning*

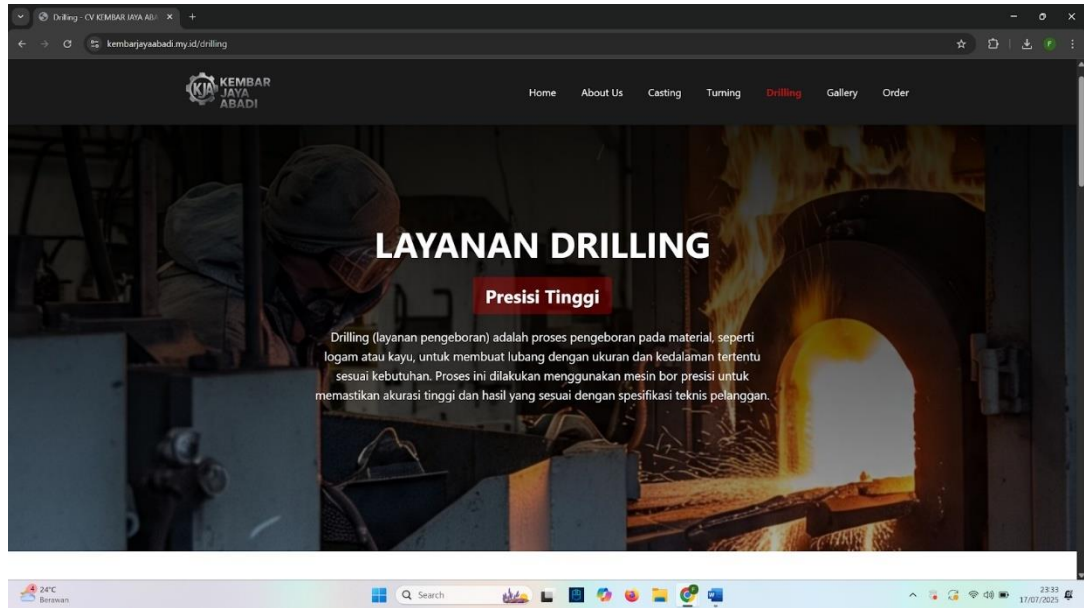
Gambar 3.33 merupakan halaman *turning* atau pembubutan yang ditujukan untuk kebutuhan industri, otomotif, dan teknik lainnya. Turning adalah proses pembentukan benda kerja, biasanya logam, dengan cara diputar menggunakan mesin bubut dan pahat. Di halaman ini juga dijelaskan beberapa jenis layanan bubut, seperti Conventional Turning untuk mengurangi diameter benda kerja, Facing untuk meratakan permukaan, Taper Turning untuk membuat bentuk kerucut, dan Grooving untuk membuat alur pada benda kerja. Semua jenis layanan ditampilkan dengan gambar dan penjelasan singkat agar mudah dipahami. Kemudian juga terdapat keunggulan dari layanan bubut CV Kembar Jaya Abadi pada bagian bawah halaman *turning*.



Gambar 3.33 Halaman Turning

e. Tampilan Drilling

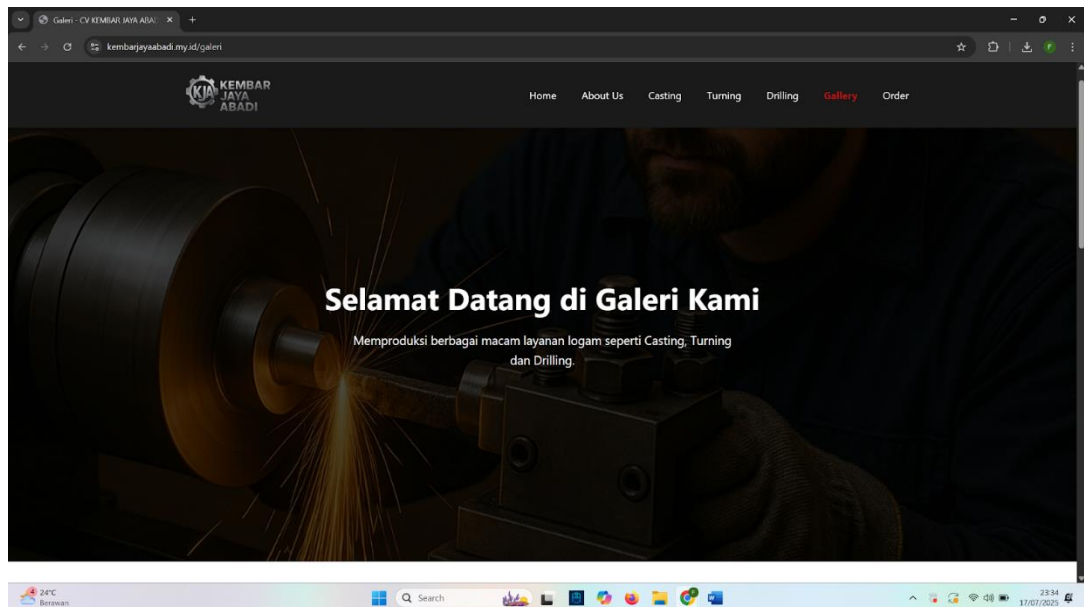
Gambar 3.34 merupakan halaman *drilling* atau pengeboran material seperti logam dan kayu dengan tingkat presisi tinggi. Di halaman ini terdapat informasi dari jenis-jenis drilling yang tersedia seperti tapping, reaming, counterboring, dan countersink. Lalu juga terdapat informasi tentang keunggulan dari layanan *drilling* seperti penggunaan mesin berteknologi tinggi, kontrol kualitas yang ketat, pengerjaan cepat, dan kemampuan mengerjakan berbagai jenis material. Halaman ini menunjukkan bahwa Kembar Jaya Abadi siap memberikan hasil pengeboran yang akurat dan berkualitas.



Gambar 3.34 Halaman Drilling

f. Tampilan Galeri

Gambar 3.35 merupakan halaman *gallery* yang berfungsi untuk menampilkan dokumentasi visual dari berbagai hasil pengerjaan layanan mereka, seperti casting, turning, dan drilling. Namun, saat ini halaman tersebut masih dalam tahap pengembangan, ditandai dengan label "Coming Soon" pada semua gambar produk seperti *Sand Casting*, *Komponen Roda*, *As Roda*, dan *Manifold Casting*. Meski belum tersedia konten visual, halaman ini sudah memiliki struktur dan navigasi yang jelas, termasuk filter kategori berdasarkan jenis layanan.



Gambar 3.35 Halaman Galeri

g. Tampilan Order

Gambar 3.36 merupakan halaman *order* yang berfungsi sebagai formulir pemesanan layanan. Pengguna diminta untuk mengisi data berupa email, nama, nomor telepon, dan alamat. Terdapat validasi panjang nomor telepon antara 11 hingga 13 digit untuk memastikan data yang dimasukkan benar. Halaman ini dirancang untuk mempermudah pelanggan dalam mengajukan permintaan layanan secara langsung dan efisien.

Gambar 3.36 Halaman Order

h. Config Database

Kode Program 3.1 berfungsi menghubungkan website dengan database agar data yang tersimpan dapat dikelola secara dinamis. Pada baris 3–24 terdapat bagian manajemen sesi yang berfungsi untuk mengatur durasi aktivitas pengguna. Jika pengguna tidak aktif selama 30 menit, maka sesi akan otomatis berakhir untuk menjaga keamanan akses, terutama pada halaman admin. Kode ini juga memastikan sesi baru dimulai dengan ID yang aman melalui `session_regenerate_id()`. Selanjutnya, pada baris 26–49, terdapat konfigurasi utama untuk koneksi ke database yang dibungkus dalam sebuah kelas bernama `Database`. Di dalam kelas ini, informasi seperti host, nama database, username, dan password dideklarasikan secara privat, kemudian fungsi `getConnection()` digunakan untuk membangun koneksi ke database MySQL menggunakan metode PDO. Koneksi ini juga dilengkapi dengan pengaturan mode error dan encoding UTF-8 untuk mendukung kompatibilitas data. Kemudian, pada baris 51–

64, terdapat fungsi validasi autentikasi admin. Fungsi `isLoggedIn()` memeriksa apakah sesi admin masih aktif, dan `requireLogin()` akan mengarahkan pengguna ke halaman login apabila belum melakukan autentikasi. Ini digunakan untuk membatasi akses hanya kepada pengguna yang memiliki hak admin. Pada baris 66–68, disediakan fungsi `sanitizeInput()` yang bertugas untuk membersihkan input pengguna dari potensi ancaman seperti script HTML atau JavaScript yang berbahaya (XSS). Terakhir, pada baris 70–94, terdapat fungsi `uploadImage()` yang digunakan untuk mengunggah file gambar. Fungsi ini memeriksa apakah direktori tujuan sudah tersedia, memvalidasi jenis dan ukuran file yang diizinkan, serta memberi nama file secara unik agar tidak tertimpa. Setelah semua pengecekan berhasil, file akan dipindahkan ke direktori penyimpanan. Fungsi ini sangat bermanfaat untuk fitur galeri, manajemen produk, atau profil perusahaan yang memerlukan unggahan gambar.

```

1. <?php
2.
3. if (session_status() == PHP_SESSION_NONE) {
4.     session_start();
5.
6.     $timeout_duration = 1800;
7.
8.     if (isset($_SESSION['last_activity']) && (time() -
        $_SESSION['last_activity']) > $timeout_duration) {
9.         session_unset();
10.        session_destroy();
11.        $protocol = isset($_SERVER['HTTPS']) &&
            $_SERVER['HTTPS'] === 'on' ? 'https' : 'http';
12.        $host = $_SERVER['HTTP_HOST'];
13.        $base_path = '/admin/';
14.        header('Location: ' . $protocol . '://' . $host .
            $base_path . 'login');
15.        exit();
16.    }
17.
18.    $_SESSION['last_activity'] = time();
19.
20.    if (!isset($_SESSION['initiated'])) {

```

```

21.         session_regenerate_id(true);
22.         $_SESSION['initiated'] = true;
23.     }
24. }
25.
26. class Database {
27.     private $host = 'localhost';
28.     private $db_name = 'veaephso_kembar_jaya_db';
29.     private $username = 'veaephso_kembarjayaabadi';
30.     private $password = 'kembarjayaabadi@12';
31.     public $conn;
32.
33.     public function getConnection() {
34.         $this->conn = null;
35.         try {
36.             $this->conn = new PDO(
37.                 "mysql:host=" . $this->host . ";dbname=" .
38.                 $this->db_name,
39.                 $this->username,
40.                 $this->password
41.             );
42.             $this->conn->setAttribute(PDO::ATTR_ERRMODE,
43.                 PDO::ERRMODE_EXCEPTION);
44.             $this->conn->exec("set names utf8");
45.         } catch(PDOException $exception) {
46.             error_log("Database connection error: " .
47.                 $exception->getMessage());
48.             die("Terjadi kesalahan koneksi database. Silakan
49.                 coba lagi nanti atau hubungi administrator.");
50.         }
51.         return $this->conn;
52.     }
53. }
54.
55. function isLoggedIn() {

```

```

52.     return isset($_SESSION['admin_id']) &&
        !empty($_SESSION['admin_id']) &&
        isset($_SESSION['last_activity']);
53. }
54.
55. function requireLogin() {
56.     if (!isLoggedIn()) {
57.         $protocol = isset($_SERVER['HTTPS']) &&
            $_SERVER['HTTPS'] === 'on' ? 'https' : 'http';
58.         $host = $_SERVER['HTTP_HOST'];
59.         $base_path = '/admin/';
60.
61.         header('Location: ' . $protocol . '://' . $host .
            $base_path . 'login');
62.         exit();
63.     }
64. }
65.
66. function sanitizeInput($data) {
67.     return htmlspecialchars(strip_tags(trim($data)));
68. }
69.
70. function uploadImage($file, $uploadDir = 'uploads/') {
71.     if (!file_exists($uploadDir)) {
72.         mkdir($uploadDir, 0755, true);
73.     }
74.
75.     $allowedTypes = ['image/jpeg', 'image/png', 'image/gif',
        'image/webp'];
76.     if (!in_array($file['type'], $allowedTypes)) {
77.         return ['success' => false, 'message' => 'Tipe file
            tidak diizinkan. Hanya JPEG, PNG, GIF, dan WebP.'];
78.     }
79.
80.     $maxSize = 5 * 1024 * 1024;

```

```

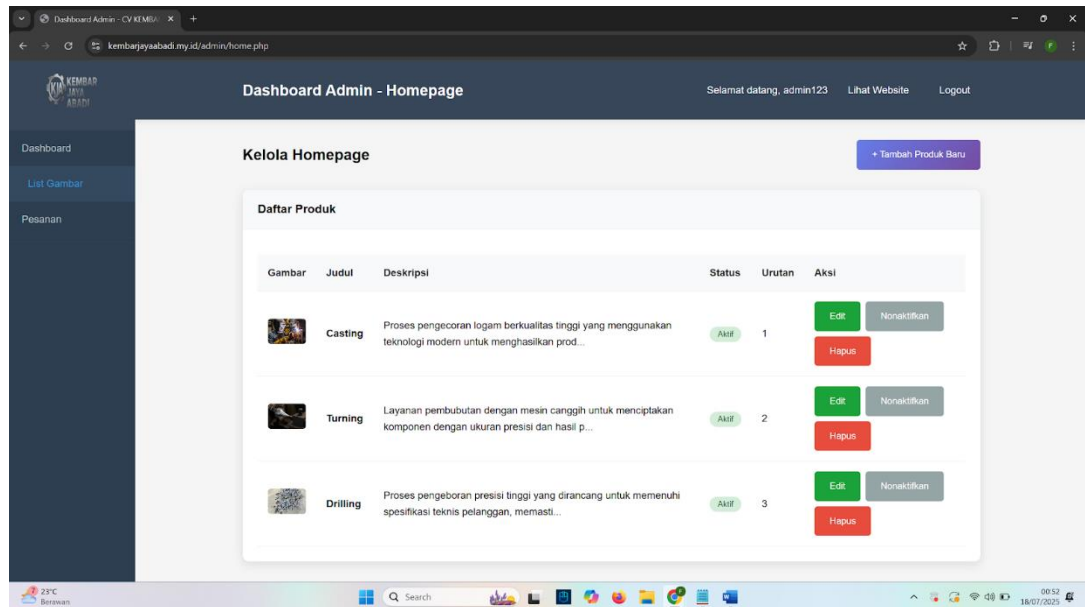
81.     if ($file['size'] > $maxSize) {
82.         return ['success' => false, 'message' => 'Ukuran file
            terlalu besar. Maksimal 5MB.'];
83.     }
84.
85.     $extension = pathinfo($file['name'], PATHINFO_EXTENSION);
86.     $filename = uniqid('img_', true) . '.' . $extension;
87.     $filepath = $uploadDir . $filename;
88.
89.     if (move_uploaded_file($file['tmp_name'], $filepath)) {
90.         return ['success' => true, 'filename' => $filename,
            'filepath' => $filepath];
91.     }
92.
93.     return ['success' => false, 'message' => 'Gagal mengunggah
            file.'];
94. }
95. ?>

```

Kode Program 3.1 Pengaturan Database

i. Tampilan Home Admin

Gambar 3.37 merupakan halaman *homepage* untuk admin dapat mengelola konten yang akan ditampilkan di halaman depan website, khususnya daftar layanan atau produk utama perusahaan seperti *Casting*, *Turning*, dan *Drilling*. Terdapat tabel berisi informasi mengenai masing-masing produk, termasuk gambar, judul, deskripsi singkat, status (aktif atau tidak), urutan tampil di halaman utama, serta tombol aksi untuk Edit, Nonaktifkan, atau Hapus. Fitur ini memudahkan admin untuk memperbarui konten secara dinamis tanpa perlu mengubah kode secara manual. Selain itu, tersedia tombol Tambah Produk Baru untuk menambahkan layanan baru ke homepage.



Gambar 3.37 Halaman Home Admin

Kode Program 3.2 berfungsi untuk admin mengelola konten yang ada pada halaman *home* di website pengguna. Pada baris 2-6 sistem memastikan bahwa sesi pengguna telah dimulai dan hanya admin yang sudah login yang bisa mengakses halaman ini. Kalau belum login, pengguna akan langsung diarahkan ke halaman login demi menjaga keamanan. Lalu pada baris 8–22, disediakan fitur hapus produk. Ketika admin menekan tombol hapus, sistem akan mengambil ID produk dari URL dan langsung menjalankan perintah untuk menghapus data tersebut dari database. Jika berhasil, halaman akan diperbarui dengan pesan bahwa produk sudah dihapus. Pada baris 24–38, ada fitur untuk mengubah status produk, misalnya dari aktif menjadi nonaktif, atau sebaliknya. Ini berguna kalau admin ingin menyembunyikan produk sementara

```

1. <?php
2. if (session_status() == PHP_SESSION_NONE) {
3.     session_start();
4. }
5. require_once '../config/database.php';
6. requireLogin();
7.
8. if (isset($_GET['delete'])) {
9.     $id = (int)$_GET['delete'];
10.    $database = new Database();
11.    $conn = $database->getConnection();
12.
13.    try {
14.        $query = "DELETE FROM products WHERE id = ?";
15.        $stmt = $conn->prepare($query);
16.        $stmt->execute([$id]);
17.        header('Location: home.php?success=deleted');

```



```

18.         exit();
19.     } catch(PDOException $e) {
20.         $error = 'Gagal menghapus produk';
21.     }
22. }
23.
24. if (isset($_GET['toggle'])) {
25.     $id = (int)$_GET['toggle'];
26.     $database = new Database();
27.     $conn = $database->getConnection();
28.
29.     try {
30.         $query = "UPDATE products SET status = CASE WHEN status
= 'active' THEN 'inactive' ELSE 'active' END WHERE id = ?";
31.         $stmt = $conn->prepare($query);
32.         $stmt->execute([$id]);
33.         header('Location: home.php?success=updated');
34.         exit();
35.     } catch(PDOException $e) {
36.         $error = 'Gagal mengubah status produk';
37.     }
38. }
39.
40. $database = new Database();
41. $conn = $database->getConnection();
42.
43. try {
44.     $query = "SELECT * FROM products ORDER BY sort_order ASC,
created_at DESC";
45.     $stmt = $conn->prepare($query);
46.     $stmt->execute();
47.     $products = $stmt->fetchAll(PDO::FETCH_ASSOC);
48. } catch(PDOException $e) {
49.     $products = [];
50.     $error = 'Gagal mengambil data produk';
51. }
52. ?>

```

Kode Program 3.2 Halaman Home Admin

Kode Program 3.3 merupakan kode program untuk mengedit layanan yang ada di bagian *home* website pengguna. Pada baris 2–3, sistem menghubungkan database dan memastikan pengguna telah login. Validasi id produk dilakukan di baris 8–11, lalu data produk diambil dari database (baris 17–20). Jika produk tidak ditemukan, pengguna akan diarahkan kembali ke *home.php* (baris 22–25). Ketika form dikirim (baris 31), data input seperti title, description, dan lainnya dibersihkan untuk mencegah input berbahaya (baris 32–36) dan data gambar lama disiapkan (baris 37). Jika ada file gambar baru yang valid (baris 42–51), maka gambar

tersebut diupload dan gambar lama dihapus. Jika tidak, URL gambar bisa diganti secara manual (baris 52–54). Jika tidak ada error, data produk diperbarui di database (baris 56–59) dan pengguna diarahkan kembali ke halaman utama (baris 61–62).

```

1. <?php
2. require_once '../config/database.php';
3. requireLogin();
4.
5. $error = '';
6. $product = null;
7.
8. if (!isset($_GET['id']) || !is_numeric($_GET['id'])) {
9.     header('Location: home.php?error=invalid_id');
10.    exit();
11. }
12.
13. $product_id = (int)$_GET['id'];
14. $database = new Database();
15. $conn = $database->getConnection();
16.
17. try {
18.     $stmt = $conn->prepare("SELECT * FROM products WHERE id =
        ?");
19.     $stmt->execute([$product_id]);
20.     $product = $stmt->fetch(PDO::FETCH_ASSOC);
21.
22.     if (!$product) {
23.         header('Location: home.php?error=product_not_found');
24.         exit();
25.     }
26. } catch (PDOException $e) {
27.     header('Location: home.php?error=database_error');
28.     exit();
29. }
30.

```

```

31. if ($_SERVER['REQUEST_METHOD'] === 'POST') {
32.     $title = sanitizeInput($_POST['title']);
33.     $description = sanitizeInput($_POST['description']);
34.     $page_url = sanitizeInput($_POST['page_url']);
35.     $sort_order = (int)$_POST['sort_order'];
36.     $status = $_POST['status'];
37.     $image_url = $product['image_url'];
38.
39.     if (empty($title) || empty($description) ||
        empty($page_url)) {
40.         $error = 'Harap isi semua field yang wajib';
41.     } else {
42.         if (isset($_FILES['image']) && $_FILES['image']['error']
            === UPLOAD_ERR_OK) {
43.             $upload_result = uploadImage($_FILES['image'],
                '../uploads/');
44.             if ($upload_result['success']) {
45.                 if (!filter_var($product['image_url'],
                    FILTER_VALIDATE_URL) && file_exists('../' .
                    $product['image_url'])) {
46.                     unlink('../' . $product['image_url']);
47.                 }
48.                 $image_url = 'uploads/' .
                    $upload_result['filename'];
49.             } else {
50.                 $error = 'Gagal upload gambar: ' .
                    $upload_result['message'];
51.             }
52.             } elseif (!empty($_POST['image_url_manual']) &&
                $_POST['image_url_manual'] !== $product['image_url']) {
53.                 $image_url =
                    sanitizeInput($_POST['image_url_manual']);
54.             }
55.
56.             if (empty($error)) {

```

```

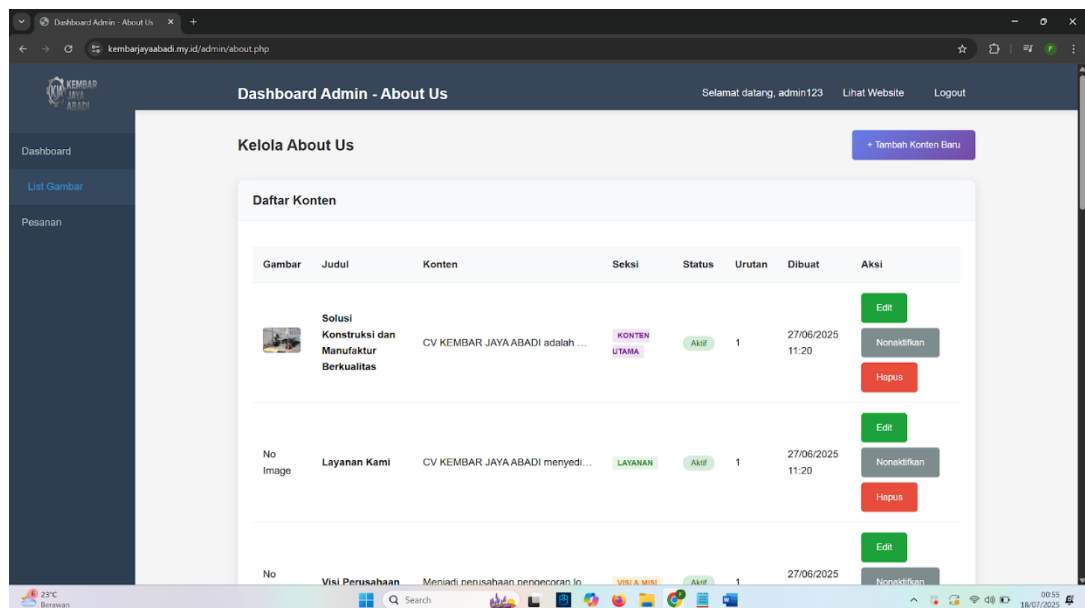
57.         try {
58.             $stmt = $conn->prepare("UPDATE products SET
            title = ?, description = ?, image_url = ?, page_url = ?, status
            = ?, sort_order = ?, updated_at = NOW() WHERE id = ?");
59.             $stmt->execute([$title, $description,
            $image_url, $page_url, $status, $sort_order, $product_id]);
60.
61.             header('Location: home.php?success=updated');
62.             exit();
63.         } catch (PDOException $e) {
64.             $error = 'Gagal mengupdate produk: ' . $e-
            >getMessage();
65.         }
66.     }
67. }
68. }
69. ?>

```

Kode Program 3.3 Halaman Edit Konten Home Admin

j. Tampilan About Us Admin

Gambar 3.38 merupakan tampilan halaman *about us* untuk admin agar dapat mengelola konten yang ada pada website pengguna.



Gambar 3.38 Halaman About Us Admin

Kode Program 3.4 merupakan kode program untuk halaman *about us* admin. Pada baris 2–4, dilakukan inisialisasi sesi dan pengecekan login admin. Baris 6–7 membuat koneksi database. Baris 10–19 menangani penghapusan konten berdasarkan parameter delete, dan baris 22–31 menangani toggle status aktif/non aktif menggunakan SQL CASE WHEN. Data konten diambil pada baris 34–41 menggunakan query SELECT * yang diurutkan berdasarkan section_order dan created_at. Baris 54–69 menampilkan pesan sukses atau error setelah aksi dilakukan. Tabel konten ditampilkan di baris 73–105, dengan tiap baris menampilkan judul, isi, jenis seksi, status, urutan, waktu dibuat, serta tombol aksi (edit, toggle, hapus).

```

1. <?php
2. if (session_status() == PHP_SESSION_NONE) session_start();
3. require_once '../config/database.php';
4. requireLogin();
5.
6. $database = new Database();
7. $conn = $database->getConnection();
8.
9. // Hapus konten
10. if (isset($_GET['delete'])) {
11.     try {
12.         $stmt = $conn->prepare("DELETE FROM about_content WHERE
            id = ?");
13.         $stmt->execute([(int)$_GET['delete']]);
14.         header('Location: about.php?success=deleted');
15.         exit();
16.     } catch (PDOException $e) {
17.         $error = 'Gagal menghapus konten';
18.     }
19. }
20.
21. // Toggle status konten
22. if (isset($_GET['toggle'])) {
23.     try {
24.         $stmt = $conn->prepare("UPDATE about_content SET status
            = CASE WHEN status = 'active' THEN 'inactive' ELSE 'active' END
            WHERE id = ?");
25.         $stmt->execute([(int)$_GET['toggle']]);
26.         header('Location: about.php?success=updated');
27.         exit();
28.     } catch (PDOException $e) {
29.         $error = 'Gagal mengubah status konten';
30.     }
31. }
32.
33. // Ambil semua konten
34. try {

```

```

35.     $stmt = $conn->prepare("SELECT * FROM about_content ORDER BY
    section_order ASC, created_at DESC");
36.     $stmt->execute();
37.     $contents = $stmt->fetchAll(PDO::FETCH_ASSOC);
38. } catch (PDOException $e) {
39.     $contents = [];
40.     $error = 'Gagal mengambil data konten';
41. }
42. ?>
43.
44. <!DOCTYPE html>
45. <html lang="id">
46. <head>
47.     <meta charset="UTF-8">
48.     <title>Admin - About Us</title>
49.     <link rel="stylesheet" href="drilling.css">
50. </head>
51. <body>
52.     <h1>Kelola About Us</h1>
53.
54.     <?php if (isset($_GET['success'])): ?>
55.         <p style="color:green;">
56.             <?php
57.                 switch($_GET['success']) {
58.                     case 'added': echo 'Konten berhasil
    ditambahkan!'; break;
59.                     case 'updated': echo 'Konten berhasil
    diupdate!'; break;
60.                     case 'deleted': echo 'Konten berhasil
    dihapus!'; break;
61.                     default: echo 'Operasi berhasil!';
62.                 }
63.             ?>
64.         </p>
65.     <?php endif; ?>
66.
67.     <?php if (isset($error)): ?>
68.         <p style="color:red;"><?php echo $error; ?></p>
69.     <?php endif; ?>
70.
71.     <a href="add-about.php">+ Tambah Konten Baru</a>
72.
73.     <?php if (!empty($contents)): ?>
74.         <table border="1" cellpadding="8" cellspacing="0">
75.             <thead>
76.                 <tr>
77.                     <th>Judul</th>
78.                     <th>Konten</th>
79.                     <th>Seksi</th>
80.                     <th>Status</th>

```

```

81.             <th>Urutan</th>
82.             <th>Dibuat</th>
83.             <th>Aksi</th>
84.         </tr>
85.     </thead>
86.     <tbody>
87.         <?php foreach ($contents as $c): ?>
88.             <tr>
89.                 <td><strong><?php echo
htmlspecialchars($c['title']); ?></strong></td>
90.                 <td><?php echo
htmlspecialchars(mb_strimwidth($c['content'], 0, 100, '...'));
?></td>
91.                 <td><?php echo
htmlspecialchars($c['section_type']); ?></td>
92.                 <td><?php echo $c['status'] === 'active' ?
'Aktif' : 'Nonaktif'; ?></td>
93.                 <td><?php echo $c['section_order']; ?></td>
94.                 <td><?php echo $c['created_at'] ?
date('d/m/Y H:i', strtotime($c['created_at'])) : '-'; ?></td>
95.                 <td>
96.                     <a href="edit-about.php?id=<?php echo
$c['id']; ?>">Edit</a> |
97.                     <a href="?toggle=<?php echo $c['id'];
?>" onclick="return confirm('Ubah status konten?')">
98.                         <?php echo $c['status'] === 'active'
? 'Nonaktifkan' : 'Aktifkan'; ?>
99.                     </a> |
100.                    <a href="?delete=<?php echo $c['id'];
?>" onclick="return confirm('Yakin ingin menghapus?')">Hapus</a>
101.                </td>
102.            </tr>
103.        <?php endforeach; ?>
104.    </tbody>
105. </table>
106. <?php else: ?>
107.     <p>Belum ada konten. <a href="add-about.php">Tambah
konten pertama</a>.</p>
108. <?php endif; ?>
109. </body>
110. </html>

```

Kode Program 3.4 Halaman About Us Admin

Kode Program 3.5 merupakan kode program untuk menangani edit konten untuk halaman *about us* admin. Pada baris 2–5, kode dimulai dengan session, memastikan user login, dan membuat koneksi ke database. Baris 7–8 memeriksa apakah id konten tersedia di URL, jika tidak, diarahkan ke about.php. Lalu di baris 10–12, data konten dengan ID tersebut diambil dari tabel about_content. Jika tidak ditemukan, kembali diarahkan (baris 13). Jika form dikirim (baris 15), maka konten akan diperbarui di database lewat query UPDATE (baris 16–25)

menggunakan data dari form (title, content, section_type, dll). Setelah berhasil update, pengguna diarahkan kembali ke halaman utama dengan notifikasi sukses (baris 26–27).

```

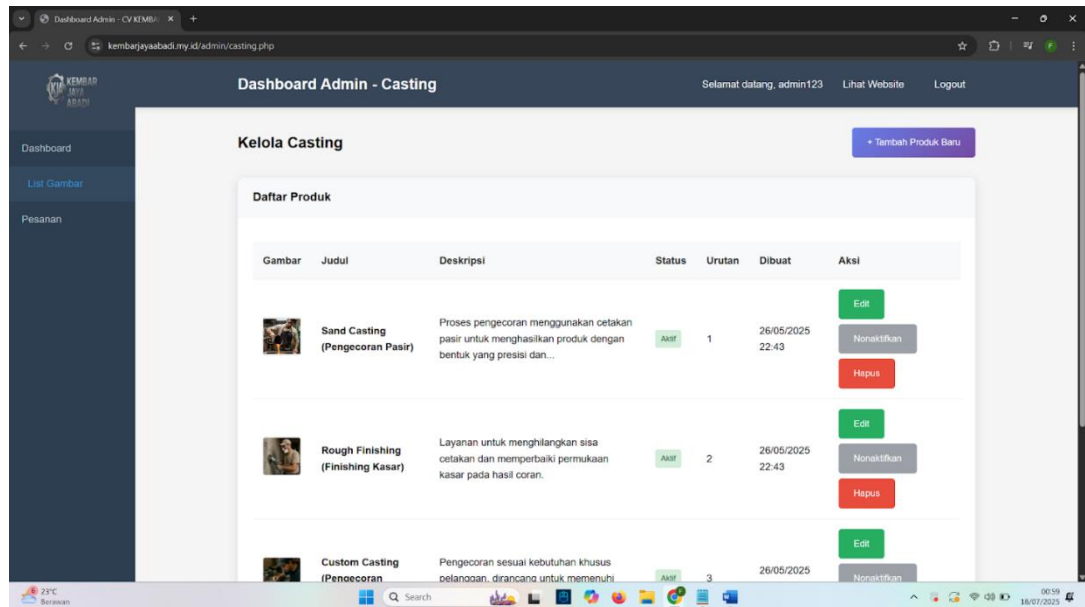
1. <?php
2. if (session_status() == PHP_SESSION_NONE) session_start();
3. require_once '../config/database.php';
4. requireLogin();
5. $conn = (new Database())->getConnection();
6.
7. if (!isset($_GET['id'])) header('Location: about.php');
8. $id = (int)$_GET['id'];
9.
10. $stmt = $conn->prepare("SELECT * FROM about_content WHERE id = ?");
11. $stmt->execute([$id]);
12. $content = $stmt->fetch(PDO::FETCH_ASSOC);
13. if (!$content) header('Location: about.php');
14.
15. if ($_SERVER['REQUEST_METHOD'] === 'POST') {
16.     $stmt = $conn->prepare("UPDATE about_content SET title=?,
        content=?, section_type=?, section_order=?, status=?,
        image_url=? WHERE id=?");
17.     $stmt->execute([
18.         $_POST['title'],
19.         $_POST['content'],
20.         $_POST['section_type'],
21.         $_POST['section_order'],
22.         $_POST['status'],
23.         $_POST['image_url'] ?? '',
24.         $id
25.     ]);
26.     header("Location: about.php?success=updated");
27.     exit();
28. }
29. ?>

```

Kode Program 3.5 Halaman Edit About Us Admin

k. Tampilan Casting Admin

Gambar 3.39 merupakan tampilan halaman *casting* untuk admin dapat mengelola konten *casting* yang ada di website pengguna. Disini admin bisa menambah, mengedit informasi, menonaktif atau mengaktifkan konten yang sudah ada, dan menghapus konten tentang *casting*.



Gambar 3.39 Halaman Casting Admin

Kode program 3.6 merupakan kode program untuk halaman *casting* admin. Pada baris 3–5, dilakukan inisialisasi *session* dan pengecekan login menggunakan `requireLogin()`, serta koneksi database dibuat pada baris 7–8. Kode baris 11–18 menangani aksi hapus produk. Jika parameter `delete` ada di URL, maka data dengan ID tersebut dihapus dari tabel `casting_services`. Setelah sukses, diarahkan kembali ke halaman utama dengan notifikasi. Kemudian, baris 21–28 menangani aksi *toggle status* aktif/nonaktif. Jika parameter `toggle` terdeteksi, status produk diubah antara *'active'* dan *'inactive'*. Pada baris 31–38, data semua produk diambil dari database dan diurutkan berdasarkan `sort_order` dan `created_at`. Bagian HTML dimulai dari baris 40. Pada baris 49–55, ditampilkan notifikasi jika ada pesan sukses atau error. Tombol untuk menambah produk ada di baris 57, lalu tabel data ditampilkan di baris 60–80. Setiap baris berisi gambar produk, judul, status, dan tombol aksi: Edit, Toggle, dan Hapus (baris 73–75).

```

1. <?php
2. // admin/casting.php (ringkas)
3. if (session_status() == PHP_SESSION_NONE) session_start();
4. require_once '../config/database.php';
5. requireLogin();
6.
7. $database = new Database();
8. $conn = $database->getConnection();
9.
10. // Hapus produk
11. if (isset($_GET['delete'])) {
12.     $id = (int)$_GET['delete'];

```

```

13.     try {
14.         $stmt = $conn->prepare("DELETE FROM casting_services
WHERE id = ?");
15.         $stmt->execute([$id]);
16.         header('Location: casting.php?success=deleted'); exit();
17.     } catch(PDOException $e) { $error = 'Gagal menghapus
produk'; }
18. }
19.
20. // Toggle status aktif/nonaktif
21. if (isset($_GET['toggle'])) {
22.     $id = (int)$_GET['toggle'];
23.     try {
24.         $stmt = $conn->prepare("UPDATE casting_services SET
status = CASE WHEN status = 'active' THEN 'inactive' ELSE
'active' END WHERE id = ?");
25.         $stmt->execute([$id]);
26.         header('Location: casting.php?success=updated'); exit();
27.     } catch(PDOException $e) { $error = 'Gagal mengubah status
produk'; }
28. }
29.
30. // Ambil data produk
31. try {
32.     $stmt = $conn->prepare("SELECT * FROM casting_services ORDER
BY sort_order ASC, created_at DESC");
33.     $stmt->execute();
34.     $products = $stmt->fetchAll(PDO::FETCH_ASSOC);
35. } catch(PDOException $e) {
36.     $products = [];
37.     $error = 'Gagal mengambil data produk';
38. }
39. ?>
40. <!DOCTYPE html>
41. <html lang="id">
42. <head>
43.     <title>Admin Casting</title>
44.     <link rel="stylesheet" href="casting.css" />
45. </head>
46. <body>
47.     <h1>Dashboard Admin - Casting</h1>
48.
49.     <?php if (isset($_GET['success'])): ?>
50.         <div><?php echo htmlspecialchars($_GET['success']);
?></div>
51.     <?php endif; ?>
52.
53.     <?php if (isset($error)): ?>
54.         <div><?php echo $error; ?></div>
55.     <?php endif; ?>

```

```

56.
57.     <a href="add-casting.php">+ Tambah Produk</a>
58.
59.     <?php if (!empty($products)): ?>
60.         <table>
61.             <thead>
62.                 <tr>
63.                     <th>Gambar</th><th>Judul</th><th>Status</th>
        <th>Aksi</th>
64.                 </tr>
65.             </thead>
66.             <tbody>
67.                 <?php foreach ($products as $p): ?>
68.                     <tr>
69.                         <td></td>
70.                         <td><?php echo
        htmlspecialchars($p['title']); ?></td>
71.                         <td><?php echo $p['status'] === 'active' ?
        'Aktif' : 'Nonaktif'; ?></td>
72.                         <td>
73.                             <a href="edit-casting.php?id=<?php echo
        $p['id']; ?>">Edit</a>
74.                             <a href="?toggle=<?php echo $p['id'];
        ?>">Toggle</a>
75.                             <a href="?delete=<?php echo $p['id'];
        ?>" onclick="return confirm('Hapus produk ini?')">Hapus</a>
76.                         </td>
77.                     </tr>
78.                 <?php endforeach; ?>
79.             </tbody>
80.         </table>
81.     <?php else: ?>
82.         <p>Belum ada produk. <a href="add-casting.php">Tambah
        sekarang</a>.</p>
83.     <?php endif; ?>
84. </body>
85. </html>

```

Kode Program 3.6 Halaman Casting Admin

Kode Program 3.7 merupakan kode program untuk menambahkan layanan *casting*. Kode ini digunakan admin menambahkan produk layanan casting baru melalui form (baris 53–77). Saat form dikirim (POST), sistem akan memproses data input berupa judul, deskripsi, urutan tampil, status, dan gambar (baris 6–10). Jika judul atau deskripsi kosong, maka muncul peringatan error (baris 13–14). Jika data valid, koneksi ke database dibuka (baris 16–17). Gambar bisa diunggah (baris 19–25) atau diisi secara manual lewat URL (baris 26–28). Jika gambar berhasil didapat, data layanan disimpan ke database (baris 30–33). Setelah berhasil,

admin langsung diarahkan ke halaman utama dengan pesan sukses (baris 34–35). Namun, jika terjadi kesalahan saat menyimpan atau gambar tidak tersedia, maka akan muncul pesan error (baris 24, 36–40). Seluruh logika diproses di awal (baris 2–43), sedangkan bagian bawah adalah tampilan HTML untuk form input (baris 46–79).

```

1. <?php
2. require_once '../config/database.php';
3. requireLogin();
4.
5. $error = '';
6. if ($_SERVER['REQUEST_METHOD'] === 'POST') {
7.     $title = sanitizeInput($_POST['title']);
8.     $description = sanitizeInput($_POST['description']);
9.     $sort_order = (int)$_POST['sort_order'];
10.    $status = $_POST['status'];
11.    $image_url = '';
12.
13.    if (empty($title) || empty($description)) {
14.        $error = 'Harap isi semua field yang wajib';
15.    } else {
16.        $database = new Database();
17.        $conn = $database->getConnection();
18.
19.        if (isset($_FILES['image']) && $_FILES['image']['error']
20.            === UPLOAD_ERR_OK) {
21.            $upload = uploadImage($_FILES['image'],
22.                '../uploads/');
23.            if ($upload['success']) {
24.                $image_url = 'uploads/' . $upload['filename'];
25.            } else {
26.                $error = 'Upload gagal: ' . $upload['message'];
27.            }
28.        } else {
29.            $image_url =
30.                sanitizeInput($_POST['image_url_manual']);
31.        }
32.
33.        if (empty($error) && !empty($image_url)) {
34.            try {
35.                $stmt = $conn->prepare("INSERT INTO
36.                    casting_services (title, description, image_url, status,
37.                    sort_order) VALUES (?, ?, ?, ?, ?)");
38.                $stmt->execute([$title, $description,
39.                    $image_url, $status, $sort_order]);
40.                header('Location: casting.php?success=added');
41.                exit;
42.            } catch (PDOException $e) {
43.                $error = 'Gagal menyimpan: ' . $e->getMessage();
44.            }
45.        }
46.    }
47.}
48.
49.// Tampilan HTML form input (baris 46–79)

```

```

39.         } elseif (empty($image_url)) {
40.             $error = 'Upload atau URL gambar wajib diisi';
41.         }
42.     }
43. }
44. ?>
45.
46. <!DOCTYPE html>
47. <html>
48. <head><title>Tambah Produk</title></head>
49. <body>
50.     <h2>Tambah Produk</h2>
51.     <?php if ($error): ?><p style="color:red;"><?php echo
        $error; ?></p><?php endif; ?>
52.
53.     <form method="POST" enctype="multipart/form-data">
54.         <label>Judul Produk *</label><br>
55.         <input type="text" name="title" required><br><br>
56.
57.         <label>Deskripsi *</label><br>
58.         <textarea name="description"
        required></textarea><br><br>
59.
60.         <label>Upload Gambar</label><br>
61.         <input type="file" name="image"
        accept="image/*"><br><br>
62.
63.         <label>atau Masukkan URL Gambar</label><br>
64.         <input type="url" name="image_url_manual"><br><br>
65.
66.         <label>Urutan Tampil</label><br>
67.         <input type="number" name="sort_order"
        value="0"><br><br>
68.
69.         <label>Status</label><br>
70.         <select name="status">
71.             <option value="active">Aktif</option>
72.             <option value="inactive">Nonaktif</option>
73.         </select><br><br>
74.
75.         <button type="submit">Simpan</button>
76.         <a href="casting.php">Batal</a>
77.     </form>
78. </body>
79. </html>

```

Kode Program 3.7 Halaman Tambah Konten Casting Admin

Kode Program 3.8 merupakan kode program untuk mengedit konten *casting*. Pada baris 2–3, dilakukan pemanggilan file konfigurasi database dan fungsi `requireLogin()` untuk membatasi

akses hanya kepada pengguna yang sudah login. Baris 6–17 menangani pengambilan ID dari parameter URL, lalu mengambil data layanan berdasarkan ID tersebut dari database. Jika ID tidak valid atau data tidak ditemukan, pengguna akan diarahkan kembali ke halaman sebelumnya. Baris 19–56 memproses form saat di submit, termasuk validasi input, pengunggahan atau penghapusan gambar (baris 29–44), serta update data ke database (baris 48–49). Sementara itu, bagian HTML pada baris 59–118 menampilkan form yang sudah terisi data lama, memungkinkan pengguna mengganti teks atau gambar, dan menyediakan tombol hapus gambar (baris 80–82). Fungsi JavaScript pada baris 105–116 memungkinkan penghapusan gambar secara interaktif.

```

1. <?php
2. require_once '../config/database.php';
3. requireLogin();
4.
5. $error = '';
6. $id = isset($_GET['id']) ? (int)$_GET['id'] : 0;
7. if ($id <= 0) { header('Location:
   casting.php?error=invalid_id'); exit(); }
8.
9. $conn = (new Database())->getConnection();
10. try {
11.     $stmt = $conn->prepare("SELECT * FROM casting_services WHERE
   id = ?");
12.     $stmt->execute([$id]);
13.     $product = $stmt->fetch(PDO::FETCH_ASSOC);
14.     if (!$product) { header('Location:
   casting.php?error=not_found'); exit(); }
15. } catch (PDOException $e) {
16.     header('Location: casting.php?error=database_error');
   exit();
17. }
18.
19. if ($_SERVER['REQUEST_METHOD'] === 'POST') {
20.     $title = sanitizeInput($_POST['title']);
21.     $description = sanitizeInput($_POST['description']);
22.     $sort_order = (int)$_POST['sort_order'];
23.     $status = $_POST['status'];
24.     $image_url = $product['image_url'];
25.
26.     if (empty($title) || empty($description)) {
27.         $error = 'Harap isi semua field yang wajib';
28.     } else {
29.         if (isset($_FILES['image']) && $_FILES['image']['error']
   === UPLOAD_ERR_OK) {
30.             $upload = uploadImage($_FILES['image'],
   '../uploads/');
31.             if ($upload['success']) {

```

```

32.         if (!empty($product['image_url']) &&
!filter_var($product['image_url'], FILTER_VALIDATE_URL)) {
33.             @unlink('../' . $product['image_url']);
34.         }
35.         $image_url = 'uploads/' . $upload['filename'];
36.     } else {
37.         $error = 'Gagal upload gambar: ' .
$upload['message'];
38.     }
39.     } elseif (!empty($_POST['image_url_manual'])) {
40.         $image_url =
sanitizeInput($_POST['image_url_manual']);
41.     } elseif (!empty($_POST['remove_current_image'])) {
42.         @unlink('../' . $product['image_url']);
43.         $image_url = '';
44.     }
45.
46.     if (empty($error)) {
47.         try {
48.             $stmt = $conn->prepare("UPDATE casting_services
SET title=?, description=?, image_url=?, status=?, sort_order=?,
updated_at=NOW() WHERE id=?");
49.             $stmt->execute([$title, $description,
$image_url, $status, $sort_order, $id]);
50.             header('Location: casting.php?success=updated');
exit();
51.         } catch (PDOException $e) {
52.             $error = 'Gagal mengupdate layanan: ' . $e-
>getMessage();
53.         }
54.     }
55. }
56. }
57. ?>
58.
59. <!DOCTYPE html>
60. <html lang="id">
61. <head>
62.     <meta charset="UTF-8">
63.     <title>Edit Layanan Casting</title>
64. </head>
65. <body>
66.     <h2>Edit Layanan</h2>
67.
68.     <?php if ($error): ?><p style="color:red"><?= $error
?></p><?php endif; ?>
69.
70.     <form method="POST" enctype="multipart/form-data">
71.         <label>Judul *<br>
72.         <input type="text" name="title" required value="<?=
htmlspecialchars($_POST['title']) ?? $product['title']">?>">

```

```

73.         </label><br><br>
74.
75.         <label>Deskripsi *<br>
76.             <textarea name="description" required><?=
htmlspecialchars($_POST['description']) ??
$product['description']) ?></textarea>
77.         </label><br><br>
78.
79.         <label>Gambar Produk<br>
80.             <?php if (!empty($product['image_url'])): ?>
81.                 <br>
82.                 <button type="button"
onclick="removeImage()">Hapus Gambar Saat Ini</button>
83.             <?php endif; ?>
84.             <input type="file" name="image"><br>
85.             <input type="url" name="image_url_manual"
placeholder="Atau masukkan URL gambar"><br>
86.         </label><br>
87.
88.         <label>Urutan Tampil<br>
89.             <input type="number" name="sort_order" value="<?=
$_POST['sort_order'] ?? $product['sort_order'] ?>">
90.         </label><br><br>
91.
92.         <label>Status<br>
93.             <select name="status">
94.                 <option value="active" <?= ($product['status']
=== 'active') ? 'selected' : '' ?>>Aktif</option>
95.                 <option value="inactive" <?= ($product['status']
=== 'inactive') ? 'selected' : '' ?>>Nonaktif</option>
96.             </select>
97.         </label><br><br>
98.
99.         <input type="hidden" name="current_image" value="<?=
htmlspecialchars($product['image_url']) ?>">
100.
101.         <button type="submit">Simpan</button>
102.         <a href="casting.php">Batal</a>
103.     </form>
104.
105.     <script>
106.         function removeImage() {
107.             if (confirm('Yakin ingin menghapus gambar saat
ini?')) {
108.                 const input = document.createElement('input');
109.                 input.type = 'hidden';
110.                 input.name = 'remove_current_image';
111.                 input.value = '1';
112.                 document.forms[0].appendChild(input);
113.                 document.forms[0].submit();

```



```

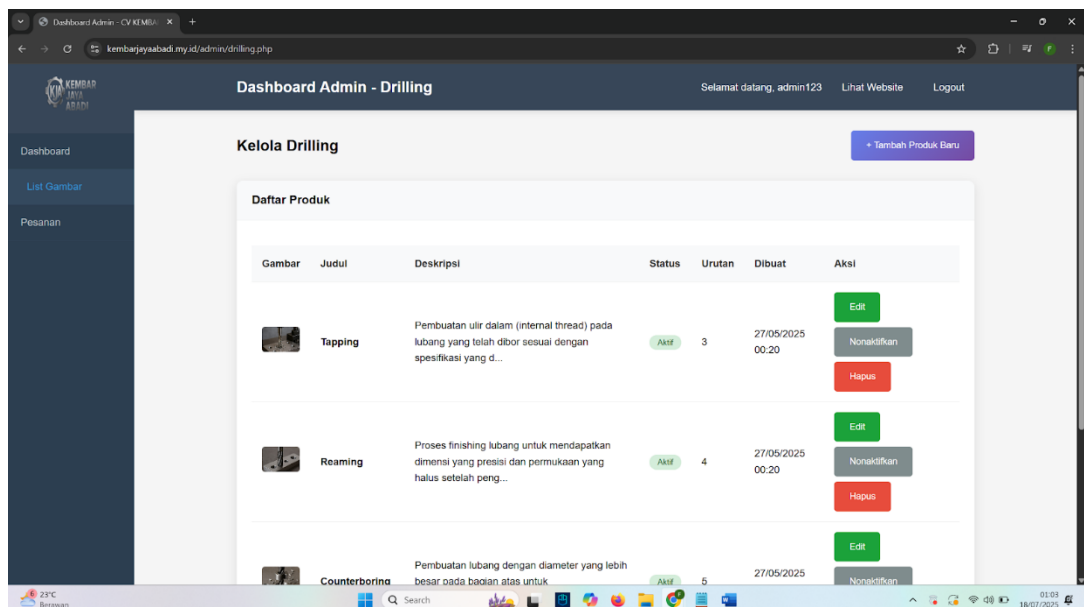
114.         }
115.     }
116.     </script>
117. </body>
118. </html>

```

Kode Program 3.8 Halaman Edit Konten Casting Admin

I. Tampilan Drilling Admin

Gambar 3.40 merupakan tampilan halaman drilling admin. Di halaman ini, admin dapat mengelola konten drilling yang ada di website pengguna. Disini admin bisa menambah, mengedit informasi, menonaktif atau mengaktifkan konten yang sudah ada, dan menghapus konten tentang *drilling*.



Gambar 3.40 Halaman Drilling Admin

Kode program 3.9 merupakan kode program untuk halaman *drilling* admin. Pada baris 3–5, dilakukan inisialisasi sesi, koneksi database, dan pengecekan login dengan `requireLogin()` untuk membatasi akses. Fitur penghapusan produk terdapat pada baris 8–19, di mana jika parameter `delete` terdeteksi di URL, maka data produk akan dihapus dari tabel `services` berdasarkan ID-nya. Sedangkan baris 22–33 menangani logika untuk mengubah status produk antara `active` dan `inactive` melalui parameter `toggle`. Kemudian, pada baris 36–43, semua data produk diambil dari database dan diurutkan berdasarkan `sort_order` dan `created_at`, lalu disimpan ke dalam array `$products`. Tampilan HTML-nya dimulai dari baris 47, di mana jika ada parameter `success` di URL, akan ditampilkan notifikasi keberhasilan (baris 48–50). Selanjutnya, baris 54–76 menampilkan daftar produk dalam bentuk tabel, termasuk kolom

gambar, judul, deskripsi singkat (maks. 100 karakter), status, dan aksi (edit, toggle, hapus). Tautan hapus dilengkapi konfirmasi JavaScript pada baris 71. Jika tidak ada data, maka ditampilkan pesan kosong pada baris 78.

```

1. <?php
2. // admin/drilling.php
3. session_start();
4. require_once '../config/database.php';
5. requireLogin();
6.
7. // Hapus produk
8. if (isset($_GET['delete'])) {
9.     $id = (int)$_GET['delete'];
10.    try {
11.        (new Database())->getConnection()
12.            ->prepare("DELETE FROM services WHERE id = ?")
13.            ->execute([$id]);
14.        header('Location: drilling.php?success=deleted');
15.        exit();
16.    } catch(PDOException $e) {
17.        $error = 'Gagal menghapus produk';
18.    }
19. }
20.
21. // Toggle status produk
22. if (isset($_GET['toggle'])) {
23.     $id = (int)$_GET['toggle'];
24.    try {
25.        (new Database())->getConnection()
26.            ->prepare("UPDATE services SET status = CASE WHEN
status = 'active' THEN 'inactive' ELSE 'active' END WHERE id =
?")
27.            ->execute([$id]);
28.        header('Location: drilling.php?success=updated');
29.        exit();
30.    } catch(PDOException $e) {
31.        $error = 'Gagal mengubah status produk';
32.    }
33. }
34.
35. // Ambil semua data produk
36. try {
37.     $products = (new Database())->getConnection()
38.         ->query("SELECT * FROM services ORDER BY sort_order ASC,
created_at DESC")
39.         ->fetchAll(PDO::FETCH_ASSOC);
40. } catch(PDOException $e) {
41.     $products = [];
42.     $error = 'Gagal mengambil data produk';

```

```

43. }
44. ?>
45.
46. <!-- HTML TAMPILAN UTAMA -->
47. <h1>Dashboard Admin - Drilling</h1>
48. <?php if (isset($_GET['success'])): ?>
49.     <div><?php echo htmlspecialchars($_GET['success']); ?></div>
50. <?php endif; ?>
51.
52. <a href="add-drilling.php">+ Tambah Produk Baru</a>
53.
54. <?php if (!empty($products)): ?>
55. <table>
56.     <thead>
57.         <tr>
58.             <th>Gambar</th><th>Judul</th><th>Deskripsi</th><th>S
tatus</th><th>Aksi</th>
59.         </tr>
60.     </thead>
61.     <tbody>
62.         <?php foreach ($products as $p): ?>
63.         <tr>
64.             <td></td>
65.             <td><?php echo htmlspecialchars($p['title']);
?></td>
66.             <td><?php echo
substr(htmlspecialchars($p['description']), 0, 100); ?></td>
67.             <td><?php echo $p['status']; ?></td>
68.             <td>
69.                 <a href="edit-drilling.php?id=<?php echo
$p['id']; ?>">Edit</a>
70.                 <a href="?toggle=<?php echo $p['id'];
?>">Toggle</a>
71.                 <a href="?delete=<?php echo $p['id']; ?>"
onclick="return confirm('Yakin?')">Hapus</a>
72.             </td>
73.         </tr>
74.         <?php endforeach; ?>
75.     </tbody>
76. </table>
77. <?php else: ?>
78. <p>Belum ada produk. <a href="add-drilling.php">Tambah
sekarang</a></p>
79. <?php endif; ?>

```

Kode Program 3. 9 Halaman Drilling Admin

Kode Program 3.10 merupakan kode program untuk menambah konten *drilling* admin. Pada baris 10–13, input seperti judul, deskripsi, status, dan urutan tampil diambil dan dibersihkan. Validasi dilakukan di baris 16–17 untuk memastikan judul dan deskripsi tidak kosong. Proses

upload gambar dilakukan di baris 20–41, baik melalui file langsung atau URL. File dicek ekstensi dan ukurannya sebelum disimpan ke folder uploads/drilling/. Jika berhasil, URL gambar disiapkan untuk dimasukkan ke database. Pada baris 44–56, data disimpan ke database jika tidak ada error. Jika berhasil, pengguna diarahkan ke halaman daftar produk. Nilai default sort_order ditentukan di baris 61–68 berdasarkan data sebelumnya. Form input ditampilkan di baris 73–91, berisi kolom judul, deskripsi, gambar, status, dan urutan tampil. Pesan error ditampilkan jika proses gagal (baris 93–95).

```

1. <?php
2. // admin/add-drilling.php
3. require_once '../config/database.php';
4. requireLogin();
5.
6. $error = '';
7. $success = '';
8.
9. if ($_SERVER['REQUEST_METHOD'] === 'POST') {
10.     $title = trim($_POST['title'] ?? '');
11.     $description = trim($_POST['description'] ?? '');
12.     $status = $_POST['status'] ?? 'active';
13.     $sort_order = (int)($_POST['sort_order'] ?? 0);
14.     $image_url = '';
15.
16.     if (empty($title) || empty($description)) {
17.         $error = 'Judul dan deskripsi harus diisi';
18.     } else {
19.         // Upload gambar
20.         if (!empty($_FILES['image']['name']) &&
21.             $_FILES['image']['error'] === UPLOAD_ERR_OK) {
22.             $upload_dir = '../uploads/drilling/';
23.             if (!is_dir($upload_dir)) mkdir($upload_dir, 0755,
24.                 true);
25.             $ext = strtolower(pathinfo($_FILES['image']['name'],
26.                 PATHINFO_EXTENSION));
27.             $allowed = ['jpg', 'jpeg', 'png', 'gif', 'webp'];
28.             if (in_array($ext, $allowed)) {
29.                 $filename = 'drilling_' . time() . rand(1000,
30.                     9999) . ".$ext";
31.                 $target = $upload_dir . $filename;
32.                 if
33.                     (move_uploaded_file($_FILES['image']['tmp_name'], $target)) {
34.                     $image_url = 'uploads/drilling/' .
35.                         $filename;
36.                 } else {
37.                     $error = 'Gagal upload gambar';
38.                 }
39.             }
40.         }
41.     }
42. }
43.
44. // Simpan ke database
45. $data = [
46.     'title' => $title,
47.     'description' => $description,
48.     'status' => $status,
49.     'sort_order' => $sort_order,
50.     'image_url' => $image_url
51. ];
52.
53. // Simpan ke database
54. $db->insert($data);
55.
56. // Redirect ke halaman daftar produk
57. header('Location: ../produk.php');
58. exit;
59.
60. // Default sort_order
61. $default_sort_order = 0;
62.
63. // Ambil data sebelumnya
64. $previous_data = $db->get('produk');
65.
66. // Tentukan default sort_order
67. if ($previous_data) {
68.     $default_sort_order = $previous_data[0]['sort_order'];
69. }
70.
71. // Form input
72.
73. <div>
74.     <div>
75.         <input type="text" value="Judul Produk" />
76.     </div>
77.     <div>
78.         <input type="text" value="Deskripsi Produk" />
79.     </div>
80.     <div>
81.         <input type="text" value="Status Produk" />
82.     </div>
83.     <div>
84.         <input type="text" value="Urutan Tampil" />
85.     </div>
86.     <div>
87.         <input type="text" value="Gambar Produk" />
88.     </div>
89. </div>
90.
91. // Pesan error
92.
93. <div>
94.     <div>
95.         <div>
96.             <div>
97.                 <div>
98.                     <div>
99.                         <div>
100.                             <div>
101.                                 <div>
102.                                     <div>
103.                                         <div>
104.                                             <div>
105.                                                 <div>
106.                                                     <div>
107.                                                         <div>
108.                                                             <div>
109.                                                                 <div>
110.                                                                    <div>
111.                                                                        <div>
112.                                                                            <div>
113.                                                                                <div>
114.                                                                                    <div>
115.                                                                                        <div>
116.                                                                                            <div>
117.                                                                                                <div>
118.                                                                                                    <div>
119.                                                                                                        <div>
120.                                                                                                            <div>
121.                                                                                                                <div>
122.                                                                                                                    <div>
123.                                                                                                                        <div>
124.                                                                                                                            <div>
125.                                                                                                                                <div>
126.                                                                                                                                    <div>
127.                                                                                                                                        <div>
128.                                                                                                                                            <div>
129.                                                                                                                                                <div>
130.                                                                                                                                                    <div>
131.                                                                                                                                                        <div>
132.                                                                                                                                                            <div>
133.                                                                                                                                                                <div>
134.                                                                                                    </div>
135.                                                                                                </div>
136.                                                                                            </div>
137.                                                                                        </div>
138.                                                                                    </div>
139.                                                                                </div>
140.                                                                            </div>
141.                                                                        </div>
142.                                                                    </div>
143.                                                                </div>
144.                                                            </div>
145.                                                        </div>
146.                                                    </div>
147.                                                </div>
148.                                            </div>
149.                                        </div>
150.                                    </div>
151.                                </div>
152.                            </div>
153.                        </div>
154.                    </div>
155.                </div>
156.            </div>
157.        </div>
158.    </div>
159. </div>

```

```

35.         }
36.     } else {
37.         $error = 'Format gambar tidak valid';
38.     }
39. } elseif (!empty(trim($_POST['image_url']))) {
40.     $image_url = trim($_POST['image_url']);
41. }
42.
43. // Simpan ke DB
44. if (empty($error)) {
45.     $db = new Database();
46.     $conn = $db->getConnection();
47.
48.     try {
49.         $stmt = $conn->prepare("INSERT INTO services
        (title, description, image_url, status, sort_order, created_at)
        VALUES (?, ?, ?, ?, ?, NOW())");
50.         $stmt->execute([$title, $description,
        $image_url, $status, $sort_order]);
51.         header('Location: drilling.php?success=added');
52.         exit;
53.     } catch (PDOException $e) {
54.         $error = 'Gagal menyimpan: ' . $e->getMessage();
55.     }
56. }
57. }
58. }
59.
60. // Hitung urutan berikutnya
61. $db = new Database();
62. $conn = $db->getConnection();
63. $next_sort_order = 1;
64. try {
65.     $stmt = $conn->query("SELECT MAX(sort_order) as max_order
        FROM services");
66.     $row = $stmt->fetch();
67.     $next_sort_order = ($row['max_order'] ?? 0) + 1;
68. } catch (PDOException $e) {}
69.
70. ?>
71.
72. <!-- FORM HTML -->
73. <form method="POST" enctype="multipart/form-data">
74.     <input type="text" name="title" required placeholder="Judul"
        value="<?= htmlspecialchars($_POST['title'] ?? '') ?>">
75.     <textarea name="description" required
        placeholder="Deskripsi"><?=
        htmlspecialchars($_POST['description'] ?? '') ?></textarea>
76.
77.     <!-- Upload gambar -->

```

```

78.     <input type="file" name="image" accept="image/*">
79.     <input type="url" name="image_url" placeholder="https://..."
      value="<?= htmlspecialchars($_POST['image_url'] ?? '') ?>">
80.
81.     <!-- Status & Urutan -->
82.     <select name="status">
83.         <option value="active" <?= ($_POST['status'] ??
      'active') === 'active' ? 'selected' : '' ?>>Aktif</option>
84.         <option value="inactive" <?= ($_POST['status'] ?? '')
      === 'inactive' ? 'selected' : '' ?>>Nonaktif</option>
85.     </select>
86.
87.     <input type="number" name="sort_order" min="1" value="<?=
      $_POST['sort_order'] ?? $next_sort_order ?>">
88.
89.     <!-- Submit -->
90.     <button type="submit">Simpan</button>
91. </form>
92.
93. <?php if (!empty($error)): ?>
94.     <div class="error"><?= $error ?></div>
95. <?php endif; ?>

```

Kode Program 3.10 Halaman Tambah Konten Drilling Admin

Kode program 3.11 merupakan kode program yang digunakan untuk mengedit konten halaman drilling. Pertama, kode akan memastikan pengguna sudah login (baris 3) dan mengecek apakah parameter id tersedia di URL (baris 7–9). Jika ID valid, data layanan diambil dari database berdasarkan ID tersebut (baris 17–19). Setelah data berhasil diambil, sistem menampilkan form yang sudah terisi data lama (baris 68–90). Jika pengguna mengirimkan form (baris 28), maka input akan divalidasi dan diproses, termasuk menangani upload gambar baru atau URL gambar manual (baris 38–44). Jika tidak ada error, data layanan akan di-*update* ke database (baris 48–49), dan pengguna akan diarahkan kembali ke halaman utama dengan pesan sukses.

```

1. <?php
2. require_once '../config/database.php';
3. requireLogin();
4.
5. $error = ''; $success = ''; $product = null;
6.
7. if (!isset($_GET['id']) || empty($_GET['id'])) {
8.     header('Location: drilling.php?error=invalid_id'); exit();
9. }
10.
11. $product_id = (int)$_GET['id'];
12. $database = new Database();
13. $conn = $database->getConnection();

```

```

14.
15. // Ambil data produk
16. try {
17.     $stmt = $conn->prepare("SELECT * FROM services WHERE id =
    ?");
18.     $stmt->execute([$product_id]);
19.     $product = $stmt->fetch(PDO::FETCH_ASSOC);
20.     if (!$product) {
21.         header('Location: drilling.php?error=not_found');
        exit();
22.     }
23. } catch(PDOException $e) {
24.     $error = 'Gagal mengambil data produk';
25. }
26.
27. // Proses update
28. if ($_SERVER['REQUEST_METHOD'] === 'POST') {
29.     $title = sanitizeInput($_POST['title']);
30.     $description = sanitizeInput($_POST['description']);
31.     $sort_order = (int)$_POST['sort_order'];
32.     $status = $_POST['status'];
33.     $image_url = $product['image_url'];
34.
35.     if (empty($title) || empty($description)) {
36.         $error = 'Harap isi semua field yang wajib';
37.     } else {
38.         if (isset($_FILES['image']) && $_FILES['image']['error']
    === UPLOAD_ERR_OK) {
39.             $upload = uploadImage($_FILES['image'],
    '../uploads/');
40.             if ($upload['success']) $image_url = 'uploads/' .
    $upload['filename'];
41.             else $error = 'Upload gagal: ' . $upload['message'];
42.         } elseif (!empty($_POST['image_url_manual'])) {
43.             $image_url =
    sanitizeInput($_POST['image_url_manual']);
44.         }
45.
46.         if (empty($error)) {
47.             try {
48.                 $stmt = $conn->prepare("UPDATE services SET
    title=?, description=?, image_url=?, status=?, sort_order=?,
    updated_at=NOW() WHERE id=?");
49.                 $stmt->execute([$title, $description,
    $image_url, $status, $sort_order, $product_id]);
50.                 header('Location:
    drilling.php?success=updated'); exit();
51.             } catch(PDOException $e) {
52.                 $error = 'Gagal mengupdate produk: ' . $e-
    >getMessage();
53.             }

```

```

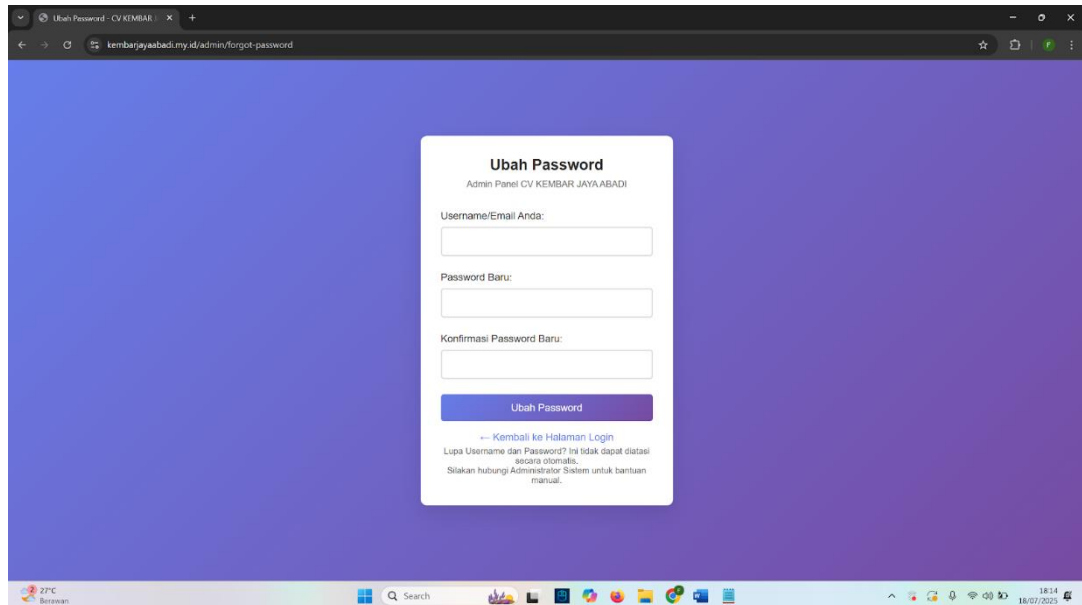
54.     }
55. }
56. }
57. ?>
58.
59. <!-- HTML Form -->
60. <!DOCTYPE html>
61. <html>
62. <head><title>Edit Drilling</title></head>
63. <body>
64.     <h1>Edit Drilling</h1>
65.
66.     <?php if ($error): ?><div style="color:red"><?= $error
    ?></div><?php endif; ?>
67.     <?php if ($product): ?>
68.         <form method="POST" enctype="multipart/form-data">
69.             <label>Judul *</label><br>
70.             <input type="text" name="title" value="<?=
    htmlspecialchars($product['title']) ?>"><br><br>
71.
72.             <label>Deskripsi *</label><br>
73.             <textarea name="description"><?=
    htmlspecialchars($product['description']) ?></textarea><br><br>
74.
75.             <label>Gambar</label><br>
76.             <input type="file" name="image"><br>
77.             <input type="url" name="image_url_manual"
    placeholder="URL gambar"><br><br>
78.
79.             <label>Urutan</label><br>
80.             <input type="number" name="sort_order" value="<?=
    $product['sort_order'] ?>"><br><br>
81.
82.             <label>Status</label><br>
83.             <select name="status">
84.                 <option value="active" <?= $product['status'] ===
    'active' ? 'selected' : '' ?>>Aktif</option>
85.                 <option value="inactive" <?= $product['status'] ===
    'inactive' ? 'selected' : '' ?>>Nonaktif</option>
86.             </select><br><br>
87.
88.             <button type="submit">Simpan</button>
89.             <a href="drilling.php">Batal</a>
90.         </form>
91.     <?php endif; ?>
92. </body>
93. </html>

```

Kode Program 3.11 Halaman Edit Konten Drilling Admin

m. Tampilan Forgot Password Admin

Gambar 3.41 menunjukkan tampilan halaman reset password pada website admin. Pada halaman ini, admin diminta untuk memasukkan username atau email terlebih dahulu. Setelah itu, admin perlu memasukkan password baru dan mengkonfirmasinya dengan mengulang password tersebut sekali lagi.



Gambar 3.41 Halaman Ubah Password

Kode Program 3.12 merupakan kode program yang menangani proses reset password admin jika lupa password. Di bagian awal (baris 8–11), sistem menerima input dari form, yaitu username atau email, password baru, dan konfirmasi password baru. Setelah itu, sistem melakukan pengecekan (baris 13–19) untuk memastikan bahwa semua data sudah diisi, password dan konfirmasinya sama, serta panjang password minimal 6 karakter. Kalau semua syarat sudah terpenuhi, sistem akan mencari data admin berdasarkan username atau email (baris 24–27). Kalau ditemukan, password baru akan diamankan dengan cara di-hash (baris 30), lalu disimpan ke database (baris 32–34). Kalau proses ini berhasil, akan muncul pesan bahwa password sudah berhasil diubah (baris 36). Tapi kalau user-nya tidak ditemukan atau ada kesalahan sistem, maka akan muncul pesan error (baris 38 dan 42). Form yang digunakan untuk input data ditampilkan di baris 69–87, dan terdiri dari tiga bagian: username/email, password baru, dan konfirmasi password. Di bagian bawah (baris 89–95), ada link untuk kembali ke halaman login dan catatan bahwa kalau lupa username dan password sekaligus, harus hubungi admin secara manual.

1. <?php
2. // admin/forgot-password.php

```

3. require_once '../config/database.php';
4.
5. $message = '';
6. $error = '';
7.
8. if ($_SERVER['REQUEST_METHOD'] === 'POST') {
9.     $identifier = sanitizeInput($_POST['identifier']);
10.    $new_password = $_POST['new_password'];
11.    $confirm_new_password = $_POST['confirm_new_password'];
12.
13.    if (empty($identifier) || empty($new_password) ||
        empty($confirm_new_password)) {
14.        $error = 'Harap isi semua field.';
15.    } elseif ($new_password !== $confirm_new_password) {
16.        $error = 'Password baru dan konfirmasi password tidak
        cocok.';
17.    } elseif (strlen($new_password) < 6) { // Minimal 6 karakter
18.        $error = 'Password baru minimal harus 6 karakter.';
19.    } else {
20.        $database = new Database();
21.        $conn = $database->getConnection();
22.
23.        try {
24.            $query = "SELECT id FROM admin_users WHERE username
            = ? OR email = ?";
25.            $stmt = $conn->prepare($query);
26.            $stmt->execute([$identifier, $identifier]);
27.            $user = $stmt->fetch(PDO::FETCH_ASSOC);
28.
29.            if ($user) {
30.                $hashed_new_password =
                password_hash($new_password, PASSWORD_DEFAULT);
31.
32.                $updateQuery = "UPDATE admin_users SET password
                = ? WHERE id = ?";
33.                $updateStmt = $conn->prepare($updateQuery);
34.                $updateStmt->execute([$hashed_new_password,
                $user['id']]);
35.
36.                $message = 'Password Anda telah berhasil
                diperbarui. Silakan login dengan password baru Anda.';
37.            } else {
38.                $error = 'Identifier (Username/Email) tidak
                ditemukan.';
39.            }
40.        } catch(PDOException $e) {
41.            error_log("Password reset error: " . $e-
            >getMessage());
42.            $error = 'Terjadi kesalahan sistem, silakan coba
            lagi nanti.';
43.        }

```

```

44.     }
45. }
46. ?>
47. <!DOCTYPE html>
48. <html lang="id">
49. <head>
50.     <meta charset="UTF-8">
51.     <meta name="viewport" content="width=device-width, initial-
scale=1.0">
52.     <title>Ubah Password - CV KEMBAR JAYA ABADI</title>
53.     <link rel="stylesheet" href="style.css">
54. </head>
55. <body>
56.     <div class="login-container">
57.         <div class="logo">
58.             <h1>Ubah Password</h1>
59.             <p>Admin Panel CV KEMBAR JAYA ABADI</p>
60.         </div>
61.
62.         <?php if ($message): ?>
63.             <div style="background: #eef; color: #33c; padding:
0.75rem; border-radius: 5px; margin-bottom: 1rem; border: 1px
solid #ccf;"><?php echo $message; ?></div>
64.         <?php endif; ?>
65.         <?php if ($error): ?>
66.             <div class="error"><?php echo $error; ?></div>
67.         <?php endif; ?>
68.
69.         <form method="POST" action="">
70.             <div class="form-group">
71.                 <label for="identifier">Username/Email
Anda:</label>
72.                 <input type="text" id="identifier"
name="identifier" required
73.                     value="<?php echo
isset($_POST['identifier']) ?
htmlspecialchars($_POST['identifier']) : ''; ?>">
74.             </div>
75.
76.             <div class="form-group">
77.                 <label for="new_password">Password Baru:</label>
78.                 <input type="password" id="new_password"
name="new_password" required>
79.             </div>
80.
81.             <div class="form-group">
82.                 <label for="confirm_new_password">Konfirmasi
Password Baru:</label>
83.                 <input type="password" id="confirm_new_password"
name="confirm_new_password" required>
84.             </div>

```

```

85.
86.         <button type="submit" class="btn">Ubah
      Password</button>
87.     </form>
88.
89.     <div class="back-link">
90.         <a href="index">← Kembali ke Halaman Login</a>
91.     </div>
92.     <div class="back-link" style="margin-top: 5px;">
93.         <p style="color:#555; font-size:0.85em;">
94.             Lupa Username dan Password? Ini tidak dapat
      diatasi secara otomatis. <br>
95.             Silakan hubungi Administrator Sistem untuk
      bantuan manual.
96.         </p>
97.     </div>
98. </div>
99. </body>
100. </html>

```

Kode Program 3.12 Halaman Forgot Password

n. Tampilan Ubah Username dan Password

Gambar 3.42 merupakan tampilan halaman untuk merubah username dan password admin.

Halaman ini terdiri dari dua bagian utama. Bagian pertama digunakan untuk mengubah username atau email, di mana admin harus mengisi data baru serta memasukkan password saat ini sebagai konfirmasi. Bagian kedua digunakan untuk mengubah password, dengan mengisi password baru, konfirmasi password baru, dan kembali memasukkan password lama sebagai verifikasi. Masing-masing bagian dilengkapi tombol “Batal” dan “Simpan/Ubah”. Di pojok kanan bawah terdapat tombol “Selesai” untuk keluar dari halaman pengaturan.

Gambar 3.42 Halaman Ubah Username dan Password Admin

Kode Program 3.13 merupakan kode yang digunakan untuk merubah password untuk akun admin. Saat halaman dibuka melalui link email (baris 9–32), sistem akan memeriksa apakah token dan email yang dikirim valid dan belum kadaluarsa. Kalau tidak valid, akan muncul pesan kesalahan. Jika admin mengirimkan form (baris 33–76), sistem akan mengecek apakah password baru dan konfirmasinya sudah diisi dengan benar. Kalau semua sesuai, password admin akan diganti dengan yang baru, dan token yang sudah dipakai akan dihapus dari database. Tampilan halaman (baris 78–125) berisi form untuk memasukkan password baru dan konfirmasi password. Jika token valid, form akan muncul disertai pesan jika proses berhasil atau gagal.

```

1. <?php
2. require_once '../config/database.php';
3.
4. $message = '';
5. $error = '';
6. $email = '';
7. $token = '';
8.
9. if ($_SERVER['REQUEST_METHOD'] === 'GET') {
10.     if (isset($_GET['token']) && isset($_GET['email'])) {
11.         $token = sanitizeInput($_GET['token']);
12.         $email = sanitizeInput($_GET['email']);
13.
14.         $database = new Database();
15.         $conn = $database->getConnection();
16.
17.         try {

```

```

18.         $query = "SELECT * FROM password_resets WHERE email
= ? AND token = ? AND expires_at > NOW()";
19.         $stmt = $conn->prepare($query);
20.         $stmt->execute([$email, $token]);
21.         $reset_request = $stmt->fetch(PDO::FETCH_ASSOC);
22.
23.         if (!$reset_request) {
24.             $error = 'Link reset password tidak valid atau
telah kedaluwarsa.';
25.         }
26.     } catch(PDOException $e) {
27.         error_log("Reset password GET error: " . $e-
>getMessage());
28.         $error = 'Terjadi kesalahan sistem, silakan coba
lagi nanti.';
29.     }
30. } else {
31.     $error = 'Link reset password tidak lengkap.';
32. }
33. } elseif ($_SERVER['REQUEST_METHOD'] === 'POST') {
34.     $email = sanitizeInput($_POST['email']);
35.     $token = sanitizeInput($_POST['token']);
36.     $new_password = $_POST['new_password'];
37.     $confirm_password = $_POST['confirm_password'];
38.
39.     if (empty($new_password) || empty($confirm_password)) {
40.         $error = 'Harap isi semua field password.';
41.     } elseif ($new_password !== $confirm_password) {
42.         $error = 'Password baru dan konfirmasi password tidak
cocok.';
43.     } elseif (strlen($new_password) < 6) {
44.         $error = 'Password minimal harus 6 karakter.';
45.     } else {
46.         $database = new Database();
47.         $conn = $database->getConnection();
48.
49.         try {
50.             $query = "SELECT * FROM password_resets WHERE email
= ? AND token = ? AND expires_at > NOW()";
51.             $stmt = $conn->prepare($query);
52.             $stmt->execute([$email, $token]);
53.             $reset_request = $stmt->fetch(PDO::FETCH_ASSOC);
54.
55.             if ($reset_request) {
56.                 $hashed_password = password_hash($new_password,
PASSWORD_DEFAULT);
57.
58.                 $updateUserQuery = "UPDATE admin_users SET
password = ? WHERE email = ?";
59.                 $updateUserStmt = $conn-
>prepare($updateUserQuery);

```

```

60.             $updateUserStmt->execute([$hashed_password,
        $email]);
61.
62.             $deleteTokenQuery = "DELETE FROM password_resets
        WHERE email = ? AND token = ?";
63.             $deleteTokenStmt = $conn-
        >prepare($deleteTokenQuery);
64.             $deleteTokenStmt->execute([$email, $token]);
65.
66.             $message = 'Password Anda telah berhasil
        direset. Silakan login dengan password baru Anda.';
67.             header('Refresh: 3; URL=login');
68.         } else {
69.             $error = 'Link reset password tidak valid atau
        telah kedaluwarsa.';
70.         }
71.     } catch(PDOException $e) {
72.         error_log("Reset password POST error: " . $e-
        >getMessage());
73.         $error = 'Terjadi kesalahan sistem, silakan coba
        lagi nanti.';
74.     }
75. }
76. }
77. ?>
78. <!DOCTYPE html>
79. <html lang="id">
80. <head>
81.     <meta charset="UTF-8">
82.     <meta name="viewport" content="width=device-width, initial-
        scale=1.0">
83.     <title>Reset Password - CV KEMBAR JAYA ABADI</title>
84.     <style>
85.     <?php include 'login_style.css'; ?>
86.     </style>
87. </head>
88. <body>
89.     <div class="login-container">
90.         <div class="logo">
91.             <h1>Reset Password</h1>
92.             <p>Admin Panel CV KEMBAR JAYA ABADI</p>
93.         </div>
94.
95.         <?php if ($message): ?>
96.         <div style="background: #eef; color: #33c; padding:
        0.75rem; border-radius: 5px; margin-bottom: 1rem; border: 1px
        solid #ccf;"><?php echo $message; ?></div>
97.         <?php endif; ?>
98.         <?php if ($error): ?>
99.         <div class="error"><?php echo $error; ?></div>
100.        <?php endif; ?>

```

```

101.
102.         <?php if (!$error || $message): ?>
103.         <form method="POST" action="">
104.             <input type="hidden" name="email" value="<?php
105.                 echo htmlspecialchars($email); ?>">
106.             <input type="hidden" name="token" value="<?php
107.                 echo htmlspecialchars($token); ?>">
108.             <div class="form-group">
109.                 <label for="new_password">Password
110.                 Baru:</label>
111.                 <input type="password" id="new_password"
112.                 name="new_password" required>
113.             </div>
114.             <div class="form-group">
115.                 <label for="confirm_password">Konfirmasi
116.                 Password Baru:</label>
117.                 <input type="password" id="confirm_password"
118.                 name="confirm_password" required>
119.             </div>
120.             <button type="submit" class="btn">Reset
121.             Password</button>
122.         </form>
123.         <?php endif; ?>
124.         <div class="back-link">
125.             <a href="login">← Kembali ke Halaman Login</a>
126.         </div>
127.     </div>
128. </body>
129. </html>

```

Kode Program 3.13 Ubah Password

Kode Program 3.14 merupakan kode program untuk melakukan perubahan pada username akun admin. Pada baris 2–4, sesi dimulai jika belum aktif, lalu login dicek dengan `requireLogin()` (baris 6). Koneksi database dibuat (baris 8–9) dan `admin_id` diambil dari sesi (baris 11). Jika admin mengirim form untuk ganti username dan email (baris 13–14), data baru diambil (baris 15–17) dan password diverifikasi (baris 20–26). Jika cocok, data diperbarui dan pesan sukses/gagal disimpan di sesi (baris 32–43). Untuk ganti password (baris 48–51), sistem cek kecocokan password baru dan konfirmasi (baris 53–57), lalu validasi password lama (baris 60–66). Jika valid, password di-hash dan diperbarui dengan pesan hasil (baris 72–83). Input yang tidak sesuai dianggap tidak valid (baris 88–90). Setelah proses selesai, diarahkan kembali ke `settings.php` (baris 92–93).

```

1. <?php
2. if (session_status() == PHP_SESSION_NONE) {

```



```

3.     session_start();
4. }
5. require_once '../config/database.php';
6. requireLogin();
7.
8. $database = new Database();
9. $conn = $database->getConnection();
10.
11. $admin_id = $_SESSION['admin_id'];
12.
13. if ($_SERVER['REQUEST_METHOD'] == 'POST') {
14.     if (isset($_POST['username']) && isset($_POST['email']) &&
        isset($_POST['current_password'])) {
15.         $new_username = sanitizeInput($_POST['username']);
16.         $new_email = sanitizeInput($_POST['email']);
17.         $current_password =
            sanitizeInput($_POST['current_password']);
18.
19.         try {
20.             $query = "SELECT password FROM admin_users WHERE id
                = :id LIMIT 0,1";
21.             $stmt = $conn->prepare($query);
22.             $stmt->bindParam(':id', $admin_id);
23.             $stmt->execute();
24.             $admin_data = $stmt->fetch(PDO::FETCH_ASSOC);
25.
26.             if (!$admin_data ||
                !password_verify($current_password, $admin_data['password'])) {
27.                 $_SESSION['profile_error'] = 'Password saat ini
                    salah.';
28.                 header('Location: settings.php');
29.                 exit();
30.             }
31.
32.             $update_query = "UPDATE admin_users SET username =
                :username, email = :email WHERE id = :id";
33.             $update_stmt = $conn->prepare($update_query);
34.             $update_stmt->bindParam(':username', $new_username);
35.             $update_stmt->bindParam(':email', $new_email);
36.             $update_stmt->bindParam(':id', $admin_id);
37.
38.             if ($update_stmt->execute()) {
39.                 $_SESSION['admin_username'] = $new_username;
40.                 $_SESSION['profile_message'] = 'Username dan
                    email berhasil diperbarui.';
41.             } else {
42.                 $_SESSION['profile_error'] = 'Gagal memperbarui
                    username atau email. Silakan coba lagi.';
43.             }
44.

```

```

45.         } catch (PDOException $e) {
46.             $_SESSION['profile_error'] = 'Terjadi kesalahan
database: ' . $e->getMessage();
47.         }
48.     } elseif (isset($_POST['new_password']) &&
isset($_POST['confirm_new_password']) &&
isset($_POST['current_password'])) {
49.         $new_password = $_POST['new_password'];
50.         $confirm_new_password = $_POST['confirm_new_password'];
51.         $current_password =
sanitizeInput($_POST['current_password']);
52.
53.         if ($new_password !== $confirm_new_password) {
54.             $_SESSION['profile_error'] = 'Password baru dan
konfirmasi password tidak cocok.';
55.             header('Location: settings.php');
56.             exit();
57.         }
58.
59.         try {
60.             $query = "SELECT password FROM admins WHERE id = :id
LIMIT 0,1";
61.             $stmt = $conn->prepare($query);
62.             $stmt->bindParam(':id', $admin_id);
63.             $stmt->execute();
64.             $admin_data = $stmt->fetch(PDO::FETCH_ASSOC);
65.
66.             if (!$admin_data ||
!password_verify($current_password, $admin_data['password'])) {
67.                 $_SESSION['profile_error'] = 'Password saat ini
salah.';
68.                 header('Location: settings.php');
69.                 exit();
70.             }
71.
72.             $hashed_new_password = password_hash($new_password,
PASSWORD_BCRYPT);
73.
74.             $update_query = "UPDATE admins SET password =
:password WHERE id = :id";
75.             $update_stmt = $conn->prepare($update_query);
76.             $update_stmt->bindParam(':password',
$hashed_new_password);
77.             $update_stmt->bindParam(':id', $admin_id);
78.
79.             if ($update_stmt->execute()) {
80.                 $_SESSION['profile_message'] = 'Password
berhasil diperbarui.';
81.             } else {
82.                 $_SESSION['profile_error'] = 'Gagal memperbarui
password. Silakan coba lagi.';

```

```

83.         }
84.
85.         } catch (PDOException $e) {
86.             $_SESSION['profile_error'] = 'Terjadi kesalahan
database: ' . $e->getMessage();
87.         }
88.     } else {
89.         $_SESSION['profile_error'] = 'Permintaan tidak valid.';
90.     }
91.
92.     header('Location: settings.php');
93.     exit();
94. } else {
95.     header('Location: settings.php');
96.     exit();
97. }
98. ?>

```

Kode Program 3.14 Update Username

Kode Program 3.15 merupakan kode program tampilan halaman pengaturan akun admin.

```

1. <?php
2. if (session_status() == PHP_SESSION_NONE) {
3.     session_start();
4. }
5. require_once '../config/database.php';
6. requireLogin();
7.
8. $message = '';
9. $error = '';
10.
11. $database = new Database();
12. $conn = $database->getConnection();
13.
14. try {
15.     $admin_id = $_SESSION['admin_id'];
16.     $query = "SELECT username, email FROM admin_users WHERE id =
:id LIMIT 0,1";
17.     $stmt = $conn->prepare($query);
18.     $stmt->bindParam(':id', $admin_id);
19.     $stmt->execute();
20.     $admin_data = $stmt->fetch(PDO::FETCH_ASSOC);
21.
22.     if (!$admin_data) {
23.         session_unset();
24.         session_destroy();
25.         header('Location: login');
26.         exit();
27.     }
28.     $current_username =
htmlspecialchars($admin_data['username']);

```

```

29.     $current_email = htmlspecialchars($admin_data['email']);
30.
31. } catch (PDOException $e) {
32.     $error = "Terjadi kesalahan saat mengambil data admin: " .
        $e->getMessage();
33.     $current_username = '';
34.     $current_email = '';
35. }
36.
37. if (isset($_SESSION['profile_message'])) {
38.     $message = $_SESSION['profile_message'];
39.     unset($_SESSION['profile_message']);
40. }
41. if (isset($_SESSION['profile_error'])) {
42.     $error = $_SESSION['profile_error'];
43.     unset($_SESSION['profile_error']);
44. }
45.
46. ?>
47. <!DOCTYPE html>
48. <html lang="id">
49. <head>
50.     <meta charset="UTF-8">
51.     <meta name="viewport" content="width=device-width, initial-
        scale=1.0">
52.     <title>Pengaturan Akun Admin - CV KEMBAR JAYA ABADI</title>
53.     <link rel="stylesheet" href="admin_layout.css" />
54.     <link rel="stylesheet" href="settings_style.css" />
55. </head>
56. <body>
57.     <div class="admin-wrapper">
58.         <?php include 'includes/sidebar_admin.php'; ?>
59.
60.         <div class="main-content">
61.             <?php $page_title = "Pengaturan Akun"; include
                'includes/header_admin.php'; ?>
62.
63.             <div class="container">
64.                 <?php if ($message): ?>
65.                     <div class="alert alert-success"><?php echo
                        $message; ?></div>
66.                 <?php endif; ?>
67.                 <?php if ($error): ?>
68.                     <div class="alert alert-error"><?php echo
                        $error; ?></div>
69.                 <?php endif; ?>
70.
71.                 <div class="form-container">
72.                     <h2>Ubah Username atau Email</h2>

```

```

73.         <form action="update_profile.php"
method="POST">
74.             <div class="form-group">
75.                 <label for="username">Username
Baru:</label>
76.                 <input type="text" id="username"
name="username" value="<?php echo $current_username; ?>"
required>
77.             </div>
78.             <div class="form-group">
79.                 <label for="email">Email
Baru:</label>
80.                 <input type="email" id="email"
name="email" value="<?php echo $current_email; ?>" required>
81.             </div>
82.             <div class="form-group">
83.                 <label
for="current_password">Password Saat Ini (Konfirmasi
Perubahan):</label>
84.                 <input type="password"
id="current_password" name="current_password" required>
85.             </div>
86.             <div class="form-actions">
87.                 <a href="dashboard.php" class="btn-
cancel">Batal</a>
88.                 <button type="submit" class="btn-
submit">Simpan Perubahan</button>
89.             </div>
90.         </form>
91.
92.         <h2 style="margin-top: 40px;">Ubah
Password</h2>
93.         <form action="update_profile.php"
method="POST">
94.             <div class="form-group">
95.                 <label for="new_password">Password
Baru:</label>
96.                 <input type="password"
id="new_password" name="new_password" required>
97.             </div>
98.             <div class="form-group">
99.                 <label
for="confirm_new_password">Konfirmasi Password Baru:</label>
100.                 <input type="password"
id="confirm_new_password" name="confirm_new_password" required>
101.             </div>
102.             <div class="form-group">
103.                 <label
for="current_password_for_password">Password Saat Ini
(Konfirmasi Perubahan):</label>

```

```

104.             <input type="password"
            id="current_password_for_password" name="current_password"
            required>
105.             </div>
106.             <div class="form-actions">
107.                 <a href="dashboard.php"
            class="btn-cancel">Batal</a>
108.                 <button type="submit" class="btn-
            submit">Ubah Password</button>
109.             </div>
110.         </form>
111.     </div>
112.
113.         <div class="bottom-actions">
114.             <a href="dashboard.php" class="btn-
            success-done">Selesai</a>
115.         </div>
116.
117.     </div>
118. </div>
119. </div>
120. </body>
121. </html>

```

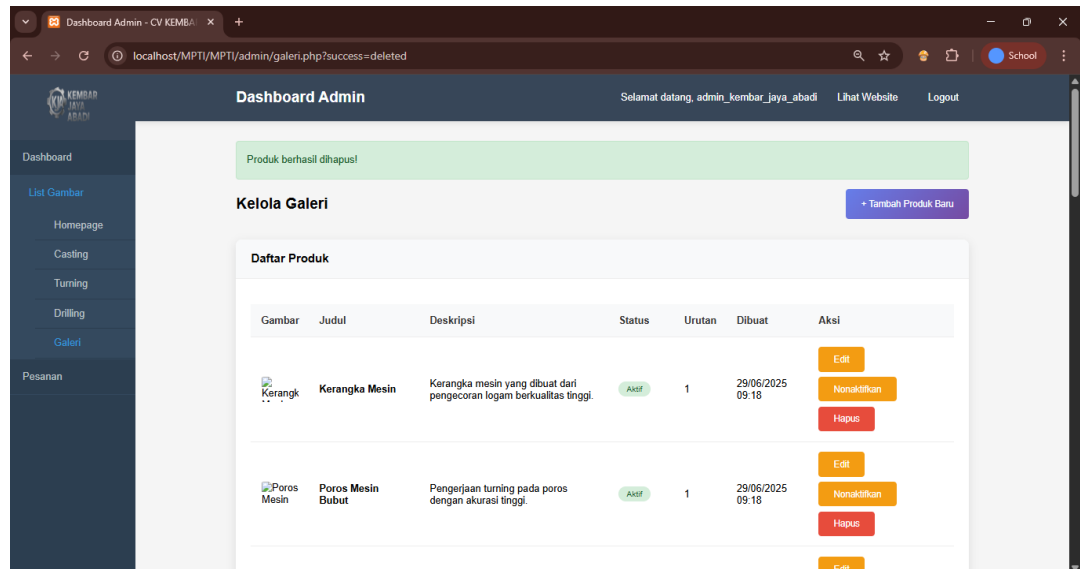
Kode Program 3.15 Halaman Pengaturan Profile

4. Aisyah Dwi Puspita

(Backend developer)

a. Manajemen Data Galeri

Gambar 3.43 merupakan tampilan halaman Kelola Galeri pada dashboard admin yang berfungsi untuk mengelola data gambar produk perusahaan. Pada tabel Daftar Produk, admin dapat melihat informasi setiap item galeri seperti gambar produk, judul, deskripsi singkat, status aktif atau nonaktif, urutan tampilan, serta waktu unggah. Setiap data galeri disertai tombol aksi berupa Edit, Nonaktifkan/Aktifkan, dan Hapus, yang memungkinkan admin untuk memperbarui informasi produk, mengubah status tampil, atau menghapus produk secara langsung dari antarmuka.



Gambar 3.43 Halaman Manajemen List Gambar Data Galeri

Kode Program 3.16 merupakan script PHP yang menangani manajemen data galeri produk dalam sistem admin. Pada baris 1–3, file konfigurasi database dimuat dan autentikasi admin dilakukan melalui `requireLogin()`. Baris 5 mendefinisikan tabel yang diizinkan untuk dimanipulasi, yaitu `'galeri_services'`. Baris 7–21 menangani penghapusan data galeri; jika parameter `'delete'` dan `'source'` ditemukan, data akan dihapus berdasarkan ID yang diberikan setelah validasi sumber, dan jika berhasil akan diarahkan ulang ke halaman dengan notifikasi. Baris 23–36 mengelola perubahan status galeri dari aktif ke nonaktif atau sebaliknya menggunakan query `'UPDATE'` dengan `'CASE'`, dan hasilnya juga diarahkan ulang ke halaman utama. Selanjutnya, baris 38–48 melakukan pengambilan data dari database menggunakan perintah `'SELECT'`, menyimpan hasilnya ke dalam array `'$products'`, serta menampilkan pesan kesalahan jika query gagal. Seluruh proses dalam kode ini mendukung fitur CRUD dasar (Read, Delete, dan Update status) untuk galeri produk perusahaan.

```

1. <?php
2. require_once '../config/database.php';
3. requireLogin();
4.
5. $error = '';
6. $success = '';
7.
8. $allowedTables = ['galeri_services'];
9.
10. // Hapus data galeri

```

```

11. if (isset($_GET['delete']) && isset($_GET['source'])) {
12.     $id = (int)$_GET['delete'];
13.     $source = $_GET['source'];
14.
15.     if (in_array($source, $allowedTables)) {
16.         $database = new Database();
17.         $conn = $database->getConnection();
18.
19.         try {
20.             $query = "DELETE FROM {$source} WHERE id = ?";
21.             $stmt = $conn->prepare($query);
22.             $stmt->execute([$id]);
23.             header('Location: galeri.php?success=deleted');
24.             exit();
25.         } catch(PDOException $e) {
26.             $error = 'Gagal menghapus data';
27.         }
28.     }
29. }
30.
31. // Aktif/Nonaktifkan status galeri
32. if (isset($_GET['toggle']) && isset($_GET['source'])) {
33.     $id = (int)$_GET['toggle'];
34.     $source = $_GET['source'];
35.
36.     if (in_array($source, $allowedTables)) {
37.         $database = new Database();
38.         $conn = $database->getConnection();
39.
40.         try {
41.             $query = "UPDATE {$source}
42.                 SET status = CASE
43.                     WHEN status = 'active' THEN
44.                         'inactive'
45.                     ELSE 'active'
46.                 END
47.                 WHERE id = ?";
48.             $stmt = $conn->prepare($query);
49.             $stmt->execute([$id]);
50.             header('Location: galeri.php?success=updated');
51.             exit();
52.         } catch(PDOException $e) {
53.             $error = 'Gagal mengubah status';
54.         }
55.     }
56.
57. // Ambil data galeri
58. $database = new Database();

```



```

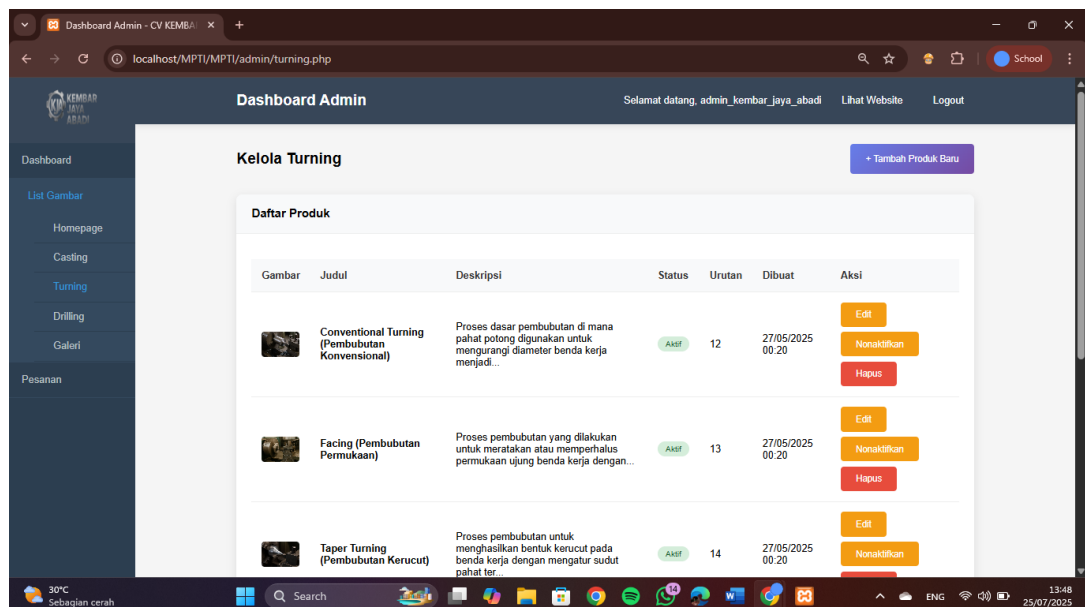
59. $conn = $database->getConnection();
60.
61. try {
62.     $query = "
63.         SELECT id, title, description, image_url, sort_order,
64.         status, created_at, updated_at,
65.         'galeri_services' AS source
66.         FROM galeri_services
67.         ORDER BY sort_order ASC, created_at DESC;
68.     ";
69.     $stmt = $conn->prepare($query);
70.     $stmt->execute();
71.     $products = $stmt->fetchAll(PDO::FETCH_ASSOC);
72. } catch(PDOException $e) {
73.     $products = [];
74.     $error = 'Gagal mengambil data produk';
75. }
76. ?>

```

Kode Program 3. 16 Manajemen Data Galeri

b. Tampilan Turning Admin

Gambar 3.44 merupakan tampilan halaman *turning* admin. Di halaman ini, admin dapat mengelola konten *turning* yang ada di website pengguna. Disini admin bisa menambah, mengedit informasi, menonaktif atau mengaktifkan konten yang sudah ada, dan menghapus konten tentang *turning*.



Gambar 3.44 Halaman List Gambar Turning Admin

Kode Program 3.17 merupakan kode program untuk halaman turning admin. Pada baris 2-5, dilakukan inisialisasi sesi, koneksi database, dan pengecekan login dengan `requireLogin()` untuk membatasi akses. Fitur penghapusan produk terdapat pada baris 7-18, di mana jika parameter `delete` terdeteksi di URL, maka data produk akan dihapus dari tabel `turning_services` berdasarkan ID-nya. Sedangkan baris 20-31 menangani logika untuk mengubah status produk antara `active` dan `inactive` melalui parameter `toggle` menggunakan query `CASE WHEN`. Kemudian, pada baris 33-41, semua data produk diambil dari database dan diurutkan berdasarkan `sort_order` dan `created_at`, lalu disimpan ke dalam array `$products`. Tampilan HTML dimulai dari baris 44, di mana jika ada parameter `success` di URL, akan ditampilkan notifikasi keberhasilan (baris 45-47). Tautan untuk menambah produk baru terdapat pada baris 49. Selanjutnya, baris 51-73 menampilkan daftar produk dalam bentuk tabel, termasuk kolom gambar, judul, deskripsi singkat (maksimal 100 karakter), status, urutan, dan aksi (edit, toggle, hapus). Tautan hapus dilengkapi konfirmasi JavaScript pada baris 70. Jika tidak ada data produk, maka ditampilkan pesan kosong dengan tautan untuk menambah produk pada baris 75.

```

1. <?php
2. // admin/turning.php
3. session_start();
4. require_once '../config/database.php';
5. requireLogin();
6.
7. // Hapus produk
8. if (isset($_GET['delete'])) {
9.     $id = (int)$_GET['delete'];
10.    try {
11.        (new Database())->getConnection()
12.            ->prepare("DELETE FROM turning_services WHERE id =
13.                ?")
14.            ->execute([$id]);
15.        header('Location: turning.php?success=deleted');
16.        exit();
17.    } catch(PDOException $e) {
18.        $error = 'Gagal menghapus produk';
19.    }
20.
21. // Toggle status produk
22. if (isset($_GET['toggle'])) {
23.     $id = (int)$_GET['toggle'];
24.     try {
25.         (new Database())->getConnection()

```

```

26.         ->prepare("UPDATE turning_services SET status = CASE
    WHEN status = 'active' THEN 'inactive' ELSE 'active' END WHERE
    id = ?")
27.         ->execute([$id]);
28.         header('Location: turning.php?success=updated');
29.         exit();
30.     } catch(PDOException $e) {
31.         $error = 'Gagal mengubah status produk';
32.     }
33. }
34.
35. // Ambil semua data produk
36. try {
37.     $products = (new Database())->getConnection()
38.         ->query("SELECT * FROM turning_services ORDER BY
    sort_order ASC, created_at DESC")
39.         ->fetchAll(PDO::FETCH_ASSOC);
40. } catch(PDOException $e) {
41.     $products = [];
42.     $error = 'Gagal mengambil data produk';
43. }
44. ?>
45.
46. <!-- HTML TAMPILAN UTAMA -->
47. <h1>Dashboard Admin - Turning</h1>
48. <?php if (isset($_GET['success'])): ?>
49.     <div><?php echo htmlspecialchars($_GET['success']); ?></div>
50. <?php endif; ?>
51.
52. <a href="add-turning.php">+ Tambah Produk Baru</a>
53.
54. <?php if (!empty($products)): ?>
55. <table>
56.     <thead>
57.         <tr>
58.             <th>Gambar</th><th>Judul</th><th>Deskripsi</th><th>S
    tatus</th><th>Urutan</th><th>Aksi</th>
59.         </tr>
60.     </thead>
61.     <tbody>
62.         <?php foreach ($products as $p): ?>
63.         <tr>
64.             <td></td>
65.             <td><?php echo htmlspecialchars($p['title']);
    ?></td>
66.             <td><?php echo
    substr(htmlspecialchars($p['description']), 0, 100); ?></td>
67.             <td><?php echo $p['status']; ?></td>
68.             <td><?php echo $p['sort_order']; ?></td>

```

```

69.             <td>
70.                 <a href="edit-turning.php?id=<?php echo
    $p['id']; ?>">Edit</a>
71.                 <a href="?toggle=<?php echo $p['id'];
    ?>">Toggle</a>
72.                 <a href="?delete=<?php echo $p['id']; ?>"
    onclick="return confirm('Yakin?')">Hapus</a>
73.             </td>
74.         </tr>
75.     <?php endforeach; ?>
76. </tbody>
77. </table>
78. <?php else: ?>
79. <p>Belum ada produk. <a href="add-turning.php">Tambah
    sekarang</a></p>
80. <?php endif; ?>

```

Kode Program 3.17 Tampilan Turning Admin

Kode Program 3.18 merupakan kode program yang digunakan untuk menambah produk turning di bagian admin. Pada baris 2-4, file database diimpor dan akses dibatasi hanya untuk admin yang login. Baris 6-7 menginisialisasi variabel error dan success untuk menangani pesan notifikasi. Baris 9-49 menangani proses form submission: mengambil data dari form (baris 10-14), memvalidasi input wajib (baris 16-18), mengelola upload gambar lokal dengan validasi ekstensi (baris 20-38) atau menggunakan URL eksternal (baris 39-41), kemudian menyimpan data ke tabel turning_services (baris 44-48). Baris 52-60 menghitung urutan berikutnya dengan mengambil nilai maksimal dari kolom sort_order dan menambahkannya dengan 1. Bagian HTML di baris 64-82 menampilkan form tambah produk dengan field judul, deskripsi, upload gambar atau URL, status, dan urutan, serta tombol simpan. Pesan error ditampilkan di baris 84-86 jika terjadi kesalahan dalam proses penyimpanan.

```

1. <?php
2. // admin/add-turning.php
3. require_once '../config/database.php';
4. requireLogin();
5.
6. $error = '';
7. $success = '';
8.
9. if ($_SERVER['REQUEST_METHOD'] === 'POST') {
10.     $title = trim($_POST['title'] ?? '');
11.     $description = trim($_POST['description'] ?? '');
12.     $status = $_POST['status'] ?? 'active';
13.     $sort_order = (int)($_POST['sort_order'] ?? 0);
14.     $image_url = '';
15.
16.     if (empty($title) || empty($description)) {

```

```

17.         $error = 'Judul dan deskripsi harus diisi';
18.     } else {
19.         // Upload gambar
20.         if (!empty($_FILES['image']['name']) &&
            $_FILES['image']['error'] === UPLOAD_ERR_OK) {
21.             $upload_dir = '../uploads/turning/';
22.             if (!is_dir($upload_dir)) mkdir($upload_dir, 0755,
                true);
23.
24.             $ext = strtolower(pathinfo($_FILES['image']['name'],
                PATHINFO_EXTENSION));
25.             $allowed = ['jpg', 'jpeg', 'png', 'gif', 'webp'];
26.
27.             if (in_array($ext, $allowed)) {
28.                 $filename = 'turning_' . time() . rand(1000,
                9999) . ".$ext";
29.                 $target = $upload_dir . $filename;
30.
31.                 if
                (move_uploaded_file($_FILES['image']['tmp_name'], $target)) {
32.                     $image_url = 'uploads/turning/' . $filename;
33.                 } else {
34.                     $error = 'Gagal upload gambar';
35.                 }
36.             } else {
37.                 $error = 'Format gambar tidak valid';
38.             }
39.         } elseif (!empty(trim($_POST['image_url']))) {
40.             $image_url = trim($_POST['image_url']);
41.         }
42.
43.         // Simpan ke DB
44.         if (empty($error)) {
45.             $db = new Database();
46.             $conn = $db->getConnection();
47.
48.             try {
49.                 $stmt = $conn->prepare("INSERT INTO
                turning_services (title, description, image_url, status,
                sort_order, created_at, updated_at) VALUES (?, ?, ?, ?, ?,
                NOW(), NOW())");
50.                 $stmt->execute([$title, $description,
                $image_url, $status, $sort_order]);
51.                 header('Location: turning.php?success=added');
52.                 exit;
53.             } catch (PDOException $e) {
54.                 $error = 'Gagal menyimpan: ' . $e->getMessage();
55.             }
56.         }
57.     }
58. }

```

```

59.
60. // Hitung urutan berikutnya
61. $db = new Database();
62. $conn = $db->getConnection();
63. $next_sort_order = 1;
64. try {
65.     $stmt = $conn->query("SELECT MAX(sort_order) as max_order
        FROM turning_services");
66.     $row = $stmt->fetch();
67.     $next_sort_order = ($row['max_order'] ?? 0) + 1;
68. } catch (PDOException $e) {}
69.
70. ?>
71.
72. <!-- FORM HTML -->
73. <form method="POST" enctype="multipart/form-data">
74.     <input type="text" name="title" required placeholder="Judul"
        value="<?= htmlspecialchars($_POST['title'] ?? '') ?>">
75.     <textarea name="description" required
        placeholder="Deskripsi"><?=
        htmlspecialchars($_POST['description'] ?? '') ?></textarea>
76.
77.     <!-- Upload gambar -->
78.     <input type="file" name="image" accept="image/*">
79.     <input type="url" name="image_url" placeholder="https://..."
        value="<?= htmlspecialchars($_POST['image_url'] ?? '') ?>">
80.
81.     <!-- Status & Urutan -->
82.     <select name="status">
83.         <option value="active" <?= ($_POST['status'] ??
            'active') === 'active' ? 'selected' : '' ?>>Aktif</option>
84.         <option value="inactive" <?= ($_POST['status'] ?? '')
            === 'inactive' ? 'selected' : '' ?>>Nonaktif</option>
85.     </select>
86.
87.     <input type="number" name="sort_order" min="1" value="<?=
        $_POST['sort_order'] ?? $next_sort_order ?>">
88.
89.     <!-- Submit -->
90.     <button type="submit">Simpan</button>
91. </form>
92.
93. <?php if (!empty($error)): ?>
94.     <div class="error"><?= $error ?></div>
95. <?php endif; ?>

```

Kode Program 3.18 Tampilan Tambah Produk Turning Admin

Kode Program 3.19 merupakan kode program untuk mengedit konten *turning*. Pada baris 2–3, dilakukan pemanggilan file konfigurasi database dan fungsi `requireLogin()` untuk membatasi akses hanya kepada pengguna yang sudah login. Baris 6–17 menangani

pengambilan ID dari parameter URL, lalu mengambil data layanan berdasarkan ID tersebut dari database. Jika ID tidak valid atau data tidak ditemukan, pengguna akan diarahkan kembali ke halaman turning.php dengan pesan error. Baris 20–56 memproses form saat di submit, termasuk validasi input (baris 26–28), pengunggahan gambar baru atau penghapusan gambar lama (baris 31–45), serta update data ke database menggunakan query UPDATE pada tabel turning_services (baris 49–50). Jika berhasil, pengguna akan diarahkan ke turning.php dengan parameter success=updated. Bagian HTML pada baris 59–97 menampilkan form yang sudah terisi data lama, memungkinkan pengguna mengganti judul, deskripsi, gambar, urutan tampil, dan status. Form menyediakan tiga opsi untuk gambar: upload file baru, input URL manual, atau menghapus gambar saat ini (baris 74–80). Fungsi JavaScript pada baris 100–109 memungkinkan penghapusan gambar secara interaktif dengan konfirmasi, yang akan menambahkan input hidden dan submit form otomatis.

```

1. <?php
2. require_once '../config/database.php';
3. requireLogin();
4.
5. $error = '';
6. $id = isset($_GET['id']) ? (int)$_GET['id'] : 0;
7. if ($id <= 0) { header('Location:
   turning.php?error=invalid_id'); exit(); }
8.
9. $conn = (new Database())->getConnection();
10. try {
11.     $stmt = $conn->prepare("SELECT * FROM turning_services WHERE
   id = ?");
12.     $stmt->execute([$id]);
13.     $product = $stmt->fetch(PDO::FETCH_ASSOC);
14.     if (!$product) { header('Location:
   turning.php?error=not_found'); exit(); }
15. } catch (PDOException $e) {
16.     header('Location: turning.php?error=database_error');
   exit();
17. }
18.
19. if ($_SERVER['REQUEST_METHOD'] === 'POST') {
20.     $title = sanitizeInput($_POST['title']);
21.     $description = sanitizeInput($_POST['description']);
22.     $sort_order = (int)$_POST['sort_order'];
23.     $status = $_POST['status'];
24.     $image_url = $product['image_url'];
25.
26.     if (empty($title) || empty($description)) {
27.         $error = 'Harap isi semua field yang wajib';
28.     } else {

```

```

29.         if (isset($_FILES['image']) && $_FILES['image']['error']
=== UPLOAD_ERR_OK) {
30.             $upload = uploadImage($_FILES['image'],
'../uploads/');
31.             if ($upload['success']) {
32.                 if (!empty($product['image_url']) &&
!filter_var($product['image_url'], FILTER_VALIDATE_URL)) {
33.                     @unlink('../' . $product['image_url']);
34.                 }
35.                 $image_url = 'uploads/' . $upload['filename'];
36.             } else {
37.                 $error = 'Gagal upload gambar: ' .
$upload['message'];
38.             }
39.             } elseif (!empty($_POST['image_url_manual'])) {
40.                 $image_url =
sanitizeInput($_POST['image_url_manual']);
41.             } elseif (!empty($_POST['remove_current_image'])) {
42.                 @unlink('../' . $product['image_url']);
43.                 $image_url = '';
44.             }
45.
46.             if (empty($error)) {
47.                 try {
48.                     $stmt = $conn->prepare("UPDATE turning_services
SET title=?, description=?, image_url=?, status=?, sort_order=?,
updated_at=NOW() WHERE id=?");
49.                     $stmt->execute([$title, $description,
$image_url, $status, $sort_order, $id]);
50.                     header('Location: turning.php?success=updated');
exit();
51.                 } catch (PDOException $e) {
52.                     $error = 'Gagal mengupdate layanan: ' . $e-
>getMessage();
53.                 }
54.             }
55.         }
56.     }
57. ?>
58.
59. <!DOCTYPE html>
60. <html lang="id">
61. <head>
62.     <meta charset="UTF-8">
63.     <title>Edit Layanan Turning</title>
64. </head>
65. <body>
66.     <h2>Edit Layanan</h2>
67.
68.     <?php if ($error): ?><p style="color:red"><?= $error
?></p><?php endif; ?>

```



```

69.
70.     <form method="POST" enctype="multipart/form-data">
71.         <label>Judul *<br>
72.             <input type="text" name="title" required value="<?=
htmlspecialchars($_POST['title']) ?? $product['title']> ?>">
73.         </label><br><br>
74.
75.         <label>Deskripsi *<br>
76.             <textarea name="description" required><?=
htmlspecialchars($_POST['description']) ??
$product['description']> ?></textarea>
77.         </label><br><br>
78.
79.         <label>Gambar Produk<br>
80.             <?php if (!empty($product['image_url'])): ?>
81.                 <br>
82.                 <button type="button"
onclick="removeImage()">Hapus Gambar Saat Ini</button>
83.             <?php endif; ?>
84.             <input type="file" name="image"><br>
85.             <input type="url" name="image_url_manual"
placeholder="Atau masukkan URL gambar"><br>
86.         </label><br>
87.
88.         <label>Urutan Tampil<br>
89.             <input type="number" name="sort_order" value="<?=
$_POST['sort_order'] ?? $product['sort_order'] ?>">
90.         </label><br><br>
91.
92.         <label>Status<br>
93.             <select name="status">
94.                 <option value="active" <?= ($product['status']
=== 'active') ? 'selected' : '' ?>>Aktif</option>
95.                 <option value="inactive" <?= ($product['status']
=== 'inactive') ? 'selected' : '' ?>>Nonaktif</option>
96.             </select>
97.         </label><br><br>
98.
99.         <input type="hidden" name="current_image" value="<?=
htmlspecialchars($product['image_url']) ?>">
100.
101.         <button type="submit">Simpan</button>
102.         <a href="turning.php">Batal</a>
103.     </form>
104.
105.     <script>
106.         function removeImage() {
107.             if (confirm('Yakin ingin menghapus gambar saat
ini?')) {
108.                 const input = document.createElement('input');

```

```
109.         input.type = 'hidden';
110.         input.name = 'remove_current_image';
111.         input.value = '1';
112.         document.forms[0].appendChild(input);
113.         document.forms[0].submit();
114.     }
115. }
116. </script>
117. </body>
118. </html>
```

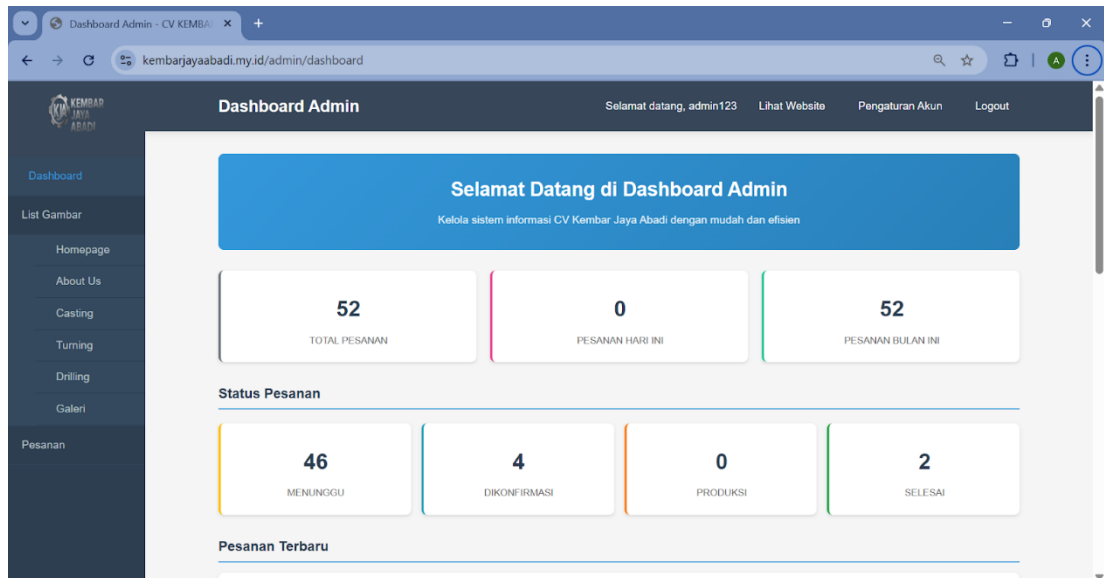
Kode Program 3.19 Tampilan Edit Turning Admin

5. Annisa Aulia

(Backend developer)

a. Statistik Data Pesanan

Pada gambar 3.45 merupakan tampilan statistik data pesanan pada dashboard admin untuk menampilkan ringkasan statistik pesanan seperti total keseluruhan pesanan, pesanan hari ini, pesanan bulan ini, serta menampilkan status pesanan pengguna yang didalamnya terdapat status menunggu, dikonfirmasi, produksi, dan selesai. Statistik data ini akan berubah sesuai dengan pembaruan data secara *real-time*, contoh pada 'pesanan hari' ini akan bertambah sesuai dengan pesanan pelanggan yang dilakukan melalui *form order* pada website. Pada bagian status pesanan juga akan berubah secara *real-time* sesuai dengan aksi yang dilakukan oleh admin dalam mengelola status pesanan.



Gambar 3. 45 Halaman Statistik Data Pesanan Admin

Kode program 3. 20 merupakan kode program yang menampilkan statistik data pesanan pada dashboard admin. Pada baris 8-21 menampilkan tiga statistik utama yaitu total pesanan yang menampilkan total keseluruhan pesanan, pesanan hari ini yaitu total pesanan yang masuk pada hari ini, dan pesanan bulan yaitu pesanan yang dilakukan dalam bulan ini. Kode ini dibuat dalam bentuk balok kartu untuk memudahkan dalam melihat statistik. Pada baris 24-44 merupakan kode yang menampilkan status pesanan dalam bentuk balok kartu, guna menghitung total setiap status pesanan yang telah dikelola oleh admin

```

1.  <!-- Welcome Message -->
2.      <div class="welcome-message">
3.          <h2>Selamat Datang di Dashboard Admin</h2>
4.          <p>Kelola sistem informasi CV Kembar Jaya
      Abadi dengan mudah dan efisien</p>
5.      </div>
6.
7.      <!-- Main Statistics Cards -->
8.      <div class="stats-cards">
9.          <div class="stat-card total">
10.             <div class="stat-number"><?php echo
                $totalOrders; ?></div>
11.             <div class="stat-label">Total
                Pesanan</div>
12.         </div>
13.         <div class="stat-card today">
14.             <div class="stat-number"><?php echo
                $todayOrders; ?></div>
15.             <div class="stat-label">Pesanan Hari
                Ini</div>
16.         </div>
17.         <div class="stat-card month">

```

```

18.         <div class="stat-number"><?php echo
    $monthOrders; ?></div>
19.         <div class="stat-label">Pesanan Bulan
    Ini</div>
20.     </div>
21. </div>
22.
23.     <!-- Status Statistics -->
24.     <div class="dashboard-section">
25.         <div class="section-title">Status
    Pesanan</div>
26.         <div class="quick-stats">
27.             <div class="stat-card pending">
28.                 <div class="stat-number"><?php echo
    $statusCounts['pending'] ?? 0; ?></div>
29.                 <div class="stat-
    label">Menunggu</div>
30.             </div>
31.             <div class="stat-card confirmed">
32.                 <div class="stat-number"><?php echo
    $statusCounts['confirmed'] ?? 0; ?></div>
33.                 <div class="stat-
    label">Dikonfirmasi</div>
34.             </div>
35.             <div class="stat-card progress">
36.                 <div class="stat-number"><?php echo
    $statusCounts['in_progress'] ?? 0; ?></div>
37.                 <div class="stat-
    label">Produksi</div>
38.             </div>
39.             <div class="stat-card completed">
40.                 <div class="stat-number"><?php echo
    $statusCounts['completed'] ?? 0; ?></div>
41.                 <div class="stat-
    label">Selesai</div>
42.             </div>
43.         </div>
44.     </div>

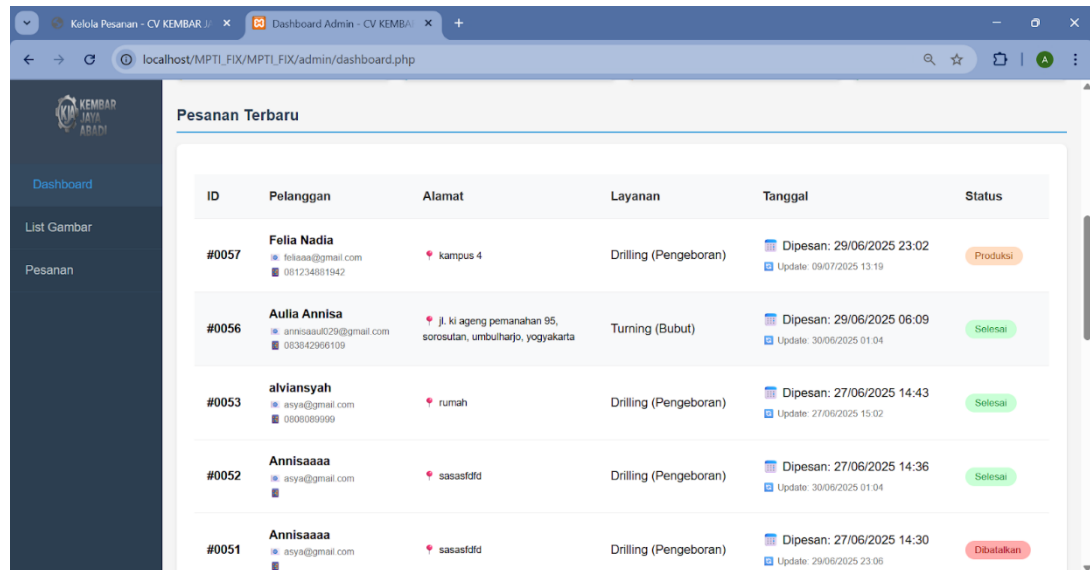
```

Kode Program 3.20 Statistik Data Pelanggan

b. Data Pesanan Pelanggan

Pada gambar 3.46 merupakan tampilan dashboard admin yang menampilkan seluruh data pelanggan yang diambil melalui form order pada website. Data tersebut berisi atribut id yang berisi nomor urut pelanggan sesuai dengan urutan pengisian form, atribut pelanggan yang berisi nama, email, telepon, atribut alamat berupa alamat pelanggan yang telah diisi pada form order, atribut layanan yang dipilih, atribut tanggal berupa tanggal dipesan oleh pelanggan dan tanggal admin melakukan pembaruan pada pesanan tersebut, dan atribut

status berupa status pesanan pelanggan. Ketika pelanggan melakukan pesanan, maka pesanan terbaru akan ditampilkan pada baris paling atas.



The screenshot shows a web browser window with two tabs: 'Kelola Pesanan - CV KEMBAR JAYA ARADI' and 'Dashboard Admin - CV KEMBAR JAYA ARADI'. The address bar shows 'localhost/MPTI_FIX/MPTI_FIX/admin/dashboard.php'. The dashboard has a sidebar with 'Dashboard', 'List Gambar', and 'Pesanan'. The main content area is titled 'Pesanan Terbaru' and displays a table of recent orders.

ID	Pelanggan	Alamat	Layanan	Tanggal	Status
#0057	Felia Nadia feliasad@gmail.com 081234881942	kampus 4	Drilling (Pengeboran)	Dipesan: 29/06/2025 23:02 Update: 09/07/2025 13:19	Produksi
#0056	Aulia Annisa annisaaul029@gmail.com 083842966109	Jl. Ki Ageng pamanahan 95, sorosutan, umbulharjo, yogyakarta	Turning (Bubut)	Dipesan: 29/06/2025 06:09 Update: 30/06/2025 01:04	Selesai
#0053	alviansyah asya@gmail.com 0808089999	rumah	Drilling (Pengeboran)	Dipesan: 27/06/2025 14:43 Update: 27/06/2025 15:02	Selesai
#0052	Annisaaaa asya@gmail.com	sasasfdd	Drilling (Pengeboran)	Dipesan: 27/06/2025 14:36 Update: 30/06/2025 01:04	Selesai
#0051	Annisaaaa asya@gmail.com	sasasfdd	Drilling (Pengeboran)	Dipesan: 27/06/2025 14:30 Update: 29/06/2025 23:06	Dibatalkan

Gambar 3. 46 Halaman Daftar Pesanan Admin

Kode program 3. 21 merupakan kode program PHP yang digunakan pada daftar pesanan pelanggan pada dashboard admin. Pada baris 1-5 merupakan query untuk mengambil data pelanggan dari form order kemudian data terbaru akan tampil pada baris paling atas serta mengambil 15 orderan terbaru yang tampil pada daftar. Pada baris 28-29 merupakan atribut ID dengan nomor urut pelanggan sebanyak 4 digit, sesuai pesanan yang masuk melalui form order. Baris 32-58 merupakan kode yang menampilkan isi dari tabel atau daftar pesanan pelanggan. Pada baris 66-69 berfungsi untuk membuat dashboard untuk refresh setiap 2 menit sekali agar dashboard admin tetap mengambil data terbaru.

```

1. // Get all orders - removed duplicate query
2. $query = "SELECT id, name, email, phone, address,
   service_type, message, status, created_at, updated_at FROM
   pesanan ORDER BY created_at DESC LIMIT 15";
3. $stmt = $conn->prepare($query);
4. $stmt->execute();
5. $recent_orders = $stmt->fetchAll(PDO::FETCH_ASSOC);
6. <!-- Recent Orders -->
7. <div class="dashboard-section">
8.     <div class="section-title">Pesanan
   Terbaru</div>
9.     <div class="card recent-orders">
10.         <div class="card-body">
11.             <?php if (!empty($recent_orders)):
12.                 <div class="table-responsive">
13.                     <table class="table">

```

```

14.         <thead>
15.             <tr>
16.                 <th>ID</th>
17.                 <th>Pelanggan</th>
18.                 <th>Alamat</th>
19.                 <th>Layanan</th>
20.                 <th>Tanggal</th>
21.                 <th>Status</th>
22.             </tr>
23.         </thead>
24.         <tbody>
25.             <?php foreach
($recent_orders as $order): ?>
26.                 <tr>
27.                     <td><strong>#<?php
echo str_pad($order['id'], 4, '0', STR_PAD_LEFT); ?></strong>
28.                     </td>
29.                     <td>
30.                         <strong><?php
echo htmlspecialchars($order['name']); ?></strong>
31.                         <div
class="order-contact">
32.                             ✉ <?php
echo htmlspecialchars($order['email']); ?><br>
33.                             📞 <?php
echo htmlspecialchars($order['phone']); ?>
34.                         </div>
35.                     </td>
36.                     <td>
37.                         <div
class="order-address">
38.                             🔑
39.                             <?php echo
htmlspecialchars(substr($order['address'], 0, 100)); ?>
40.                             <?php if
(strlen($order['address']) > 100): ?>...<?php endif; ?>
41.                         </div>
42.                     </td>
43.                     <td><?php echo
htmlspecialchars($order['service_type']); ?></td>
44.                     <td>
45.                         <div
class="date-info">
46.                             📅 Dipesan:
47.                             <?php echo
date('d/m/Y H:i', strtotime($order['created_at'])); ?>
48.                         </div>
49.                         <div
class="order-details">
50.                             🕒 Update:
51.                             <?php echo
date('d/m/Y H:i', strtotime($order['updated_at'])); ?>
52.                         </div>
53.                     </td>

```

```

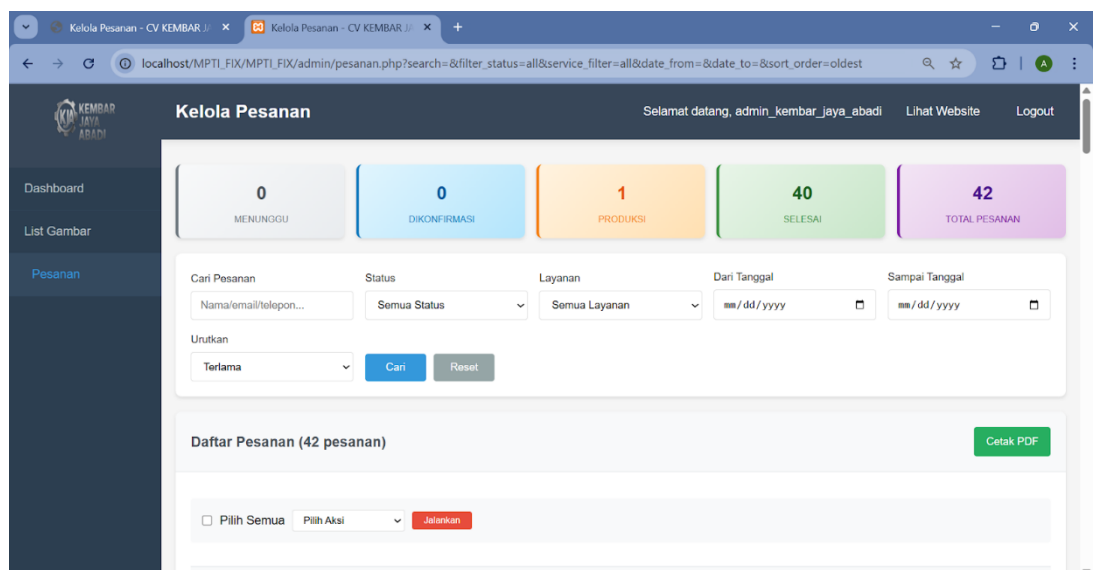
54.                                     <td>
55.                                     <span
56.                                         class="statu
s-badge <?php echo getStatusBadgeClass($order['status']); ?>">
57.                                     <?php echo
getStatusText($order['status']); ?>
58.                                     </span>
59.                                     </td>
60.                                 </tr>
61.                                 <?php endforeach; ?>
62.                             </tbody>
63.                         </table>
64.                     </div>
65. // Auto refresh setiap 2 menit untuk update data dashboard
66.     setInterval(function() {
67.         window.location.reload();
68.     }, 120000);

```

Kode Program 3. 21 Daftar Pesanan Pelanggan

c. Kelola Pesanan Pelanggan

Pada gambar 3.47 merupakan tampilan kelola pesanan pada dashboard admin yang mendukung admin untuk mengelola daftar pesanan seperti, terdapat fitur filter dan pencarian pesanan pelanggan. Fitur filter yang tersedia yaitu antara lain, melakukan filter untuk pesanan berdasarkan status produksi, melakukan filter untuk pesanan berdasarkan layanan, melakukan filter untuk mencari pesanan berdasarkan rentang waktu pesanan pelanggan, dan fitur untuk membuat list pesanan menjadi antara dari pesanan terbaru atau terlama. Fitur *search* atau cari mendukung admin untuk melakukan pencarian berdasarkan nama, email, atau nomor telepon pelanggan.



Gambar 3.47 Halaman Kelola Pesanan Dashboard Admin

Kode program 3.22 merupakan kode program yang berisi kode PHP yang menampilkan halaman kelola pesanan dashboard admin. Pada baris 18-21 kode untuk melakukan filter berdasarkan status produksi yang berfungsi untuk mengambil parameter filter status. Pada baris 126-127 merupakan kode HTML yang menampilkan dropdown opsi yang menampilkan daftar status pesanan, dan baris 7 dan 39-42 merupakan kode yang merupakan parameter untuk mengambil filter layanan dan melakukan filter berdasarkan layanan, dan baris 139-147 merupakan kode HTML yang menampilkan dropdown opsi pilihan yang menampilkan daftar layanan pesanan. Baris 5-6 dan 29-36 merupakan parameter untuk membuat filter tanggal 'dari-ke' sesuai keinginan untuk difilter. Dan filter untuk mengurutkan daftar dari terbaru sampai terlama terdapat pada baris 48-56, dan pada baris 245-248. Fitur search atau cari terdapat pada baris 3 dan 23-27 berfungsi untuk mengambil parameter pencarian.

```

1. <?php
2. // Handle search and filters
3. $search = $_GET['search'] ?? '';
4. $status_filter = $_GET['filter_status'] ?? 'all';
5. $date_from = $_GET['date_from'] ?? '';
6. $date_to = $_GET['date_to'] ?? '';
7. $service_filter = $_GET['service_filter'] ?? 'all';
8. $sort_order = $_GET['sort_order'] ?? 'newest';
9.
10. $database = new Database();
11. $conn = $database->getConnection();
12.
13. try {
14.     // Build WHERE clause
15.     $whereConditions = [];
16.     $params = [];
17.
18.     if ($status_filter !== 'all') {
19.         $whereConditions[] = "status = ?";
20.         $params[] = $status_filter;
21.     }
22.
23.     if (!empty($search)) {
24.         $whereConditions[] = "(name LIKE ? OR email LIKE ? OR
phone LIKE ? OR message LIKE ?)";
25.         $searchTerm = "%{$search}%";
26.         $params = array_merge($params, [$searchTerm,
$searchTerm, $searchTerm, $searchTerm]);
27.     }
28.
29.     if (!empty($date_from)) {
30.         $whereConditions[] = "DATE(created_at) >= ?";
31.         $params[] = $date_from;
32.     }
33.
34.     if (!empty($date_to)) {

```



```

35.         $whereConditions[] = "DATE(created_at) <= ?";
36.         $params[] = $date_to;
37.     }
38.
39.     if ($service_filter !== 'all') {
40.         $whereConditions[] = "service_type = ?";
41.         $params[] = $service_filter;
42.     }
43.
44.     $whereClause = !empty($whereConditions) ? 'WHERE ' .
implode(' AND ', $whereConditions) : '';
45.
46.     // Get orders with sorting
47.     $orderClause = '';
48.     switch($sort_order) {
49.         case 'oldest':
50.             $orderClause = 'ORDER BY created_at ASC';
51.             break;
52.         case 'newest':
53.             default:
54.             $orderClause = 'ORDER BY created_at DESC';
55.             break;
56.     }
57.
58.     $query = "SELECT * FROM pesanan {$whereClause}
{$orderClause}";
59.     $stmt = $conn->prepare($query);
60.     $stmt->execute($params);
61.     $pesanan = $stmt->fetchAll(PDO::FETCH_ASSOC);
62.
63.     // Get order counts for each status
64.     $countQuery = "SELECT status, COUNT(*) as count FROM pesanan
GROUP BY status";
65.     $countStmt = $conn->prepare($countQuery);
66.     $countStmt->execute();
67.     $statusCounts = $countStmt->fetchAll(PDO::FETCH_KEY_PAIR);
68.
69.     // Get service counts
70.     $serviceQuery = "SELECT service_type, COUNT(*) as count FROM
pesanan GROUP BY service_type";
71.     $serviceStmt = $conn->prepare($serviceQuery);
72.     $serviceStmt->execute();
73.     $serviceCounts = $serviceStmt->
>fetchAll(PDO::FETCH_KEY_PAIR);
74.
75. } catch(PDOException $e) {
76.     $pesanan = [];
77.     $error = 'Gagal mengambil data pesanan: ' . $e-
>getMessage();
78. }
79. ?>
80.
81. <!DOCTYPE html>
82. <html lang="id">

```

```

83. <head>
84.     <meta charset="UTF-8">
85.     <meta name="viewport" content="width=device-width, initial-
      scale=1.0">
86.     <title>Kelola Pesanan - Filter dan Pencarian</title>
87.     <link rel="stylesheet" href="css/orders-style.css">
88. </head>
89. <body>
90.     <div class="container">
91.         <h1>Kelola Pesanan</h1>
92.
93.         <!-- Statistics Cards -->
94.         <div class="stats-cards">
95.             <div class="stat-card">
96.                 <div class="stat-number"><?php echo
          $statusCounts['pending'] ?? 0; ?></div>
97.                 <div class="stat-label">Menunggu</div>
98.             </div>
99.             <div class="stat-card">
100.                 <div class="stat-number"><?php echo
          $statusCounts['confirmed'] ?? 0; ?></div>
101.                 <div class="stat-label">Dikonfirmasi</div>
102.             </div>
103.             <div class="stat-card">
104.                 <div class="stat-number"><?php echo
          $statusCounts['in_progress'] ?? 0; ?></div>
105.                 <div class="stat-label">Produksi</div>
106.             </div>
107.             <div class="stat-card">
108.                 <div class="stat-number"><?php echo
          $statusCounts['completed'] ?? 0; ?></div>
109.                 <div class="stat-label">Selesai</div>
110.             </div>
111.             <div class="stat-card">
112.                 <div class="stat-number"><?php echo
          array_sum($statusCounts); ?></div>
113.                 <div class="stat-label">Total Pesanan</div>
114.             </div>
115.         </div>
116.
117.         <!-- Search and Filter Form -->
118.         <div class="search-filters">
119.             <form method="GET" action="" class="search-form">
120.                 <div class="form-group">
121.                     <label for="search">Cari Pesanan</label>
122.                     <input type="text" id="search"
          name="search"
123.                     value="<?php echo
          htmlspecialchars($search); ?>"
124.                     placeholder="Nama/email/telepon...">
125.                 </div>
126.
127.                 <div class="form-group">
128.                     <label for="filter_status">Status</label>

```

```

129.             <select id="filter_status"
                name="filter_status">
130.                 <option value="all" <?php echo
                    $status_filter === 'all' ? 'selected' : ''; ?>>Semua
                    Status</option>
131.                 <option value="pending" <?php echo
                    $status_filter === 'pending' ? 'selected' : '';
                    ?>>Menunggu</option>
132.                 <option value="confirmed" <?php echo
                    $status_filter === 'confirmed' ? 'selected' : '';
                    ?>>Dikonfirmasi</option>
133.                 <option value="in_progress" <?php
                    echo $status_filter === 'in_progress' ? 'selected' : '';
                    ?>>Produksi</option>
134.                 <option value="completed" <?php echo
                    $status_filter === 'completed' ? 'selected' : '';
                    ?>>Selesai</option>
135.                 <option value="cancelled" <?php echo
                    $status_filter === 'cancelled' ? 'selected' : '';
                    ?>>Dibatalkan</option>
136.             </select>
137.         </div>
138.
139.         <div class="form-group">
140.             <label
                for="service_filter">Layanan</label>
141.             <select id="service_filter"
                name="service_filter">
142.                 <option value="all" <?php echo
                    $service_filter === 'all' ? 'selected' : ''; ?>>Semua
                    Layanan</option>
143.                 <option value="Casting (Pengecoran)"
                    <?php echo $service_filter === 'Casting (Pengecoran)' ?
                    'selected' : ''; ?>>Casting (Pengecoran)</option>
144.                 <option value="Turning (Bubut)" <?php
                    echo $service_filter === 'Turning (Bubut)' ? 'selected' : '';
                    ?>>Turning (Bubut)</option>
145.                 <option value="Drilling (Pengeboran)"
                    <?php echo $service_filter === 'Drilling (Pengeboran)' ?
                    'selected' : ''; ?>>Drilling (Pengeboran)</option>
146.             </select>
147.         </div>
148.
149.         <div class="form-group">
150.             <label for="date_from">Dari
                Tanggal</label>
151.             <input type="date" id="date_from"
                name="date_from"
152.                 value="<?php echo
                    htmlspecialchars($date_from); ?>">
153.         </div>
154.
155.         <div class="form-group">
156.             <label for="date_to">Sampai
                Tanggal</label>

```

```

157.                <input type="date" id="date_to"
    name="date_to"
158.                value="<?php echo
    htmlspecialchars($date_to); ?>"
159.            </div>
160.
161.            <div class="form-group">
162.                <label for="sort_order">Urutkan</label>
163.                <select id="sort_order"
    name="sort_order">
164.                    <option value="newest" <?php echo
    $sort_order === 'newest' ? 'selected' : ''; ?>>Terbaru</option>
165.                    <option value="oldest" <?php echo
    $sort_order === 'oldest' ? 'selected' : ''; ?>>Terlama</option>
166.                </select>
167.            </div>
168.
169.            <div class="form-group">
170.                <div class="form-buttons">
171.                    <button type="submit" class="search-
    btn">Cari</button>
172.                    <a href="?" class="clear-
    btn">Reset</a>
173.                </div>
174.            </div>
175.        </form>
176.    </div>
177.
178.    <!-- Orders Table -->
179.    <div class="card">
180.        <div class="card-header">
181.            <h3>Daftar Pesanan (<?php echo
    count($pesanan); ?> pesanan)</h3>
182.            <a href="export_orders.php<?php echo
    !empty($_SERVER['QUERY_STRING']) ? '?' : '' .
    $_SERVER['QUERY_STRING'] : ''; ?>"
183.                class="export-btn">Cetak PDF</a>
184.        </div>
185.        <div class="card-body">
186.            <?php if (!empty($pesanan)): ?>
187.                <div class="table-responsive">
188.                    <table class="table">
189.                        <thead>
190.                            <tr>
191.                                <th>ID</th>
192.                                <th>Pelanggan</th>
193.                                <th>Layanan</th>
194.                                <th>Jenis Logam</th>
195.                                <th>Jumlah</th>
196.                                <th>Tanggal</th>
197.                                <th>Status</th>
198.                            </tr>
199.                        </thead>
200.                        <tbody>

```

```

201.                                <?php foreach ($pesanan as
                                $order): ?>
202.                                <tr>
203.                                    <td><strong>#<?php echo
                                str_pad($order['id'], 4, '0', STR_PAD_LEFT); ?></strong></td>
204.                                    <td>
205.                                        <strong><?php echo
                                htmlspecialchars($order['name']); ?></strong><br>
206.                                        <small><?php echo
                                htmlspecialchars($order['email']); ?></small><br>
207.                                        <small><?php echo
                                htmlspecialchars($order['phone']); ?></small>
208.                                    </td>
209.                                    <td><?php echo
                                htmlspecialchars($order['service_type']); ?></td>
210.                                    <td><?php echo
                                htmlspecialchars($order['metal_type'] ?? '-'); ?></td>
211.                                    <td><?php echo
                                htmlspecialchars($order['quantity'] ?? '-'); ?></td>
212.                                    <td><?php echo date('d/m/Y
                                H:i', strtotime($order['created_at'])); ?></td>
213.                                    <td>
214.                                        <span class="status-badge
                                status-<?php echo $order['status']; ?>">
215.                                            <?php
216.                                            $statusText = [
217.                                                'pending' =>
218.                                                    'Menunggu',
219.                                                'confirmed' =>
220.                                                    'Dikonfirmasi',
221.                                                'in_progress' =>
222.                                                    'Produksi',
223.                                                'completed' =>
224.                                                    'Selesai',
225.                                                'cancelled' =>
226.                                                    'Dibatalkan'
227.                                            ];
228.                                            echo
229.                                            $statusText[$order['status']] ?? 'Menunggu';
230.                                            ?>
231.                                        </span>
232.                                    </td>
233.                                </tr>
234.                                <?php endforeach; ?>
235.                            </tbody>
236.                        </table>
237.                    </div>
238.                    <?php else: ?>
239.                    <div class="no-data">
240.                        <p>Tidak ada pesanan ditemukan.</p>
241.                        <?php if (!empty($search) ||
242.                            $status_filter !== 'all' || !empty($date_from) ||
243.                            !empty($date_to) || $service_filter !== 'all'): ?>
244.                            <a href="#" class="btn btn-primary">Reset
245.                            Filter</a>

```

```

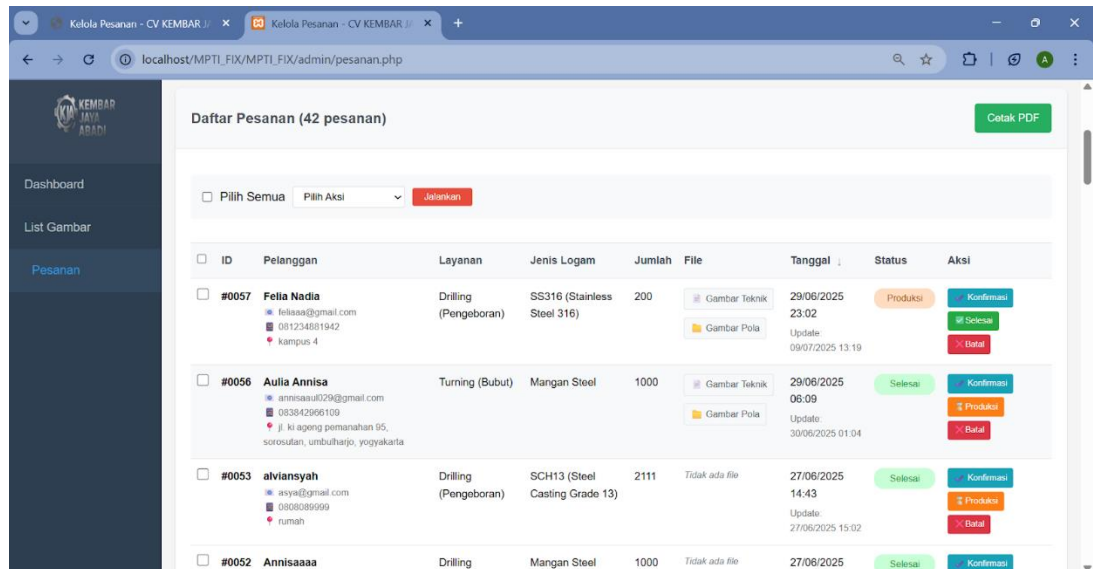
237.             <?php endif; ?>
238.         </div>
239.     <?php endif; ?>
240. </div>
241. </div>
242. </div>
243.
244.     <script>
245.         // Auto submit form ketika sort order berubah
246.         document.getElementById('sort_order').addEventListener
247.         r('change', function() {
248.             this.form.submit();
249.         });
250.         // Prevent form submission on Enter in search fields
251.         dan submit otomatis
252.         document.querySelectorAll('.search-form
253.         input').forEach(input => {
254.             input.addEventListener('keydown', function(e) {
255.                 if (e.key === 'Enter') {
256.                     e.preventDefault();
257.                     document.querySelector('.search-
258.                     btn').click();
259.                 }
260.             });
261.         });
262.     </script>
263. </body>
264. </html>

```

Kode Program 3. 22 Filter Search Bar Dashboard Admin

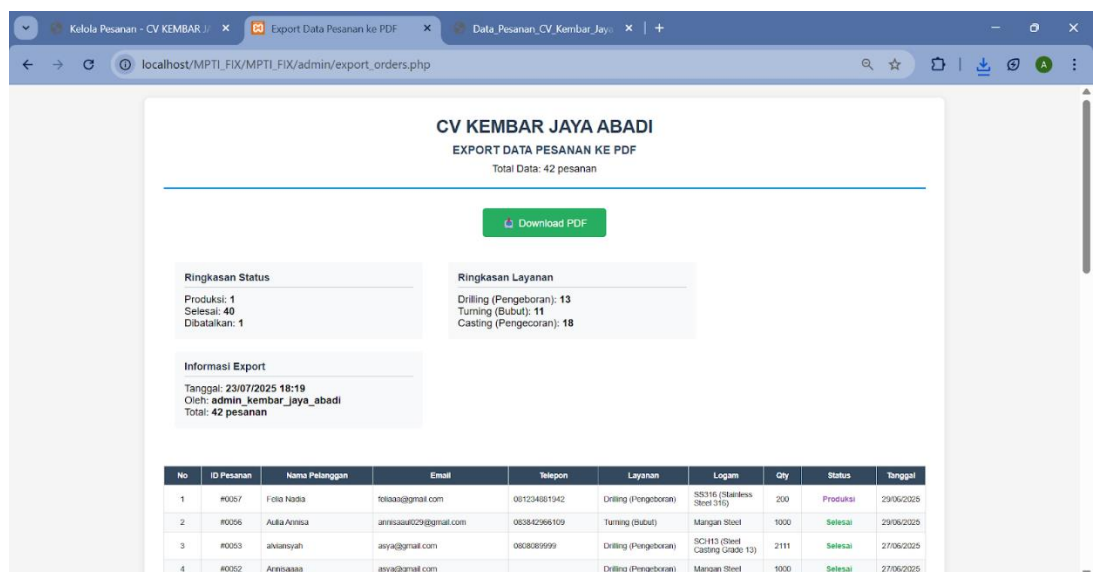
d. Daftar Pesanan

Pada gambar 3.48 merupakan tampilan pada dashboard admin yang menampilkan daftar pesanan pelanggan yang berisi detail pesanan pelanggan, serta dengan file gambar pola dan gambar teknik sesuai dengan permintaan yang diinginkan oleh pelanggan, kemudian detail tanggal serta tanggal dan jam pesanan dibuat oleh pelanggan dan juga kapan pesanan diperbarui oleh admin. Pada halaman ini, admin juga dapat melakukan aksi untuk mengubah status pesanan pelanggan, halaman ini juga disertai dengan *checkbox* untuk memudahkan admin ketika melakukan perubahan data dalam jumlah banyak, yang memungkinkan admin melakukan perubahan dalam satu kali klik. Fitur tambahan disini juga admin bisa untuk membuat file PDF berisi pesanan pelanggan dengan cara 'Cetak PDF', isi detail yang akan dicetak adalah sesuai dengan daftar yang terlihat pada halaman, apabila admin menggunakan filter maka file yang dicetak akan berisi data yang masuk ke dalam filter tersebut.



Gambar 3.48 Halaman Kelola Daftar Pesanan Admin

Gambar 3.49 merupakan contoh tampilan cetak PDF laporan, ketika tombol cetak PDF di klik maka akan dialihkan ke halaman HTML untuk dapat di review dahulu sebelum di download file PDF. Ketika tombol download PDF di klik maka file PDF akan otomatis di download dan tersimpan pada device admin.



Gambar 3.49 Tampilan Cetak Pdf Laporan

3.3. Penjamin Kualitas Proyek

Tim pengembang telah merancang dan melaksanakan serangkaian pengujian sistem untuk menjamin kualitas Sistem Informasi CV Kembar Jaya Abadi. Pengujian ini bertujuan untuk memastikan

semua fungsionalitas situs web publik dan Dashboard Admin berjalan secara optimal dan memenuhi standar kualitas yang diharapkan. Metode pengujian yang digunakan meliputi:

1. Black Box Testing

Pengujian *black box* merupakan pendekatan pengujian perangkat lunak yang berfokus pada verifikasi keluaran dan fungsionalitas aplikasi tanpa perlu mendalami detail dari kode sumber, struktur internal, atau alur proses di balik layar program tersebut. Tabel 3.2 merupakan skenario dan hasil pengujian yang telah dilakukan:

Tabel 3. 2 Hasil Pengujian Blackbox Testing

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Keterangan
1	Admin memasukkan username dan password yang valid lalu mengklik tombol login.	Sistem memvalidasi kredensial dan mengarahkan admin ke halaman Dashboard utama.	Admin berhasil login dan diarahkan ke halaman Dashboard.	Sesuai
2	Pengguna mengisi semua data pada form di halaman "Order" dan mengklik "Kirim Pesanan".	Data pesanan berhasil terkirim dan tercatat sebagai pesanan baru dengan status "Menunggu" di halaman "Kelola Pesanan" admin.	Pesanan baru berhasil masuk ke sistem admin.	Sesuai
3	Admin mengklik tombol + Tambah Produk Baru di halaman "Kelola Homepage".	Sistem menampilkan halaman formulir kosong untuk menambahkan layanan baru di homepage.	Halaman formulir tambah konten berhasil ditampilkan.	Sesuai
4	Admin mengklik tombol Edit pada salah satu konten di halaman "Kelola About Us".	Sistem menampilkan halaman formulir yang sudah terisi dengan data dari konten yang dipilih untuk diubah.	Halaman formulir edit berhasil ditampilkan dengan data yang sesuai.	Sesuai
5	Admin mengklik tombol Nonaktifkan pada salah satu layanan di halaman "Kelola Turning".	Status layanan tersebut berubah menjadi "Nonaktif" di dalam tabel.	Status berhasil diubah menjadi "Nonaktif".	Sesuai
6	Admin mengklik tombol Hapus pada item di "Kelola Galeri", lalu mengklik "OK" pada jendela konfirmasi.	Item tersebut terhapus secara permanen dari daftar dan database.	Item berhasil dihapus dari sistem.	Sesuai
7	Admin memilih "Casting" pada filter Layanan dan "Dikonfirmasi" pada filter Status di halaman "Kelola Pesanan", lalu klik Cari.	Tabel hanya menampilkan pesanan untuk layanan Casting yang berstatus "Dikonfirmasi".	Tabel berhasil memfilter data pesanan dengan benar.	Sesuai
8	Admin mengklik ikon "Produksi" pada salah satu pesanan yang berstatus "Dikonfirmasi".	Status pesanan tersebut langsung berubah menjadi "Produksi" di dalam tabel.	Status pesanan berhasil diperbarui secara instan.	Sesuai
9	Admin memilih 3 pesanan berstatus "Menunggu" menggunakan checkbox, memilih aksi "Konfirmasi" dari dropdown, lalu klik Jalankan.	Ketiga pesanan yang dipilih statusnya berubah serentak menjadi "Dikonfirmasi".	Aksi massal berhasil mengubah status beberapa pesanan sekaligus.	Sesuai
10	Admin mengklik tombol Cetak PDF di halaman "Kelola Pesanan".	Sistem menghasilkan pratinjau cetak dan memungkinkan admin mengunduh file PDF berisi daftar pesanan yang tampil di layar.	File PDF berhasil dibuat dan diunduh sesuai data yang ditampilkan.	Sesuai

2. Pengujian System Usability Scale (SUS)

Gambar 3.50 merupakan pengujian yang dilakukan dengan menggunakan Google Form sebagai media penilaian pengguna terhadap Dashboard Admin yang sudah dibuat. Hasil dari pengujian ini mendapatkan skor akhir 81 dengan 10 jawaban responden dan dihitung menggunakan alat hitung SUS, yang berarti sistem dapat diterima dengan sangat baik oleh pengguna.

Timestamp	Nama	Skor	Saya merasa mudah untuk belajar menggunakan sistem ini	Saya merasa nyaman saat menggunakan sistem ini	Saya merasa puas dengan sistem ini	Saya merasa senang dengan sistem ini	Saya merasa puas dengan sistem ini	Saya merasa senang dengan sistem ini	Saya merasa puas dengan sistem ini	Saya merasa senang dengan sistem ini
2023-08-01 14:00:00	Adi	10	5	5	5	5	5	5	5	5
2023-08-01 14:00:01	Nisa Nurfarida	10	5	5	5	5	5	5	5	5
2023-08-01 14:00:02	Adi Nurfarida	10	5	5	5	5	5	5	5	5
2023-08-01 14:00:03	Adi Nurfarida	10	5	5	5	5	5	5	5	5
2023-08-01 14:00:04	Adi Nurfarida	10	5	5	5	5	5	5	5	5
2023-08-01 14:00:05	Adi Nurfarida	10	5	5	5	5	5	5	5	5
2023-08-01 14:00:06	Adi Nurfarida	10	5	5	5	5	5	5	5	5
2023-08-01 14:00:07	Adi Nurfarida	10	5	5	5	5	5	5	5	5
2023-08-01 14:00:08	Adi Nurfarida	10	5	5	5	5	5	5	5	5
2023-08-01 14:00:09	Adi Nurfarida	10	5	5	5	5	5	5	5	5

Gambar 3.50 Daftar Responden

SUS.xlsx - Excel

FileHomeInsertPage LayoutFormulasDataReviewViewHelp

Paste

Clipboard

Calibri

11

A⁺

Font

Alignment

General

Number

Conditional Formatting

Format as Table

Cell Styles

Styles

Insert

Delete

Format

Cells

Sort & Filter

Find & Select

Editing

Share

Add-ins

Add-ins

L18

<

Gambar 3.51 Hasil Perhitungan

3.4. Keberlanjutan Proyek

Sebagai tahap akhir penyelesaian, Sistem Informasi CV Kembar Jaya Abadi yang mencakup situs web publik dan Dashboard Admin akan diserahkan kepada manajemen perusahaan, yang dalam hal ini diwakili oleh Bapak Umar Raharjo. Setelah serah terima, tim pengembang berkomitmen untuk memberikan dukungan pasca-proyek guna memastikan sistem dapat beroperasi secara efektif dan berkelanjutan. Komitmen ini merupakan bagian dari tanggung jawab tim terhadap keberhasilan jangka panjang proyek:

1. Kolaborasi Berkelanjutan:

Menjaga jalur komunikasi yang terbuka dengan staf admin CV Kembar Jaya Abadi untuk bantuan teknis, pemecahan masalah (troubleshooting), serta diskusi mengenai potensi pengembangan fitur di masa depan.

2. Pelatihan dan Pendampingan:

Memberikan sesi pelatihan serta materi panduan kepada staf admin untuk memastikan mereka mampu mengelola konten situs web dan alur kerja pesanan melalui Dashboard Admin secara mandiri dan optimal.

BAB IV

KESIMPULAN DAN SARAN

4.1. Kesimpulan

Proyek pengembangan Sistem Informasi CV Kembar Jaya Abadi berhasil diselesaikan, dengan sukses mencapai dua tujuan utamanya: membangun kehadiran digital yang profesional melalui situs web publik dan meningkatkan efisiensi operasional melalui Dashboard Admin yang terintegrasi.

Keberhasilan ini divalidasi melalui serangkaian pengujian. Uji *Black Box* membuktikan bahwa seluruh fungsionalitas utama mulai dari formulir pemesanan, manajemen konten, hingga alur kerja pesanan berjalan sesuai spesifikasi. Selain itu, uji *System Usability Scale* (SUS) pada Dashboard Admin menghasilkan skor akhir 81, menandakan bahwa sistem sangat mudah digunakan dan diterima dengan baik oleh admin.

Secara konkret, proyek ini telah menghasilkan situs web yang informatif (mencakup profil, layanan, dan galeri) dan sebuah Dashboard Admin yang komprehensif. Dashboard tersebut berfungsi sebagai pusat kendali yang efektif, memungkinkan manajemen konten yang dinamis dan pengelolaan alur kerja pesanan secara menyeluruh, lengkap dengan fitur pencarian dan filter yang canggih. Dengan demikian, sistem yang dibangun tidak hanya menjadi alat pemasaran modern tetapi juga fondasi untuk manajemen bisnis yang lebih terstruktur bagi CV Kembar Jaya Abadi.

4.2. Saran

Berdasarkan hasil pengembangan dan evaluasi proyek, berikut adalah beberapa saran untuk optimalisasi dan pengembangan sistem di masa mendatang:

1. Pengembangan Fitur Analitik dan Pelaporan.

Menambahkan modul analitik pada Dashboard Admin untuk menghasilkan laporan otomatis (misalnya, tren pesanan bulanan atau layanan terpopuler). Fitur ini akan mendukung manajemen dalam pengambilan keputusan strategis yang berbasis data.

2. Integrasi Notifikasi Real-Time.

Membangun sistem notifikasi (via email atau WhatsApp) untuk memberitahu admin secara langsung setiap kali ada pesanan baru masuk. Hal ini akan mempercepat waktu respons kepada pelanggan secara signifikan.

3. Peningkatan Interaktivitas pada Sisi Pengguna.

Mempertimbangkan penambahan fitur bernilai tambah seperti kalkulator estimasi harga untuk layanan standar atau mengubah galeri menjadi portofolio studi kasus yang lebih mendalam untuk meningkatkan kepercayaan calon klien.

4. Pemeliharaan dan Keamanan Berkala.

Melakukan pemeliharaan rutin dan audit keamanan secara terjadwal untuk memastikan sistem tetap aman dari ancaman siber, berjalan dengan performa optimal, dan selalu relevan dengan perkembangan teknologi.

LAMPIRAN

Lampiran 1. Proposal Proyek

PROPOSAL PENAWARAN PEMBUATAN WEBSITE



Dosen Pembimbing :

Guntur Maulana Zamroni, B.Sc., M.Kom.



Disusun Oleh :

Aisyah Dwi Puspita	2200018104
Felia Nadia Fikarda	2200018159
Naufal Muzakki	2200018160
Gia Sulipri	2200018171
Anisa Aulia	2200018184

PROGRAM STUDI S1 INFORMATIKA

FAKULTAS TEKNOLOGI INDUSTRI

UNIVERSITAS AHMAD DAHLAN

YOGYAKARTA

2025

Lampiran : 1 (satu) set Proposal
Perihal : Penawaran Pembuatan Website

Kepada Yth.
Direktur CV. Kembar Jaya Abadi
Di Tempat.

Dengan hormat,

Melalui surat ini, kami dari tim TechNova ingin mengajukan penawaran kerjasama untuk mengembangkan sebuah sistem informasi berbasis web yang modern dan fungsional bagi perusahaan Anda. Kami memahami bahwa kehadiran digital yang kuat bukan hanya tentang tampilan, tetapi juga tentang efisiensi operasional, dan kami siap menjadi mitra Anda untuk mewujudkan kedua hal tersebut.

Terlampir dalam surat ini adalah proposal terperinci yang mencakup tujuan, manfaat, rincian paket, dan tahapan pelaksanaan. Dalam proposal ini, kami telah merinci fitur dan fungsionalitas yang akan disertakan, yang tidak hanya mencakup situs web korporat yang dinamis, tetapi juga sistem manajemen (Dashboard Admin) yang terintegrasi untuk mengelola konten dan alur pesanan secara efisien.

Kami berkomitmen untuk menghasilkan sistem yang responsif, modern, dan mudah digunakan, serta dioptimalkan untuk meningkatkan visibilitas online dan menyederhanakan proses bisnis internal Anda.

Terima kasih atas perhatian Anda. Kami sangat antusias dan menantikan kesempatan untuk bekerja sama dengan CV Kembar Jaya Abadi.

Yogyakarta, 20 April 2025



TachNova

PENDAHULUAN

Pada era digital saat ini, kemajuan Teknologi Informasi berkembang sangat pesat. Keberadaan website telah menjadi kebutuhan krusial bagi setiap pelaku bisnis untuk meningkatkan minat calon konsumen serta membangun kepercayaan. Perusahaan yang memiliki website profesional cenderung dipandang lebih kredibel dan bonafid, memperkuat citra profesionalisme dan kepercayaan publik.

Website adalah kumpulan halaman web yang dapat diakses secara online melalui jaringan internet, menjadi sarana untuk menyimpan dan menyajikan informasi kepada publik secara luas dan cepat. Oleh karena itu, website kini menjadi salah satu media paling diandalkan dalam dunia bisnis. Tidak hanya perusahaan besar, usaha kecil dan menengah pun telah banyak memanfaatkannya sebagai media promosi dan komunikasi dengan konsumen.

Selain itu, kebutuhan akan perangkat lunak dan aplikasi berbasis web maupun desktop juga menjadi aspek penting dalam operasional sebuah perusahaan. Dengan sistem dan aplikasi yang tepat, proses kerja di dalam perusahaan dapat berjalan lebih efisien, efektif, dan terstruktur. Perusahaan yang memiliki website profesional dengan domain sendiri cenderung dipandang lebih kredibel dan bonafid. Bahkan, keberadaan website yang dicantumkan dalam media promosi seperti iklan, kop surat, dan kartu nama akan memperkuat citra profesional dan meningkatkan kepercayaan publik.

Berdasarkan hal tersebut, kami dari TechNova menawarkan kerjasama kepada CV Kembar Jaya Abadi dalam bentuk layanan pengembangan sistem informasi berbasis web yang disesuaikan dengan kebutuhan dan visi perusahaan Anda. Kami siap membantu membangun sebuah platform yang tidak hanya modern dan fungsional, tetapi juga mampu menunjang pencapaian target bisnis Anda secara optimal.

TUJUAN

Melalui pembuatan website untuk perusahaan pengecoran logam, kami bertujuan untuk:

1. Menyediakan platform digital yang merepresentasikan profil, kapabilitas, dan portofolio proyek CV Kembar Jaya Abadi di dunia maya.
2. Meningkatkan visibilitas perusahaan untuk menjangkau calon klien baru yang lebih luas.
3. Membangun citra profesional dan kepercayaan di mata klien industri melalui situs web yang informatif dan terstruktur.
4. Menyediakan Dashboard Admin untuk memusatkan dan menyederhanakan proses pengelolaan pesanan, mulai dari penerimaan hingga pelacakan status produksi.
5. Menyediakan saluran yang efektif bagi calon pelanggan untuk melakukan pemesanan dan konsultasi teknis.

MANFAAT

Manfaat yang akan diperoleh dari pembangunan website untuk CV Kembar Jaya Abadi antara lain :

1. Menyediakan informasi yang lengkap dan akurat mengenai profil serta layanan perusahaan.
2. Meningkatkan citra (branding) perusahaan dan memperluas jangkauan pemasaran kepada masyarakat luas.
3. Menyederhanakan alur kerja internal melalui Dashboard Admin, sehingga mengurangi pekerjaan manual dan meningkatkan produktivitas.

MENGAPA SISTEM INFORMASI BERBASIS WEBSITE?

Di industri manufaktur, kecepatan dan akurasi adalah kunci. Memiliki sistem informasi berbasis web memberikan dua keuntungan strategis:

1. Dari Sisi Eksternal (Website Publik):
 - a. Website memungkinkan bisnis Anda lebih dikenal oleh calon klien karena dapat diakses tanpa batas waktu dan lokasi, melampaui batasan geografis.
 - b. Website yang profesional dan informatif akan membangun kredibilitas dan kepercayaan klien terhadap kualitas produk dan layanan yang ditawarkan, yang menjadi faktor penting dalam pengambilan keputusan.
 - c. Website dapat menyajikan informasi detail mengenai profil perusahaan, jenis material, proses produksi, dan portofolio proyek secara jelas dan menarik.
2. Dari Sisi Internal (Dashboard Admin):
 - a. Menggantikan proses manual (telepon/email) dengan sistem terpusat untuk mengelola pesanan, mengurangi risiko miskomunikasi dan mempercepat respons kepada pelanggan.
 - b. Admin dapat dengan mudah memperbarui informasi layanan, galeri, atau profil perusahaan secara dinamis tanpa memerlukan bantuan teknis.
 - c. Semua data pesanan tersimpan rapi dalam satu sistem, memudahkan pelacakan, analisis, dan pembuatan laporan di masa depan.

JENIS PAKET DAN BIAYA LAYANAN

1. PAKET STANDARD

IDR 1.500.000

- a. Website Publik (Front-End):
 - 1. 7 Halaman Utama: Beranda, Tentang Kami, Layanan Casting, Layanan Turning, Layanan Drilling, Galeri, dan Pemesanan.
 - 2. Galeri Proyek dengan sistem filter berdasarkan kategori layanan.
 - 3. Formulir Pemesanan Lanjutan dengan fitur unggah file (Gambar Teknik & Pola).
- b. Sistem Manajemen (Back-End / Dashboard Admin):
 - 1. Halaman Login Admin.
 - 2. Dashboard Utama dengan ringkasan statistik pesanan (Total, Harian, Status).
 - 3. Manajemen Pesanan Lanjutan: Fitur untuk mencari, memfilter, dan mengubah status pesanan.
- c. Layanan Tambahan:
 - 1. 3x Sesi Revisi Desain.
 - 2. Gratis Pemeliharaan Teknis selama 1 Bulan setelah serah terima.

2. PAKET PREMIUM

IDR 2.250.000

- a. Website Publik (Front-End):
 - 1. Penambahan hingga 10 Halaman Utama, termasuk halaman Artikel/Blog dan Testimoni Pelanggan.
 - 2. Galeri Proyek dengan sistem filter berdasarkan kategori layanan.
 - 3. Formulir Pemesanan Lanjutan dengan fitur unggah file (Gambar Teknik & Pola).
 - 4. Integrasi langsung dengan media sosial (misalnya, menampilkan feed Instagram).
- b. Sistem Manajemen (Back-End / Dashboard Admin):

1. Modul Manajemen Blog/Artikel untuk publikasi konten secara man
 2. Modul Manajemen Testimoni untuk menampilkan ulasan pelanggan.
 3. Halaman Login Admin.
 4. Dashboard Utama dengan ringkasan statistik pesanan (Total, Harian, Status).
 5. Manajemen Pesanan Lanjutan: Fitur untuk mencari, memfilter, dan mengubah status pesanan.
- c. Layanan Tambahan:
1. 5x Sesi Revisi Desain dan Fungsionalitas.
 2. Gratis Pemeliharaan Teknis selama 3 Bulan setelah serah terima.

PEMELIHARAAN

Website merupakan etalase digital bagi bisnis pengecoran logam Anda. Oleh karena itu, penting untuk memastikan bahwa website selalu dalam kondisi terbaik agar memberikan pengalaman yang menarik dan menguntungkan bagi calon pelanggan. Namun, pemeliharaan website bukanlah tugas yang mudah. Kami memahami bahwa sebagai pemilik bisnis, waktu dan sumber daya Anda terbatas. Dengan mempercayakan pemeliharaan website kepada kami, Anda dapat fokus pada inti bisnis Anda, yaitu memberikan produk pengecoran logam terbaik kepada pelanggan. Layanan pemeliharaan kami mencakup:

1. **Pembaruan Berkala:**

Kami akan memastikan konten, perangkat lunak, dan plugin website Anda selalu up-to-date untuk menjaga performa, keamanan, dan daya tarik website.

2. **Optimasi Performa:**

Kami akan melakukan optimasi kecepatan, caching, dan kompresi gambar agar website Anda memberikan pengalaman pengguna yang cepat dan lancar.

3. Keamanan Terjamin:

Tim kami akan memantau dan menerapkan langkah-langkah keamanan terbaik untuk melindungi website dari ancaman malware, hacking, atau serangan lainnya.

4. Dukungan Teknis:

Jika terjadi masalah atau kendala teknis, tim kami siap memberikan dukungan dan penanganan cepat untuk meminimalkan gangguan pada website.

Investasi dalam pemeliharaan website adalah langkah penting untuk menjaga reputasi, keamanan, dan kelangsungan bisnis pengecoran logam Anda dalam era digital saat ini. Dengan layanan pemeliharaan kami, Anda dapat berkonsentrasi pada pertumbuhan bisnis sambil memastikan website Anda selalu berada di jalur yang tepat.

TAHAP PELAKSANAAN PEMBUATAN WEBSITE

1. Perencanaan Project

Diawali dengan proses wawancara yang bertujuan untuk mendapatkan informasi berupa gambaran secara umum dari website pengecoran logam yang akan dibuat serta data yang dibutuhkan untuk proses pembuatan website.

2. Pembuatan Prototype

Selanjutnya, pembuatan sebuah model awal atau gambaran visual dari website pengecoran logam yang akan dibuat.

3. Pembuatan Website

Proses inti dari pembuatan sebuah website, termasuk pengembangan halaman-halaman, penataan konten, dan integrasi fitur yang diperlukan untuk memenuhi kebutuhan bisnis pengecoran logam Anda.

4. Peluncuran

Tahapan akhir setelah pembuatan website selesai, di mana website dipublikasikan dan dapat diakses oleh publik, serta siap untuk digunakan oleh calon pelanggan.

5. Evaluasi

Tahapan penting yang dilakukan untuk menilai dan memastikan bahwa website yang dibuat telah memenuhi kebutuhan klien dan sesuai dengan kesepakatan bersama.

FREQUENTLY ASKED QUESTIONS

1. Bagaimana prosedur pemesanan website?

Jika kedua belah pihak sudah menyetujui, maka klien dapat melakukan proses pembayaran DP sebesar 50% dan dapat melakukan konsultasi terlebih dahulu kepada kami mengenai gambaran secara umum dari website pengecoran logam yang akan dibuat, serta memberikan data yang dibutuhkan untuk proses pembuatan.

2. Berapa lama proses pembuatan website hingga dapat ditampilkan di internet?

Dari kedua paket yang kami berikan, estimasi proses pembuatannya adalah ± 50 hari, menyesuaikan dengan tingkat kesulitan pada proses pembuatan website dan terhitung sejak proses pembayaran DP dilakukan.

3. Berapa harga perpanjangan website setelah 1 tahun?

Biaya perpanjangan website sewaktu-waktu dapat berubah tanpa pemberitahuan sesuai ketentuan dari penyedia hosting dan domain, serta tergantung jenis paket yang dipilih.

4. Apakah dapat menambahkan fitur website sewaktu-waktu?

Kami bersedia memfasilitasi jika klien ingin menambahkan fitur sesuai dengan kebutuhan dan sumber daya. Namun, terdapat penambahan biaya yang menyesuaikan dengan jenis fitur yang ditambahkan.

PENUTUP

Demikian proposal penawaran jasa pembuatan website ini kami buat. Kami sangat senang jika Bapak/Ibu berkenan untuk meluangkan waktu untuk berkonsultasi dengan kami mengenai informasi pembuatan website pengecoran logam ini, dan kami berharap dapat bekerja sama dalam mewujudkan website yang sesuai dengan kebutuhan perusahaan.

Lampiran 2. MOU

**MEMORIUM OF UNDRESTANDING (MOU)
SURAT PERJANJIAN KERJASAMA ANTARA TECHNOVA
DENGAN CV. KEMBAR JAYA ABADI**



Dosen Pembimbing :

Guntur Maulana Zamroni, B.Sc., M.Kom.



Disusun Oleh :

Aisyah Dwi Puspita	2200018104
Felia Nadia Fikarda	2200018159
Naufal Muzakki	2200018160
Gia Sulipri	2200018171
Anisa Aulia	2200018184

**PROGRAM STUDI S1 INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS AHMAD DAHLAN
YOGYAKARTA
2025**

Pada hari Sabtu tanggal 3 Mei 2024, kami yang bertanda tangan di bawah ini :

1. Nama : Umar Raharja
Jabatan : Pemilik/Direktur
No. Hp : 091226177600
Alamat : Drono, Ngawen Klaten, Jawa tengah

Selanjutnya disebut sebagai **PIHAK PERTAMA**

2. Nama : Naufal Muzakki
Jabatan : Project Manajer
No. Hp : 081226597850
Alamat : Nitikan, Sorosutan, Yogyakarta, DI Yogyakarta

Selanjutnya disebut sebagai **PIHAK KEDUA**

Kedua belah pihak telah sepakat untuk melaksanakan perjanjian dalam bentuk kerjasama. Adapun kewajiban dan tanggung jawab kedua belah pihak diatur dalam pasal-pasal di bawah ini:

PASAL 1

RUANG LINGKUP Pengerjaan

1. Dalam hal ini, PIHAK KEDUA bertindak untuk melakukan pembuatan Sistem Informasi Berbasis Web sesuai dengan rincian Paket Standard.
2. Paket Premium yang dipilih meliputi:
 - a. Website Publik (Front-End):
 1. 7 Halaman Utama: Beranda, Tentang Kami, Layanan Casting, Layanan Turning, Layanan Drilling, Galeri, dan Pemesanan.
 2. Galeri Proyek dengan sistem filter berdasarkan kategori layanan.
 3. Formulir Pemesanan Lanjutan dengan fitur unggah file (Gambar Teknik & Pola).
 - b. Sistem Manajemen (Back-End / Dashboard Admin):
 1. Halaman Login Admin.
 2. Dashboard Utama dengan ringkasan statistik pesanan (Total, Harian, Status).

3. Manajemen Pesanan Lanjutan: Fitur untuk mencari, memfilter, dan mengubah status pesanan.
- c. Layanan Tambahan:
 1. 3x Sesi Revisi Desain.
 2. Gratis Pemeliharaan Teknis selama 1 Bulan setelah serah terima.

PASAL 2

HAK DAN KEWAJIBAN PARA PIHAK

Hak **PIHAK PERTAMA** dalam perjanjian ini:

1. Mendapatkan informasi yang sebenar-benarnya mengenai website yang dikerjakan dalam perjanjian ini.
2. Mendapatkan hasil website sesuai dengan ketentuan yang telah dibicarakan oleh **PIHAK PERTAMA** dan **PIHAK KEDUA**.

Hak **PIHAK KEDUA** dalam perjanjian ini:

1. Mendapatkan upah kerja sesuai dengan nominal yang telah tercantum dalam Pasal 3 perjanjian ini.
2. Mendapatkan data-data pendukung dari **PIHAK PERTAMA** yang diperlukan dalam pembuatan website sesuai dengan permintaan awal.
3. Mendapatkan rasa aman dan bebas dari tuntutan hukum atas hasil karya yang telah dibuat sesuai dengan kesepakatan bersama.

Kewajiban **PIHAK PERTAMA** dalam perjanjian ini:

1. Melakukan pembayaran upah kerja kepada **PIHAK KEDUA** sesuai dengan nominal yang telah disebutkan dalam Pasal 3 perjanjian ini.
2. Menyerahkan data-data yang dibutuhkan sebagai isi konten website kepada **PIHAK KEDUA**, dalam bentuk apapun, selama tidak termasuk data yang bersifat pribadi atau rahasia tanpa persetujuan kedua belah pihak.

Kewajiban **PIHAK KEDUA** dalam perjanjian ini:

1. Menyelesaikan pekerjaan sesuai dengan rincian permintaan **PIHAK PERTAMA** serta dalam jangka waktu yang telah ditentukan pada Pasal 3 perjanjian ini.

PASAL 3

BIAYA DAN PEMBAYARAN

Biaya untuk pengerjaan website tersebut adalah sebagai berikut:

- Biaya jasa pengembangan sistem adalah sebesar Rp 1.500.000 (Satu Juta Lima Ratus Ribu Rupiah) untuk Paket Standard.

Biaya dibayarkan oleh **PIHAK PERTAMA** 50% dari total biaya setelah surat ini ditandatangani dan akan dilunasi setelah **PIHAK KEDUA** menyelesaikan pembuatan website dan menyerahkan website kepada **PIHAK PERTAMA**.

Pembayaran dapat dilakukan secara transfer bank:

BANK BRI

Atas Nama : Naufal Muzakki

No. Rekening : 675301002072209

PASAL 4

KETENTUAN AWAL

Ketentuan awal yang telah disepakati oleh kedua belah pihak untuk permulaan kerja sama:

1. **PIHAK KEDUA** akan membuat "website untuk perusahaan pengecoran logam dan jasa bubut" untuk **PIHAK PERTAMA**.
2. **KEDUA BELAH PIHAK** telah setuju dengan tampilan website tersebut sesuai kesepakatan melalui sarana komunikasi digital seperti Whatsapp/Media Sosial Sejenis.

3. Pembayaran awal oleh **PIHAK PERTAMA** sebesar 50% dari biaya total merupakan tanda jadi dan kesepakatan pengembangan website tersebut.
4. Pelunasan pembayaran oleh **PIHAK PERTAMA** akan dilakukan secara bersamaan dengan proses serah terima yang melibatkan website yang telah selesai dibuat.

PASAL 5

JAMINAN PENGEMBALIAN

1. Pembatalan oleh **PIHAK PERTAMA** yang dilakukan setelah perjanjian, maka uang tanda jadi tidak akan dikembalikan.

PENUTUP

Demikianlah surat perjanjian ini dibuat oleh kedua belah pihak tanpa ada paksaan dari pihak manapun. Perjanjian ini dibuat untuk menghindari terjadinya permasalahan dalam kerja sama maupun proyek yang berlangsung, serta menghindari adanya penipuan antara kedua belah pihak. Perjanjian ini berbentuk file dokumen PDF dan dipegang oleh kedua belah pihak sehingga memiliki kekuatan hukum yang mengikat antara keduanya.

PIHAK PERTAMA

CV. Kembar Jaya Abadi



Umar Raharja

PIHAK KEDUA

Tim TechNova



Naufal Muzakki

Lampiran 3. LogBook Kelompok

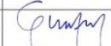
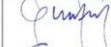
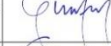
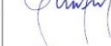
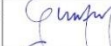
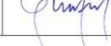
**LOG BOOK MANAJEMEN PROYEK TEKNOLOGI INFORMASI MAHASISWA
PROGRAM STUDI S1 INFORMATIKA UNIVERSITAS AHMAD DAHLAN T.A
2021/2022 (WAJIB DIISI DAN MASUK DALAM PENILAIAN)**

Nama Kelompok MPTI :		
No	NIM	NAMA
1	2200018104	Aisyah Dwi Puspita
2	2200018159	Felia Nadia Fikarda
3	2200018160	Naufal Muzakki
4	2200018171	Gia Sulipri
5	2200018184	Annisa Aulia
6		
7		
Judul Manajemen Tugas Proyek : Pembuatan Website dengan Sistem Manajemen Informasi pada Perusahaan Pengecoran Logam		
Dosen Pembimbing : Guntur Maulana Zamroni, B.Sc., M.Kom.		

Petunjuk Pengisian Log Book

1. Log book dibuat per minggu
2. Log book ditulis tangan
3. Setiap kegiatan di paraf oleh Mitra atau Client / dosen pembimbing MPTI
4. Log book per minggu diparaf oleh dosen pengampu kelas MPTI
5. Jumlah bimbingan minimal 4x Sebelum UTS

Logbook Minggu 4 sd 7 (sebelum UTS)

No	Kegiatan	Waktu Pelaksanaan		Hasil	Kendala, Rencana Perubahan (Jika ada)	Paraf MITRA/CLIENT	Paraf Dosen Pembimbing MPTI
		Hari/TGL	Durasi (Jam)				
1.	Pembuatan Proposal	19/04/2025	4 Jam	Laporan Proposal	Tidak Ada		
2.	Revisi Proposal	21/04/2025	2 Jam	Revisi Laporan Proposal	Perubahan rancangan harga		
3.	Pembuatan MOU	26/04/2025	4 Jam	Laporan MOU	Tidak Ada		
4.	Bimbingan Hasil Proposal dan MOU	Senin, 28/04/2025	2 Jam	Perubahan Proposal dan MOU	Perubahan di beberapa poin di paket dan rancangan harga serta beberapa poin di MOU		
5.	Pitching Bersama Mitra	Sabtu, 03/05/2025	1 Jam	Penyerahan Proposal & MOU	Tidak Ada		
6.	Rapat Internal	Senin, 05/05/2025	4 Jam	Pembuatan Analisis Kebutuhan	Tidak Ada		
7.	Rapat Internal	Sabtu, 10/5/2025	5 Jam	Pembuatan UI dan Implementasi Rancangan ke Kode Program	Tidak Ada		

Catatan Pembimbing Lapangan/Dosen Pembimbing MPI/Dosen Pengampu Kelas MPI :

.....

.....

.....

.....

Yogyakarta, 23 Mei 2022

Dosen Pengampu Kelas MPI

Ketua MPI


24/05/25
Guntur Maulana Zamroni, B.Sc., M.Kom.
NID: 0509038402


(Naufal Muzakki)

LOG BOOK MANAJEMEN PROYEK TEKNOLOGI INFORMASIMAHASISWA
PROGRAM STUDI S1 INFORMATIKA UNIVERSITAS AHMAD DAHLAN T.A 2024/2025
(WAJIB DIISI DAN MASUK DALAM PENILAIAN)

Nama Kelompok MPI :		
No	NIM	NAMA
1	2200018104	Aisyah Dwi Puspita
2	2200018159	Felia Nadia Fikarda
3	2200018160	Naufal Muzakki
4	2200018171	Gia Sulipri
5	2200018184	Annisa Aulia
6		
7		
Judul Manajemen Tugas Proyek : Sistem Manajemen Informasi Berbasis Web pada Perusahaan Pengecoran Logam		
Dosen Pembimbing : Guntur Maulana Zamroni, B.Sc., M.Kom.		

Petunjuk Pengisian Log Book

1. Log book dibuat per minggu
2. Log book ditulis tangan
3. Setiap kegiatan di paraf oleh Mitra Atau Client / dosen pembimbing MPI
4. Log book per minggu di paraf oleh dosen pengampu kelas MPI
5. Jumlah bimbingan minimal 7x Sebelum Daftar Seminar MPI

Logbook Minggu 9 sd 11 (setelah UTS)

No	Kegiatan	Waktu Pelaksanaan		Hasil	Kendala, Rencana Perubahan (Jika ada)	Paraf MITRA/CLIENT	Paraf Dosen Pembimbing MPI
		Hari/TGL	Jam				
1	Rapat Internal Setelah UTS	Sabtu 07/06/2025	4 Jam	Melanjutkan Pembuatan UI dan Implementasi Rancangan ke Kode Program	Tidak ada		
2	Rapat Internal	Sabtu 14/06/2025	4 Jam	Implementasi UI ke tampilan web dan Pengembangan Kode Program	Tidak ada		
3	Rapat Internal	Sabtu 21/06/2025	4 Jam	Pengembangan Kode Program dan Perbaikan Halaman Website Admin	Kendala pada saat menggabungkan kode program dari masing-		

					masing jobdesk		
4	Rapat Internal	Sabtu 28/06/2025	4 Jam	Pengembangan Program untuk isi konten dari Website	Tidak ada		
5	Rapat Internal	Sabtu 5/06/2025	4 Jam	Mulai pembuatan Laporan Akhir MPTI	Tidak ada		
6	Rapat Internal	Sabtu 12/06/2025	4 Jam	Pembuatan Laporan akhir, Poster, Video Profile, dan PPT	Tidak ada		
7	Rapat Internal	Sabtu 19/06/2025	4 Jam	Pembuatan Laporan Akhir dan Fiksasi Program	Fiksasi pada halaman list gambar homepage admin		

Catatan Pembimbing Lapangan/Dosen Pembimbing MPTI/Dosen Pengampu Kelas MPTI :

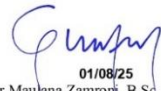
.....

.....

.....

.....

Dosen Pengampu Kelas MPTI


01/08/25
Guntur Maulana Zamroni, B.Sc., M.Kom.
NIPM. 19840309 201810 111 1205917

Yogyakarta, 31 Juli 2025

Ketua MPTI


(Naufal Muzakki)

Lampiran 4. LogBook Individu

1. Aisyah Dwi Puspita (2200018104)

LOGBOOK INDIVIDU MANAJEMEN PROYEK TEKNOLOGI INFORMASI
MAHASISWA PROGRAM STUDI S1 INFORMATIKA UNIVERSITAS AHMAD DAHLAN T.A 2024/2025
 (WAJIB DIISI DAN TERLAMPIR DALAM LAPORAN)

Nama kelompok MPTI : TechNova
Nim : 2200018104
Nama : Aisyah Dwi Puspita Asriningtyas
Job Desk : BackEnd Developer
Judul MPTI : Pembuatan Website dengan Sistem Manajemen Informasi pada perusahaan Pengcoran Logam
Dosen Pembimbing MPTI : Guntur Maulana Zamroni, B.Sc., M.Kom.

No	Kegiatan	Waktu Pelaksanaan		Hasil	Catatan/Rencana Perubahan (jika ada)
		Tanggal	Durasi (Jam)		
1.	Diskusi Perencanaan Judul Proyek	15/03/2025	4 Jam	Judul Awal	-
2.	Pengajuan Judul	17/03/2025	2 Jam	ACC Judul	-
3.	Revisi Topic	18/03/2025	2 Jam	Topik Baru	Rencana Perubahan Topik
4.	Pembuatan Proposal	19/04/2025	4 Jam	Proposal	-
5.	Revisi Proposal	21/04/2025	2 Jam	Revisi Proposal Bagian Paket	Rencana Perubahan Harga Paket
6.	Pembuatan MOU	26/04/2025	4 Jam	MOU	-
7.	Bimbingan Proposal Dan MOU	28/04/2025	2 Jam	Revisi MOU Pasal 1 Ayat 2	Rencana Perubahan MOU
8.	Pitching Bersama Mitra	03/04/2025	1 Jam	Mendapatkan Persetujuan dalam Pengajuan Proposal	-
9.	Pembuatan Dashboard Admin dan BackEnd Login	10/05/2025	5 Jam	Dashboard Admin, Database Admin, dan BackEnd Fitur Login beserta Database	-
10.	Membuat Database Pesanan, Database Homepage, Database Galeri, Menyambungkan Database, Fitur Edit, List Gambar Galeri, Homepage, dan Pesanan.	12/05/2025 - 13/05/2025	48 Jam	Database Pesanan, Database Homepage, Database Galeri, Integrasi Database, Fitur Edit, List Gambar Galeri, Homepage, dan Pesanan.	-

11.	Membuat halaman turning untuk bagian admin	23/06/2025	2 Jam	Halaman turning untuk admin mengelola konten	
12.	Membuat halaman kelola galeri untuk admin dapat mengelola gambar produk	26/06/2025	4 Jam	Admin dapat mengedit atau menghapus gambar galeri, mengatur status aktif/nonaktif	
13.	Menambahkan fitur upload dan manajemen gambar galeri services	29/06/2025	3 Jam	Fitur upload gambar dengan validasi, pengaturan urutan tampil, dan manajemen metadata gambar	
14.	Mengimplementasikan sistem toggle status dan soft delete untuk galeri	03/07/2025	2 Jam	Admin dapat mengaktifkan/menonaktifkan gambar galeri dan menghapus dengan konfirmasi	
15.	Menambahkan responsive design dan interface yang user-friendly untuk halaman galeri	04/07/2025	3 Jam	Desain responsif untuk mobile, tablet dan desktop dengan button styling yang konsisten dan navigasi yang intuitif	

Catatan Ketua Tim (Project Manager) MPTI :

Note : Untuk ketua tim di ttd oleh anggota lain dalam satu kelompok

Yogyakarta, 31 Juli 2025


Naufal Muzakki

2. Felia Nadia Fikarda (2200018159)

LOGBOOK INDIVIDU MANAJEMEN PROYEK TEKNOLOGI INFORMASI
MAHASISWA PROGRAM STUDI S1 INFORMATIKA UNIVERSITAS AHMAD DAHLAN T.A 2024/2025
 (WAJIB DIISI DAN TERLAMPIR DALAM LAPORAN)

Nama kelompok MPTI : TechNova
Nim : 2200018159
Nama : Felia Nadia Fikarda
Job Desk : FrontEnd and BackEnd Developer
Judul MPTI : Pembuatan Website dengan Sistem Manajemen Informasi pada perusahaan Pengecoran Logam
Dosen Pembimbing MPTI : Guntur Maulana Zamroni, B.Sc., M.Kom.

No	Kegiatan	Waktu Pelaksanaan		Hasil	Catatan/Rencana Perubahan (jika ada)
		Tanggal	Durasi (Jam)		
1.	Diskusi Perencanaan Judul Proyek	15/03/2025	4 Jam	Judul Awal	-
2.	Pengajuan Judul	17/03/2025	2 Jam	ACC Judul	-
3.	Revisi Topic	18/03/2025	2 Jam	Topik Baru	Rencana Perubahan Topik
4.	Pembuatan Proposal	19/04/2025	4 Jam	Proposal	-
5.	Revisi Proposal	21/04/2025	2 Jam	Revisi Proposal Bagian Paket	Rencana Perubahan Harga Paket
6.	Pembuatan MOU	26/04/2025	4 Jam	MOU	-
7.	Bimbingan Proposal Dan MOU	28/04/2025	2 Jam	Revisi MOU Pasal 1 Ayat 2	Rencana Perubahan MOU
8.	Pitching Bersama Mitra	03/04/2025	1 Jam	Mendapatkan Persetujuan dalam Pengajuan Proposal	-
9.	Pembuatan Halaman Beranda dan Tentang Kami menggunakan HTML, CSS dan Javascript	10/05/2025	5 Jam	Halaman Beranda dan Tentang Kami	-
10.	Pembuatan Halaman Casting, Turning, Drilling Galeri, dan Pesanan menggunakan HTML, CSS dan Javascript	12/05/2025 13/05/2025	48 Jam	Halaman Casting, Turning, dan Drilling	-

11.	Membuat halaman admin dan menghubungkannya ke database untuk pengelolaan data.	26/05/2025	3 Jam	Halaman admin berhasil terhubung ke database dan dapat digunakan untuk mengelola data	-
12.	Mengedit tampilan website user	01/06/2025	3 Jam	Tata letak isi konten pada website user diperbarui	-
13.	Membuat halaman home, casting, dan drilling untuk bagian admin	23/06/2025	5 Jam	Halaman home, casting dan drilling untuk admin mengelola konten berhasil dibuat	-
14.	Memperbaiki error halaman turning bagian admin	26/06/2025	3 Jam	Error berhasil diperbaiki	-
15.	Membuat halaman about us bagian admin	02/07/2025	2 Jam	Halaman edit about us untuk admin mengedit konten berhasil di buat	-
16.	Melakukan hosting dan memperbaiki bug setelah hosting	09/07/2025	2 Jam	Website user dan admin berhasil di hosting dan beberapa bug berhasil diperbaiki	-
17.	Memperbaiki halaman galeri bagian admin akibat bug hosting	13/07/2025	2 Jam	Halaman galeri bagian admin berhasil muncul	-
18.	Membuat halaman forgot password dan ubah username bagian admin	16/07/2025	4 Jam	Halaman forgot password dan halaman merubah username berhasil dibuat	-

Catatan Ketua Tim (Project Manager) MPTI :

Note : Untuk ketua tim di ttd oleh anggota lain dalam satu kelompok

Yogyakarta, 31 Juli 2025


 Naufal Muzakki

3. Naufal Muzakki (2200018160)

LOGBOOK INDIVIDU MANAJEMEN PROYEK TEKNOLOGI INFORMASI
MAHASISWA PROGRAM STUDI S1 INFORMATIKA UNIVERSITAS AHMAD DAHLAN T.A 2024/2025
(WAJIB DIISI DAN TERLAMPIR DALAM LAPORAN)

Nama kelompok MPTI : TechNova
Nim : 2200018160
Nama : Naufal Muzakki
Job Desk : Project Manager dan Designer
Judul MPTI : Pembuatan Website dengan Sistem Manajemen Informasi pada perusahaan Pengecoran Logam
Dosen Pembimbing MPTI : Guntur Maulana Zamroni, B.Sc., M.Kom.

No	Kegiatan	Waktu Pelaksanaan		Hasil	Catatan/Rencana Perubahan (jika ada)
		Tanggal	Durasi (Jam)		
1.	Diskusi Perencanaan Judul Proyek	15/03/2025	4 Jam	Judul Awal	-
2.	Pengajuan Judul	17/03/2025	2 Jam	ACC Judul	-
3.	Revisi Topic	18/03/2025	2 Jam	Topik Baru	Rencana Perubahan Topik
4.	Pembuatan Proposal	19/04/2025	4 Jam	Proposal	-
5.	Revisi Proposal	21/04/2025	2 Jam	Revisi Proposal Bagian Paket	Rencana Perubahan Harga Paket
6.	Pembuatan MOU	26/04/2025	4 Jam	MOU	-
7.	Bimbingan Proposal Dan MOU	28/04/2025	2 Jam	Revisi MOU Pasal 1 Ayat 2	Rencana Perubahan MOU
8.	Pitching Bersama Mitra	03/04/2025	1 Jam	Mendapatkan Persetujuan dalam Pengajuan Proposal	-
9.	Kebutuhan Fungsional, dan Kebutuhan Non Fungsional	05/03/2025	4 Jam	Kebutuhan Fungsional(KF), Kebutuhan Non Fungsional, dan Pembuatan UI Design Beranda	-
10.	Pembuatan UI Design, Pengembangan UI/UX	06/05/2025 - 10/05/2025	120 Jam	Pembuatan UI Design Casting, Turning, Halaman UI Dashboard Admin dan Login	-
11.	Rapat Evaluasi Progress Mingguan dengan Tim	16/06/2025	2 Jam	Catatan progress dan identifikasi hambatan	Penyesuaian tugas back-end

12.	Masukan dalam Pengembangan Kode Halaman Admin	23/06/2025 - 25/06/2025	8 Jam	Halaman Admin fungsional	-
13.	Perancangan Skenario Pengujian (Black Box)	07/07/2025	4 Jam	Dokumen skenario pengujian untuk semua fitur	-
14.	Pelaksanaan Pengujian Sistem (Black Box & Usability)	14/07/2025 - 16/07/2025	10 Jam	Daftar temuan bug dan feedback usability	-
15.	Serah Terima & Finalisasi Laporan	28/07/2025	6 Jam	Laporan akhir Proyek	-

Catatan Ketua Tim (Project Manager) MPTI :

Note : Untuk ketua tim di ttd oleh anggota lain dalam satu kelompok

Yogyakarta, 31 Juli 2025



Felia Nadia Fikarda

4. Gia Sulipri (2200018171)

LOGBOOK INDIVIDU MANAJEMEN PROYEK TEKNOLOGI INFORMASI
MAHASISWA PROGRAM STUDI S1 INFORMATIKA UNIVERSITAS AHMAD DAHLAN T.A 2024/2025
(WAJIB DIISI DAN TERLAMPIR DALAM LAPORAN)

Nama kelompok MPTI : TechNova
Nim : 2200018171
Nama : Gia Sulipri
Job Desk : System Analyst dan Designer
Judul MPTI : Pembuatan Website dengan Sistem Manajemen Informasi pada perusahaan Pengecoran Logam
Dosen Pembimbing MPTI : Guntur Maulana Zamroni, B.Sc., M.Kom.

No	Kegiatan	Waktu Pelaksanaan		Hasil	Catatan/Rencana Perubahan (jika ada)
		Tanggal	Durasi (Jam)		
1.	Diskusi Perencanaan Judul Proyek	15/03/2025	4 Jam	Judul Awal	-
2.	Pengajuan Judul	17/03/2025	2 Jam	ACC Judul	-
3.	Revisi Topic	18/03/2025	2 Jam	Topik Baru	Rencana Perubahan Topik
4.	Pembuatan Proposal	19/04/2025	4 Jam	Proposal	-
5.	Revisi Proposal	21/04/2025	2 Jam	Revisi Proposal Bagian Paket	Rencana Perubahan Harga Paket
6.	Pembuatan MOU	26/04/2025	4 Jam	MOU	-
7.	Bimbingan Proposal Dan MOU	28/04/2025	2 Jam	Revisi MOU Pasal 1 Ayat 2	Rencana Perubahan MOU
8.	Pitching Bersama Mitra	03/04/2025	1 Jam	Mendapatkan Persetujuan dalam Pengajuan Proposal	-
9.	Pembuatan Analisis Kebutuhan Website (Class Diagram, Activity Diagram, Use Case Diagram, FR & NFR, Use Case Description)	05/05/2025	4 Jam	Analisis Kebutuhan Website	-
10.	Pengembangan UI/UX	09/05/2025 - 10/05/2025	48 Jam	Pembuatan UI Halaman Tentang Kami, Galeri, Drilling, dan Pesanan	-

11.	Analisis Kebutuhan Sistem	25/05/2025	6 jam	Daftar kebutuhan fungsional dan non-fungsional dirinci	-
12.	Penyusunan Use Case, Activity Diagram, BPMN	01/06/2025	4 jam	Diagram sistem selesai dibuat	-
13.	Revisi Desain UI berdasarkan masukan mitra	20/06/2025	4 jam	Penyesuaian pada warna tombol, ukuran teks, dan posisi konten	Disesuaikan dengan preferensi mitra yang lebih sederhana
14.	Pemetaan BPMN dan Flowchart Terintegrasi (User - Sistem - Admin)	26/06/2025	3 jam	BPMN proses pemesanan, pengelolaan data & galeri selesai dibuat	Siap diimplementasikan tim developer
15.	Uji Coba Alur Navigasi di Prototipe	26/06/2025	2 jam	Simulasi penggunaan prototipe dilakukan oleh rekan kelompok	kesesuaian
16.	Konsultasi Akhir Hasil UI/UX dan Sistem Flow	29/06/2025	2 jam	Disetujui dosen PM dan mitra untuk dilanjutkan ke tahap implementasi.	-

Catatan Ketua Tim (Project Manager) MPTI :

Note : Untuk ketua tim di ttd oleh anggota lain dalam satu kelompok

Yogyakarta, 31 Juli 2025


Naufal Muzakki

5. Annisa Aulia (2200018184)

LOGBOOK INDIVIDU MANAJEMEN PROYEK TEKNOLOGI INFORMASI
MAHASISWA PROGRAM STUDI S1 INFORMATIKA UNIVERSITAS AHMAD DAHLAN T.A 2024/2025
(WAJIB DIISI DAN TERLAMPIR DALAM LAPORAN)

Nama kelompok MPTI : TechNova
Nim : 2200018184
Nama : Annisa Aulia
Job Desk : BackEnd Developer
Judul MPTI : Pembuatan Website dengan Sistem Manajemen Informasi pada perusahaan Pengecoran Logam
Dosen Pembimbing MPTI : Guntur Maulana Zamroni, B.Sc., M.Kom.

No	Kegiatan	Waktu Pelaksanaan		Hasil	Catatan/Rencana Perubahan (jika ada)
		Tanggal	Durasi (Jam)		
1.	Diskusi Perencanaan Judul Proyek	15/03/2025	4 Jam	Judul Awal	-
2.	Pengajuan Judul	17/03/2025	2 Jam	ACC Judul	-
3.	Revisi Topic	18/03/2025	2 Jam	Topik Baru	Rencana Perubahan Topik
4.	Pembuatan Proposal	19/04/2025	4 Jam	Proposal	-
5.	Revisi Proposal	21/04/2025	2 Jam	Revisi Proposal Bagian Paket	Rencana Perubahan Harga Paket
6.	Pembuatan MOU	26/04/2025	4 Jam	MOU	-
7.	Bimbingan Proposal Dan MOU	28/04/2025	2 Jam	Revisi MOU Pasal 1 Ayat 2	Rencana Perubahan MOU
8.	Pitching Bersama Mitra	03/04/2025	1 Jam	Mendapatkan Persetujuan dalam Pengajuan Proposal	-
9.	Membuat Dashboard Admin dan BackEnd Login	10/05/2025	5 Jam	Dashboard Admin, Database Admin, dan BackEnd Fitur Login beserta Database	-
10.	Membuat Database Casting, Database Turning, Database Drilling, Tambah Gambar, Hapus Gambar, List Gambar Casting, Turning, dan Drilling	12/05/2025 - 13/05/2025	48 Jam	Database Galeri, Database Casting, Database Turning, Database Drilling, Fitur Tambah Data, Hapus Data, List Gambar Casting, Turning, dan Drilling	-
11.	Membuat halaman dashboard utama admin dan menghubungkan dari database	26/06/2025	3 Jam	Halaman dashboard admin yang menampilkan statistik dan pesanan terbaru	-
12.	Membuat halaman kelola pesanan untuk admin dapat mengelola pesanan	01/06/2025	3 Jam	Admin dapat mengedit atau memperbarui status pesanan	-
13.	Menambahkan fitur filter dan cari pada halaman kelola pesanan	23/06/2025	2 Jam	Fitur cari dan filter berdasarkan status pesanan, tanggal, dll	-
14.	Menambahkan fitur cetak laporan pesanan	26/06/2025	2 Jam	Admin dapat melakukan cetak laporan pesanan	-

Catatan Ketua Tim (Project Manager) MPTI :

.....
.....
.....
.....
.....

Note : Untuk ketua tim di ttd oleh anggota lain dalam satu kelompok

Yogyakarta, 31 Juli 2025

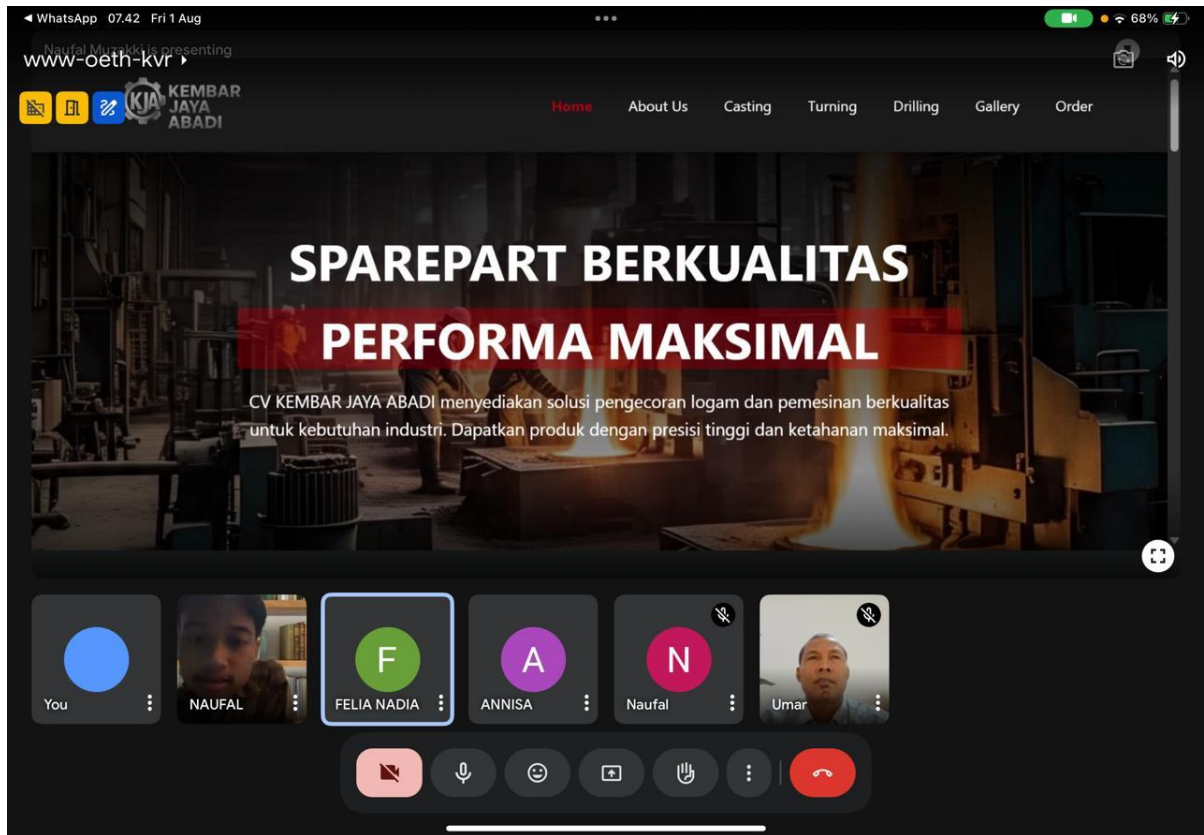

Naufal Muzakki

Lampiran 5. Dokumentasi





Lampiran 6. Bukti Serah Terima



Lampiran 7. Bukti Pembayaran

1. Pembayaran DP

11.43 📶 🕒 🗨 •

4G 📶 75% 🔋



Transfer berhasil

Rp1.000.000

26 Mei 2025 • 15:42:47 WIB •

Ref ID: 20250526154218292941

Penerima

NAUFAL MUZAKKI

BRI • 675301002072509

Sumber dana

UMAR

TAPLUS • *****353

Detail transfer

Nominal	Rp1.000.000
Biaya transaksi	Rp2.500
Metode transfer	BI-FAST
BIZ ID	20250526BNINIDJA0100 0240823401
Tujuan transaksi	Lainnya
Catatan	DP pembuatan website CV Kembar Jaya Abadi



2. Pelunasan

13.07  75**Transfer berhasil****Rp500.000**

02 Agu 2025 • 13:07:02 WIB •

Ref ID: 20250802130625740186

**Simpan
Penerima****Bukti Transaksi****Bagikan****Penerima**NAUFAL MUZAKKI
BRI • 675301002072509**Sumber dana**UMAR
TAPLUS • *****353**Kembali ke Beranda**

Lampiran 8. Source Code

1. Pemesanan.php

```
1. <?php
2. // pemesanan.php
3.
4. if (session_status() == PHP_SESSION_NONE) {
5.     session_start();
6. }
7.
8. require_once 'config/database.php';
9.
10. if (!function_exists('uploadImage')) {
11.     function uploadImage($file, $uploadDir) {
12.         $response = ['success' => false, 'filename' => null,
13.             'message' => ''];
14.         $allowedTypes = ['jpg', 'jpeg', 'png'];
15.         $maxSize = 1 * 1024 * 1024;
16.         if ($file['error'] === UPLOAD_ERR_NO_FILE) {
17.             $response['message'] = 'Tidak ada file yang diunggah.';
18.             return $response;
19.         }
20.
21.         if ($file['error'] !== UPLOAD_ERR_OK) {
22.             $response['message'] = 'Upload error: ' .
23.                 $file['error'];
24.             return $response;
25.         }
26.
27.         $fileName = basename($file['name']);
28.         $fileExt = strtolower(pathinfo($fileName,
29.             PATHINFO_EXTENSION));
30.
31.         if (!in_array($fileExt, $allowedTypes)) {
```

```

30.         $response['message'] = 'Hanya file JPG, JPEG, dan PNG
        yang diperbolehkan.';
31.         return $response;
32.     }
33.
34.     if ($file['size'] > $maxSize) {
35.         $response['message'] = 'Ukuran file melebihi 1 MB.';
36.         return $response;
37.     }
38.
39.     $newFileName = uniqid('upload_', true) . '.' . $fileExt;
40.     $targetFilePath = $uploadDir . $newFileName;
41.
42.     if (move_uploaded_file($file['tmp_name'], $targetFilePath))
    {
43.         $response['success'] = true;
44.         $response['filename'] = $newFileName;
45.     } else {
46.         $response['message'] = 'Gagal memindahkan file yang
        diunggah.';
47.     }
48.     return $response;
49. }
50. }
51.
52. $success_message = '';
53. $error_message = '';
54. $form_data = [];
55. $last_order_data = null;
56. $validation_errors = [];
57. if (isset($_SESSION['form_data'])) {
58.     $form_data = $_SESSION['form_data'];
59.     unset($_SESSION['form_data']);
60. }

```

```

61.
62. if (isset($_GET['success']) && $_GET['success'] == '1' &&
    isset($_SESSION['last_order_data'])) {
63.     $last_order_data = $_SESSION['last_order_data'];
64.     $success_message = "Harap konfirmasi pesanan Anda melalui
        WhatsApp.";
65.     unset($_SESSION['last_order_data']);
66. }
67.
68. if ($_SERVER['REQUEST_METHOD'] === 'POST') {
69.     try {
70.         $database = new Database();
71.         $conn = $database->getConnection();
72.
73.         $form_data = $_POST;
74.         $name = sanitizeInput($_POST['nama']);
75.         $email = sanitizeInput($_POST['email']);
76.         $phone = sanitizeInput($_POST['telepon']);
77.         $address = sanitizeInput($_POST['alamat']);
78.         $service_type = sanitizeInput($_POST['layanan']);
79.         $metal_type = sanitizeInput($_POST['jenis_logam'] ?? '');
80.         $quantity = $_POST['jumlah'] ?? '';
81.
82.         if (empty($email)) {
83.             $validation_errors[] = "Email wajib diisi.";
84.         } elseif (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
85.             $validation_errors[] = "Format email tidak valid.";
86.         } elseif (!preg_match('/^[a-zA-Z0-9._%+-]+@(gmail\.com|[a-
            zA-Z0-9.-]+\co\.id)$/ ', $email)) {
87.             $validation_errors[] = "Email harus menggunakan domain
                @gmail.com atau @co.id.";
88.         }
89.
90.         // Validasi Nama
91.         if (empty($name)) {

```

```

92.         $validation_errors[] = "Nama wajib diisi.";
93.     }
94.
95.     // Validasi Nomor Telepon
96.     if (empty($phone)) {
97.         $validation_errors[] = "Nomor telepon wajib diisi.";
98.     } elseif (!preg_match('/^08\d{9,11}$/', $phone)) {
99.         $validation_errors[] = "Nomor telepon harus diawali
    dengan 08 dan terdiri dari 11-13 digit.";
100.    }
101.
102.    // Validasi Alamat
103.    if (empty($address)) {
104.        $validation_errors[] = "Alamat wajib diisi.";
105.    }
106.
107.    // Validasi Layanan
108.    if (empty($service_type)) {
109.        $validation_errors[] = "Layanan wajib dipilih.";
110.    }
111.
112.    // Validasi Jumlah
113.    if (empty($quantity)) {
114.        $validation_errors[] = "Jumlah wajib diisi.";
115.    } elseif (!is_numeric($quantity) || intval($quantity) <=
    0) {
116.        $validation_errors[] = "Jumlah harus berupa angka
    positif.";
117.    } else {
118.        $quantity = intval($quantity);
119.    }
120.
121.    // Validasi File Upload (Gambar Teknik)
122.    if (!isset($_FILES['gambar_teknik']) ||
    $_FILES['gambar_teknik']['error'] === UPLOAD_ERR_NO_FILE) {

```

```

123.         $validation_errors[] = "Gambar Teknik wajib
        diunggah.";
124.     } else {
125.         // Panggil uploadImage untuk validasi dan penanganan
        upload
126.         $uploadResultGambarTeknik =
        uploadImage($_FILES['gambar_teknik'], 'uploads/orders/');
127.         if (!$uploadResultGambarTeknik['success']) {
128.             $validation_errors[] = "Upload Gambar Teknik: "
        . $uploadResultGambarTeknik['message'];
129.         } else {
130.             $gambar_teknik =
        $uploadResultGambarTeknik['filename'];
131.         }
132.     }
133.
134.     if (!isset($_FILES['pola']) || $_FILES['pola']['error']
    === UPLOAD_ERR_NO_FILE) {
135.         $validation_errors[] = "Pola wajib diunggah.";
136.     } else {
137.
138.         $uploadResultPola = uploadImage($_FILES['pola'],
        'uploads/orders/');
139.         if (!$uploadResultPola['success']) {
140.             $validation_errors[] = "Upload Pola: " .
        $uploadResultPola['message'];
141.         } else {
142.             $pattern_file = $uploadResultPola['filename'];
143.         }
144.     }
145.
146.     if (!empty($validation_errors)) {
147.         $error_message = implode('<br>',
        $validation_errors);
148.         $_SESSION['form data'] = $form data;

```

```

149.             $uploadDir = 'uploads/orders/';
150.             if (isset($gambar_teknik) && file_exists($uploadDir
151.                 . $gambar_teknik)) {
152.                 unlink($uploadDir . $gambar_teknik);
153.             }
154.             if (isset($pattern_file) && file_exists($uploadDir .
155.                 $pattern_file)) {
156.                 unlink($uploadDir . $pattern_file);
157.             }
158.             } else {
159.
160.                 $query = "INSERT INTO pesanan (name, email, phone,
161.                 address, service_type, metal_type, quantity, gambar_teknik,
162.                 gambar_pola)
163.
164.                 VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?)";
165.
166.                 $stmt = $conn->prepare($query);
167.                 $result = $stmt->execute([
168.                     $name, $email, $phone, $address, $service_type,
169.                     $metal_type, $quantity, $gambar_teknik,
170.                     $pattern_file
171.                 ]);
172.
173.                 if ($result) {
174.                     $order_id = $conn->lastInsertId();
175.
176.                     $_SESSION['last_order_data'] = [
177.                         'order_id' => $order_id,
178.                         'name' => $name,
179.                         'email' => $email,
180.                         'phone' => $phone,
181.                         'address' => $address,
182.                         'service_type' => $service_type,
183.                         'metal_type' => $metal_type,
184.                         'quantity' => $quantity,

```

```

179.             'gambar_teknik' => $gambar_teknik,
180.             'pattern_file' => $pattern_file,
181.             'order_date' => date('Y-m-d H:i:s')
182.         ];
183.
184.         session_write_close();
185.
186.         header("Location: " . $_SERVER['PHP_SELF'] .
187.             "?success=1");
188.         exit();
189.     } else {
190.         $error_message = "Gagal menyimpan pesanan ke
191.             database.";
192.
193.         $uploadDir = 'uploads/orders/';
194.         if (isset($gambar_teknik) &&
195.             file_exists($uploadDir . $gambar_teknik)) {
196.             unlink($uploadDir . $gambar_teknik);
197.         }
198.         if (isset($pattern_file) &&
199.             file_exists($uploadDir . $pattern_file)) {
200.             unlink($uploadDir . $pattern_file);
201.         }
202.     }
203. }
204.
205. } catch (PDOException $e) {
206.     $error_message = "Terjadi kesalahan database: " . $e-
207.         >getMessage();
208. } catch (Exception $e) {
209.     $error_message = "Terjadi kesalahan: " . $e-
210.         >getMessage();
211. }
212. }
213. }
214. }
215. }
216. }
217. }
218. }
219. }
220. }
221. }
222. }
223. }
224. }
225. }
226. }
227. }
228. }
229. }
230. }
231. }
232. }
233. }
234. }
235. }
236. }
237. }
238. }
239. }
240. }
241. }
242. }
243. }
244. }
245. }
246. }
247. }
248. }
249. }
250. }
251. }
252. }
253. }
254. }
255. }
256. }
257. }
258. }
259. }
260. }
261. }
262. }
263. }
264. }
265. }
266. }
267. }
268. }
269. }
270. }
271. }
272. }
273. }
274. }
275. }
276. }
277. }
278. }
279. }
280. }
281. }
282. }
283. }
284. }
285. }
286. }
287. }
288. }
289. }
290. }
291. }
292. }
293. }
294. }
295. }
296. }
297. }
298. }
299. }
300. }
301. }
302. }
303. }
304. }
305. }
306. }
307. }
308. }
309. }
310. }
311. }
312. }
313. }
314. }
315. }
316. }
317. }
318. }
319. }
320. }
321. }
322. }
323. }
324. }
325. }
326. }
327. }
328. }
329. }
330. }
331. }
332. }
333. }
334. }
335. }
336. }
337. }
338. }
339. }
340. }
341. }
342. }
343. }
344. }
345. }
346. }
347. }
348. }
349. }
350. }
351. }
352. }
353. }
354. }
355. }
356. }
357. }
358. }
359. }
360. }
361. }
362. }
363. }
364. }
365. }
366. }
367. }
368. }
369. }
370. }
371. }
372. }
373. }
374. }
375. }
376. }
377. }
378. }
379. }
380. }
381. }
382. }
383. }
384. }
385. }
386. }
387. }
388. }
389. }
390. }
391. }
392. }
393. }
394. }
395. }
396. }
397. }
398. }
399. }
400. }
401. }
402. }
403. }
404. }
405. }
406. }
407. }
408. }
409. }
410. }
411. }
412. }
413. }
414. }
415. }
416. }
417. }
418. }
419. }
420. }
421. }
422. }
423. }
424. }
425. }
426. }
427. }
428. }
429. }
430. }
431. }
432. }
433. }
434. }
435. }
436. }
437. }
438. }
439. }
440. }
441. }
442. }
443. }
444. }
445. }
446. }
447. }
448. }
449. }
450. }
451. }
452. }
453. }
454. }
455. }
456. }
457. }
458. }
459. }
460. }
461. }
462. }
463. }
464. }
465. }
466. }
467. }
468. }
469. }
470. }
471. }
472. }
473. }
474. }
475. }
476. }
477. }
478. }
479. }
480. }
481. }
482. }
483. }
484. }
485. }
486. }
487. }
488. }
489. }
490. }
491. }
492. }
493. }
494. }
495. }
496. }
497. }
498. }
499. }
500. }
501. }
502. }
503. }
504. }
505. }
506. }
507. }
508. }
509. }
510. }
511. }
512. }
513. }
514. }
515. }
516. }
517. }
518. }
519. }
520. }
521. }
522. }
523. }
524. }
525. }
526. }
527. }
528. }
529. }
530. }
531. }
532. }
533. }
534. }
535. }
536. }
537. }
538. }
539. }
540. }
541. }
542. }
543. }
544. }
545. }
546. }
547. }
548. }
549. }
550. }
551. }
552. }
553. }
554. }
555. }
556. }
557. }
558. }
559. }
560. }
561. }
562. }
563. }
564. }
565. }
566. }
567. }
568. }
569. }
570. }
571. }
572. }
573. }
574. }
575. }
576. }
577. }
578. }
579. }
580. }
581. }
582. }
583. }
584. }
585. }
586. }
587. }
588. }
589. }
590. }
591. }
592. }
593. }
594. }
595. }
596. }
597. }
598. }
599. }
600. }
601. }
602. }
603. }
604. }
605. }
606. }
607. }
608. }
609. }
610. }
611. }
612. }
613. }
614. }
615. }
616. }
617. }
618. }
619. }
620. }
621. }
622. }
623. }
624. }
625. }
626. }
627. }
628. }
629. }
630. }
631. }
632. }
633. }
634. }
635. }
636. }
637. }
638. }
639. }
640. }
641. }
642. }
643. }
644. }
645. }
646. }
647. }
648. }
649. }
650. }
651. }
652. }
653. }
654. }
655. }
656. }
657. }
658. }
659. }
660. }
661. }
662. }
663. }
664. }
665. }
666. }
667. }
668. }
669. }
670. }
671. }
672. }
673. }
674. }
675. }
676. }
677. }
678. }
679. }
680. }
681. }
682. }
683. }
684. }
685. }
686. }
687. }
688. }
689. }
690. }
691. }
692. }
693. }
694. }
695. }
696. }
697. }
698. }
699. }
700. }
701. }
702. }
703. }
704. }
705. }
706. }
707. }
708. }
709. }
710. }
711. }
712. }
713. }
714. }
715. }
716. }
717. }
718. }
719. }
720. }
721. }
722. }
723. }
724. }
725. }
726. }
727. }
728. }
729. }
730. }
731. }
732. }
733. }
734. }
735. }
736. }
737. }
738. }
739. }
740. }
741. }
742. }
743. }
744. }
745. }
746. }
747. }
748. }
749. }
750. }
751. }
752. }
753. }
754. }
755. }
756. }
757. }
758. }
759. }
760. }
761. }
762. }
763. }
764. }
765. }
766. }
767. }
768. }
769. }
770. }
771. }
772. }
773. }
774. }
775. }
776. }
777. }
778. }
779. }
780. }
781. }
782. }
783. }
784. }
785. }
786. }
787. }
788. }
789. }
790. }
791. }
792. }
793. }
794. }
795. }
796. }
797. }
798. }
799. }
800. }
801. }
802. }
803. }
804. }
805. }
806. }
807. }
808. }
809. }
810. }
811. }
812. }
813. }
814. }
815. }
816. }
817. }
818. }
819. }
820. }
821. }
822. }
823. }
824. }
825. }
826. }
827. }
828. }
829. }
830. }
831. }
832. }
833. }
834. }
835. }
836. }
837. }
838. }
839. }
840. }
841. }
842. }
843. }
844. }
845. }
846. }
847. }
848. }
849. }
850. }
851. }
852. }
853. }
854. }
855. }
856. }
857. }
858. }
859. }
860. }
861. }
862. }
863. }
864. }
865. }
866. }
867. }
868. }
869. }
870. }
871. }
872. }
873. }
874. }
875. }
876. }
877. }
878. }
879. }
880. }
881. }
882. }
883. }
884. }
885. }
886. }
887. }
888. }
889. }
890. }
891. }
892. }
893. }
894. }
895. }
896. }
897. }
898. }
899. }
900. }
901. }
902. }
903. }
904. }
905. }
906. }
907. }
908. }
909. }
910. }
911. }
912. }
913. }
914. }
915. }
916. }
917. }
918. }
919. }
920. }
921. }
922. }
923. }
924. }
925. }
926. }
927. }
928. }
929. }
930. }
931. }
932. }
933. }
934. }
935. }
936. }
937. }
938. }
939. }
940. }
941. }
942. }
943. }
944. }
945. }
946. }
947. }
948. }
949. }
950. }
951. }
952. }
953. }
954. }
955. }
956. }
957. }
958. }
959. }
960. }
961. }
962. }
963. }
964. }
965. }
966. }
967. }
968. }
969. }
970. }
971. }
972. }
973. }
974. }
975. }
976. }
977. }
978. }
979. }
980. }
981. }
982. }
983. }
984. }
985. }
986. }
987. }
988. }
989. }
990. }
991. }
992. }
993. }
994. }
995. }
996. }
997. }
998. }
999. }
1000. }

```

```
208. <!DOCTYPE html>
209. <html lang="id">
210.
211. <head>
212.     <meta charset="UTF-8" />
213.     <meta name="viewport" content="width=device-width, initial-
    scale=1.0" />
214.     <title>Pemesanan - CV KEMBAR JAYA ABADI Logam</title>
215.     <link rel="stylesheet" href="pemesanan.css" />
216.     <style>
217.
218.         .alert-success {
219.             background-color: #d4edda;
220.             color: #155724;
221.             border: 1px solid #c3e6cb;
222.             padding: 15px;
223.             margin-bottom: 20px;
224.             border-radius: 5px;
225.             display: flex;
226.             flex-direction: column;
227.             align-items: flex-start;
228.             gap: 10px;
229.         }
230.
231.         .alert-error {
232.             background-color: #f8d7da;
233.             color: #721c24;
234.             border: 1px solid #f5c6cb;
235.             padding: 15px;
236.             margin-bottom: 20px;
237.             border-radius: 5px;
238.         }
239.
240.         .whatsapp-btn {
```



```
241.         background-color: #25D366;
242.         color: white;
243.         padding: 10px 20px;
244.         border: none;
245.         border-radius: 5px;
246.         cursor: pointer;
247.         font-size: 16px;
248.         display: inline-flex;
249.         align-items: center;
250.         gap: 8px;
251.         text-decoration: none;
252.     }
253.
254.     .whatsapp-btn:hover {
255.         background-color: #1da851;
256.     }
257.
258.     .file-upload-btn {
259.         display: flex;
260.         align-items: center;
261.         gap: 8px;
262.         padding: 10px 15px;
263.         border: 1px solid #ccc;
264.         border-radius: 5px;
265.         background-color: #f9f9f9;
266.         cursor: pointer;
267.         transition: background-color 0.3s ease;
268.     }
269.
270.     .file-upload-btn:hover {
271.         background-color: #e9e9e9;
272.     }
273.
274.     .file-preview {
```

```

275.         margin-top: 10px;
276.         display: flex;
277.         align-items: center;
278.         gap: 10px;
279.         flex-wrap: wrap; /* Untuk menangani banyak file */
280.     }
281.
282.     .file-preview img {
283.         max-width: 100px;
284.         max-height: 100px;
285.         border: 1px solid #ddd;
286.         border-radius: 4px;
287.         padding: 5px;
288.     }
289. </style>
290. </head>
291.
292. <body>
293.     <header>
294.         <div class="container nav-container">
295.             <div class="nav-logo">
296.                 
298.             </div>
299.             <div class="hamburger-menu" id="hamburger-menu">
300.                 <div class="bar"></div>
301.                 <div class="bar"></div>
302.                 <div class="bar"></div>
303.             </div>
304.             <ul class="nav-links" id="nav-links">
305.                 <li><a href="index">Home</a></li>
306.                 <li><a href="tentangKami">About Us</a></li>
307.                 <li><a href="pengecoran">Casting</a></li>
308.                 <li><a href="bubut">Turning</a></li>

```

```

308.         <li><a href="drilling">Drilling</a></li>
309.         <li><a href="galeri">Gallery</a></li>
310.         <li><a href="pemesanan"
      class="active">Order</a></li>
311.     </ul>
312. </div>
313. </header>
314.
315.     <section class="hero">
316.         <div class="hero-content">
317.             <h1 class="hero-title">PEMESANAN</h1>
318.             <p class="hero-description">Isi formulir di bawah
      ini untuk melakukan pemesanan produk atau layanan
319.             pengecoran logam dan pemesinan. Tim kami akan
      segera menghubungi Anda.</p>
320.         </div>
321.     </section>
322.
323.     <section class="form-section">
324.         <div class="container">
325.             <?php if (!empty($success_message)): ?>
326.                 <div class="alert-success">
327.                     <?php echo $success_message; ?>
328.                     <?php if ($last_order_data): ?>
329.                         <button type="button" id="whatsappBtn"
      class="whatsapp-btn">
330.                             <svg
      xmlns="http://www.w3.org/2000/svg" width="16" height="16"
      fill="currentColor" class="bi bi-whatsapp" viewBox="0 0 16 16">
331.                                 <path d="M13.673 2.912A7.994
      7.994 0 0 0 8.01 0a8 8 0 0 0-8 8c0 1.487.36 2.906 1.05 4.148L.5
      16l3.924-1.298A7.98 7.98 0 0 0 8.01 16c4.417 0 8-3.582 8-8a7.99
      7.99 0 0 0-2.327-5.088zM8.01 1.5c3.58 0 6.5 2.92 6.5 6.5a6.51 6.51
      0 0 1-1.88 4.678l-.1162-1.163 3.82L8.52 14.88c-1.104.594-2.328.92-
      3.6.92A6.51 6.51 0 0 1 1.51 8a6.5 6.5 0 0 1 6.5-6.5zm-1.8 1.46c-
      .225 0-.41.077-.55.23-.14.153-.21.364-

```

```

.21.636s.07.483.21.636c.14.153.325.23.55.23h.01c.225 0 .41-.077.55-
.23.14-.153.21-.364.21-.636s-.07-.483-.21-.636c-.14-.153-.325-.23-
.55-.23zm3.6 0c-.225 0-.41.077-.55.23-.14.153-.21.364-
.21.636s.07.483.21.636c.14.153.325.23.55.23h.01c-.225 0 .41-
.077.55-.23.14-.153.21-.364.21-.636s-.07-.483-.21-.636c-.14-.153-
.325-.23-.55-.23z"/>
332.                                </svg>
333.                                Konfirmasi via WhatsApp
334.                                </button>
335.                                <?php endif; ?>
336.                                </div>
337.                                <?php endif; ?>
338.
339.                                <?php if (!empty($error_message)): ?>
340.                                <div class="alert alert-error">
341.                                <?php echo $error_message; ?>
342.                                </div>
343.                                <?php endif; ?>
344.
345.                                <form id="orderForm" action="" method="POST"
enctype="multipart/form-data" novalidate>
346.                                <div class="form-card">
347.                                <div class="form-group">
348.                                <label for="email"
class="required">Email <span class="required-field"></span></label>
349.                                <input type="email" id="email"
name="email" class="form-control" placeholder="Jawaban Anda"
350.                                value="<?php echo
isset($form_data['email']) ? htmlspecialchars($form_data['email'])
: ''; ?>" required />
351.                                <small class="form-text text-
muted">Email harus menggunakan domain @gmail.com atau
@.co.id</small>
352.                                </div>
353.                                </div>
354.

```

```

355.             <div class="form-card">
356.                 <div class="form-group">
357.                     <label for="nama" class="required">Nama
                        <span class="required-field"></span></label>
358.                     <input type="text" id="nama" name="nama"
                        class="form-control" placeholder="Jawaban Anda"
359.                         value="<?php echo
                            isset($form_data['nama']) ? htmlspecialchars($form_data['nama']) :
                            ''; ?>" required />
360.                     </div>
361.                 </div>
362.
363.             <div class="form-card">
364.                 <div class="form-group">
365.                     <label for="telepon"
                        class="required">No. Telepon <span class="required-
                        field"></span></label>
366.                     <input type="tel" id="telepon"
                        name="telepon" class="form-control" placeholder="Jawaban Anda"
367.                         pattern="08[0-9]{9,11}"
                        maxlength="13" inputmode="numeric"
368.                         value="<?php echo
                            isset($form_data['telepon']) ?
                            htmlspecialchars($form_data['telepon']) : ''; ?>" required />
369.                     <small class="form-text text-
                        muted">Nomor telepon harus diawali dengan 08 dan terdiri dari 11-13
                        digit</small>
370.                     </div>
371.                 </div>
372.
373.             <div class="form-card">
374.                 <div class="form-group">
375.                     <label for="alamat"
                        class="required">Alamat <span class="required-
                        field"></span></label>

```

```

376.             <textarea id="alamat" name="alamat"
class="form-control" placeholder="Jawaban Anda"
377.             rows="3" required><?php echo
isset($form_data['alamat']) ?
htmlspecialchars($form_data['alamat']) : ''; ?></textarea>
378.             </div>
379.         </div>
380.
381.         <div class="form-card">
382.             <div class="form-group">
383.                 <label class="required">Layanan <span
class="required-field"></span></label>
384.                 <div class="radio-group">
385.                     <div class="radio-option">
386.                         <input type="radio" id="bubut"
name="layanan" value="Turning (Bubut)" required
387.                         <?php echo
(isset($form_data['layanan']) && $form_data['layanan'] === 'Turning
(Bubut)') ? 'checked' : ''; ?> />
388.                         <label for="bubut">Turning
(Bubut)</label>
389.                     </div>
390.                     <div class="radio-option">
391.                         <input type="radio" id="casting"
name="layanan" value="Casting (Pengecoran)" required
392.                         <?php echo
(isset($form_data['layanan']) && $form_data['layanan'] === 'Casting
(Pengecoran)') ? 'checked' : ''; ?> />
393.                         <label for="casting">Casting
(Pengecoran)</label>
394.                     </div>
395.                     <div class="radio-option">
396.                         <input type="radio"
id="drilling" name="layanan" value="Drilling (Pengeboran)" required

```

[illegible]

```

421.                                     <?php echo
      (isset($form_data['jenis_logam']) && $form_data['jenis_logam'] ===
      $metal) ? 'checked' : ''; ?> />
422.                                     <label for="metal"<?php echo
      $index; ?>"><?php echo $metal; ?></label>
423.                                     </div>
424.                                     <?php endforeach; ?>
425.                                     </div>
426.                                     </div>
427.                                     </div>
428.
429.                                     <div class="form-card">
430.                                         <div class="form-group">
431.                                             <label for="jumlah"
      class="required">Jumlah <span class="required-
      field"></span></label>
432.                                             <input type="number" id="jumlah"
      name="jumlah" class="form-control" placeholder="Jawaban Anda"
      min="1"
433.                                             step="1"
434.                                             value="<?php echo
      isset($form_data['jumlah']) ?
      htmlspecialchars($form_data['jumlah']) : ''; ?>"
435.                                             required />
436.                                             <small class="form-text text-
      muted">Jumlah harus berupa angka positif</small>
437.                                         </div>
438.                                     </div>
439.
440.                                     <div class="form-card">
441.                                         <div class="form-group">
442.                                             <label for="gambar_teknik"
      class="required">Upload Gambar Teknik <span class="required-
      field"></span></label>
443.                                             <div class="file-info">Upload 1 file
      yang didukung. Maks 1 MB. (JPG, PNG)</div>

```



```

444.                <div class="file-upload">
445.                    <button type="button" class="file-
upload-btn"
446.                        onclick="document.getElementById
('gambar_teknik').click()">
447.                    <svg
xmlns="http://www.w3.org/2000/svg" width="16" height="16"
fill="currentColor"
448.                        viewBox="0 0 16 16">
449.                        <path
450.                            d="M.5 9.9a.5.5 0 0 1
.5.5v2.5a1 1 0 0 0 1 1h12a1 1 0 0 0 1-1v-2.5a.5.5 0 0 1 1 0v2.5a2 2
0 0 1-2 2H2a2 2 0 0 1-2-2v-2.5a.5.5 0 0 1 .5-.5z" />
451.                        <path
452.                            d="M7.646 1.146a.5.5 0 0
1 .708 0l3 3a.5.5 0 0 1-.708.708L8.5 2.707V11.5a.5.5 0 0 1-1
0V2.707L5.354 4.854a.5.5 0 1 1-.708-.708l3-3z" />
453.                    </svg>
454.                    <span
id="gambar_teknik_text">Tambahkan file</span>
455.                </button>
456.                <input type="file"
id="gambar_teknik" name="gambar_teknik" hidden accept="image/jpeg,
image/png" required />
457.                <div id="gambar_teknik_preview"
class="file-preview"></div>
458.            </div>
459.        </div>
460.    </div>
461.
462.        <div class="form-card">
463.            <div class="form-group">
464.                <label for="pola"
class="required">Upload Pola <span class="required-
field"></span></label>

```

```

465.                <div class="file-info">Upload 1 file
                    yang didukung. Maks 1 MB. (JPG, PNG)</div>
466.                <div class="file-upload">
467.                    <button type="button" class="file-
                        upload-btn"
468.                                onclick="document.getElementById
                            ('pola').click()">
469.                <svg
                    xmlns="http://www.w3.org/2000/svg" width="16" height="16"
                    fill="currentColor"
470.                                viewBox="0 0 16 16">
471.                    <path
472.                                d="M.5 9.9a.5.5 0 0 1
                        .5.5v2.5a1 1 0 0 0 1 1h12a1 1 0 0 0 1-1v-2.5a.5.5 0 0 1 1 1
                        0v2.5a2 2
                        0 0 1-2 2H2a2 2 0 0 1-2-2v-2.5a.5.5 0 0 1 .5-.5z" />
473.                    <path
474.                                d="M7.646 1.146a.5.5 0 0
                        1 .708 0l3 3a.5.5 0 0 1-.708.708L8.5 2.707V11.5a.5.5 0 0 1-1
                        0V2.707L5.354 4.854a.5.5 0 1 1-.708-.708l3-3z" />
475.                </svg>
476.                <span id="pola_text">Tambahkan
                    file</span>
477.                </button>
478.                <input type="file" id="pola"
                    name="pola" hidden accept="image/jpeg, image/png" required />
479.                <div id="pola_preview" class="file-
                    preview"></div>
480.                </div>
481.            </div>
482.        </div>
483.
484.        <div class="submit-button-wrapper">
485.            <button type="submit" class="submit-
                btn">Kirim Pesanan</button>
486.        </div>
487.    </form>

```

```

488.         </div>
489.     </section>
490.
491.     <footer>
492.         <div class="container">
493.             <div class="footer-container">
494.                 <div class="footer-col">
495.                     <h3>CV KEMBAR JAYA ABADI</h3>
496.                     <p>Spesialis casting, turning, dan drilling
berkualitas untuk kebutuhan industri.</p>
497.                 </div>
498.                 <div class="footer-col">
499.                     <h3>Layanan Kami</h3>
500.                     <a href="pengecoran">Casting</a>
501.                     <a href="bubut">Turning</a>
502.                     <a href="drilling">Drilling</a>
503.                 </div>
504.                 <div class="footer-col">
505.                     <h3>Kontak</h3>
506.                     <div class="contact-icon">📍</div>
507.                     <p>Jl. Tegalsari Baru Rt 39 Rw 16 Ngawonggo
Ceper Klaten</p>
508.                     <div class="contact-icon">☎️</div>
509.                     <p>Telp: 0812-2617-7600</p>
510.                     <div class="contact-icon">✉️</div>
511.                     <p>Email: umarraharjo4@gmail.com</p>
512.                 </div>
513.                 <div class="footer-col">
514.                     <h3>Jam Kerja</h3>
515.                     <p>Senin - Jumat: 08.00 - 17.00</p>
516.                     <p>Sabtu: 08.00 - 13.00</p>
517.                     <p>Minggu: Tutup</p>
518.                 </div>
519.             </div>

```

```

520.         <div class="copyright">
521.             <p>&copy; 2025 CV KEMBAR JAYA ABADI. All Rights
Reserved.</p>
522.         </div>
523.     </div>
524. </footer>
525.
526.     <script>
527.         function createWhatsAppMessage(orderData) {
528.             let message = "🔔 PESANAN BARU - CV KEMBAR JAYA
ABADI\n\n";
529.
530.             message += "📄 DETAIL PESANAN:\n";
531.             message += ID:
#`${String(orderData.order_id).padStart(4, '0')}\n;
532.             message += Nama: `${orderData.name}\n;
533.             message += Email: `${orderData.email}\n;
534.             message += Telepon: `${orderData.phone || '-'}\n;
535.             message += Alamat: `${orderData.address || '-'}\n;
536.             message += Layanan: `${orderData.service_type}\n;
537.             message += Jenis Logam: `${orderData.metal_type || '-'}\n;
538.             message += Jumlah: `${orderData.quantity || '-'}\n\n;
539.
540.             message += "\nMohon konfirmasi pesanan ini. Terima
kasih!";
541.
542.             return message;
543.         }
544.
545.         document.addEventListener('DOMContentLoaded', function()
{
546.             const lastOrderData = <?php echo
isset($last_order_data) ? json_encode($last_order_data) : 'null';
?>;

```

```

547.             const whatsappNumber = '6281226177600'; // Nomor
                WhatsApp tanpa '+'
548.
549.             function openWhatsApp(orderData) {
550.                 const message =
                    createWhatsAppMessage(orderData);
551.                 const isMobile =
                    /Android|iPhone|iPad|iPod|BlackBerry|IEMobile|Opera
                    Mini/i.test(navigator.userAgent);
552.
553.                 let whatsappUrl;
554.                 if (isMobile) {
555.                     whatsappUrl =
                        whatsapp://send?phone=${whatsappNumber}&text=${encodeURIComponent(m
                        essage)};
556.                 } else {
557.                     whatsappUrl =
                        https://web.whatsapp.com/send?phone=${whatsappNumber}&text=${encode
                       URIComponent(message)};
558.                 }
559.                 window.open(whatsappUrl, '_blank');
560.             }
561.
562.             // Client-side validation functions
563.             function validateEmail(email) {
564.                 const emailPattern = /^[a-zA-Z0-9._%+-
                    ]+@(gmail\.com|[a-zA-Z0-9.-]+\co\.id)$/;
565.                 return emailPattern.test(email);
566.             }
567.
568.             function validatePhone(phone) {
569.                 const phonePattern = /^08\d{9,11}$/; // Starts
                    with 08, 11-13 digits total
570.                 return phonePattern.test(phone);
571.             }

```

```

572.
573.         // Max file size is 1MB for consistency
574.         function validateFileSize(file, maxSizeMB = 1) {
575.             const maxSizeBytes = maxSizeMB * 1024 * 1024;
576.             return file.size <= maxSizeBytes;
577.         }
578.
579.         // Add real-time validation feedback (blur/input
events)
580.         const emailInput = document.getElementById('email');
581.         const phoneInput =
document.getElementById('telepon');
582.         const quantityInput =
document.getElementById('jumlah');
583.         const submitBtn = document.querySelector('.submit-
btn');
584.
585.         emailInput.addEventListener('blur', function() {
586.             if (this.value && !validateEmail(this.value)) {
587.                 this.setCustomValidity('Email harus
menggunakan domain @gmail.com atau @co.id. ');
588.             } else {
589.                 this.setCustomValidity('');
590.             }
591.             this.reportValidity(); // Tampilkan pesan
validasi langsung
592.         });
593.
594.         phoneInput.addEventListener('input', function() {
595.             this.value = this.value.replace(/[^\0-9]/g, '');
// Hapus karakter non-angka
596.             if (this.value && !validatePhone(this.value)) {
597.                 this.setCustomValidity('Nomor telepon harus
diawali dengan 08 dan terdiri dari 11-13 digit. ');
598.             } else {

```

```

599.                this.setCustomValidity('');
600.            }
601.            this.reportValidity();
602.        });
603.
604.        quantityInput.addEventListener('input', function() {
605.            // Ensure only digits are entered for
        type="number" on some mobile browsers
606.            this.value = this.value.replace(/^[^0-9]/g, '');
607.            if (this.value === '' || parseFloat(this.value)
        <= 0) {
608.                this.setCustomValidity('Jumlah harus berupa
        angka positif.');
```

```

        } else {
609.            } else {
610.                this.setCustomValidity('');
611.            }
612.            this.reportValidity();
613.        });
614.
615.
616.        function setupFileUpload(inputId, textId, previewId)
        {
617.            const fileInput =
        document.getElementById(inputId);
618.            const textSpan =
        document.getElementById(textId);
619.            const previewDiv =
        document.getElementById(previewId);
620.
621.            // Initial check for required fields
622.            if (fileInput.hasAttribute('required') &&
        fileInput.files.length === 0) {
623.                fileInput.setCustomValidity('Harap unggah
        file.');
```

```

        }
624.        }

```

```

625.
626.             fileInput.addEventListener('change', function()
        {
627.                 previewDiv.innerHTML = ''; // Bersihkan
        preview sebelumnya
628.
629.                 if (this.files.length > 0) {
630.                     const file = this.files[0];
631.
632.                     if (!validateFileSize(file)) {
633.                         fileInput.value = ''; // Hapus file
        jika ukuran terlalu besar
634.                         textSpan.textContent = 'Tambahkan
        file';
635.                         fileInput.setCustomValidity('Ukuran
        file tidak boleh lebih dari 1 MB.');
```

```

636.                         previewDiv.innerHTML = <span
        style="color: red;">Ukuran file terlalu besar. Maks 1 MB.</span>;
637.                     } else {
638.                         fileInput.setCustomValidity('');
639.                         textSpan.textContent = file.name;
640.
641.                         // Tampilkan preview gambar
642.                         if (file.type.startsWith('image/'))
        {
643.                             const img =
        document.createElement('img');
644.                             img.src =
        URL.createObjectURL(file);
645.                             img.style.maxWidth = '100px';
646.                             img.style.maxHeight = '100px';
647.                             img.style.marginTop = '10px';
648.                             img.style.marginRight = '10px';
649.                             img.style.verticalAlign =
        'middle';

```



```

650.                previewDiv.appendChild(img);
651.                }
652.                const fileInfoText =
document.createElement('span');
653.                fileInfoText.textContent = File
dipilih: ${file.name} (${(file.size / 1024 / 1024).toFixed(2)} MB);
654.                previewDiv.appendChild(fileInfoText)
;
655.                }
656.                } else {
657.                textSpan.textContent = 'Tambahkan file';
658.                if (fileInput.hasAttribute('required'))
{
659.                fileInput.setCustomValidity('Harap
unggah file. ');
660.                }
661.                }
662.                fileInput.reportValidity(); // Tampilkan
pesan validasi HTML5
663.                });
664.                }
665.
666.                setupFileUpload('gambar_teknik',
'gambar_teknik_text', 'gambar_teknik_preview');
667.                setupFileUpload('pola', 'pola_text',
'pola_preview');
668.
669.
670.                const form = document.getElementById('orderForm');
671.                form.addEventListener('submit', function(e) {
672.                // Reset custom validity messages before re-
checking
673.                emailInput.setCustomValidity('');
674.                phoneInput.setCustomValidity('');
675.                quantityInput.setCustomValidity('');

```

```

676.         document.getElementById('gambar_teknik').setCust
           omValidity('');
677.         document.getElementById('pola').setCustomValidit
           y('');
678.
679.         let isValid = true;
680.         let errorMessageList = [];
681.
682.         // Re-run all client-side validations
683.         const email = emailInput.value;
684.         if (!email) {
685.             errorMessageList.push('Email wajib diisi.');
```

```

686.             isValid = false;
687.         } else if (!validateEmail(email)) {
688.             errorMessageList.push('Format email tidak
           valid atau bukan domain @gmail.com/@.co.id.');
```

```

689.             isValid = false;
690.         }
691.
692.         const name =
           document.getElementById('nama').value;
693.         if (!name) {
694.             errorMessageList.push('Nama wajib diisi.');
```

```

695.             isValid = false;
696.         }
697.
698.         const phone = phoneInput.value;
699.         if (!phone) {
700.             errorMessageList.push('Nomor telepon wajib
           diisi.');
```

```

701.             isValid = false;
702.         } else if (!validatePhone(phone)) {
703.             errorMessageList.push('Nomor telepon harus
           diawali dengan 08 dan terdiri dari 11-13 digit.');
```

```

704.             isValid = false;

```

```

705.                }
706.
707.                const address =
document.getElementById('alamat').value;
708.                if (!address) {
709.                    errorMessageList.push('Alamat wajib
diisi.');
```

```

710.                    isValid = false;
711.                }
712.
713.                const serviceRadios =
document.querySelectorAll('input[name="layanan"]');
```

```

714.                let serviceSelected = false;
715.                for (const radio of serviceRadios) {
716.                    if (radio.checked) {
717.                        serviceSelected = true;
718.                        break;
719.                    }
720.                }
721.                if (!serviceSelected) {
722.                    errorMessageList.push('Layanan wajib
dipilih.');
```

```

723.                    isValid = false;
724.                }
725.
726.                const quantity = quantityInput.value;
727.                if (!quantity) {
728.                    errorMessageList.push('Jumlah wajib
diisi.');
```

```

729.                    isValid = false;
730.                } else if (isNaN(quantity) ||
parseFloat(quantity) <= 0) {
731.                    errorMessageList.push('Jumlah harus berupa
angka positif.');
```

```

732.                    isValid = false;

```

```

733.                }
734.
735.                const gambarTeknikFile =
document.getElementById('gambar_teknik').files[0];
736.                if (!gambarTeknikFile) {
737.                    errorMessageList.push('Gambar Teknik wajib
diunggah.');
```

```

738.                    isValid = false;
739.                } else if (!validateFileSize(gambarTeknikFile))
{
740.                    errorMessageList.push('Ukuran file gambar
teknik tidak boleh lebih dari 1 MB.');
```

```

741.                    isValid = false;
742.                }
743.
744.                const polaFile =
document.getElementById('pola').files[0];
745.                if (!polaFile) {
746.                    errorMessageList.push('Pola wajib
diunggah.');
```

```

747.                    isValid = false;
748.                } else if (!validateFileSize(polaFile)) {
749.                    errorMessageList.push('Ukuran file pola
tidak boleh lebih dari 1 MB.');
```

```

750.                    isValid = false;
751.                }
752.
753.                // Cek validasi HTML5 bawaan browser
754.                if (!form.checkValidity()) {
755.                    // Jika ada error HTML5 yang belum
terdeteksi oleh JS kustom, ambil dari situ
756.                    Array.from(form.elements).forEach(element =>
{
```

```

757.             if (!element.validity.valid &&
                element.validationMessage &&
                !errorMessageList.includes(element.validationMessage)) {
758.                 errorMessageList.push(element.validationMessage);
759.             }
760.         });
761.         isValid = false;
762.     }
763.
764.
765.         if (!isValid) {
766.             e.preventDefault(); // Mencegah pengiriman
                formulir
767.             // Tampilkan pesan error di div alert-error
768.             const errorDiv =
                document.querySelector('.alert-error');
769.             if (errorDiv) {
770.                 errorDiv.innerHTML =
                errorMessageList.join('<br>');
771.                 errorDiv.style.display = 'block'; //
                Pastikan div error terlihat
772.             } else {
773.                 // Jika div tidak ada, buat secara
                dinamis atau gunakan alert (fallback)
774.                 alert('Validasi Gagal:\n' +
                errorMessageList.join('\n'));
775.             }
776.             submitBtn.innerHTML = 'Kirim Pesanan'; //
                Kembalikan teks tombol
777.             submitBtn.disabled = false; // Aktifkan
                kembali tombol
778.             return false;
779.         }
780.

```

```

781.                // Jika validasi berhasil, ubah teks tombol dan
                nonaktifkan
782.                submitBtn.innerHTML = 'Mengirim...';
783.                submitBtn.disabled = true;
784.            });
785.
786.            // Handler untuk tombol WhatsApp
787.            if (lastOrderData) {
788.                const whatsappBtn =
                document.getElementById('whatsappBtn');
789.                if (whatsappBtn) {
790.                    whatsappBtn.addEventListener('click',
                function() {
791.                        openWhatsApp(lastOrderData);
792.                    });
793.                }
794.            }
795.
796.            // Hamburger menu functionality
797.            const hamburgerMenu =
                document.getElementById('hamburger-menu');
798.            const navLinks = document.getElementById('nav-
                links');
799.
800.            if (hamburgerMenu && navLinks) {
801.                hamburgerMenu.addEventListener('click',
                function() {
802.                    navLinks.classList.toggle('active');
803.                    hamburgerMenu.classList.toggle('active');
804.                });
805.
806.                navLinks.querySelectorAll('a').forEach(link => {
807.                    link.addEventListener('click', function() {
808.                        if
                (navLinks.classList.contains('active')) {

```



```

18.             <div class="stat-number"><?php echo
                $monthOrders; ?></div>
19.             <div class="stat-label">Pesanan Bulan
                Ini</div>
20.         </div>
21.     </div>
22.
23.     <!-- Status Statistics -->
24.     <div class="dashboard-section">
25.         <div class="section-title">Status Pesanan</div>
26.         <div class="quick-stats">
27.             <div class="stat-card pending">
28.                 <div class="stat-number"><?php echo
                    $statusCounts['pending'] ?? 0; ?></div>
29.                 <div class="stat-label">Menunggu</div>
30.             </div>
31.             <div class="stat-card confirmed">
32.                 <div class="stat-number"><?php echo
                    $statusCounts['confirmed'] ?? 0; ?></div>
33.                 <div class="stat-
                    label">Dikonfirmasi</div>
34.             </div>
35.             <div class="stat-card progress">
36.                 <div class="stat-number"><?php echo
                    $statusCounts['in_progress'] ?? 0; ?></div>
37.                 <div class="stat-label">Produksi</div>
38.             </div>
39.             <div class="stat-card completed">
40.                 <div class="stat-number"><?php echo
                    $statusCounts['completed'] ?? 0; ?></div>
41.                 <div class="stat-label">Selesai</div>
42.             </div>
43.         </div>
44.     </div>
45. // Get all orders - removed duplicate query

```


[illegible]

```

76.                                      <?php echo
    htmlspecialchars($order['email']); ?><br>
77.                                      <?php echo
    htmlspecialchars($order['phone']); ?>
78.                                     </div>
79.                                     </td>
80.                                     <td>
81.                                     <div class="order-
    address">
82.                                     
83.                                     <?php echo
    htmlspecialchars(substr($order['address'], 0, 100)); ?>
84.                                     <?php if
    (strlen($order['address']) > 100): ?>...<?php endif; ?>
85.                                     </div>
86.                                     </td>
87.                                     <td><?php echo
    htmlspecialchars($order['service_type']); ?></td>
88.                                     <td>
89.                                     <div class="date-
    info">
90.                                      Dipesan:
91.                                     <?php echo
    date('d/m/Y H:i', strtotime($order['created_at'])); ?>
92.                                     </div>
93.                                     <div class="order-
    details">
94.                                      Update:
95.                                     <?php echo
    date('d/m/Y H:i', strtotime($order['updated_at'])); ?>
96.                                     </div>
97.                                     </td>
98.                                     <td>
99.                                     <span
100.                                     class="statu
    s-badge <?php echo getStatusBadgeClass($order['status']); ?>">

```

```

101.                                     <?php echo
    getStatusText($order['status']); ?>
102.                                     </span>
103.                                     </td>
104.                                 </tr>
105.                             <?php endforeach; ?>
106.                             </tbody>
107.                         </table>
108.                     </div>
109.         // Auto refresh setiap 2 menit untuk update data dashboard
110.         setInterval(function() {
111.             window.location.reload();
112.         }, 120000);

```

3. casting.php

```

1. <?php
2. // admin/casting.php (ringkas)
3. if (session_status() == PHP_SESSION_NONE) session_start();
4. require_once '../config/database.php';
5. requireLogin();
6.
7. $database = new Database();
8. $conn = $database->getConnection();
9.
10. // Hapus produk
11. if (isset($_GET['delete'])) {
12.     $id = (int)$_GET['delete'];
13.     try {
14.         $stmt = $conn->prepare("DELETE FROM casting_services WHERE
    id = ?");
15.         $stmt->execute([$id]);
16.         header('Location: casting.php?success=deleted'); exit();
17.     } catch(PDOException $e) { $error = 'Gagal menghapus produk'; }
18. }
19.
20. // Toggle status aktif/nonaktif
21. if (isset($_GET['toggle'])) {
22.     $id = (int)$_GET['toggle'];
23.     try {
24.         $stmt = $conn->prepare("UPDATE casting_services SET status
    = CASE WHEN status = 'active' THEN 'inactive' ELSE 'active' END
    WHERE id = ?");
25.         $stmt->execute([$id]);
26.         header('Location: casting.php?success=updated'); exit();

```

```

27.     } catch(PDOException $e) { $error = 'Gagal mengubah status
    produk'; }
28. }
29.
30. // Ambil data produk
31. try {
32.     $stmt = $conn->prepare("SELECT * FROM casting_services ORDER BY
    sort_order ASC, created_at DESC");
33.     $stmt->execute();
34.     $products = $stmt->fetchAll(PDO::FETCH_ASSOC);
35. } catch(PDOException $e) {
36.     $products = [];
37.     $error = 'Gagal mengambil data produk';
38. }
39. ?>
40. <!DOCTYPE html>
41. <html lang="id">
42. <head>
43.     <title>Admin Casting</title>
44.     <link rel="stylesheet" href="casting.css" />
45. </head>
46. <body>
47.     <h1>Dashboard Admin - Casting</h1>
48.
49.     <?php if (isset($_GET['success'])): ?>
50.         <div><?php echo htmlspecialchars($_GET['success']);
    ?></div>
51.     <?php endif; ?>
52.
53.     <?php if (isset($error)): ?>
54.         <div><?php echo $error; ?></div>
55.     <?php endif; ?>
56.
57.     <a href="add-casting.php">+ Tambah Produk</a>
58.
59.     <?php if (!empty($products)): ?>
60.         <table>
61.             <thead>
62.                 <tr>
63.                     <th>Gambar</th><th>Judul</th><th>Status</th><th>
    Aksi</th>
64.                 </tr>
65.             </thead>
66.             <tbody>
67.                 <?php foreach ($products as $p): ?>
68.                 <tr>
69.                     <td></td>
70.                     <td><?php echo htmlspecialchars($p['title']);
    ?></td>

```

```

71.             <td><?php echo $p['status'] === 'active' ?
    'Aktif' : 'Nonaktif'; ?></td>
72.             <td>
73.                 <a href="edit-casting.php?id=<?php echo
    $p['id']; ?>">Edit</a>
74.                 <a href="?toggle=<?php echo $p['id'];
    ?>">Toggle</a>
75.                 <a href="?delete=<?php echo $p['id']; ?>"
    onclick="return confirm('Hapus produk ini?')">Hapus</a>
76.             </td>
77.         </tr>
78.     <?php endforeach; ?>
79. </tbody>
80. </table>
81. <?php else: ?>
82.     <p>Belum ada produk. <a href="add-casting.php">Tambah
    sekarang</a>.</p>
83. <?php endif; ?>
84. </body>
85. </html>

```

4. galeri.php

```

1. <?php
2. require_once '../config/database.php';
3. requireLogin();
4.
5. $error = '';
6. $success = '';
7.
8. $allowedTables = ['galeri_services'];
9.
10. // Hapus data galeri
11. if (isset($_GET['delete']) && isset($_GET['source'])) {
12.     $id = (int)$_GET['delete'];
13.     $source = $_GET['source'];
14.
15.     if (in_array($source, $allowedTables)) {
16.         $database = new Database();
17.         $conn = $database->getConnection();
18.
19.         try {
20.             $query = "DELETE FROM {$source} WHERE id = ?";
21.             $stmt = $conn->prepare($query);
22.             $stmt->execute([$id]);
23.             header('Location: galeri.php?success=deleted');
24.             exit();
25.         } catch(PDOException $e) {
26.             $error = 'Gagal menghapus data';
27.         }

```

```

28.     }
29. }
30.
31. // Aktif/Nonaktifkan status galeri
32. if (isset($_GET['toggle']) && isset($_GET['source'])) {
33.     $id = (int)$_GET['toggle'];
34.     $source = $_GET['source'];
35.
36.     if (in_array($source, $allowedTables)) {
37.         $database = new Database();
38.         $conn = $database->getConnection();
39.
40.         try {
41.             $query = "UPDATE {$source}
42.                 SET status = CASE
43.                     WHEN status = 'active' THEN 'inactive'
44.                     ELSE 'active'
45.                 END
46.                 WHERE id = ?";
47.             $stmt = $conn->prepare($query);
48.             $stmt->execute([$id]);
49.             header('Location: galeri.php?success=updated');
50.             exit();
51.         } catch(PDOException $e) {
52.             $error = 'Gagal mengubah status';
53.         }
54.     }
55. }
56.
57. // Ambil data galeri
58. $database = new Database();
59. $conn = $database->getConnection();
60.
61. try {
62.     $query = "
63.         SELECT id, title, description, image_url, sort_order,
64.         status, created_at, updated_at,
65.         'galeri_services' AS source
66.         FROM galeri_services
67.         ORDER BY sort_order ASC, created_at DESC;
68.     ";
69.     $stmt = $conn->prepare($query);
70.     $stmt->execute();
71.     $products = $stmt->fetchAll(PDO::FETCH_ASSOC);
72. } catch(PDOException $e) {
73.     $products = [];
74.     $error = 'Gagal mengambil data produk';
75. }

```

5. pesanan.php

```

1. <?php
2. // Handle search and filters
3. $search = $_GET['search'] ?? '';
4. $status_filter = $_GET['filter_status'] ?? 'all';
5. $date_from = $_GET['date_from'] ?? '';
6. $date_to = $_GET['date_to'] ?? '';
7. $service_filter = $_GET['service_filter'] ?? 'all';
8. $sort_order = $_GET['sort_order'] ?? 'newest';
9.
10. $database = new Database();
11. $conn = $database->getConnection();
12.
13. try {
14.     // Build WHERE clause
15.     $whereConditions = [];
16.     $params = [];
17.
18.     if ($status_filter !== 'all') {
19.         $whereConditions[] = "status = ?";
20.         $params[] = $status_filter;
21.     }
22.
23.     if (!empty($search)) {
24.         $whereConditions[] = "(name LIKE ? OR email LIKE ? OR phone
LIKE ? OR message LIKE ?)";
25.         $searchTerm = "%{$search}%";
26.         $params = array_merge($params, [$searchTerm, $searchTerm,
$searchTerm, $searchTerm]);
27.     }
28.
29.     if (!empty($date_from)) {
30.         $whereConditions[] = "DATE(created_at) >= ?";
31.         $params[] = $date_from;
32.     }
33.
34.     if (!empty($date_to)) {
35.         $whereConditions[] = "DATE(created_at) <= ?";
36.         $params[] = $date_to;
37.     }
38.
39.     if ($service_filter !== 'all') {
40.         $whereConditions[] = "service_type = ?";
41.         $params[] = $service_filter;
42.     }
43.
44.     $whereClause = !empty($whereConditions) ? 'WHERE ' . implode('
AND ', $whereConditions) : '';
45.
46.     // Get orders with sorting
47.     $orderClause = '';
48.     switch($sort_order) {

```

```

49.         case 'oldest':
50.             $orderClause = 'ORDER BY created_at ASC';
51.             break;
52.         case 'newest':
53.         default:
54.             $orderClause = 'ORDER BY created_at DESC';
55.             break;
56.     }
57.
58.     $query = "SELECT * FROM pesanan {$whereClause} {$orderClause}";
59.     $stmt = $conn->prepare($query);
60.     $stmt->execute($params);
61.     $pesanan = $stmt->fetchAll(PDO::FETCH_ASSOC);
62.
63.     // Get order counts for each status
64.     $countQuery = "SELECT status, COUNT(*) as count FROM pesanan
GROUP BY status";
65.     $countStmt = $conn->prepare($countQuery);
66.     $countStmt->execute();
67.     $statusCounts = $countStmt->fetchAll(PDO::FETCH_KEY_PAIR);
68.
69.     // Get service counts
70.     $serviceQuery = "SELECT service_type, COUNT(*) as count FROM
pesanan GROUP BY service_type";
71.     $serviceStmt = $conn->prepare($serviceQuery);
72.     $serviceStmt->execute();
73.     $serviceCounts = $serviceStmt->fetchAll(PDO::FETCH_KEY_PAIR);
74.
75. } catch(PDOException $e) {
76.     $pesanan = [];
77.     $error = 'Gagal mengambil data pesanan: ' . $e->getMessage();
78. }
79. ?>
80.
81. <!DOCTYPE html>
82. <html lang="id">
83. <head>
84.     <meta charset="UTF-8">
85.     <meta name="viewport" content="width=device-width, initial-
scale=1.0">
86.     <title>Kelola Pesanan - Filter dan Pencarian</title>
87.     <link rel="stylesheet" href="css/orders-style.css">
88. </head>
89. <body>
90.     <div class="container">
91.         <h1>Kelola Pesanan</h1>
92.
93.         <!-- Statistics Cards -->
94.         <div class="stats-cards">
95.             <div class="stat-card">
96.                 <div class="stat-number"><?php echo
$statusCounts['pending'] ?? 0; ?></div>
97.                 <div class="stat-label">Menunggu</div>
98.             </div>

```



```

136.             </select>
137.         </div>
138.
139.         <div class="form-group">
140.             <label for="service_filter">Layanan</label>
141.             <select id="service_filter"
name="service_filter">
142.                 <option value="all" <?php echo
$service_filter === 'all' ? 'selected' : ''; ?>>Semua
Layanan</option>
143.                 <option value="Casting (Pengecoran)"
<?php echo $service_filter === 'Casting (Pengecoran)' ? 'selected'
: ''; ?>>Casting (Pengecoran)</option>
144.                 <option value="Turning (Bubut)" <?php
echo $service_filter === 'Turning (Bubut)' ? 'selected' : '';
?>>Turning (Bubut)</option>
145.                 <option value="Drilling (Pengeboran)"
<?php echo $service_filter === 'Drilling (Pengeboran)' ? 'selected'
: ''; ?>>Drilling (Pengeboran)</option>
146.             </select>
147.         </div>
148.
149.         <div class="form-group">
150.             <label for="date_from">Dari Tanggal</label>
151.             <input type="date" id="date_from"
name="date_from"
152.                 value="<?php echo
htmlspecialchars($date_from); ?>">
153.         </div>
154.
155.         <div class="form-group">
156.             <label for="date_to">Sampai Tanggal</label>
157.             <input type="date" id="date_to"
name="date_to"
158.                 value="<?php echo
htmlspecialchars($date_to); ?>">
159.         </div>
160.
161.         <div class="form-group">
162.             <label for="sort_order">Urutkan</label>
163.             <select id="sort_order" name="sort_order">
164.                 <option value="newest" <?php echo
$sort_order === 'newest' ? 'selected' : ''; ?>>Terbaru</option>
165.                 <option value="oldest" <?php echo
$sort_order === 'oldest' ? 'selected' : ''; ?>>Terlama</option>
166.             </select>
167.         </div>
168.
169.         <div class="form-group">
170.             <div class="form-buttons">
171.                 <button type="submit" class="search-
btn">Cari</button>
172.                 <a href="?" class="clear-btn">Reset</a>
173.             </div>
174.         </div>

```

```

175.         </form>
176.     </div>
177.
178.     <!-- Orders Table -->
179.     <div class="card">
180.         <div class="card-header">
181.             <h3>Daftar Pesanan (<?php echo count($pesanan);
182.             <a href="export_orders.php">?php echo
!empty($_SERVER['QUERY_STRING']) ? '?' . $_SERVER['QUERY_STRING'] :
''; ?>"
183.                 class="export-btn">Cetak PDF</a>
184.         </div>
185.         <div class="card-body">
186.             <?php if (!empty($pesanan)): ?>
187.             <div class="table-responsive">
188.                 <table class="table">
189.                     <thead>
190.                         <tr>
191.                             <th>ID</th>
192.                             <th>Pelanggan</th>
193.                             <th>Layanan</th>
194.                             <th>Jenis Logam</th>
195.                             <th>Jumlah</th>
196.                             <th>Tanggal</th>
197.                             <th>Status</th>
198.                         </tr>
199.                     </thead>
200.                     <tbody>
201.                         <?php foreach ($pesanan as $order):
?>
202.                             <tr>
203.                                 <td><strong>#<?php echo
str_pad($order['id'], 4, '0', STR_PAD_LEFT); ?></strong></td>
204.                                 <td>
205.                                     <strong><?php echo
htmlspecialchars($order['name']); ?></strong><br>
206.                                     <small><?php echo
htmlspecialchars($order['email']); ?></small><br>
207.                                     <small><?php echo
htmlspecialchars($order['phone']); ?></small>
208.                                 </td>
209.                                 <td><?php echo
htmlspecialchars($order['service_type']); ?></td>
210.                                 <td><?php echo
htmlspecialchars($order['metal_type'] ?? '-'); ?></td>
211.                                 <td><?php echo
htmlspecialchars($order['quantity'] ?? '-'); ?></td>
212.                                 <td><?php echo date('d/m/Y H:i',
strtotime($order['created_at'])); ?></td>
213.                                 <td>
214.                                     <span class="status-badge
status-<?php echo $order['status']; ?>">
215.                                     <?php
216.                                     $statusText = [

```

```

217.                                     'pending' =>
    'Menunggu',
218.                                     'confirmed' =>
    'Dikonfirmasi',
219.                                     'in_progress' =>
    'Produksi',
220.                                     'completed' =>
    'Selesai',
221.                                     'cancelled' =>
    'Dibatalkan'
222.                                     ];
223.                                     echo
    $statusText[$order['status']] ?? 'Menunggu';
224.                                     ?>
225.                                     </span>
226.                                     </td>
227.                               </tr>
228.                               <?php endforeach; ?>
229.                           </tbody>
230.                       </table>
231.                   </div>
232.                   <?php else: ?>
233.                       <div class="no-data">
234.                           <p>Tidak ada pesanan ditemukan.</p>
235.                           <?php if (!empty($search) || $status_filter
    !== 'all' || !empty($date_from) || !empty($date_to) ||
    $service_filter !== 'all'): ?>
236.                               <a href="#" class="btn btn-primary">Reset
    Filter</a>
237.                           <?php endif; ?>
238.                       </div>
239.                   <?php endif; ?>
240.               </div>
241.           </div>
242.       </div>
243.
244.       <script>
245.           // Auto submit form ketika sort order berubah
246.           document.getElementById('sort_order').addEventListener('
    change', function() {
247.               this.form.submit();
248.           });
249.
250.           // Prevent form submission on Enter in search fields dan
    submit otomatis
251.           document.querySelectorAll('.search-form
    input').forEach(input => {
252.               input.addEventListener('keydown', function(e) {
253.                   if (e.key === 'Enter') {
254.                       e.preventDefault();
255.                       document.querySelector('.search-
    btn').click();
256.                   }
257.               });
258.           });

```

```
259.         </script>  
260.     </body>  
261. </html>
```

Lampiran 9. Link Video Profile

<https://drive.google.com/file/d/1zMgrKTuiNjDOPha8QkqF8gqoOHkIf4qy/view?usp=drivesdk>

Lampiran 10. Poster



WEBSITE PENGECORAN LOGAM & PERMESINAN

KENAPA PILIH KAMI?

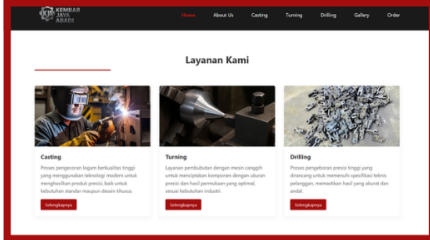
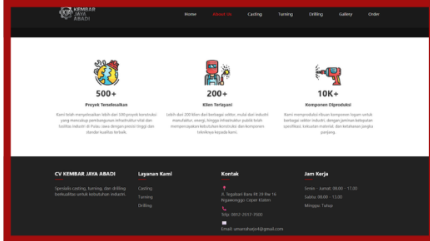
- ✔ Harga Kompetitif Langsung dari Pabrik
- ✔ Kualitas Produk yang Terjamin
- ✔ Pengerjaan Sesuai Estimasi
- ✔ Layanan Cepat dan Responsi

FITUR WEBSITE KAMI:

- ✔ Detail jenis layanan
- ✔ Formulir orderan
- ✔ Galeri perusahaan
- ✔ Informasi kontak perusahaan

HUBUNGI

- ☎ 0812-2617-7600
- 🌐 kembarjayaabadi.my.id/abadi

Lampiran 11. Slide Presentasi

https://www.canva.com/design/DAGuojsTP-s/AVTHHDJU7YrgVpmzSq4_9g/edit?utm_content=DAGuojsTP-s&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton