

ISBN:



# **BUKU AJAR**

# **MATA KULIAH**

---

**INTERAKSI MANUSIA DAN KOMPUTER**

**PENYUSUN: MUREIN MIKSA MARDHIA**

**NAMA PENERBIT**

## HAK CIPTA

### BUKU AJAR INTERAKSI MANUSIA DAN KOMPUTER

Copyright© 2020,

Murein Miksa Mardhia, S.T., M.T.

#### Hak Cipta dilindungi Undang-Undang

Dilarang mengutip, memperbanyak atau mengedarkan isi buku ini, baik sebagian maupun seluruhnya, dalam bentuk apapun, tanpa izin tertulis dari pemilik hak cipta dan penerbit.

#### Diterbitkan oleh:

##### Program Studi Teknik Informatika

Fakultas Teknologi Industri

Universitas Ahmad Dahlan

Jalan Ring Road Selatan, Tamanan, Banguntapan, Bantul Yogyakarta 55166

**Penulis** : Murein Miksa Mardhia, S.T., M.T.

**Editor** : Laboratorium Teknik Informatika, Universitas Ahmad Dahlan

**Desain sampul** : Laboratorium Teknik Informatika, Universitas Ahmad Dahlan

**Tata letak** : Laboratorium Teknik Informatika, Universitas Ahmad Dahlan

**Ukuran/Halaman** : 21 x 29,7 cm / 32 halaman

#### Didistribusikan oleh:



##### Laboratorium Teknik Informatika

Universitas Ahmad Dahlan

Jalan Ring Road Selatan, Tamanan, Banguntapan, Bantul Yogyakarta 55166

Indonesia

## **KATA PENGANTAR**

Terima kasih saya sampaikan kepada hadirat Allah SWT yang telah melimpahkan rahmat dan bimbingan-Nya sehingga penulis dapat menyelesaikan Buku Ajar Matakuliah Interaksi Manusia dan Komputer (IMK) di Program Studi Teknik Informatika, Universitas Ahmad Dahlan. Topik yang disusun dalam buku ajar ini telah disesuaikan dengan perencanaan silabus perkuliahan matakuliah IMK sehingga harapannya mahasiswa dapat memiliki referensi belajar untuk tiap materi yang disampaikan saat tatap muka di kelas.

Buku ini merupakan buku pertama yang disusun sebagai pengantar pembelajaran IMK. Penulis menyadari masih banyak ketidaksempurnaan dalam penulisan ini dan penulis selalu menerima kritik dan saran untuk meningkatkan kualitas materi yang disajikan di edisi dan tahun berikutnya.

Terimakasih kepada seluruh pihak yang terlibat dalam pembuatan buku ajar ini. Semoga hasil yang diperoleh dari implementasi pembelajaran mahasiswa melalui buku ajar ini dapat memberikan manfaat dan kontribusi dalam kemajuan ilmu pengetahuan.

Yogyakarta, September 2020

Penulis  
Murein Miksa Mardhia

## DAFTAR ISI

HAK CIPTA .....	ii
KATA PENGANTAR .....	iv
DAFTAR ISI .....	iv
BAB 1. METODE PENGEMBANGAN PERANGKAT LUNAK: AGILE DAN LEAN .....	1
1.1. Tujuan Instruksional .....	1
1.2. Definisi Agile dan Lean .....	1
1.3. Metode dan Teknik Agile dan Lean .....	3
1.4. Latihan dan Evaluasi .....	4
BAB 2. USER RESEARCH .....	5
2.1. Tujuan Instruksional .....	5
2.2. Definisi User Research .....	5
2.3. Metode dan Teknik dalam User Research .....	5
2.4. Latihan dan Evaluasi .....	7
BAB 3. DESAIN INTERAKSI .....	8
3.1. Tujuan Instruksional .....	8
3.2. Definisi Desain Interaksi .....	8
3.3. Prinsip Desain Interaksi .....	8
3.4. Lingkungan Desain Interaksi .....	11
3.5. Prinsip Desain dan Kebergunaan .....	12
3.6. Identifikasi Kebutuhan Desain .....	14
3.7. Latihan dan Evaluasi .....	14
BAB 4. USER STORYBOARD DAN PROTOTYPING .....	16
4.1. Tujuan Instruksional .....	16
4.2. Definisi <i>Storyboard</i> dalam Desain Antarmuka .....	16
4.3. Narasi dalam Visual .....	19
4.4. Latihan dan Evaluasi .....	20
BAB 5. PENGUJIAN KEBERGUNAAN (USABILITY TESTING) .....	21
5.1. Tujuan Instruksional .....	21
5.2. Pengantar Pengujian Produk .....	21
5.3. Evaluasi Sumatif .....	23
5.4. Eksperimen Terkontrol .....	24
5.5. Latihan dan Evaluasi .....	25
BAB 6. PERKEMBANGAN PENELITIAN DI BIDANG IMK .....	27
6.1. Tujuan Instruksional .....	27
6.2. Virtual dan Augmented reality .....	27
6.3. Tangible User Interface .....	27
6.4. Machine Vision .....	28
6.5. Paper Interfaces .....	28
6.6. Mixed Reality .....	28
6.7. Eye Tracking dan Gaze Control .....	28
6.8. Surface dan Interaksi Tabletop .....	29
6.9. Latihan dan Evaluasi .....	29
DAFTAR PUSTAKA .....	31

# **BAB 1. METODE PENGEMBANGAN PERANGKAT LUNAK: AGILE DAN LEAN**

## **1.1. Tujuan Instruksional**

### **A. Tujuan Instruksional Umum**

Mahasiswa memahami konsep metode pengembangan perangkat lunak yang populer dalam IMK.

### **B. Tujuan Instruksional Khusus**

Mahasiswa memahami konsep alur metode Agile dan Lean dalam mengerjakan tugas proyek IMK.

## **1.2. Definisi Agile dan Lean**

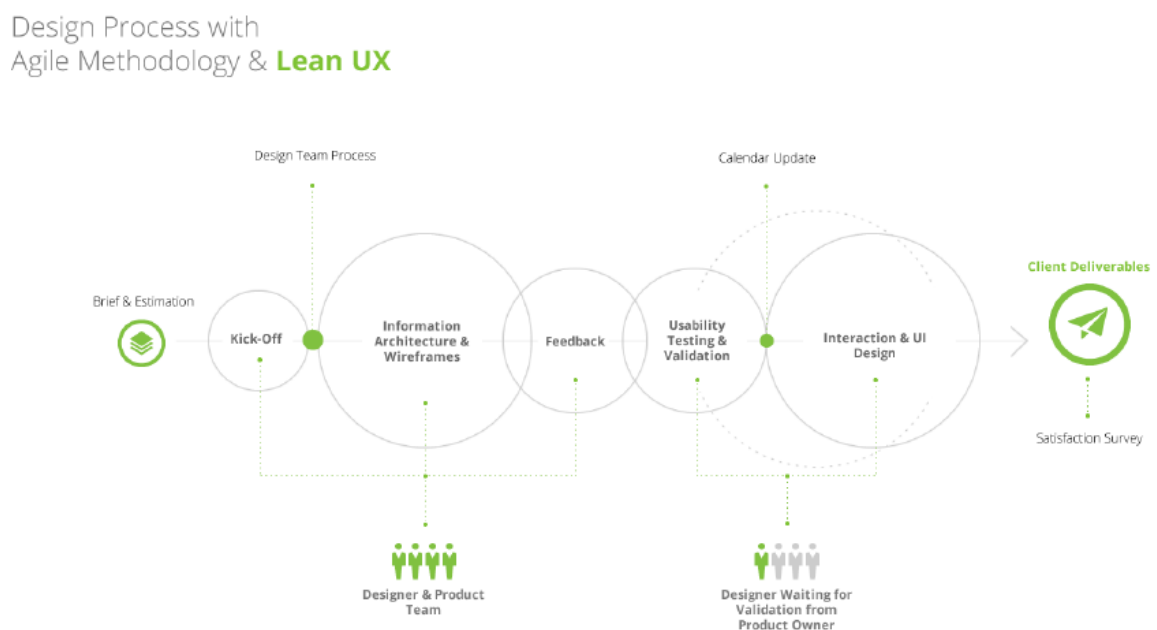
Sebelum memulai proyek pengembangan antarmuka dalam IMK, kita perlu mengenal terlebih dahulu metode yang akan diadaptasi selama proses berjalan. Metode pengembangan perangkat lunak yang saat ini banyak digunakan di perusahaan rintisan maupun IT Corporate antara lain Agile dan Lean UX. Agile Scrum memiliki tujuan untuk memberikan kemampuan perangkat lunak baru setiap 2-4 minggu. Menurut laporan State of Agile tahunan ke-12, 70% tim perangkat lunak menggunakan Scrum atau Scrum *hybrid*. Namun, Scrum telah menyebar ke fungsi bisnis lainnya termasuk TI dan pemasaran di mana ada proyek yang harus bergerak maju dengan adanya kompleksitas dan ambiguitas.

Scrum adalah subkelompok Agile, Agile adalah sekumpulan nilai dan prinsip yang menggambarkan interaksi dan aktivitas kelompok sehari-hari. Metodologi Scrum mengikuti nilai dan prinsip Agile, tetapi mencakup definisi dan spesifikasi lebih lanjut, terutama terkait praktik pengembangan perangkat lunak tertentu. Tim Scrum biasanya terdiri dari 7 +/- 2 anggota dan tidak memiliki ketua tim untuk mendelegasikan tugas atau memutuskan bagaimana suatu masalah diselesaikan. Tim sebagai satu unit memutuskan bagaimana menangani masalah dan memecahkan masalah. Setiap anggota tim Scrum merupakan bagian integral dari solusi dan diharapkan membawa produk dari awal hingga penyelesaian. Ada tiga peran kunci dalam tim Scrum:

**Pemilik Produk (*Product Owner*)** - adalah pemangku kepentingan utama proyek - biasanya pelanggan internal atau eksternal, atau juru bicara pelanggan. Hanya ada satu Pemilik Produk yang menyampaikan keseluruhan misi dan visi produk yang sedang dibangun tim. Pemilik Produk pada akhirnya bertanggung jawab untuk mengelola simpanan produk dan menerima peningkatan pekerjaan yang diselesaikan.

**Scrum Master** – adalah pemimpin yang melayani Pemilik Produk, Tim Pengembang dan Organisasi. Tanpa otoritas hierarki atas tim tetapi lebih sebagai fasilitator, Scrum Master memastikan bahwa tim mematuhi teori, praktik, dan aturan Scrum. Scrum Master melindungi tim dengan melakukan apa pun yang memungkinkan untuk membantu tim tampil di level tertinggi. Ini mungkin termasuk menghilangkan hambatan, memfasilitasi pertemuan, dan membantu Pemilik Produk mengatur *backlog*.

**Tim Pengembang (*Development Team*)** - adalah kelompok lintas fungsi yang mengatur dirinya sendiri dan dipersenjatai dengan semua keterampilan untuk memberikan peningkatan yang dapat dikirim pada penyelesaian setiap sprint. Scrum memperluas definisi istilah "pengembang" lebih dari sekedar menjadi pemrogram untuk menyertakan siapa saja yang berpartisipasi dalam pembuatan progress produk yang dihasilkan dalam sebuah siklus tahapan/*increment*.



Gambar 1.1. Metodologi Agile dan Lean UX

Sumber: Ellie Cachtette dalam “Simple Steps to Great UI UX”

### Event dalam Scrum

**Sprint** - *Sprint* adalah periode waktu terbatas di mana pekerjaan tertentu diselesaikan dan disiapkan untuk ditinjau. Sprint biasanya berlangsung selama 2-4 minggu, tetapi bisa juga sesingkat satu minggu.

**Sprint Planning** - Rapat tim perencanaan adalah acara dengan batas waktu yang menentukan item simpanan produk mana yang akan dikirimkan dan bagaimana pekerjaan akan dicapai.

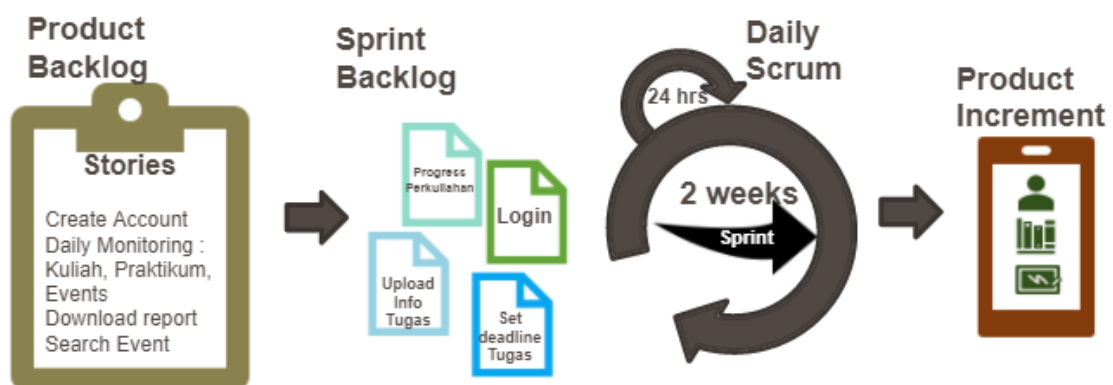
**Daily Stand-up** - adalah pertemuan komunikasi singkat (tidak lebih dari 15 menit) di mana setiap anggota tim dengan cepat dan transparan meliputi kemajuan sejak stand-up terakhir, pekerjaan yang direncanakan sebelum pertemuan berikutnya, dan segala halangan yang mungkin menghalangi atau kemajuannya.

**Sprint Review** - adalah acara "*show-and-tell*" atau demonstrasi bagi tim untuk mempresentasikan pekerjaan yang telah diselesaikan selama sprint. Pemilik Produk memeriksa pekerjaan berdasarkan kriteria penerimaan yang telah ditentukan sebelumnya dan menerima atau menolak pekerjaan tersebut. Pemangku kepentingan atau klien memberikan umpan balik untuk memastikan bahwa kenaikan yang disampaikan memenuhi kebutuhan bisnis.

**Retrospektif** - atau Retro, adalah pertemuan tim terakhir di Sprint untuk menentukan apa yang berjalan dengan baik, apa yang tidak berjalan dengan baik, dan bagaimana tim dapat meningkat di Sprint berikutnya. Dihadiri oleh tim dan ScrumMaster, Retrospektif adalah kesempatan penting bagi tim untuk fokus pada kinerja keseluruhan dan mengidentifikasi strategi untuk perbaikan berkelanjutan pada prosesnya.

### 1.3. Metode dan Teknik Agile dan Lean

Bagaimana cara mengimplementasikan Scrum dan Lean UX dalam satu waktu? Berikut contoh aplikasi mengenai monitoring studi mahasiswa yang dibangun dengan menggabungkan kedua metode tersebut.



Gambar 1.2. Alur Tahapan Metodologi Agile Scrum

Gambar 1.2 menunjukkan tahapan yang terjadi dalam metode Agile. Definisi istilah dalam metode tersebut antara lain:

- **Product Backlog** → tentukan semua fitur yang akan dikembangkan.
- **Sprint Backlog** → pembagian durasi waktu pengerjaan *backlog*. Tentukan setiap iterasi *sprint*, dan tugas apa yang harus dilakukan di masing-masing *sprint*. Terapkan dengan cepat; 2 minggu adalah waktu rata-rata, mungkin lebih lama sampai 1 bulan.
- **Daily Scrum** → adakan rapat singkat selama *sprint* untuk mendiskusikan setiap kendala yang terjadi. Perubahan akan dilakukan. Pemilik produk dan Scrum Master harus memeriksa perkembangannya, menyiapkan beberapa rencana jika ada yang tidak beres. Dalam satu kali *sprint*, aktivitas berlangsung secara parallel: implementasi dan *testing*.
  - **Lean User Testing** → dapat menjadi *Guerilla Testing*, merekrut responden acak tetapi representatif untuk menguji aplikasi dan mengamati apa yang mereka rasakan dengan membiarkan mereka berpikir keras.
  - Rekrut peserta dengan 2 cara: lingkungan yang terkendali (menggunakan lab), atau *Guerilla Testing* (silakan pilih peserta, selama dia berasal dari pengguna yang ditargetkan)

**Product Increment** → *Increment* adalah jumlah dari semua fitur/*backlog* simpanan produk yang telah diselesaikan sejak rilis perangkat lunak terakhir. Pemilik Produk berhak memutuskan kapan produk dirilis, tetapi tim bertanggung jawab untuk memastikan semua yang disertakan dalam tahapan tersebut siap untuk dirilis. Ini juga disebut sebagai *Potentially Shippable Increment* (PSI).

#### 1.4.Latihan dan Evaluasi

Untuk memperkenalkan pengetahuan dasar mengenai Agile, Lean UX dan pengujian dalam IMK, silakan mengakses file materi di link berikut:

<https://drive.google.com/file/d/1qI9usb7odHQ1uhQZBj6pdtQqD4ciznu/view?usp=sharing>.

Buat perencanaan dengan tim kelompok Anda dalam mempersiapkan pembagian pekerjaan dalam tugas proyek. Tentukan product backlog Anda masing-masing beserta jumlah dan durasi *sprint* yang ditetapkan selama total waktu pengerjaan proyek!



## **BAB 2. USER RESEARCH**

### **2.1. Tujuan Instruksional**

#### **C. Tujuan Instruksional Umum**

Mahasiswa memahami konsep utama dari memahami kebutuhan dan karakteristik target pengguna produk aplikasi.

#### **D. Tujuan Instruksional Khusus**

Mahasiswa memahami teknik dan luaran dari aktivitas riset terhadap target pengguna produk aplikasi.

### **2.2. Definisi User Research**

Manusia merupakan karakter sentral dalam sistem interaktif, sedangkan sistem komputer dirancang untuk membantu manusia sebagai penggunanya. Kebutuhan pengguna menjadi prioritas utama. Agar dapat merancang sesuatu, kita perlu memahami kapabilitas dan limitasinya. Kita harus tahu apakah sesuatu akan sulit atau mudah dilakukan pengguna atau bahkan mustahil. Bagaimana manusia memandang segala sesuatu di sekitarnya, bagaimana menyimpan dan memproses informasi serta memecahkan masalah dan bagaimana manusia mengolah obyek-obyek.

Kerangka utama yang memberikan sifat pada IMK adalah kognitif. Kognisi/kognitif merupakan proses bagaimana seseorang memperoleh pengetahuan dengan cara mengenali sesuatu/benda. Termasuk: pemahaman, mengingat, penalaran, kehadiran, kepedulian, mengasah keterampilan dan membuat ide baru. Tujuan kognitif pada IMK adalah untuk memahami dan merepresentasikan bagaimana manusia berinteraksi dengan komputer terkait bagaimana pengetahuan ditransmisikan di antara keduanya. Latar belakang teoritis untuk pendekatan ini berasal dari psikologi kognitif, yaitu untuk menjelaskan bagaimana manusia mencapai tujuan yang telah ditetapkan. Aktifitas berorientasi tujuan semacam itu terdiri dari pelaksanaan tugas-tugas kognitif yang melibatkan pemrosesan informasi.

### **2.3. Metode dan Teknik dalam User Research**

Secara umum, daftar tugas ini menjadi bekal melaksanakan riset terhadap pengguna:

1. Masalah apa yang sedang Anda coba selesaikan?
2. Apa dampak utama yang ingin kita miliki?

3. Pertanyaan Framing (5W + 1H). Hasilkan pertanyaan sebanyak yang Anda bisa.
4. Apa konteks dan batasannya?
5. Bagaimana tantangan dalam desain yang ada?
6. Tetapkan Tujuan Penelitian.

Salah satu teknik yang populer dan mudah diadaptasi saat menggali profil pengguna adalah dengan membangun User Persona. User Persona adalah Representasi semi-fiktif dari target pelanggan yang dibingkai dari penemuan nyata oleh pengguna untuk mengungkapkan kebutuhan, tujuan, dan pola perilaku yang diamati dari target pengguna. Tujuan dari pembuatan user persona adalah mengungkap berbagai cara orang menelusuri, membeli, dan menggunakan produk; guna fokus pada upaya meningkatkan pengalaman untuk pengguna yang sebenarnya melalui data fakta yang nyata.

User persona yang dibuat harus dapat mewakili profil dari populasi pengguna secara umum. Komponen yang dijelaskan dalam User Persona meliputi profil demografi (nama, usia, jenis kelamin, hobi, domisili lokasi tinggal atau bekerja) dari target pengguna, serta yang lebih spesifik seperti apa yang menjadi motivasi mereka dalam melakukan aktivitas yang terkait suatu fokus topik domain dan hal apa yang mereka ingin selesaikan dari persoalan yang mereka alami (Gambar 2.1).



Gambar 2.1. Komposisi Informasi dalam User Persona

Dengan menyusun sebuah user persona, pengembang aplikasi akan menyelaraskan fokus fungsionalitas aplikasi dengan bagaimana karakter dan kebiasaan target konsumen. Melalui user persona, faktor budaya dan kepercayaan suatu target konsumen akan dapat terlihat. Sehingga semakin detail karakteristik calon pengguna dapat digali, semakin baik persiapan yang dapat dilakukan pengembang aplikasi dalam menyempurnakan arsitektur aplikasinya. Contoh sebuah User Persona ditampilkan pada Gambar 2.2.



Gambar 2.2. Contoh User Persona

Selain User Persona, metode berikut ini juga digunakan dalam fase memahami pengguna:

**Card Sorting** - Memungkinkan pengguna untuk mengelompokkan informasi situs Anda. Ini membantu memastikan bahwa struktur situs sesuai dengan cara berpikir pengguna.

**Contextual Interviews** - Memungkinkan Anda untuk mengamati pengguna di lingkungan alami mereka, memberi Anda pemahaman yang lebih baik tentang cara kerja pengguna.

**First Click Testing** - Metode pengujian yang berfokus pada navigasi, yang dapat dilakukan pada situs web yang berfungsi, prototipe, atau gambar rangka.

**Focus Group** - Diskusi yang dimoderasi dengan sekelompok pengguna, memungkinkan Anda mempelajari tentang sikap, ide, dan keinginan pengguna.

**Heuristic Evaluation** - Sekelompok ahli kegunaan mengevaluasi situs web Anda berdasarkan daftar pedoman yang ditetapkan.

**Individual Interviews** - Diskusi empat mata dengan pengguna menunjukkan cara kerja pengguna tertentu. Mereka memungkinkan Anda mendapatkan informasi mendetail tentang sikap, keinginan, dan pengalaman pengguna.

**Parallel Design** - Sebuah metodologi desain yang melibatkan beberapa desainer yang mengejar upaya yang sama secara bersamaan, tetapi secara independen, dengan maksud untuk menggabungkan aspek terbaik dari masing-masing untuk solusi akhir.

## 2.4. Latihan dan Evaluasi

Dari topik aplikasi yang Anda rencanakan, tetapkan siapa target konsumen yang paling potensial! Profilkan target konsumen tersebut dengan menyimpulkan range usia, sebaran jenis kelamin, pekerjaan dan teknologi/*gadget* yang saat ini digunakan!

## BAB 3. DESAIN INTERAKSI

*Desain Interaksi berfokus pada momen ketika pengguna berinteraksi dengan produk dan tujuan mereka adalah meningkatkan pengalaman interaktif.*

### 3.1. Tujuan Instruksional

#### E. Tujuan Instruksional Umum

Mahasiswa memahami konsep utama dari desain interaksi dalam pengembangan antarmuka.

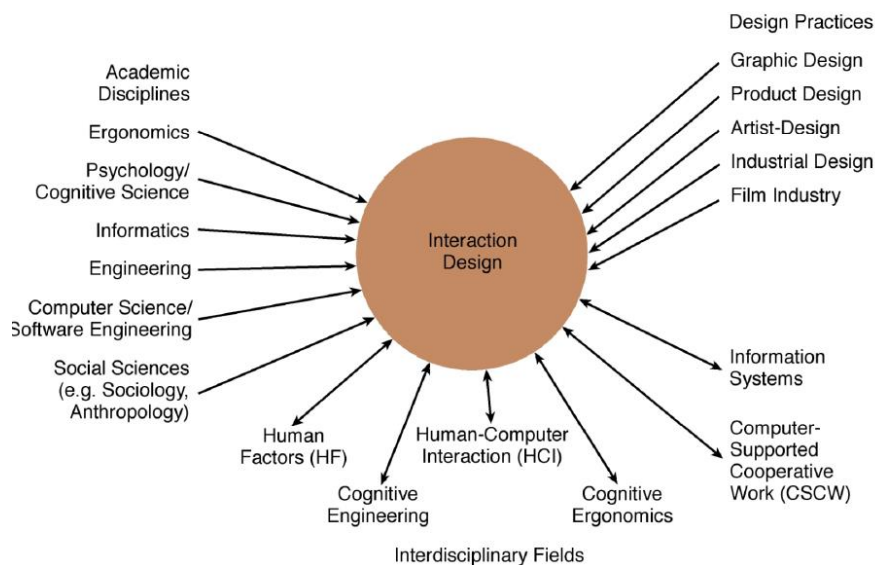
#### F. Tujuan Instruksional Khusus

Mahasiswa memahami konsep contoh desain interaksi pada antarmuka produk maupun aplikasi.

### 3.2. Definisi Desain Interaksi

Definisi Desain Interaksi:

- Mendesain produk-produk interaktif untuk membantu manusia berkomunikasi dan berinteraksi dalam kehidupan sehari-hari dan pekerjaan mereka
- Istilah “Desain Interaksi” pertama kali dicetuskan oleh Bill Moggridge, pembuat laptop pertama di dunia



**Gambar 3.1. Hubungan antara ilmu-ilmu yang berkontribusi dalam Desain Interaksi**

*“By interaction design, we mean designing interactive products to support people in their everyday and working lives”*

### 3.3. Prinsip Desain Interaksi

Prinsip desain berasal dari perpaduan pengetahuan berbasis teori, pengalaman, dan akal sehat. Prinsip desain dimaksudkan untuk membantu desainer produk dalam membangun interaksi. Namun, mereka tidak dimaksudkan untuk menentukan bagaimana merancang antarmuka yang sebenarnya (misal memberi tahu desainer cara mendesain ikon tertentu atau cara menyusun struktur portal website) tetapi lebih bertindak seperti pengingat untuk desainer, memastikan bahwa mereka telah menyediakan hal-hal tertentu di antarmuka.

Sejumlah prinsip desain telah diperkenalkan, dimana yang paling terkenal yaitu memperhatikan bagaimana cara menentukan apa yang harus dilihat dan dilakukan pengguna saat menjalankan pekerjaan/*task* menggunakan suatu produk yang interaktif. Berikut akan dijelaskan secara singkat mengenai prinsip-prinsip yang paling umum: visibility, feedback, constraints, mapping, consistency, dan affordances. (Norman, 2016)

### **Visibility**

Norman (1988) menjelaskan tentang control dari sebuah mobil untuk menekankan hal ini. Kontrol untuk operasi yang berbeda jelas terlihat (misalnya, indikator, lampu depan, klakson, lampu peringatan bahaya), menunjukkan apa bisa diselesaikan. Hubungan antara cara pengendalian diposisikan di dalam mobil dan apa yang mereka lakukan memudahkan pengemudi untuk menemukan kontrol yang sesuai untuk tugas yang ada.

### **Feedback**

Umpan balik adalah tentang mengirimkan kembali informasi tentang tindakan apa yang telah dilakukan dan apa yang telah dicapai, memungkinkan orang tersebut untuk melanjutkan aktivitas. Berbagai jenis umpan balik tersedia untuk audio desain interaksi, taktil, verbal, visual, dan kombinasinya. Memutuskan kombinasi mana yang sesuai untuk berbagai jenis kegiatan dan interaktivitas adalah pusatnya. Menggunakan masukan dalam cara yang benar juga dapat memberikan visibilitas yang diperlukan untuk interaksi pengguna.

### **Constraints**

Norman (1999) mengklasifikasikan Constraints (Batasan) menjadi tiga kategori: fisik, logis, dan kultural. Batasan fisik mengacu pada cara benda-benda fisik membatasi pergerakan. Misalnya, cara disk eksternal ditempatkan di dalam disk drive secara fisik dibatasi oleh bentuk dan ukurannya, sehingga hanya dapat disisipkan dengan satu jalan.

Batasan logis bergantung pada pemahaman orang tentang cara dunia bekerja. Batasan ini mengandalkan bagaimana logika nalar manusia berproses tentang suatu tindakan dan konsekuensinya. Adanya aksi yang dapat terlihat dengan jelas memungkinkan manusia untuk menyimpulkan secara logis mengenai tindak lanjut apa yang diperlukan. Menonaktifkan opsi menu bila tidak sesuai untuk pekerjaan yang sedang berlangsung memberikan batasan logis. Hal ini memungkinkan pengguna untuk berpikir mengapa (atau mengapa tidak) mereka dirancang seperti ini dan pilihan aksi apa yang tersedia.

Batasan budaya bergantung pada kebiasaan yang dipelajari, seperti penggunaan warna merah untuk peringatan, penggunaan jenis sinyal audio tertentu untuk bahaya, dan penggunaan *smiley* wajah untuk mewakili emosi bahagia. Sebagian besar batasan budaya bersifat di luar akal nalar logika sehingga direpresentasikan secara abstrak, dan bisa berevolusi dalam bentuk lain (misalnya, penggunaan warna kuning –sebagai ganti warna merah– untuk peringatan). Oleh karena itu, Batasan budaya harus dipelajari. Setelah dipelajari dan diterima oleh suatu kelompok budaya, mereka menjadi kebiasaan yang diterima secara universal. Dua kebiasaan dalam antarmuka yang diterima secara universal adalah penggunaan jendela untuk menampilkan informasi dan penggunaan ikon pada desktop untuk mewakili operasi dan dokumen.

## **Mapping**

Prinsip ini mengacu pada hubungan antara kendali dan efek yang diberikannya. Hampir semua artefak membutuhkan semacam pemetaan antara kontrol dan efek, apakah itu senter, mobil, pembangkit listrik, atau kokpit. Contoh pemetaan yang bagus antara kontrol dan efek adalah panah atas dan bawah yang digunakan untuk mewakili naik dan gerakan kursor ke bawah, masing-masing, di keyboard komputer. Itu pemetaan posisi relatif kontrol dan efeknya juga penting. Mempertimbangkan berbagai perangkat pemutaran musik (misalnya, MP3, pemutar CD, perekam kaset). Bagaimana kontrol bermain, memutar ulang, dan maju cepat dipetakan ke yang diinginkan efek? Mereka biasanya mengikuti konvensi umum dalam menyediakan urutan tombol, dengan tombol putar di tengah, tombol putar ulang di kiri dan maju cepat di sebelah kanan. Konfigurasi ini memetakan langsung ke arah tindakan (lihat Gambar 1.9a). Bayangkan betapa sulitnya jika pemetaan masuk Gambar 1.9b digunakan. Perhatikan Gambar 1.10 dan tentukan dari berbagai pemetaan mana yang baik dan yang akan menimbulkan masalah bagi orang yang menggunakannya.

**Consistency** - mengacu pada mendesain antarmuka untuk memiliki operasi dan penggunaan yang serupa elemen serupa untuk mencapai tugas serupa. Secara khusus, antarmuka yang konsisten adalah salah satu yang mengikuti aturan, seperti menggunakan operasi yang sama untuk memilih semua objek. Misalnya, operasi yang konsisten menggunakan tindakan masukan yang sama untuk menyerot salah satu objek grafis di antarmuka, seperti selalu mengklik tombol kiri mouse. Tidak konsisten antarmuka, di sisi lain, memungkinkan pengecualian untuk suatu aturan. Contoh dari ini adalah tempat objek grafis tertentu (mis., pesan email yang disajikan dalam tabel) dapat disorot hanya dengan menggunakan tombol kanan mouse, sementara semua operasi lainnya disorot menggunakan tombol kiri. Masalah dengan ketidakkonsistenan semacam ini adalah bahwa itu cukup sewenang-wenang, sehingga menyulitkan pengguna untuk mengingat dan membuat file pengguna lebih rentan terhadap kesalahan.

**Keterjangkauan/Affordance** - adalah istilah yang digunakan untuk merujuk pada atribut dari suatu objek yang memungkinkan orang untuk mengetahui cara menggunakannya. Misalnya, tombol *mouse* mengundang dorongan (saat melakukan aktivasi klik) dengan cara dibatasi secara fisik dalam cangkang plastiknya. Sangat tingkat sederhana, untuk membeli berarti "memberi petunjuk" (Norman, 1988). Saat kemampuan objek fisik secara persepsi jelas, mudah untuk mengetahui bagaimana berinteraksi dengan itu. Misalnya, pegangan pintu bisa ditarik, pegangan cangkir bisa ditarik menggenggam, dan tombol mouse memungkinkan dorongan. Norman memperkenalkan konsep ini di akhir 80an dalam pembahasannya tentang desain benda sehari-hari. Sejak itu, itu terjadi telah banyak dipopulerkan, digunakan untuk menjelaskan bagaimana objek antarmuka harus dirancang sehingga mereka menjelaskan apa yang bisa dilakukan terhadap mereka. Misalnya, grafis elemen seperti tombol, ikon, tautan, dan *scroll bar* untuk bagaimana membuatnya tampak jelas bagaimana mereka harus digunakan: ikon harus dirancang untuk membeli klik, *scroll bar* untuk bergerak ke atas dan ke bawah, tombol untuk membeli mendorong.

### 3.4. Lingkungan Desain Interaksi

Pada dasarnya, proses desain interaksi melibatkan empat aktivitas dasar:

1. Mengidentifikasi kebutuhan dan menetapkan kebutuhan.
2. Mengembangkan alternatif desain yang memenuhi kebutuhan tersebut.
3. Membangun versi interaktif dari desain sehingga dapat dikomunikasikan dan dinilai.
4. Mengevaluasi apa yang sedang dibangun selama proses berlangsung.

Kegiatan ini dimaksudkan untuk saling berhubungan dan prosesnya dapat diulang. Misalnya, mengukur kegunaan dari apa yang telah dibangun dalam hal apakah mudah digunakan dapat berarti perubahan tertentu harus dilakukan atau kebutuhan tertentu belum dipenuhi. Mengevaluasi apa yang telah dibangun merupakan inti dari desain interaksi. Fokusnya adalah memastikan bahwa produk tersebut dapat digunakan. Ini biasanya diketahui melalui pendekatan *User-Centered Design* yang seperti namanya, berusaha untuk melibatkan pengguna selama proses desain. Ada banyak cara untuk mencapai ini, misalnya dengan mengamati pengguna, berbicara dengan mereka, mewawancarai mereka, menguji mereka menggunakan pemberian *task*, memodelkan kinerja mereka (dalam mengerjakan *task* tersebut), meminta mereka untuk mengisi kuesioner, dan bahkan meminta mereka untuk menjadi rekan dalam perancangan (*co-designer*). Hal yang sama pentingnya dengan melibatkan pengguna dalam mengevaluasi suatu produk yang interaktif adalah memahami apa yang saat ini dilakukan atau digemari masyarakat. Bentuk penelitian seperti ini harus dilakukan sebelum membangun produk interaktif apapun. Alasan utama perlu memiliki pemahaman yang lebih baik tentang pengguna adalah bahwa pengguna memiliki kebutuhan yang berbeda dan produk interaktif perlu dirancang sesuai dengan itu. Misalnya, anak-anak memiliki ekspektasi yang berbeda tentang bagaimana mereka ingin belajar atau bermain dari orang dewasa. Mereka mungkin mendapati kuis interaktif dan karakter kartun membantu mereka menjadi sangat memotivasi, sedangkan kebanyakan orang dewasa menganggapnya menjengkelkan. Sebaliknya, orang dewasa sering kali menyukai forum diskusi tentang suatu topik, tetapi anak-anak menganggapnya membosankan. Seperti halnya objek sehari-hari seperti pakaian, makanan, dan permainan dirancang secara berbeda untuk anak-anak, remaja, dan orang dewasa, demikian juga produk interaktif harus dirancang untuk memenuhi kebutuhan berbagai jenis pengguna.

Selain empat aktivitas dasar desain di atas, terdapat tiga karakteristik utama dari proses desain interaksi:

1. Pengguna harus dilibatkan dalam pengembangan proyek.
2. Kegunaan khusus dan tujuan pengalaman pengguna harus diidentifikasi, didokumentasikan dengan jelas, dan disepakati di awal proyek.
3. Iterasi melalui empat aktivitas merupakan suatu kewajiban yang harus dilalui.

### **3.5. Prinsip Desain dan Kebergunaan**

Di bawah ini adalah sepuluh prinsip kebergunaan/*usability* yang dikembangkan oleh (Nielsen, 2000) dan rekan-rekannya. Perhatikan bagaimana beberapa di antaranya tumpang tindih dengan prinsip desain.



1. **Visibilitas status sistem** - selalu berikan informasi kepada pengguna tentang apa yang sedang terjadi, dengan memberikan umpan balik yang sesuai dalam waktu yang wajar.
2. **Memadukan bahasa yang digunakan antara sistem dan dunia nyata** - ucapkan bahasa pengguna, menggunakan kata, frasa, dan konsep yang akrab bagi pengguna, bukan istilah berorientasi sistem.
3. **Memastikan adanya kontrol oleh pengguna** - untuk mudah menemukan jalan keluar dari tempat yang mereka temukan secara tak terduga, dengan menggunakan 'pintu keluar darurat' yang ditandai dengan jelas.
4. **Konsistensi dan penggunaan standar** - hindari membuat pengguna bertanya-tanya apakah kata-kata, situasi, atau tindakan yang berbeda memiliki arti yang sama.
5. Bantu pengguna mengenali, mendiagnosis, dan pulih dari kesalahan - gunakan bahasa sederhana untuk menggambarkan sifat masalah dan menyarankan cara untuk menyelesaikannya.
6. **Pencegahan kesalahan** - jika memungkinkan dahulukan mencegah kesalahan yang mungkin terjadi.
7. **Mengutamakan pengenalan daripada ingatan** - membuat objek, aksi, dan opsi dapat dilihat.
8. **Fleksibilitas dan efisiensi penggunaan** - sediakan akselerator yang tidak terlihat oleh pengguna pemula, tetapi memungkinkan pengguna yang lebih berpengalaman untuk menjalankan tugas dengan lebih cepat.
9. **Desain estetis dan minimalis** - hindari penggunaan informasi yang tidak relevan atau jarang dibutuhkan.
10. **Bantuan dan dokumentasi** - sediakan informasi yang dapat dicari dengan mudah dan berikan bantuan dalam serangkaian langkah konkret yang dapat diikuti dengan mudah.

Prinsip desain dan kegunaan juga telah dioperasionalkan secara lebih spesifik yang disebut aturan. Ini adalah pedoman yang harus diikuti. Contohnya adalah "selalu tempatkan tombol atau menu keluar di bagian bawah menu pertama dalam aplikasi".

Desain interaksi berkaitan dengan merancang produk interaktif untuk mendukung manusia dalam kehidupan sehari-hari dan pekerjaan mereka. Desain interaksi bersifat multidisiplin, yang melibatkan banyak masukan dari disiplin ilmu dan bidang yang luas. Desain interaksi sekarang menjadi bisnis besar: banyak perusahaan menginginkannya tetapi tidak tahu bagaimana melakukannya. Mengoptimalkan interaksi antara pengguna dan produk interaktif membutuhkan sejumlah faktor yang saling bergantung, termasuk konteks penggunaan, jenis

tugas, dan jenis pengguna. Produk interaktif perlu dirancang agar sesuai dengan tujuan kegunaan seperti kemudahan penggunaan dan pembelajaran. Sasaran pengalaman pengguna berkaitan dengan pembuatan sistem yang meningkatkan pengalaman pengguna dalam hal membuatnya menyenangkan, menyenangkan, membantu, memotivasi, dan menyenangkan. Prinsip desain dan kegunaan, seperti umpan balik dan kesederhanaan, adalah heuristik yang berguna untuk menganalisis dan mengevaluasi aspek produk interaktif. (Preece, Sharp, & Rogers, 2015)

### **3.6. Identifikasi Kebutuhan Desain**

Dalam rekayasa perangkat lunak, ada dua jenis kebutuhan tradisional yang diidentifikasi: kebutuhan fungsional, yang menyatakan apa yang harus dilakukan sistem, dan kebutuhan non-fungsional, yang menyatakan batasan apa yang ada pada sistem dan pengembangannya. Misalnya, kebutuhan fungsional untuk pengolah kata mungkin itu harus mendukung berbagai *formatting style*. Kebutuhan ini kemudian dapat diuraikan menjadi kebutuhan yang lebih spesifik yang merinci *formatting style* yang diperlukan seperti berdasarkan paragraf, karakter, dan dokumen, turun ke tingkat yang sangat spesifik seperti *formatting* karakter yang harus menyertakan 20 tipografi, masing-masing dengan pilihan tebal, miring, dan standar.

Kebutuhan non-fungsional untuk pengolah kata mungkin itu harus bisa berjalan di berbagai platform seperti PC, Mac, dan mesin Unix. Atau misalkan dapat berfungsi di komputer dengan RAM 64 MB. Kebutuhan non-fungsional lain yaitu harus dapat diselesaikan dalam waktu enam bulan. Hal-hal tersebut merepresentasikan batasan pada aktivitas pengembangan itu sendiri daripada pada produk yang sedang dikembangkan. (Preece et al., 2015)

Mengidentifikasi kebutuhan pengguna tidak sesederhana seperti kedengarannya atau sekadar menulis daftar fitur yang diinginkan. Mengingat sifat desain interaksi yang berulang, aktivitas analisis kebutuhan perlu disendirikan; tidak digabung dengan kegiatan desain dan kegiatan evaluasi, karena dalam prakteknya mereka semua saling terkait: beberapa desain akan dibangun sementara analisis kebutuhan sedang ditetapkan, dan desain akan berkembang melalui serangkaian perulangan siklus evaluasi. Bagaimanapun juga, setiap aktivitas ini dapat dibedakan dari penekanan dan tekniknya masing-masing.

### **3.7. Latihan dan Evaluasi**

Dari topik tugas proyek yang telah dipilih di kelas, tuliskan kebutuhan fungsional yang harus disediakan dalam aplikasi dan diskusikan dengan teman dalam tim mengenai langkah-

langkah/*task* yang harus dikerjakan pengguna untuk mencapai tujuan agar kebutuhan fungsional tersebut selesai.

## BAB 4. USER STORYBOARD DAN PROTOTYPING

### 4.1. Tujuan Instruksional

#### A. Tujuan Instruksional Umum

Mahasiswa memahami konsep pentingnya *storyboarding* dalam proses mendesain antarmuka.

#### B. Tujuan Instruksional Khusus

Mahasiswa memahami contoh *storyboarding* dalam proses pembuatan *prototype* antarmuka.

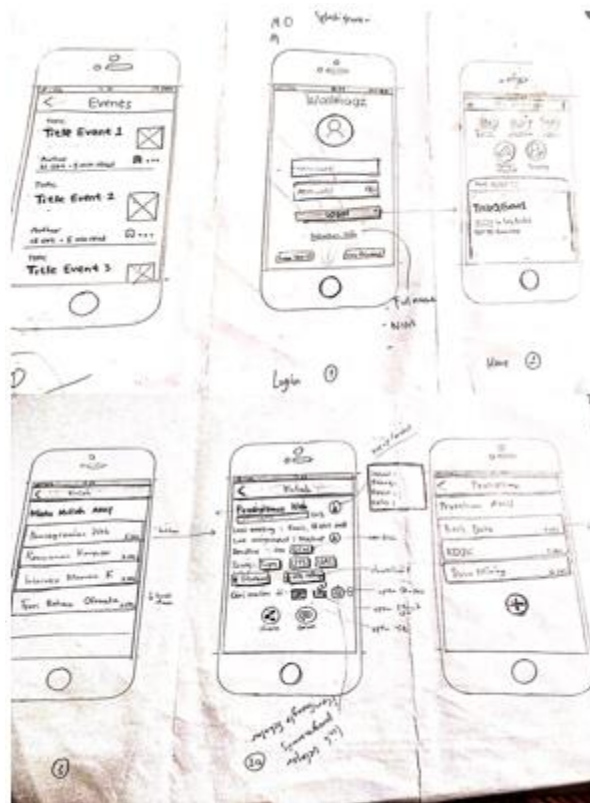
### 4.2. Definisi *Storyboard* dalam Desain Antarmuka

*Storyboard* merupakan aktivitas merancang/membuat alur (*flow*) proses seorang user mencapai *goal* yang diinginkan. Tujuan pembuatan *storyboard* yaitu untuk mendemonstrasikan alur tahapan yang harus dijalani user dalam aplikasi dari awal hingga tujuan mereka tercapai. *Storyboard* juga digunakan untuk membantu proses *prototyping* supaya fokus, terarah dan lebih cepat selesai.

Tahapan *storyboard* berada di bagian setelah define problem dan sebelum *prototyping*. Wujud (artifak) *storyboard* yaitu berupa deretan ilustrasi gambar yang terurut. Mirip dengan ilustrasi komik, *storyboard* dalam UI Design menunjukkan cerita yang terurut dari pengenalan, konflik masalah yang dihadapi, dan penyelesaian yang dilakukan. Pengenalan menunjukkan sosok profil *user*, yang memperlihatkan dimana dia tinggal/bekerja. Konflik masalah yang dihadapi menunjukkan persoalan yang sedang dihadapi *user* yang terkait dengan topik aplikasi yang dikembangkan. Penyelesaian menunjukkan apa yang bisa dilakukan oleh aplikasi untuk dapat membantu *user* menyelesaikan persoalan tersebut. Setiap tahapan bisa digambarkan melalui 1-3 screen, dimana antar screen diberi pengait. Pengait ini biasanya berbentuk garis penghubung yang menunjukkan sebab-akibat, misal: screen B tampil karena melakukan interaksi tekan tombol X di screen A.

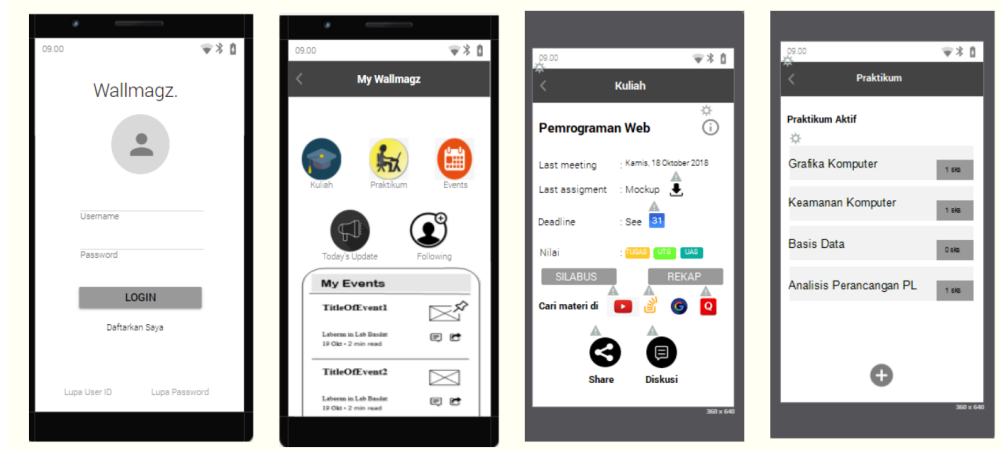
Proses *storyboarding* dilakukan secara kolaborasi seluruh anggota tim. Semua tim berdiskusi alur yang paling optimal untuk dijalani user dari start membuka aplikasi hingga menyelesaikan goal mereka. Alat yang dibutuhkan (fisik) yaitu papan tulis, alat tulis, dan kertas post-it. Apabila kolaborasi ini harus diadakan secara daring, maka alat-alat tersebut dapat digantikan dengan Google Meet dan Google Jamboard. Sediakan ruang yang cukup untuk menunjukkan hasil *storyboarding* dengan membuat sekitar 8 *screen* dari awal *user* mulai

berinteraksi dengan aplikasi hingga *goal* tercapai. Setiap *screen* biasanya berawal dari 1 buah *sticky note* yang bertuliskan 1 tahapan. INGAT: setiap *screen* harus saling berkaitan karena menunjukkan jalan cerita. Apabila membutuhkan alat untuk menggambar, maka dapat dilakukan dengan mengunggah foto ke dalam Google Jamboard. Foto bisa jadi hanya coret-coretan rancangan *screen*, yang tentu saja tidak sebagus yang seharusnya tampil di *prototype*. Tetapi, foto *storyboard* ini akan dapat membantu dalam membuat versi yang lebih estetik karena sudah terbayang komponen apa yang akan diletakkan di *prototype* calon aplikasi nantinya (misal akan ada *image list*, menu di suatu sisi, *scroll*, tombol *back*, *text field*, *icon sorting*, *filtering*, dsb).



Gambar 4.1. Low Fidelity Prototyping

Pada penelitian yang dilakukan oleh (Mardhia & Anggraini, 2019), proses *storyboard* dipresentasikan melalui fase low-fidelity dengan *paper sketch* (Gambar 4.1) lalu didemonstrasikan pada seluruh tim sehingga didapat umpan balik mengenai fitur yang akan dijadikan jalan cerita. Aplikasi *monitoring* studi mahasiswa yang dicetuskan konsepnya fokus pada pemberian informasi berkala mengenai aktivitas kuliah di kelas setiap minggu, termasuk tugas, presensi dan pemberian nilainya. Setiap pengguna dari target pengguna mahasiswa harus dapat memantau aktivitas kuliah mingguan sehingga tidak ada alasan mahasiswa tidak *up-to-date* pada materi dan rencana ujian.

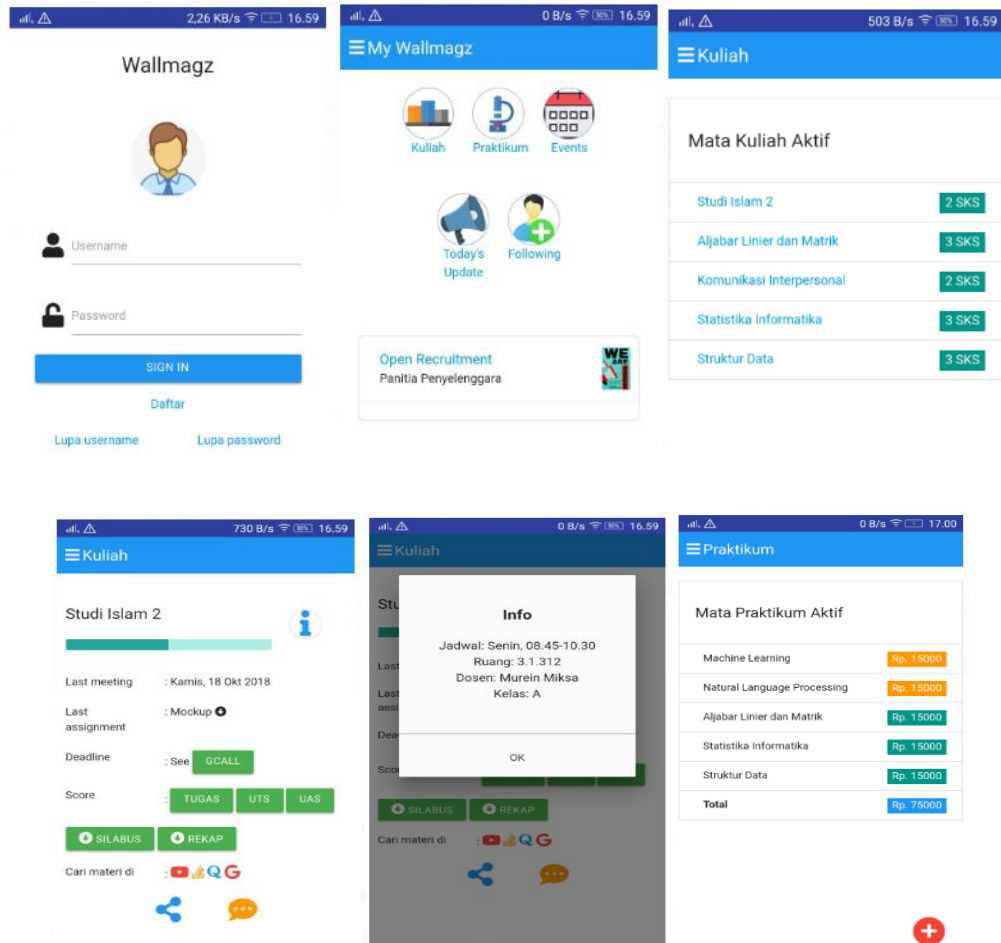


Gambar 4.2. *High Fidelity Prototyping*

Sumber: (Mardhia & Angraini, 2019)

Gambar 4.2 merupakan representasi dari fase mockup yang dihasilkan dari *low-fidelity paper sketch* sebelumnya. Tahap ini penting untuk menunjukkan kepada tim dan perwakilan target pengguna mengenai apa yang akan mereka hadapi saat berinteraksi dengan aplikasi. Beberapa terobosan diberikan di bagian detail informasi matakuliah sehingga dapat menjawab pertanyaan seperti, “dimana saya dapat mendapat bantuan untuk mempelajari materi ini atau sekedar membantu mengerjakan tugas?” dengan memberi icon website-website yang populer dan punya frekuensi tinggi diakses oleh orang-orang sedang belajar di bidang yang sama, seperti contohnya *web* atau *mobile programming*.

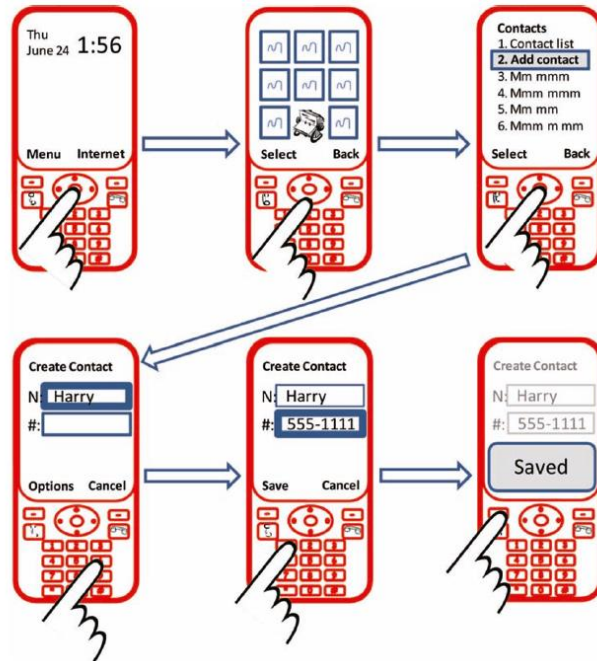
Selanjutnya anggota tim yang memegang pekerjaan sebagai *front end engineer* bertanggungjawab untuk mengimplementasikan desain mockup ke dalam platform yang disepakati (Gambar 4.3). Tahap pengembangan awal masih menggunakan pendekatan *responsive web design* dimana proses implementasi dilakukan di platform web namun tampilan dapat responsif pada ukuran layar website dan *mobile*. Tahap pengembangan di iterasi kedua baru melibatkan *native Android*.



Gambar 4.3. Implementasi UI di Platform Android

### 4.3. Narasi dalam Visual

*Storyboard* dasar hanyalah urutan gambar, di mana setiap gambar menunjukkan sebuah momen waktu - keadaan antarmuka - dalam cerita visual. Anda sudah tidak asing lagi dengan ide papan cerita. Beginilah cara kerja buku komik, dan Anda mungkin tahu itu industri film merencanakan filmnya melalui papan cerita. Salah satu cara berpikir storyboard adalah sebagai kumpulan sketsa, dimana masing-masing sketsa adalah bingkai kunci atau keadaan yang menggambarkan momen penting dalam antarmuka urutan seperti yang terungkap seiring waktu. 'Ruang' di antara bingkai-bingkai ini menangkap transisi antar status, yang biasanya terjadi sebagai hasil dari aktivitas pengguna. Dalam storyboard, transisi dapat berfungsi sebagai ruang putih, yang terserah imajinasi pemirsa untuk mengisinya.



Gambar 4.4. Mendesain *Storyboard* dengan Alur antar Layar

Sumber: *Sketching User Experience* (Greenberg, Carpendale, Marquardt, & Buxton, 2012)

Papan cerita sekuensial adalah teknik yang umum digunakan yang menceritakan kisah visual pengguna urutan pengalaman yang terungkap dari waktu ke waktu (Gambar 4.4). Tantangan utama dalam *storyboard* adalah memutuskan sketsa apa yang akan dimasukkan sebagai bingkai kunci, dan apakah pemirsa dapat mengisi ruang secara mental - transisi - antara bingkai ini. Memberi anotasi pada papan cerita dapat membantu di sini, khususnya dengan menjelaskan interaksi pengguna yang terjadi selama transisi.

#### 4.4.Latihan dan Evaluasi

Siapkan kertas A4 atau yang serupa. Lipat menjadi 8 kotak (seukuran layar *smartphone*). Jelaskan langkah-langkah ide aplikasi tugas proyek Anda, mulai dari halaman pertama pengguna memulai perjalanannya dengan aplikasi hingga pengguna mencapai tujuannya. Tutorial yang disarankan oleh dosen pengampu untuk disimak dalam membantu proses/teknik storyboarding dapat dilihat pada URL berikut: [https://youtu.be/GukAT\\_S8WH8](https://youtu.be/GukAT_S8WH8)



## BAB 5. PENGUJIAN KEBERGUNAAN (USABILITY TESTING)

*Bagaimana cara mendefinisikan kebergunaan/usability? Bagaimana, kapan, dan di mana harus meningkatkannya? Mengapa kita harus peduli? Bab ini membahas mengenai gambaran umum konsep utama usability.*

### 5.1. Tujuan Instruksional

#### A. Tujuan Instruksional Umum

Mahasiswa memahami konsep pentingnya menguji antarmuka aplikasi.

#### B. Tujuan Instruksional Khusus

Mahasiswa memahami metode dan menguji kebergunaan aplikasi.

### 5.2. Pengantar Pengujian Produk

Fase Pengujian/*Testing* adalah bagian kelima dari metode *Design Thinking Process*. Setelah fungsionalitas utama dirasa telah selesai diimplementasikan, produk aplikasi yang dibuat harus melalui proses pengujian dengan tujuan memvalidasi produk yang dihasilkan dari sudut pandang target pengguna. Sebelum melakukan proses pengujian yang sesungguhnya, tim harus mengetahui dahulu mengenai teori/konsep pengujian dalam IMK, dan mengetahui langkah-langkah apa saja yang harus disiapkan. Ketika mengujikan produk aplikasi, tim harus berusaha untuk mendapatkan responden dari kelompok target pengguna yang sesungguhnya. Hal ini bertujuan untuk memastikan hasil pengujian tetap kredibel. Bayangkan apa yang terjadi bila produk aplikasi dengan tema pendidikan anak usia dini diujikan kepada responden kelompok remaja atau mahasiswa?

Salah satu jenis pengujian dalam produk perangkat lunak adalah pengujian kegunaan atau *usability*. Kegunaan adalah atribut kualitas yang menilai seberapa mudah antarmuka pengguna digunakan. Kata "kegunaan" juga mengacu pada metode untuk meningkatkan kemudahan penggunaan selama proses desain.

Berdasarkan (Nielsen, 2000), kegunaan ditentukan oleh 5 komponen kualitas:

- **Learnability:** Seberapa mudah bagi pengguna untuk menyelesaikan tugas-tugas dasar saat pertama kali menemukan desain?

- **Efficiency:** Setelah pengguna mempelajari desain, seberapa cepat mereka dapat melakukan tugas?
- **Memorability:** Ketika pengguna kembali ke desain setelah beberapa saat tidak menggunakannya, seberapa mudah mereka dapat membangun kembali kemahirannya?
- **Errors:** Berapa banyak kesalahan yang dilakukan pengguna, seberapa parah kesalahan ini, dan seberapa mudah mereka dapat memulihkan kesalahan?
- **Satisfaction:** Seberapa menyenangkan menggunakan desain?

Ada banyak atribut kualitas penting lainnya. Kunci utamanya adalah utilitas, yang mengacu pada fungsionalitas desain: Apakah ia melakukan apa yang dibutuhkan pengguna? Kegunaan dan utilitas sama pentingnya dan bersama-sama menentukan apakah sesuatu itu berguna: Tidak masalah bahwa sesuatu itu mudah jika itu bukan yang Anda inginkan. Juga tidak baik jika sistem secara hipotetis dapat melakukan apa yang Anda inginkan, tetapi Anda tidak dapat mewujudkannya karena antarmuka pengguna terlalu sulit. Untuk mempelajari kegunaan desain, Anda dapat menggunakan metode penelitian pengguna yang sama yang meningkatkan kegunaan.

- Definisi Utilitas = apakah ia menyediakan fitur yang Anda butuhkan.
- Definisi Kegunaan = seberapa mudah & menyenangkan fitur-fitur ini digunakan.
- Definisi Berguna = kegunaan + utilitas.

Kegunaan memainkan peran dalam setiap tahap proses desain. Berikut langkah-langkah utamanya:

1. Sebelum memulai desain baru, uji desain lama untuk mengidentifikasi bagian baik yang harus Anda pertahankan atau tekankan, dan bagian buruk yang menyulitkan pengguna.
2. Uji desain pesaing/kompetitor Anda untuk mendapatkan data tentang berbagai antarmuka alternatif yang memiliki fitur serupa dengan milik Anda. (Jika Anda bekerja di intranet, baca desain intranet tahunan untuk belajar dari desain lain.)
3. Melakukan studi lapangan untuk melihat bagaimana perilaku pengguna di habitat aslinya.

4. Buatlah prototipe kertas dari satu atau lebih ide desain baru dan ujilah. Semakin sedikit waktu yang Anda investasikan dalam ide desain ini semakin baik, karena Anda harus mengubah semuanya berdasarkan hasil pengujian.
5. Sempurnakan ide desain yang paling baik diuji melalui beberapa iterasi, secara bertahap beralih dari pembuatan prototipe *low-fidelity* ke representasi *high-fidelity* yang berjalan di komputer. Uji setiap iterasi.
6. Periksa desain relatif terhadap pedoman kegunaan/*usability guidelines* yang ditetapkan baik dari penelitian Anda sebelumnya atau penelitian yang diterbitkan.
7. Setelah Anda memutuskan dan menerapkan desain akhir, ujilah lagi. Masalah kegunaan yang 'samar' selalu muncul selama implementasi.

Satu-satunya cara untuk mendapatkan pengalaman pengguna yang berkualitas tinggi adalah dengan memulai pengujian pengguna di awal proses desain dan terus menguji setiap langkahnya.

### **5.3. Evaluasi Sumatif**

Evaluasi sumatif sering dilakukan di bawah payung pengujian kegunaan dilakukan pada akhir proyek setelah sistem dibangun, untuk menilai apakah itu memenuhi spesifikasinya, atau apakah proyek itu berhasil. Hal ini berbeda dengan evaluasi formatif, dimana tujuan utamanya adalah untuk memberikan kontribusi pada desain produk, dengan menilai spesifikasi atau prototipe sebelum sistem dibangun. Evaluasi formatif seringkali bersifat analitik (hasil dari penalaran tentang desain), sedangkan evaluasi sumatif seringkali bersifat empiris (dilakukan dengan melakukan observasi atau pengukuran). Evaluasi sumatif juga sering digunakan dalam situasi penelitian, di mana teknik interaksi baru dinilai untuk publikasi ilmiah. Namun, evaluasi sumatif tidak begitu populer di lingkungan komersial seperti di lingkungan akademis.

Setelah sistem dibangun, pembuatnya cenderung tidak tertarik dengan saran lebih lanjut - banyak perusahaan kecil menganggap bahwa merilis suatu produk sangat murah sehingga mereka mungkin juga merilisnya sambil mengujinya. Masalah kegunaan apa pun dapat diselesaikan di versi 2, sebagai tanggapan atas umpan balik pengguna. Alternatifnya adalah dengan menggunakan teknik kegunaan yang tidak seketat studi akademis, tetapi tetap memberikan lebih banyak informasi daripada menyilangkan jari dan berharap pengguna akan menyukainya (meskipun itu merupakan pendekatan yang sangat umum di perusahaan kecil).

Perusahaan yang lebih besar dan lebih mapan membelanjakan lebih banyak untuk evaluasi sumatif produk baru, karena bahaya bagi reputasi mereka jika mereka merilis produk yang jauh lebih rendah. Untuk alasan ini, perusahaan seperti Microsoft melakukan studi evaluasi sumatif semua produk, bahkan sebelum mereka menjalani rilis pasar (beta) lebih awal. Masalah kegunaan kemudian dapat dilacak dan diselesaikan dengan cara yang sama seperti kerusakan perangkat lunak lainnya, menggunakan proses yang sama seperti untuk *bug* fungsional yang ditemukan selama pengujian sistem.

#### 5.4. Eksperimen Terkontrol

Metode empiris yang paling umum digunakan dalam penelitian IMK yang berasal dari faktor manusia dan psikologi eksperimental adalah eksperimen terkontrol. Eksperimen didasarkan pada sejumlah observasi (pengukuran dilakukan saat seseorang menggunakan antarmuka eksperimental). Ukuran yang umum mungkin adalah "Berapa lama User X menyelesaikan tugas A?" atau "Berapa banyak kesalahan yang dia buat?" Berbagai macam pengukuran alternatif dimungkinkan, termasuk detak jantung atau data biologis eksotis lainnya. Namun, pengamatan tunggal terhadap kecepatan tidaklah terlalu menarik. Jika X melakukan tugas itu lagi, dia akan membutuhkan waktu yang berbeda, dan jika orang lain melakukannya, itu akan membutuhkan jumlah waktu yang lebih berbeda. Oleh karena itu perlu ada beberapa set pengukuran, dan dibandingkan rata-ratanya. Kumpulan mungkin beberapa pengamatan dari satu orang yang melakukan tugas selama banyak percobaan, atau dari berbagai orang (peserta eksperimen) yang melakukan tugas yang sama dalam kondisi terkontrol.

Seperti kebanyakan kinerja manusia, hasil yang diukur biasanya diketahui memiliki distribusi normal. Eksperimen IMK tipikal melibatkan satu atau beberapa perlakuan eksperimental yang memodifikasi antarmuka pengguna. Contoh yang sangat sederhana mungkin menguji pertanyaan: "Berapa lama X menyelesaikan tugas A saat menggunakan UI yang baik, dibandingkan dengan UI yang buruk?" Hasilnya sering kali UI yang baik biasanya lebih cepat digunakan daripada yang buruk, tetapi tidak di setiap percobaan. Jika kita memplot pengukuran, kita menemukan dua distribusi normal yang tumpang tindih, dan oleh karena itu kita harus membandingkan pengaruh perlakuan relatif terhadap penyebaran dalam distribusi populasi. Kita perlu mengetahui apakah perbedaan antara rata-rata adalah hasil dari variasi acak biasa, atau efek dari perubahan yang kita buat pada antarmuka pengguna. Ini melibatkan uji signifikansi statistik seperti uji-*t*. Uji-*t* dan tes serupa lainnya menjawab pertanyaan "Berapa probabilitas perbedaan sarana yang diamati dapat terjadi hanya dengan variasi acak?". Gagasan bahwa perbedaan eksperimental mungkin hanya variasi acak disebut hipotesis nol, dan penting

untuk diingat bahwa ini selalu memungkinkan dalam eksperimen apa pun. Kami biasanya berharap probabilitasnya sangat rendah - yaitu perbedaan yang diamati adalah karena kami merancang antarmuka yang sangat bagus, bukan keberuntungan.

Dalam penelitian IMK, kami biasanya bersikeras bahwa probabilitas hasil yang disebabkan oleh variasi acak ( $p$ ) kurang dari 0,05 atau 5%. Hasil penelitian berkualitas baik biasanya didasarkan pada eksperimen dengan nilai signifikansi  $p < 0,01$ , yang dapat dinyatakan sebagai 'kami menolak hipotesis nol, dengan keyakinan 99%'. Jika sejumlah besar data tersedia (misalnya, jika Anda adalah Google dan memiliki jutaan orang yang berinteraksi dengan produk Anda), maka pengukuran biner sederhana mungkin cukup untuk memberikan evaluasi yang berguna. Ini biasanya digunakan dalam pengujian A/B, di mana dua versi desain dibuat lalu dibandingkan - apakah pengguna lebih cenderung mengklik tombol dalam desain A atau desain B? Pengujian signifikansi dengan data biner dapat dilakukan dengan menggunakan uji binomial untuk melihat apakah variasi antara keduanya lebih besar dari yang diharapkan secara kebetulan.

### **Studi *Thinking aloud* / *Think-Aloud Protocol***

Meskipun bukan benar-benar eksperimen (sering dilakukan tanpa hipotesis, dan datanya kualitatif daripada kuantitatif), studi terkontrol di IMK sering menggunakan teknik *Thinking Aloud* untuk mendapatkan wawasan tentang cara pengguna menafsirkan prototipe. Perhatian besar dilakukan untuk memastikan bahwa pengguna menyuarakan setiap pemikiran yang mereka sadari, dan rekaman ditranskripsikan dan dianalisis secara rinci untuk bukti proses mental tertentu. Namun dalam konteks komersial, protokol *think-aloud* dapat terlihat lebih seperti umpan balik evaluasi waktu nyata, di mana pengguna hanya diminta untuk membuat komentar sebanyak mungkin pada antarmuka pengguna. Ini mungkin tidak memberikan banyak wawasan ilmiah, tetapi setidaknya ini menghindari masalah pengguna yang menghabiskan waktu satu jam menggunakan sistem baru, lalu hampir tidak mengatakan apa-apa sebagai umpan balik.

## **5.5. Latihan dan Evaluasi**

Setelah mempelajari persiapan pengujian Usability, perkuliahan IMK akan membahas tentang pelaksanaan pengujian. Di bagian ini, tim harus sudah selesai menetapkan calon responden yang akan mengikuti pengujian, menyusun dokumen pengujian (yang berisi *task-based scenario*, *pre-test*, *post-test*) dan sudah membagi pekerjaan ke semua anggota dalam tim.

Selama minggu ke-13, tim fokus ke melakukan pengujian *usability*. Saat proses pengujian, tim juga harus membawa sebuah dokumen tambahan yang berupa kuesioner pengujian yang mengukur tingkat kebergunaan aplikasi. Dari berbagai macam kuesioner, perkuliahan ini akan memberikan panduan penggunaan kuesioner yang dasar dan sederhana, yaitu SUS (Software Usability Scale). Template kuesioner tersebut salah satunya disediakan di tautan berikut: <https://www.usability.gov/how-to-and-tools/resources/templates/system-usability-scale-sus.html>

Dokumen SUS ini tidak perlu diubah-ubah untuk menjaga validitas isi kuesioner yang telah baku. Metode perhitungan dan interpretasi hasil dapat disimak pada materi berikut: <https://drive.google.com/file/d/1fmcthvHJFHN93NZzvZRKJ6yemmSQLJgy/view?usp=sharing>.

Di akhir fase sprint, semua tim harus melengkapi pekerjaan proyeknya masing-masing hingga hasil pengujian produk/prototype. Komponen yang dilaporkan dari proses pengujian antara lain:

1. Success Rate
2. Time-on-Task
3. Usability Score
4. Summary Feedback Responden

Penjelasan dan contoh dari 4 poin di atas akan disampaikan melalui video tutorial di link berikut:

[https://drive.google.com/file/d/1C\\_D58paxoPSetfIO5u2wmHyaB59Nt8BJ/view?usp=sharing](https://drive.google.com/file/d/1C_D58paxoPSetfIO5u2wmHyaB59Nt8BJ/view?usp=sharing).

## **BAB 6. PERKEMBANGAN PENELITIAN DI BIDANG IMK**

Kuliah ini merangkum penelitian IMK terkini dan terkini ke dalam teknologi interaksi lanjutan untuk meninjau prinsip-prinsip yang diperkenalkan di tempat lain terkait materi ini. Bagian ini diterjemahkan dan disunting seperlunya dari (Blackwell, 2007).

### **6.1. Tujuan Instruksional**

#### **A. Tujuan Instruksional Umum**

Mahasiswa memahami penerapan teori IMK dalam topik pengembangan penelitian.

#### **B. Tujuan Instruksional Khusus**

Mahasiswa memahami contoh teknologi yang diaplikasikan dalam kebutuhan pengembangan antarmuka.

### **6.2. Virtual dan Augmented reality**

Sistem *augmented reality* (AR) menghamparkan informasi digital ke dunia nyata, baik menggunakan tampilan yang dipasang di kepala sebagian transparan seperti pada proyek Google Glass, atau dengan mengambil umpan video dari adegan aktual, dan mengomposisikannya dengan elemen yang dihasilkan komputer. Masalah teknis utama adalah registrasi - relatif baru-baru ini, ini harus dilakukan dengan mengintegrasikan GPS, orientasi kompas, akselerometer untuk orientasi gravitasi, dan seringkali giroskop, ke dalam HMD. Sekarang semua periferal ini tersedia di ponsel kelas atas, Mobile AR menjadi kata kunci pemasaran utama, mungkin dengan hilangnya fungsi aktual yang sama seperti saat VR bergeser dari idealisme penelitian ke jargon bisnis dan pemasaran.

### **6.3. Tangible User Interface**

Antarmuka pengguna berwujud (TUI) menggunakan objek fisik untuk mengontrol komputer, paling sering adalah kumpulan objek yang disusun di atas meja untuk bertindak sebagai 'ikon fisik'. Persoalan nyatanya adalah objek fisik tidak mengubah keadaan terlihatnya dengan sangat mudah. Anda dapat menyertakan motor penggerak dan tampilan di setiap objek (mahal), atau memproyeksikan informasi AR ke atas lapisannya, atau cukup gunakan sebagai beberapa mouse / pucks khusus yang mengontrol elemen dari 23 tampilan pada layar terpisah. Dalam hal ini, perlu untuk melacak posisi mereka, mungkin dengan menggunakan perangkat tablet yang besar. Jika mereka hanya digunakan sebagai token untuk memilih fungsi atau bagian data tertentu, chip RFID tertanam dapat digunakan untuk mendeteksi ketika ditempatkan dalam jarak tertentu dari pembaca.

#### **6.4. Machine Vision**

Visi mesin adalah teknologi kunci untuk AR dan TUI, sebagai cara untuk mengenali objek dunia nyata seperti bangunan (dalam kasus AR luar ruang) atau objek di atas meja (digunakan untuk TUI). Banyak prototipe AR saat ini mengenali objek khusus dari sejumlah besar fitur visual tingkat rendah, seperti dalam algoritme SIFT. Masalah utamanya adalah mempertahankan database fitur objek yang cukup besar, melacaknya dengan cukup cepat untuk memberikan umpan balik pengguna yang merespons kamera, gestur, atau gerakan objek secara realtime, dan melakukan keduanya dalam berbagai kondisi pencahayaan. Alternatifnya adalah penanda *fidusia* - penanda visual sederhana seperti kode batang, yang dapat digunakan untuk mengidentifikasi dan melacak objek dengan lebih andal dari input kamera. Mereka lebih kuat untuk perubahan sudut kamera dan pencahayaan daripada algoritma pengenalan objek.

#### **6.5. Paper Interfaces**

Inspired by the research conducted by Abigail Sellen and Richard Harper (originally at Xerox EuroPARC in Cambridge, now at Microsoft Research Cambridge), whose book 'The Myth of the Paperless Office' analyses the ways in which the properties of paper are preferable to computers for many kinds of activity. The book remains a useful resource for designers of mobile devices substituting for paper (phones and tablets), but has also inspired research in which paper is integrated with digital systems, for example with fiducial markers on the page that can be traced by cameras (the Anoto digital pen can perhaps be considered an extreme example of gesture recognition implemented with fiducial markers).

#### **6.6. Mixed Reality**

*Mixed Reality* menggabungkan objek fisik dengan tampilan informasi, misalnya dengan memproyeksikan data digital ke objek di atas meja, atau di atas kertas. Penanda *fidusia* dapat digunakan untuk menentukan identitas dan lokasi setiap lembar kertas, dan memproyeksikan informasi tambahan ke dalamnya. Seri konferensi ISMAR menyajikan hasil baru dalam Mixed and Augmented Reality, sering kali berdasar pada teknik visi mesin.

#### **6.7. Eye Tracking dan Gaze Control**

Awalnya dikembangkan untuk penelitian psikologis ke dalam proses perhatian visual, *eyetrackers* sekarang digunakan secara rutin dalam penelitian IMK untuk mempelajari posisi



apa yang dilihat oleh 24 pengguna di layar. Kamera *close-up* resolusi tinggi digunakan untuk merekam video salah satu mata pengguna, dan posisi tepat di mana mereka melihat disimpulkan dari posisi pupil, sering kali dikombinasikan dengan pantulan dari sepasang lampu sorot infra merah kecil (LED). Satu perusahaan menjual perangkat dengan kamera dan lampu sorot yang terintegrasi ke sekeliling monitor, agar tidak mengganggu. Namun, hampir semua sistem seperti ini mengharuskan pengguna untuk duduk diam, dan menjalani prosedur kalibrasi di mana mereka melihat titik-titik di layar secara berurutan. Performa bisa menjadi buruk jika ada pencahayaan sekitar yang kuat, saat pengguna memakai kacamata, memiliki mata berair atau kulit berkilau. Seringkali latihan dibutuhkan untuk mendapatkan hasil yang baik. *Eyetracker* kadang-kadang digunakan untuk membuat antarmuka yang menggunakan mata sebagai pengendali. Pada pandangan pertama, tampaknya ini sangat alami dan intuitif untuk digunakan. Dalam praktiknya, gerakan mata alami dari fiksasi dan saccades dapat membingungkan algoritme inferensi pelacak mata, sulit untuk menjaga mata Anda tetap terpaku pada lokasi kontrol untuk periode waktu yang cukup lama, dan godaan alami untuk melirik ke tempat lain (periksa pekerjaan yang sedang berlangsung, melihat waktu, melihat ke bawah ke tangan Anda dll) atau berkedip berlebihan harus terus berjuang.

### **6.8. Surface dan Interaksi Tabletop**

Surface/tabletop interaction uses large display areas, usually projected, on a flat surface such as a wall or table. User interaction takes place by touching, gesturing, or pointing at the display. Many of these systems use camera input, with more accurate recognition of the users hands possible by using infrared, rather than visible light. A low-powered infrared spotlight is often used to illuminate the scene, rather than relying on body heat (which can be confused by other hot objects in the environment – such as computers!). A popular technique at present is frustrated total internal reflection (FTIR), where infrared light is shone inside a flat transparent medium such as a glass panel, and anything touching the surface causes infrared to be scattered. This technique can be used to recognize fingertip touches, or gestures involving more skin contact, such as multiple fingers or even a flat palm. A motivating scenario for many of these systems has been the gesture-controlled projection interface in the movie *Minority Report*.

### **6.9. Latihan dan Evaluasi**

Carilah artikel jurnal/paper yang menggunakan teknologi antarmuka seperti yang dijelaskan di subbab sebelumnya, buat ringkasan supaya dapat memahami korelasi penerapan teknologi tersebut terhadap kegunaan dan utilitas yang dihasilkan.

## DAFTAR PUSTAKA

- Blackwell, A. (2007). *Human Computer Interaction Notes*. Retrieved from <https://www.cl.cam.ac.uk/teaching/0708/HCI/HCI2007.pdf>
- Greenberg, S., Carpendale, S., Marquardt, N., & Buxton, B. (2012). Sketching User Experiences. In *Sketching User Experiences*. <https://doi.org/10.1016/C2009-0-61147-8>
- Mardhia, M. M., & Anggraini, E. D. (2019). Implement a Lean UX Model: Integrating Students' Academic Monitoring through a mobile app. *Proceedings - 2019 International Conference on Advanced Informatics: Concepts, Theory, and Applications, ICAICTA 2019*. <https://doi.org/10.1109/ICAICTA.2019.8904323>
- Nielsen, J. (2000). Designing Web Usability. In *Interactive Marketing*.
- Norman, D. (2016). The Design of Everyday Things. In *The Design of Everyday Things*. <https://doi.org/10.15358/9783800648108>
- Preece, J., Sharp, H., & Rogers, Y. (2015). Interaction Design - Beyond Human-Computer Interaction, Fourth Edition. *Journal of Chemical Information and Modeling*.

