

**SKRIPSI**

**PERANCANGAN *USER INTERFACE* DAN *USER EXPERIENCE* PADA  
*VIDEO CONFERENCE* JITSI MEET MENGGUNAKAN REACTJS**

**Disusun dan diajukan oleh:**

**ULFAH ROJIYYAH**

**D421 14 512**



**DEPARTEMEN TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK**

**UNIVERSITAS HASANUDDIN**

**MAKASSAR**

**2021**

**LEMBAR PENGESAHAN SKRIPSI**

**PERANCANGAN USER INTERFACE DAN USER EXPERIENCE PADA VIDEO  
CONFERENCE JITSI MEET MENGGUNAKAN REACTJS**

Disusun dan diajukan oleh

**ULFAH ROJIYYAH**

**D421 14 512**

Telah dipertahankan di hadapan Panitia Ujian yang dibentuk dalam rangka Penyelesaian Studi

Program Sarjana Program Studi Teknik Informatika

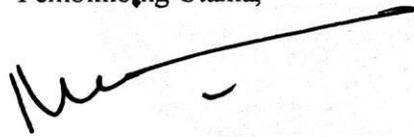
Fakultas Teknik Universitas Hasanuddin

Pada tanggal 13 Oktober 2021

dan dinyatakan telah memenuhi syarat kelulusan

Menyetujui

Pembimbing Utama,



Dr. Eng. Muhammad Niswar, S.T., M.IT.  
Nip. 19730922 199903 1 001

Pembimbing Pendamping,



Iqra Aswad, S.T., M.T  
Nip. 199011282019043001



Ketua Program Studi

Dr. Amil Ahmad Ilham, S.T., M.IT  
Nip. 19731010 199802 1 001

## **PERNYATAAN KEASLIAN**

Yang bertanda tangan dibawah ini :

Nama : ULFAH ROJIYYAH

Nim : D421 14 512

Program Studi : S1 Teknik Informatika

Menyatakan dengan sebenar-benarnya bahwa skripsi yang berjudul :

### ***PERANCANGAN USER INTERFACE DAN USER EXPERIENCE PADA VIDEO CONFERENCE JITSI MEET MENGGUNAKAN REACTJS***

Adalah karya ilmiah saya sendiri dan sepanjang pengetahuan saya di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan/ditulis/diterbitkan sebelumnya, kecuali yang secara tertulis dikutip dalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka.

Apabila dikemudian hari ternyata didalam naskah skripsi ini terdapat unsur-unsur djiplakan, saya bersedia menerima sanksi atas perbuatan tersebut dan diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2000, pasal 25 ayat 2 dan pasal 70).

Makassar, 14 Oktober 2021

Yang membuat Pernyataan

  
**ULFAH ROJIYYAH**

## ABSTRAK

Sejak Pandemi covid-19 semua kegiatan yang dilakukan di luar rumah, sekarang harus dilakukan di dalam rumah, termasuk kegiatan belajar mengajar. Salah satu teknologi yang digunakan untuk memudahkan proses belajar mengajar dari rumah dan mudah diakses menggunakan internet ialah video conference. Jitsi Meet adalah layanan *video conference* bersifat *opensource* yang sepenuhnya terenkripsi sehingga semua pembicaraan di dalam Jitsi Meet bersifat rahasia dan tidak bisa dilihat orang yang tidak berkepentingan. Dengan adanya *video conference* dengan desain yang lebih interaktif dan mudah digunakan tidak lepas dari yang namanya *user interface* dan *user experience*. *User interface* dan *user experience* berfungsi untuk menghubungkan pengguna dengan informasi yang dibutuhkan dari sistem itu sendiri. Software atau perangkat lunak yang dibutuhkan untuk mengimplementasikan rancangan sistem adalah NodeJS Visual Code, Jitsi Meet, Bootstrap, React JS, WinSCP. Perancangan sistem dengan langkah-langkah yaitu Analisis Kebutuhan, Perancangan Sistem, Desain UI/UX, Pengujian. Hasil dari pengujian yang melibatkan 25 responden mahasiswa dan 5 responden dosen dengan metode *cognitive walkthrough* dan metode SUS (*System Usability Scale*).

**Kata Kunci** : *user interface, user experience, video conference, jitsi meet, cognitive walkthrough, system usability scale.*

## KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakatuh.

Alhamdulillahirabbil'alamin Puji dan syukur penulis panjatkan kehadirat Allah SWT, atas segala karunia dan rahmat-Nya sehingga penulis dapat menyelesaikan penyusunan tugas akhir ini dengan judul "**Perancangan *User Interface Dan User Experience* Pada *Video Conference Jitsi Meet Menggunakan Reactjs***" di susun untuk memenuhi sebagian persyaratan kelulusan untuk menyelesaikan program Strata-1 Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin.

Pada kesempatan ini penulis menyadari keterbatasan dan kelemahan yang ada dalam menyelesaikan tugas akhir ini sehingga penulis ingin menyampaikan ucapan terima kasih kepada pihak-pihak yang telah memberikan dukungan, bimbingan dan motivasi kepada penulis untuk menyelesaikan tugas akhir ini, kepada:

1. Allah SWT, yang telah memberikan rahmat dan karunia-Nya yang telah diberikan kepada penulis disetiap langkah dalam penulisan dan penyusunan skripsi ini.
2. Kedua orang tua yang tak henti-hentinya memberikan dukungan kepada penulis hingga sekarang, saudara-saudari penulis beserta keluarga yang selalu memberikan doa, semangat, motivasi, dan kasih sayang kepada penulis.
3. Bapak Dr.Eng. Muhammad Niswar, S.T., M.IT., selaku pembimbing 1 dan kepala Laboratorium Ubiquitous Computing & Networking Lab atau yang

lebih dikenal dengan nama LAB UBICON yang telah banyak memberi bimbingan, inspirasi, motivasi, dan masukan yang bermanfaat selama masa pengerjaan tugas akhir penulis.

4. Bapak Iqra Aswad, S.T., M.T selaku pembimbing II yang telah banyak memberi keyakinan, perhatian, bimbingan, motivasi, dan masukan yang bermanfaat kepada penulis.
5. Bapak Ir. Christoforus Yohannes, M.T. dan Bapak Dr. Ir Zahir Zainuddin, M.Sc. Selaku penguji yang telah memberikan saran kepada penulis sehingga skripsi ini dapat berjalan dengan baik.
6. Bapak Dr. Amil Ahmad Ilham, S.T., M.IT., selaku Ketua Departemen Teknik Informatika, dan semua dosen Departemen Teknik Informatika atas semua ilmu yang telah di berikan.
7. Bapak Robert dan Bapak Zainuddin serta segenap staf Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin yang telah membantu kelancaran penyelesaian tugas akhir penulis.
8. Teman-teman dekat penulis yang selalu ada untuk mengingatkan dan membantu dalam proses penyelesaian skripsi Hermawan Safrin, Yakip, Syarif Hidayatullah, Muh. Nur Alamsyah, Auraevadne, Inka G. Mallisa Anastasia Yuki A. Sande, Tiwi Nur Safitri, Al Riefqy Dasmito, Fitriani Idrus, Winda Astiyanti, Arya Jaka Putra, Rahmat Firman, dan Cindy Oktaviani L.Bulan.

9. Seluruh teman-teman RECTIFIER'14 “Kita Luka Hari Ini Mereka Luka Selamanya”
10. Seluruh teman-teman Teknik Informatika 2014 terkhusus aryandi dan alwi telah membantu memberikan masukan dalam pembuatan sistem.
11. Kepada teman-teman ENJ UNHAS 2018, terima kasih pengalaman yang yang tidak akan terlupakan.
12. Kepada Rahmat Hidayat, Sukirman, Muh.Zulfakhri Marjusi, Muh. Isman Ismail, terima kasih selalu ada hingga sekarang dalam keadaan apapun.
13. Serta seluruh pihak yang tidak sempat disebutkan satu persatu yang telah banyak meluangkan tenaga, waktu, dan pikiran selama penyusunan laporan tugas akhir ini.

Akhir kata, penulis berharap semoga Allah SWT senantiasa membalas segala kebaikan dari semua pihak yang telah banyak membantu. Semoga skripsi ini dapat memberikan manfaat bagi pengembangan ilmu selanjutnya. Aamiin ya Rabbal ‘alamin.

Gowa, 30 April 2021

Penulis

## DAFTAR ISI

|   |      |
|---|------|
| SKRIPSI.....                                      | i    |
| ABSTRAK.....                                      | i    |
| KATA PENGANTAR .....                              | iii  |
| DAFTAR ISI.....                                   | vi   |
| DAFTAR GAMBAR .....                               | viii |
| DAFTAR TABEL.....                                 | x    |
| BAB I PENDAHULUAN.....                            | 1    |
| 1.1 Latar Belakang .....                          | 1    |
| 1.2 Rumusan Masalah .....                         | 2    |
| 1.3 Batasan Masalah.....                          | 2    |
| 1.4 Tujuan Penelitian.....                        | 3    |
| 1.5 Manfaat Penelitian.....                       | 3    |
| 1.6 Sistematika Penelitian .....                  | 3    |
| BAB II.....                                       | 5    |
| 2.1 <i>Website</i> .....                          | 5    |
| 2.2 <i>User Inteface</i> .....                    | 7    |
| 2.3 <i>User Experience</i> .....                  | 10   |
| 2.4 <i>Javascript</i> .....                       | 12   |
| 2.5 <i>React Js</i> .....                         | 13   |
| 2.6 <i>Hypertext Markup Language (HTML)</i> ..... | 14   |
| 2.7 <i>Cascading Style Sheets (CSS)</i> .....     | 15   |
| 2.8 <i>NodeJS</i> .....                           | 16   |
| 2.9 <i>Jitsi Meet</i> .....                       | 17   |
| 2.10 <i>Usability</i> .....                       | 18   |
| 2.11 <i>Cognitive Walkthrough</i> .....           | 20   |
| 2.12 <i>System Usability Scale</i> .....          | 20   |

|   |    |
|---|----|
| BAB III METODOLOGI PENELITIAN .....                       | 21 |
| 3.1 Lokasi dan Waktu Penelitian.....                      | 21 |
| 3.2 Instrumen Penelitian.....                             | 21 |
| 3.3 Prosedur Penelitian.....                              | 22 |
| 3.4 Tahap Persiapan .....                                 | 24 |
| 3.5 Perancangan Sistem.....                               | 24 |
| 3.5.1 Analisis Kebutuhan .....                            | 24 |
| 3.5.2 Perancangan Sistem.....                             | 27 |
| 3.5.3 Desain UI/UX.....                                   | 31 |
| 3.5.4. Implementasi Sistem .....                          | 40 |
| 3.5.5. Pengujian.....                                     | 40 |
| BAB IV HASIL DAN PEMBAHASAN .....                         | 51 |
| 4.1 Hasil Pembuatan Sistem .....                          | 51 |
| 4.1.1 Tampilan Halaman Utama .....                        | 51 |
| 4.1.2 Tampilan Halaman Sign In .....                      | 52 |
| 4.1.3 Tampilan Halaman Sign Up.....                       | 53 |
| 4.1.5 Tampilan Dosen dan Mahasiswa .....                  | 53 |
| 4.2 Hasil Pengujian Cognitive Walkthrough.....            | 61 |
| 4.2.1. Tingkat Keberhasilan Skenario Tugas .....          | 62 |
| 4.2.2 Jumlah Kesalahan yang Dilakukan .....               | 67 |
| 4.2.3 Jumlah Waktu Penyelesaian Skenario Tugas.....       | 71 |
| 4.2.2 Hasil Pengujian <i>System Usability Scale</i> ..... | 76 |
| BAB V PENUTUP.....  | 80 |
| 5.1 KESIMPULAN .....                                      | 80 |
| 5.2 SARAN .....   | 81 |
| DAFTAR PUSTAKA .....                                      | 82 |
| LAMPIRAN.....   | 86 |

## DAFTAR GAMBAR

|  |    |
|--|----|
| Gambar 2. 1 Perbandingan Virtual DOM dan Real DOM.....                 | 14 |
| Gambar 2. 2 Jitsi Architecture Diagram .....                           | 18 |
| Gambar 3. 1 Diagram Tahapan Penelitian .....                           | 22 |
| Gambar 3. 2 State Transition Diagram Dosen .....                       | 28 |
| Gambar 3. 3 State Transition Diagram Mahasiswa .....                   | 30 |
| Gambar 3. 4 Sketsa Halaman Utama .....                                 | 31 |
| Gambar 3. 5 Sketsa Halaman Sign Up.....                                | 32 |
| Gambar 3. 6 Sketsa Halaman Sign In .....                               | 32 |
| Gambar 3. 7 Sketsa Halaman Dosen.....                                  | 33 |
| Gambar 3. 8 Sketsa Halaman Mahasiswa.....                              | 34 |
| Gambar 3. 9 Sketsa Halaman Utama .....                                 | 35 |
| Gambar 3. 10 Sketsa Halaman Sign Up.....                               | 36 |
| Gambar 3. 11 Sketsa Halaman Sign In .....                              | 36 |
| Gambar 3. 12 Sketsa Halaman Dosen.....                                 | 37 |
| Gambar 3. 13 Sketsa Halaman Mahasiswa.....                             | 38 |
| Gambar 3.14 Hubungan jumlah permasalahan dengan jumlah responden ..... | 41 |
| Gambar 3. 15 Tahapan Pengujian Cognitive Walkthrough .....             | 42 |
| Gambar 3. 16 Skala Penilaian Pertanyaan SUS .....                      | 49 |
| Gambar 4. 1 Tampilan Halaman Utama .....                               | 51 |
| Gambar 4. 2 Tampilan Halaman Sign In .....                             | 52 |
| Gambar 4. 3 Tampilan Halaman Sign Up.....                              | 53 |

|   |    |
|---|----|
| Gambar 4. 4 Tampilan Halaman Dashboard Dosen .....        | 54 |
| Gambar 4. 5 Tampilan Halaman Kuliah .....                 | 54 |
| Gambar 4. 6 Tampilan Halaman Tambah Matakuliah.....       | 55 |
| Gambar 4. 7 Tampilan Halaman Edit Matakuliah .....        | 56 |
| Gambar 4. 8 Tampilan Halaman Hapus .....                  | 56 |
| Gambar 4. 9 Tampilan Halaman Detail Matakuliah .....      | 57 |
| Gambar 4. 10 Tombol Menampilkan Video Conference.....     | 57 |
| Gambar 4. 11 Tampilan Jitsi meet .....                    | 57 |
| Gambar 4. 12 Tampilan Halaman Dashboard Mahasiswa.....    | 58 |
| Gambar 4. 13 Menu Selengkapnya .....                      | 59 |
| Gambar 4. 14 Tampilan Halaman Daftar Kuliah .....         | 59 |
| Gambar 4. 15 Tampilan Detail Kuliah .....                 | 60 |
| Gambar 4. 16 Tombol Keluar .....                          | 60 |
| Gambar 4. 17 Nilai System Usability Scale.....            | 78 |
| Gambar 4. 18 Presentase Nilai System Usability Scale..... | 79 |

## DAFTAR TABEL

|  |    |
|--|----|
| Tabel 3. 1 Tabel Survei Perilaku Pengguna .....                        | 25 |
| Tabel 3. 2 Skenario Tugas Dosen .....                                  | 43 |
| Tabel 3. 3 Skenario Tugas Mahasiswa.....                               | 44 |
| Tabel 4. 1 Tingkat Keberhasilan Responden Dosen .....                  | 62 |
| Tabel 4. 2 Tingkat Keberhasilan Responden Mahasiswa .....              | 64 |
| Tabel 4. 3 Jumlah Kesalahan yang Dilakukan Responden Dosen .....       | 67 |
| Tabel 4. 4 Jumlah Kesalahan yang Dilakukan Responden Mahasiswa .....   | 69 |
| Tabel 4. 5 Jumlah Waktu Penyelesaian Responden Dosen (Detik) .....     | 72 |
| Tabel 4. 6 Jumlah Waktu Penyelesaian Responden Mahasiswa (Detik) ..... | 73 |
| Tabel 4. 7 Hasil Perhitungan <i>System Usability Scale</i> .....       | 77 |

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Sejak pandemi covid-19 semua kegiatan yang dilakukan di luar rumah, sekarang harus dilakukan di dalam rumah, termasuk kegiatan belajar mengajar. Salah satu teknologi yang digunakan untuk memudahkan proses belajar mengajar dari rumah dan mudah diakses menggunakan internet ialah video conference. Video conference merupakan teknologi komunikasi dua arah yang efektif bagi para pengguna yang bergerak dibidang pendidikan, bisnis maupun dibidang lainnya dan untuk penggunaan secara pribadi.

Jitsi Meet adalah layanan *video conference* bersifat *opensource* yang sepenuhnya terenkripsi sehingga semua pembicaraan di dalam Jitsi Meet bersifat rahasia dan tidak bisa dilihat orang yang tidak berkepentingan. Dengan adanya *video conference* dengan desain yang lebih interaktif dan mudah digunakan tidak lepas dari yang namanya *user interface* dan *user experience*. *User interface* dan *user experience* berfungsi untuk menghubungkan pengguna dengan informasi yang dibutuhkan dari sistem itu sendiri. Sebuah sistem memiliki desain *user interface* dan *user experience* yang berbeda-beda sesuai dengan fungsi dan kebutuhan dari pengguna. *User interface* dan *user experience* dibutuhkan dalam pembuatan sebuah sistem yang nantinya pengguna tidak harus memahami lebih lama cara kerja dari sistem tersebut. Sebuah

sistem yang memiliki *user interface* dan *user experience* yang baik akan membuat pengguna nyaman saat mencari informasi yang dibutuhkan. (Reed et, al, 1999) *User interface* yang bagus atau menarik adalah kemudahan pengguna dalam menggunakannya.

Oleh karena itu, penulis mengangkat penelitian sesuai dengan judul **“PERANCANGAN *USER INTERFACE* DAN *USER EXPERIENCE* PADA *VIDEO CONFERENCE* JITSI MEET MENGGUNAKAN REACTJS”**.

## **1.2 Rumusan Masalah**

1. Bagaimana membuat sebuah perancangan *user interface* dan *user experience* yang efektif untuk pengguna.
2. Bagaimana dosen dan mahasiswa dalam lingkup universitas hasanuddin untuk bisa menjalankan atau bisa berinteraksi dengan sistem perancangan *user interface* dan *user experience* yang telah dibuat.
3. Bagaimana tingkat kepuasan pengguna dalam menggunakan sistem yang telah dibangun.

## **1.3 Batasan Masalah**

Website yang dibuat menggunakan library javascript yaitu reactjs, video conference yaitu jitsi meet dan CSS yaitu bootstrap.

## 1.4 Tujuan Penelitian

Tujuan dari penelitian:

1. Dapat memfasilitasi pengguna yang ingin berkomunikasi menggunakan *video conference* dengan tampilan website yang *user friendly*.
2. Dapat menghasilkan tampilan media pembelajaran dengan *user interface dan user experience* yang memudahkan pengguna mendapatkan informasi yang diinginkan.

## 1.5 Manfaat Penelitian

Manfaat dari penelitian:

1. Memaksimalkan interaksi antara pengguna dan aplikasi melalui fitur-fitur yang tersedia dengan menggunakan *user experience* yang mudah dipahami.
2. Memudahkan pengguna dalam memperoleh informasi yang dibutuhkan, sehingga dapat meningkatkan kinerja dari pengguna.

## 1.6 Sistematika Penelitian

Untuk memberikan gambaran singkat mengenai isi tulisan secara keseluruhan, maka akan diuraikan beberapa tahapan dari penulisan secara sistematis, yaitu:

**BAB I PENDAHULUAN:** Bab ini menguraikan secara umum mengenai hal yang menyangkut latar belakang, perumusan masalah dan batasan masalah, tujuan, manfaat, dan sistematika penulisan.

**BAB II TINJAUAN PUSTAKA:** Bab ini berisi teori-teori tentang hal-hal yang berhubungan dengan perancangan *user interface dan user experience* dan metode yang digunakan.

**BAB III METODOLOGI PENELITIAN:** Bab ini berisi tentang kerangka kerja penelitian dan menjelaskan prosedur percobaan.

**BAB IV HASIL DAN PEMBAHASAN:** Bab ini berisi tentang hasil percobaan serta pembahasan yang disertai tabel hasil penelitian.

**BAB V PENUTUP:** Bab ini berisi tentang kesimpulan yang didapatkan berdasarkan hasil penelitian yang telah dilakukan serta saran-saran untuk pengembangan lebih lanjut.

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Website**

*Website* adalah seperangkat halaman web yang saling berhubungan, umumnya terletak di server yang sama, dan disiapkan serta digunakan sebagai kumpulan informasi seseorang, grup, atau organisasi. Web telah menjadi alat untuk bisnis, komunikasi, pembelajaran, rekreasi, dan berbagai kegiatan di seluruh spektrum populasi yang luas. Penggunaan web, telah menyebabkan proliferasi besar data dan informasi yang tersedia baik untuk sektor publik dan swasta (Germonprez dan Zigurs, 2005). Web adalah jaringan teknologi informasi yang kompleks. Hari ini terdiri dari beberapa ratus juta halaman Web dan lebih dari seratus juta pengguna. Setiap hari pengguna mencari berbagai situs Web yang dapat diakses untuk menemukan informasi yang paling nyaman, relevan, dan terkini yang mereka butuhkan. Di Web, pengguna mencari informasi dengan menavigasi dari halaman ke halaman di sepanjang tautan Web (Dragulanescu, 2007).

World Wide Web dicetuskan oleh Tim Berners-Lee pada 12 Maret 1989, yang saat itu bekerja untuk lab fisika CERN (Organisasi Eropa untuk Riset Nuklir). Ia mengusulkan sistem manajemen informasi yang terdesentralisasi. Pada saat itu, CERN memiliki banyak karyawan, dan semakin banyak dengan masuknya karyawan baru. Sangat sulit untuk mendapatkan informasi yang mungkin berkaitan, tetapi tidak diketahui, karena tempat penyimpanan yang berbeda tempat. Usulnya adalah sistem

koneksi *hypertext*, serta menggunakan kata kunci, mencari informasi-informasi terkait. Pada tahun 1990, Robert Cailliau dari Belgia membantu mengembangkan usul Berners-Lee. Penemuan ini didasarkan pada dua pilar, yaitu: *markup language* HTML, yang mendukung keberadaan situs web, serta protokol pertukaran *hypertext* HTTP, yang memungkinkan pengguna meminta, dan menerima halaman yang diinginkan. Sistem ini kemudian disebarluaskan pada April tahun 1993, dan semakin populer seiring rilisnya Mosaic pada bulan November (Harminingtyas, 2014).

Secara teknis, web adalah sebuah sistem di mana informasi dalam bentuk teks, gambar, suara, dan lain-lain yang tersimpan dalam sebuah internet web server dipresentasikan dalam bentuk *hypertext*. Informasi lainnya disajikan dalam bentuk grafis (dalam format GIF, JPG, PNG), suara (dalam format AU, WAV), dan objek multimedia lainnya (seperti MIDI, *Shockwave*, *Quicktime Movie*, 3D World). Web dapat diakses oleh perangkat lunak *web client* yang secara populer disebut sebagai *browser*. *Browser* membaca halaman-halaman yang tersimpan dalam *webserver* melalui protokol yang disebut dengan HTTP (*Hypertext Transfer Protocol*). Sebagai dokumen *hypertext*, dokumen-dokumen di web dapat memiliki *link* dengan dokumen lain, baik yang tersimpan dalam web server yang sama maupun di web server lainnya. Link memudahkan para pengakses web berpindah dari satu halaman ke halaman lainnya, dan “berkelana” dari satu server ke server lain. Kegiatan penelusuran halaman web ini biasa diistilahkan sebagai *browsing*, ada juga yang menyebutnya sebagai *surfing* (berselancar) (Safrin, 2020).

## **2.2 User Inteface**

*User Inteface* adalah bagian dari komputer dan perangkat lunaknya yang dapat dilihat, didengar, disentuh, diajak bicara, atau dipahami atau diarahkan (Galitz, 2002). Menurut Desintha dan Varian (2019), *User Interface* merupakan “wajah” dari suatu *website*. Pada *User Interface* sendiri, berdampak besar dalam menampilkan kesan maupun identitas dari web, serta dapat mempengaruhi berbagai aspek dari penggunaannya.

Antarmuka pengguna pada dasarnya memiliki dua komponen: *input* dan *output*. *Input* adalah bagaimana seseorang mengkomunikasikan kebutuhan atau keinginannya ke komputer. Beberapa komponen *input* yang umum adalah *keyboard*, *mouse*, *trackball*, satu jari (untuk layar sentuh yang sensitif), dan suara seseorang (untuk instruksi lisan). *Output* adalah bagaimana komputer menyampaikan hasil perhitungan dan persyaratannya kepada pengguna. Saat ini, mekanisme *output* komputer yang paling umum adalah tampilan layar, diikuti oleh mekanisme yang memanfaatkan kemampuan pendengaran seseorang: suara dan suara. Penggunaan indra penciuman dan sentuhan manusia dalam desain antarmuka sebagian besar masih belum dijelajahi. Desain antarmuka yang tepat akan memberikan campuran mekanisme *input* dan *output* yang dirancang dengan baik yang memenuhi kebutuhan, kemampuan, dan keterbatasan pengguna dengan cara yang seefektif mungkin. Antarmuka terbaik adalah yang tidak diperhatikan, yang memungkinkan pengguna untuk fokus pada informasi dan tugas

yang ada, bukan mekanisme yang digunakan untuk menyajikan informasi dan melakukan tugas (Galitz, 2002).

Memiliki desain *user interface* yang baik merupakan suatu hal yang penting, karena penggunaan perangkat *smartphone* tidak dapat dipungkiri sudah menjadi kebiasaan manusia sehari-hari pada umumnya. Kebutuhan desain dan pengembangan *user interface* yang mendukung tugas-tugas manusia dan dapat digunakan dengan mudah oleh berbagai jenis kalangan telah menjadi permasalahan yang penting. Dalam hal membangun citra merek, *user interface* yang baik dapat menghasilkan hasil positif seperti meningkatnya jumlah pengguna, menumbuhkan kepercayaan terhadap aplikasi dan kepuasan pengguna yang secara tidak langsung menjadi promosi melalui mulut ke mulut. Selain itu tampilan sistem yang mudah dipahami membutuhkan waktu yang lebih sedikit untuk para pengguna mempelajari sebuah aplikasi dan menghindari stres ketika pengguna mengoperasikannya (Stone dkk, 2005).

User interface harus memberikan bahasa visual yang tepat agar pengguna dapat berinteraksi dengan serangkaian bentuk visual yang akan diterjemahkan menjadi bentuk perintah. Menurut Railean (2017), strategi perancangan user interface diukur dari 5 poin berikut, yaitu:

a. *Time to Learn*

Seberapa lama waktu pengguna untuk mempelajari atau memahami user interface yang disuguhkan. Terutama pengguna-pengguna yang memiliki keterbatasan akan pengetahuan bahasa atau cara mengoperasikan suatu perangkat.

b. *Speed of Performance*

Ketika merancang user interface desainer harus memikirkan cara agar desain yang mereka buat tidak membuat performa aplikasi menurun dan membuat interaksi yang terdapat user interface tidak membuang-buang waktu pengguna.

c. *Rate of Errors by Users*

Memperhatikan seberapa banyak jumlah dan jenis kesalahan ketika pengguna menggunakan *user interface* yang telah dirancang. Karena desain *user interface* yang baik adalah tampilan yang dapat dimengerti hampir semua orang.

d. *Retention Over Time*

Seberapa baik para pengguna mempertahankan pengetahuan atau ingatan mereka pada *user interface* yang telah dirancang. Retensi terkait dengan waktu pengguna untuk belajar *user interface* dan frekuensi penggunaan juga berperan penting.

e. *Subjective Satisfaction*

Kepuasan pengguna saat menggunakan *user interface* dalam berbagai aspek merupakan keuntungan pada sebuah aplikasi. Walaupun bersifat subjektif, tapi memiliki dampak yang cukup besar yang menjadi salah satu penentu kesuksesan pada sebuah aplikasi.

### **2.3 User Experience**

*User Experience* adalah pengalaman yang diciptakan dalam produk untuk orang-orang yang menggunakannya di dunia nyata. Ketika suatu produk sedang dikembangkan, orang menaruh banyak perhatian pada apa yang dilakukannya. Pengalaman pengguna adalah sisi lain yang sering diabaikan cara kerjanya yang sering kali dapat membuat perbedaan antara produk yang berhasil dan yang gagal. Pengalaman pengguna bukanlah tentang cara kerja bagian dalam suatu produk atau layanan. Pengalaman pengguna adalah tentang cara kerjanya di luar, di mana seseorang bersentuhan dengannya. Ketika seseorang bertanya bagaimana rasanya menggunakan produk atau layanan, mereka bertanya tentang pengalaman pengguna (Garrett, 2011).

Pengalaman pengguna sangat penting untuk semua jenis produk dan layanan. Di Web, pengalaman pengguna menjadi lebih penting daripada jenis produk lainnya. Namun pelajaran yang kami pelajari dari menciptakan pengalaman pengguna di Web dapat diterapkan jauh melampaui batasnya. Situs web adalah bagian teknologi yang rumit, dan sesuatu yang lucu terjadi ketika orang mengalami kesulitan menggunakan bagian teknologi yang rumit: Mereka menyalahkan diri mereka sendiri. Situs web

adalah produk swalayan. Tidak ada instruksi manual untuk dibaca sebelumnya, tidak ada seminar pelatihan yang harus dihadiri, tidak ada perwakilan layanan pelanggan untuk membantu memandu pengguna melalui situs. Hanya ada pengguna, menghadapi situs sendirian dengan hanya akal dan pengalaman pribadi untuk membimbingnya (Garrett, 2011).

Dalam Stone dkk (2005) satu kelompok dari peneliti *Human-Interaction Computer* (HCI) melakukan survei terhadap 275 profesional dan akademisi *user experience* untuk mengetahui apa arti dari pengalaman pengguna (*user experience*). Setelah menganalisis jawaban responden, mereka mengidentifikasi beberapa fitur umum dalam definisi orang mengenai kontribusi *user experience*, berikut:

- a. *User experience* dipandang sebagai respon seseorang ketika mereka menggunakan perangkat, produk, layanan atau objek melalui *user interface*.
- b. *User experience* bersifat dinamis, sehingga dapat berubah sebelum, selama atau setelah digunakan dan itu tergantung pada konteks atau pengalaman ketika menggunakan perangkat, produk, layanan dan objek.
- c. Respon *user experience* yang bersifat subjektif. Dipengaruhi oleh latar belakang pengguna, pengalaman sebelumnya dan banyak faktor lainnya.

Saat menggunakan produk atau ketika membeli perangkat, dapat dibayangkan perasaan yang akan dimiliki pengguna lain dalam situasi yang berbeda. Perasaan positif dapat diartikan kurva pengalaman pengguna naik, perasaan negatif berarti kurva menurun. Jika kurva turun terlalu banyak atau turun berulang kali, kemungkinan besar

akan kehilangan pelanggan bahkan sebelum mereka membelinya atau pelanggan merasa sangat tidak senang ketika menggunakan produk tersebut dan yang terjadi adalah mereka tidak akan membeli produk tersebut lagi (Kraft, 2012).

#### **2.4 Javascript**

JavaScript adalah bahasa pemrograman Web. Sebagian besar situs web modern menggunakan JavaScript, dan semua browser web modern di desktop, konsol *game*, tablet, dan ponsel pintar termasuk penerjemah JavaScript, menjadikan Java-Script sebagai bahasa pemrograman yang paling banyak digunakan dalam sejarah. JavaScript adalah bagian dari tiga serangkai teknologi yang harus dipelajari oleh semua pengembang Web: HTML untuk menentukan konten halaman web, CSS untuk menentukan presentasi halaman web, dan JavaScript untuk menentukan perilaku halaman web. Nama "JavaScript" sebenarnya kurang sesuai. Kecuali untuk kemiripan sintaksis dasar, JavaScript benar-benar berbeda dari bahasa pemrograman Java. Dan JavaScript telah lama melampaui akar bahasa skripnya menjadi bahasa tujuan umum yang kuat dan efisien (Flanagan, 2011).

Menurut Permana (dalam Safrin, 2020) pada awalnya, Javascript mulai diperkenalkan di browser Netscape Navigator 2. Namun waktu itu namanya bukan Javascript, namun Livescript. Mengingat pada waktu itu teknologi Java sedang panas-panasnya atau sedang tren, maka pihak Netscape memutuskan untuk mengganti namanya menjadi javascript, yang seperti nama tersebut lebih *marketible* dibandingkan dengan Livescript. Selanjutnya pihak Microsoft (rival Netscape) pun mulai ikut-ikutan

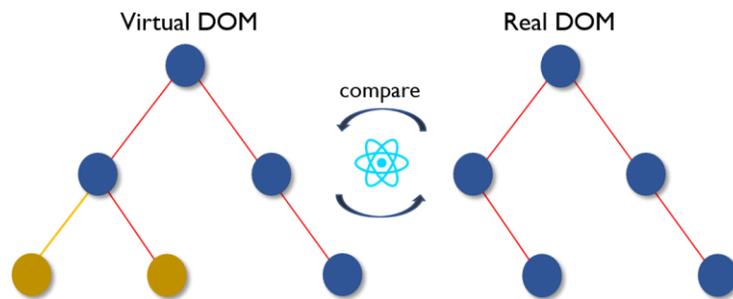
memfasilitasi web browser buatannya, 'Internet Explorer', supaya bisa mendukung Javascript. Namun mungkin karena gengsi, pihak Microsoft memberi nama bahasa lain, Jscript.

## **2.5 React Js**

React.js adalah *library* JavaScript yang dibuat oleh Facebook. Ini sering dianggap sebagai "tampilan" dalam antarmuka pengguna *model-view-controller* (MVC). Ini masuk akal ketika Anda mempertimbangkan fakta bahwa satu-satunya fungsi yang harus diimplementasikan di React adalah fungsi "*render*". Fungsi *render* menyediakan *output* yang dilihat pengguna ("tampilan"). React didasarkan pada komponen dan status. Inilah yang membuat React menjadi perpustakaan yang populer. Saat ingin membuat aplikasi, biasanya Anda memecahnya menjadi bagian-bagian yang lebih sederhana. Saat memprogram dengan React, Anda ingin memecah antarmuka Anda menjadi bagian paling dasar, dan itu akan menjadi komponen React Anda. Sifat komponen yang luar biasa karena bersifat modular dan dapat digunakan kembali (Dyrr, 2018).

Secara umum, React digunakan untuk memudahkan penalaran tentang struktur antarmuka pada saat tertentu. Ini dilakukan melalui komponen, yang dapat dianggap sebagai bagian antarmuka yang mandiri. React juga menggunakan konsep *Document Object Model* (DOM) *virtual*. Ini, pada dasarnya, adalah DOM sekunder yang secara konseptual berada di atas DOM asli dalam memori. Daripada memanipulasi DOM asli secara langsung, ini malah mengizinkan React untuk menengahi perubahan yang dapat

terjadi pada halaman. React akan memperbarui DOM virtual dan kemudian dengan cerdas akan mencari tahu, melalui algoritma diffing, paling sedikit pekerjaan DOM nyata yang dapat dilakukan untuk menyelesaikan pembaruan. Yang paling penting, ini memungkinkan React untuk mengumpulkan perubahan DOM nyata dan menerapkan semuanya sekaligus, yang jauh lebih efisien daripada melakukannya satu per satu. Hasilnya adalah kinerja yang lebih baik daripada yang biasanya dapat dicapai dengan manipulasi DOM langsung (selalu ada pengecualian, tetapi ini cenderung benar secara umum) (Zametti, 2020).



**Gambar 2. 1** Perbandingan Virtual DOM dan Real DOM

## 2.6 Hypertext Markup Language (HTML)

HTML adalah singkatan dari *Hypertext Markup Language*. HTML awalnya didasarkan pada *Standard Generalized Markup Language* (SGML), sistem pemrosesan dokumen yang jauh lebih besar dan lebih rumit. Untuk menulis halaman HTML, Anda tidak perlu tahu banyak tentang SGML. Namun, mengetahui bahwa salah satu fitur utama SGML adalah bahwa ia menggambarkan struktur umum konten di dalam dokumen daripada tampilan sebenarnya di halaman atau di layar memang membantu.

HTML mendefinisikan satu set elemen umum untuk halaman web: judul, paragraf, daftar, dan tabel. Ini juga mendefinisikan format karakter seperti huruf tebal dan contoh kode. Elemen dan format ini ditunjukkan di dalam dokumen HTML menggunakan tag. Setiap tag memiliki nama tertentu dan berangkat dari isi dokumen menggunakan notasi (Lemay dkk, 2015).

HTML pada awalnya dirancang untuk berbagi dokumen statis berbasis teks di Internet. Seiring waktu, karena pengguna web dan desainer menginginkan lebih banyak inter-aktivitas dalam dokumen HTML mereka, mereka mulai meningkatkan dokumen ini, dengan menambahkan fungsionalitas formulir dan kemampuan tipe "portal" awal. Sekarang, kumpulan dokumen statis ini, atau situs web, lebih seperti aplikasi web, berdasarkan prinsip aplikasi desktop klien/server yang kaya. Aplikasi web ini digunakan di hampir semua perangkat: laptop, ponsel pintar, tablet, semuanya (Wang, 2013).

## **2.7 Cascading Style Sheets (CSS)**

*Cascading Style Sheets* (CSS) adalah bahasa yang kita gunakan untuk menata sebuah dokumen HTML. CSS menjelaskan bagaimana elemen HTML harus ditampilkan. CSS merupakan bahasa yang digunakan untuk mengatur tampilan suatu dokumen yang ditulis dalam bahasa *markup* atau *markup language*. Jika kita berbicara dalam konteks web secara bebas, CSS bisa di artikan secara bebas sebagai bahasa yang digunakan untuk mengatur tampilan atau desain suatu halaman HTML (Sari dkk, 2019).

*Cascading Style Sheets* sebagai cara untuk dapat mengontrol tampilan dan nuansa halaman web. Tujuan utama CSS adalah untuk menggambarkan struktur dokumen. *Cascading Style Sheets* adalah cara untuk mengontrol bagaimana browser me-render elemen HTML. CSS memberikan banyak fleksibilitas tentang bagaimana Anda dapat mengubah tampilan jenis elemen apa pun, dan gaya dapat diterapkan dalam beberapa cara berbeda. Kelebihan CSS adalah dapat digunakan dengan berbagai cara (Lemay dkk, 2015).

## **2.8 NodeJS**

NodeJS adalah platform untuk mengeksekusi program Javascript di luar browser. Bekerja dengan *server-side code* yang memiliki kinerja tinggi dan mampu menangani beban permintaan besar dengan mudah. Ini didasarkan pada bahasa yang paling banyak digunakan di planet ini saat ini yaitu JavaScript. Di Node, hampir semua yang Anda lakukan adalah *nonblocking*, artinya kode tidak akan menahan permintaan pemrosesan *thread* lainnya. Sebagian besar jenis I/O, *asynchronous* di Node, baik itu panggilan jaringan atau panggilan sistem file atau panggilan basis data. Nilai lebih dari Node adalah bahwa untuk mengeksekusi kode, Node menggunakan mesin JavaScript V8 Google yang populer dan disetel tinggi, mesin yang sama yang menggerakkan *browser* Chrome-nya, membuatnya berkinerja sangat tinggi dan mampu menangani beban permintaan yang besar (Zametti, 2020).

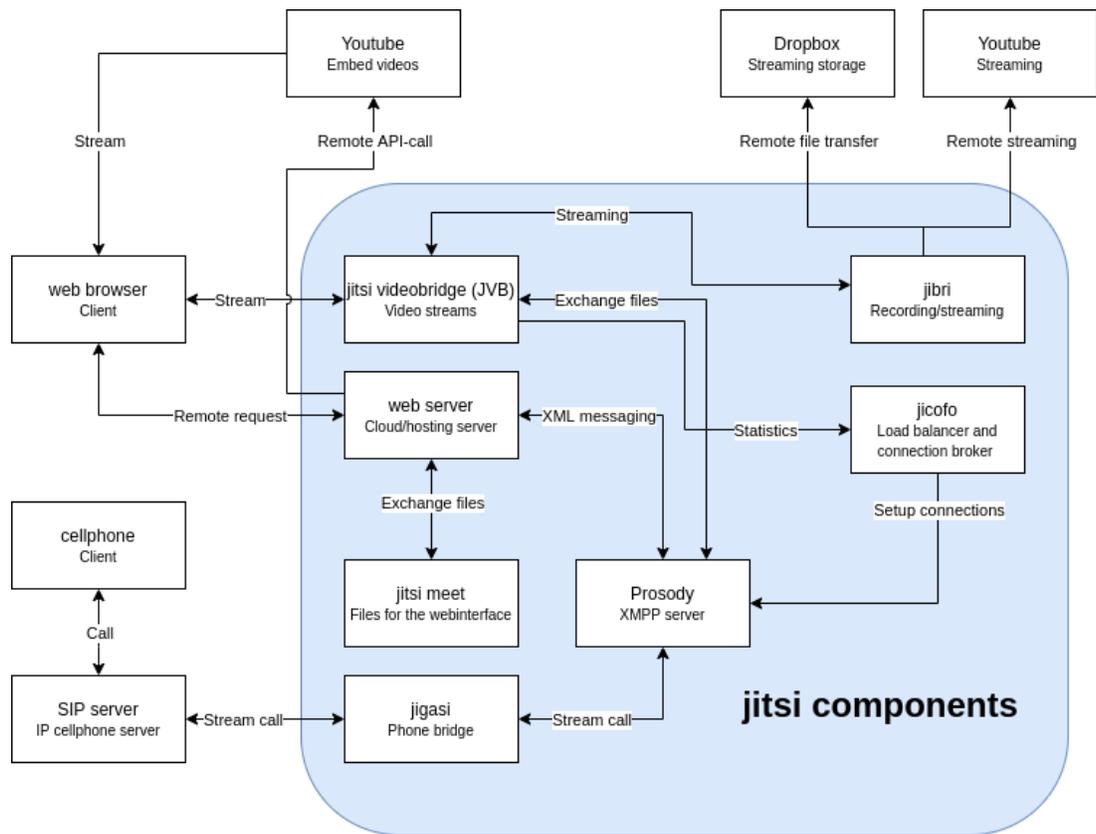
Node bisa dikatakan *first-class runtime environment*, artinya kita dapat melakukan hal-hal seperti berinteraksi dengan sistem *file* lokal, mengakses *database*

relasional, memanggil sistem jarak jauh, dan banyak lagi. Di masa lalu, Anda harus menggunakan *runtime* yang "tepat", seperti Java atau .Net untuk melakukan semua ini sedangkan JavaScript bukan pemain di ruang itu. Dengan Node, hal itu dapat diatasi. Node dapat bersaing tidak hanya pada kinerja tetapi juga dalam hal kemampuan apa yang diberikannya kepada pengembang. Jika Anda dapat memikirkannya, kemungkinan Anda dapat melakukannya dengan Node, dan itu tidak selalu terjadi dengan JavaScript (Zametti, 2020).

## **2.9 Jitsi Meet**

Jitsi Meet adalah aplikasi JavaScript WebRTC *open-source* (Apache) yang menggunakan Jitsi Videobridge untuk menyediakan konferensi video berkualitas tinggi, aman, dan skalabel. Klien dapat menggunakan Jitsi Meet dengan mudah karena berjalan di browser, tanpa meng-instal apa pun di komputer. Jitsi Meet memungkinkan kolaborasi yang sangat efisien. Pengguna dapat melakukan *streaming* desktop mereka atau hanya beberapa jendela. Ini juga mendukung pengeditan dokumen bersama dengan Etherpad.

Jitsi Meet dimulai sebagai contoh aplikasi konferensi menggunakan Jitsi Videobridge. Ini awalnya dikembangkan oleh pengembang ESTOS Philipp Hancke yang kemudian berkontribusi pada komunitas di mana pengembangan berlanjut dengan kekuatan gabungan.



**Gambar 2. 2** Jitsi Architecture Diagram

### 2.10 Usability

*Usability* adalah analisa kualitatif yang menentukan seberapa mudah *user* menggunakan antarmuka suatu aplikasi (Nielsen, 2012). Suatu aplikasi disebut *usable* jika fungsi-fungsinya dapat dijalankan secara efektif, efisien, dan memuaskan (Nielsen, 1993).

Definisi *usability* menurut ISO 9241:11 (1998) adalah sejauh mana suatu produk dapat digunakan oleh pengguna tertentu untuk mencapai target yang ditetapkan dengan efektivitas, efisiensi dan mencapai kepuasan penggunaan dalam konteks

tertentu. Konteks penggunaan terdiri dari pengguna, tugas, peralatan (*hardware*, *software* dan material).

Berdasarkan definisi tersebut *usability* diukur berdasarkan komponen:

- A. Kemudahan (*learnability*) didefinisikan seberapa cepat pengguna mahir dalam menggunakan sistem serta kemudahan dalam penggunaan menjalankan suatu fungsi serta apa yang pengguna inginkan dapat mereka dapatkan.
- B. Efisiensi (*efficiency*) didefinisikan sebagai sumber daya yang dikeluarkan guna mencapai ketepatan dan kelengkapan tujuan.
- C. Mudah diingat (*memorability*) didefinisikan bagaimana kemampuan pengguna mempertahankan pengetahuannya setelah jangka waktu tertentu, kemampuan mengingat didapatkan dari peletakkan menu yang selalu tetap.
- D. Kesalahan dan keamanan (*errors*) didefinisikan berapa banyak kesalahan-kesalahan apa saja yang dibuat pengguna, kesalahan yang dibuat pengguna mencakup ketidaksesuaian apa yang pengguna pikirkan dengan apa yang sebenarnya disajikan oleh sistem.
- E. Kepuasan (*satisfaction*) didefinisikan kebebasan dari ketidaknyamanan, dan sikap positif terhadap penggunaan produk atau ukuran subjektif sebagaimana pengguna merasa tentang penggunaan sistem.

### **2.11 Cognitive Walkthrough**

Metode evaluasi analitis yang paling sering dipakai adalah *cognitive walkthrough* (CW) yang merupakan metode inspeksi untuk mengevaluasi *usability* dalam *user interface*. Metode ini berfokus pada kesederhanaan dalam belajar, terutama lewat penyelidikan. Keuntungan penggunaan metode *cognitive walkthrough* adalah dapat digunakan untuk mengidentifikasi permasalahan *usability* yang penting dengan sangat mudah, cepat dan murah ketika sumber daya untuk melakukan *usability test* sangat terbatas. Karena *usability test* membutuhkan orang yang sangat ahli dalam permasalahan *usability* (Kaufman, 2003).

### **2.12 System Usability Scale**

SUS (*System Usability Scale*). SUS dikembangkan (Brooke, 1996) sebagai sebuah pengukuran *usability* yang “*quick and dirty*”. Survei terdiri dari 10 pertanyaan; masing-masing memiliki 5 poin Likert sebagai tanggapan. *Output* SUS berupa skor yang tampak mudah dipahami, dengan range dari 0 hingga 100, dengan semakin besar skor berarti semakin baik *usability*-nya.

System Usability Scale (SUS) metode ini paling banyak digunakan untuk melakukan uji software karena mudah untuk digunakan dan hasilnya dapat digunakan sebagai pertimbangan. Adapun alasan menggunakan metode ini karena sangat mudah dimengerti dan dalam implementasinya sangat mudah untuk digunakan sehingga metode ini populer digunakan dalam melakukan uji *usability testing* (Agus, 2019).