

**IMPLEMENTASI PETA 3 DIMENSI MENGGUNAKAN METODE
IMSDD (*INTERACTIVE MULTIMEDIA SYSTEM DESIGN AND
DEVELOPMENT*) DAN WEBGL API BERBASIS WEB**

(Studi Kasus Di Smp Karya Pembangunan 2 Majalaya)

SKRIPSI

Karya Tulis sebagai syarat memperoleh
Gelar Sarjana Komputer dari Fakultas Teknologi Informasi
Universitas Bale Bandung

Disusun oleh:

TAUFIQ JAMIL HANAFI

NPM. 301180015



**PROGRAM STRATA 1
PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS BALE BANDUNG
BANDUNG**

2022

LEMBAR PERSETUJUAN PEMBIMBING

IMPLEMENTASI PETA 3 DIMENSI MENGGUNAKAN METODE IMSDD (*INTERACTIVE MULTIMEDIA SYSTEM DESIGN AND DEVELOPMENT*) DAN WEBGL API BERBASIS WEB (Studi Kasus Di Smp Karya Pembangunan 2 Majalaya)

Disusun oleh:

TAUFIQ JAMIL HANAFI

NPM. 301180015

Telah diterima dan disetujui untuk memenuhi persyaratan mencapai gelar
SARJANA KOMPUTER

Pada

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS BALE BANDUNG**

Baleendah, Agustus 2022

Disetujui oleh:

Pembimbing Utama

Pembimbing Pendamping

Yaya Suharya, S.Kom., M.T
NIK. 01043170007

Yusuf Muharam, S.Kom., M.Kom
NIK. 04104820003

LEMBAR PERSETUJUAN PENGUJI
IMPLEMENTASI PETA 3 DIMENSI MENGGUNAKAN METODE
IMSDD (*INTERACTIVE MULTIMEDIA SYSTEM DESIGN AND*
***DEVELOPMENT*) DAN WEBGL API BERBASIS WEB**
(Studi Kasus Di Smp Karya Pembangunan 2 Majalaya)

Disusun Oleh :

TAUFIQ JAMIL HANAFI
NPM. 301180015

Telah diterima dan disetujui untuk memenuhi persyaratan mencapai gelar
SARJANA KOMPUTER

Pada

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS BALE BANDUNG

Baleendah, Agustus 2022

Disetujui oleh:

Penguji 1

Penguji 2

Denny Rusdianto, S.T, M.Kom
NIK. 04104808094

Sukiman, S.Tr.Kom., S.Pd., M.Kom
NIK. 04104821001

LEMBAR PENGESAHAN PROGRAM STUDI

IMPLEMENTASI PETA 3 DIMENSI MENGGUNAKAN METODE IMSDD (*INTERACTIVE MULTIMEDIA SYSTEM DESIGN AND DEVELOPMENT*) DAN WEBGL API BERBASIS WEB (Studi Kasus Di Smp Karya Pembangunan 2 Majalaya)

Disusun oleh:
TAUFIQ JAMIL HANAFI
NPM. 301180015

Telah diterima dan disetujui untuk memenuhi persyaratan mencapai gelar
SARJANA KOMPUTER

Pada

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS BALE BANDUNG**

Baleendah, Agustus 2022

Mengetahui,
Dekan,

Mengesahkan,
Ketua Program Studi

Yudi Herdiana, S.T., M.T
NIK. 04104808008

Yusuf Muharam, S.Kom., M.Kom
NIK. 04104820003

HALAMAN PERNYATAAN

Penyusun yang bertanda tangan dibawah ini:

Nama : TAUFIQ JAMIL HANAFI
NPM : 301180015
Judul Skripsi : Implementasi Peta 3 Dimensi Menggunakan Metode Imsdd
(*Interactive Multimedia System Design And Development*)
Dan *Webgl Api* Berbasis *Web* Studi Kasus Di Smp Karya
Pembangunan 2 Majalaya

Menyatakan dengan sebenarnya bahwa penulisan skripsi ini berdasarkan hasil penelitian, pemikiran, dan pemaparan asli dari penyusun sendiri, baik untuk naskah laporan maupun kegiatan *programming* yang tercantum sebagai bagian dari skripsi ini. jika terdapat karya orang lain, penyusun mencantumkan sumber yang jelas.

Pernyataan ini penyusun buat dengan sesungguhnya dan apabila dikemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka penyusun bersedia menerima sanksi akademik berupa pencabutan gelar yang telah diperoleh karena karya tulis ini dan sanksi lain sesuai dengan peraturan yang berlaku di Fakultas Informasi Universitas Bale Bandung.

Demikian surat pernyataan ini penyusun buat dalam keadaan sadar tanpa paksaan dari pihak manapun.

Baleendah, Agustus 2022

Yang membuat pernyataan

Taufiq Jamil Hanafi

NPM. 301180015

Abstrak

Peta merupakan hal yang dibutuhkan banyak pengguna, tidak terkecuali pendidikan menengah pertama. SMP Karya Pembanguna 2 Majalaya merupakan salah satu pendidikan sekolah menengah pertama yang mempunyai banyak bangunan fisik. Informasi peta kampus yang masih digunakan sampai sekarang adalah dalam bentuk master plan peta 2D berformat WORD. Hal ini mengakibatkan siswa/siswi dan pengunjung SMP Karya Pembanguna 2 Majalaya menjadi kebingungan dan tidak mengetahui letak bangunan serta ruangan yang ingin dituju.

Membangun sebuah peta 3D interaktif yang dapat memperlihatkan lokasi dan letak bangunan beserta ruangnya berbasis web menggunakan WebGL API merupakan cara terbaik untuk memberikan layanan informasi kampus. Penelitian ini menggunakan metode pengembangan sistem multimedia interaktif (IMSDD).

Hasil dari penelitian ini adalah sebuah peta kampus berbentuk 3D berbasis web pada kampus SMP KP 2 Majalaya yang dapat memperlihatkan bentuk fisik bangunan, lokasi, dan informasi nama Gedung beserta ruangan yang ada di SMP Karya Pembanguna 2 Majalaya dalam bentuk peta digital.

Kata kunci : SMP KP 2 Majalaya, Peta 3D, Sistem Informasi Peta Kampus, IMSDD, WebGL API , Blender 3D

Abstract

Maps are something that many users need, and junior secondary education is no exception. SMP Karya Pembanguna 2 Majalaya is one of the first high school education that has many physical buildings. Campus map information that is still used today is in the form of a 2D map master plan in WORD format. This causes students and visitors to SMP Karya Pembanguna 2 Majalaya to be confused and do not know the location of the building and the room they want to go to.

Building an interactive 3D map that can show the location and location of buildings and their spaces based on the web using the WebGL API is the best way to provide campus information services. This study uses the method of developing an interactive multimedia system (IMSDD).

The result of this research is a 3D web-based campus map on the SMP KP 2 Majalaya campus which can show the physical form of the building, location, and information on the name of the building along with the rooms in SMP Karya Pembanguna 2 Majalaya in the form of a digital map.

Keywords: SMP KP 2 Majalaya, 3D Map, Campus Map Information System, IMSDD, WebGL API, Blender 3D

KATA PENGANTAR

Puji syukur kepada Allah SWT berkar Rahmat, Hidayah, dan Karunia-Nya kepada kita semua sehingga dapat menyelesaikan laporan skripsi dengan judul “Implementasi Peta 3 Dimensi Menggunakan Metode IMSDD (*Interactive Multimedia System Design and Development*) dan WebGL API Berbasis Web Studi Kasus di SMP Karya Pembangunan 2 Majalaya”. Laporan skripsi ini disusun sebagai salah satu syarat untuk mengerjakan skripsi pada program Strata-1 di Jurusan Teknik Informatika, Fakultas Teknologi Informasi, Universitas Bale Bandung.

Penulis menyadari dalam penyusunan proposal skripsi ini tidak akan selesai tanpa bantuan dari berbagai pihak. Karena itu pada kesempatan ini kami ingin mengucapkan terima kasih kepada:

1. Bapak Yudi Herdiana, S.T., M.T selaku Dekan Fakultas Teknologi Infomasi
2. Bapak Yaya Suharya, S.Kom., M.T selaku dosen Pembimbing 1 yang telah membimbing dalam penyusunan laporan skripsi.
3. Bapak Yusuf Muharam, S.Kom., M.Kom selaku Dosen Pembimbing 2 yang telah membimbing dalam penyusunan laporan skripsi.
4. Bapak Denny Rusdianto, S.T, M.Kom selaku dosen Penguji ke 2 sidang skripsi di Fakultas Teknologi Informasi Universitas Bale Bandung.
5. Ibu Imas Ratna Sumirat, S.Pd selaku Kepala SMP KP 2 Majalaya yang telah memberikan tempat, kesempatan, dan fasilitas Kerja Praktek di SMP Karya Pembangunan 2 Majalaya.
6. Ibu Yati Ernawati, A.Md selaku pembimbing lapangan Kerja Praktek di SMP Karya Pembangunan 2 Majalaya yang telah membantu dalam pelaksanaan Kerja Praktek.
7. Ibu Nenden Rahmawati, S.Hum selaku pendamping ke-1 pengumpulan siswa dalam pengisian kuesioner.
8. Ibu Reni Rohayani, S.Pd selaku pendamping ke-2 pengumpulan dan pengisian kuesioner siswa.

9. Kedua orang tua dan saudara penyusun yang telah mendukung dan memberikan do'a restu.
10. Bapak Ibu staff administrasi SMP Karya Pembangunan 2 Majalaya yang telah membantu dan memberikan pengalamannya.
11. Semua pihak yang tidak dapat penyusun sebutkan satu persatu yang senantiasa selalu membantu baik secara moral maupun material.

Penyusun mengakui bahwa laporan skripsi ini tidak luput dari berbagai kekurangan. Penulis mengharapkan saran dan kritik demi kesempurnaan dan perbaikannya sehingga akhirnya laporan skripsi ini dapat memberikan manfaat bagi bidang Pendidikan dan penerapan di lapangan serta bisa dikembangkan lagi lebih lanjut.

Bandung, Agustus 2022

Penyusun

DAFTAR ISI

ABSTRAK	vi
ABSTRACT	vii
KATA PENGANTAR	viii
DAFTAR ISI	x
DAFTAR GAMBAR	xiii
DAFTAR TABEL	xiv
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 Metodologi	3
1.6 Sistematika Penulisan	4
BAB II TINJAUAN PUSAKA	5
2.1 Landasan Teori	5
2.2 Dasar Teori	7
2.2.1 Informasi	7
2.2.2 Multimedia	8
2.2.3 Animasi	10
2.2.4 Blender 3D	11
2.2.5 WebGL API	13
2.2.6 Three.js	13
2.2.7 Model Viewer	13
2.2.8 HTML	14
2.2.9 Java Script	14
2.2.10 CSS	14
2.2.11 Web Component	14
2.2.12 DOM	14

2.2.13	Metode Pembangunan Multimedia	15
2.2.14	Storyboard	17
2.2.15	State Transition Diagram	18
2.2.16	Beta Testing	19
2.2.17	UML	19
2.2.18	Vercel App	22
2.2.19	Vite.js	22
2.2.20	glTF	23
2.2.21	The Eight Golden Rules Of Interfaces Design	23
BAB III	METODOLOGI	26
3.1	Kerangka Pikir	26
3.2	Deskripsi	27
3.2.1	Pengumpulan Data	27
3.2.2	System Requirement	28
3.2.3	Design Consideration	30
3.2.4	Penerapan / Implementation	34
3.2.5	Evaluation	34
3.2.6	Penyelesaian Laporan Penelitian	34
BAB IV	ANALISIS DAN PERANCANGAN	35
4.1	Analisis	35
4.1.1	Analisis Masalah	35
4.1.2	Analisis Software	35
4.1.3	Analisis Pengguna	36
4.1.4	User Interface	36
4.1.5	Fitur – fitur	38
4.1.6	Analisis Data	38
4.2	Perancangan	38
4.2.1	System Requirement	38
4.2.2	Consideration Design	40
BAB V	IMPLEMENTASI DAN PERANCANGAN	49
5.1	Implementasi	49

5.1.1 Listing Program	49
5.1.2 Implementasi Sistem	89
5.1.3 Spesifikasi Sistem	90
5.2 Pengujian	106
5.3 Evaluasi	107
BAB VI KESIMPULAN DAN SARAN	109
6.1 Kesimpulan	109
6.2 Saran	110
DAFTAR PUSTAKA	111
LAMPIRAN	112

DAFTAR GAMBAR

Gambar 2.1 Contoh Sketsa gambar untuk 2D	11
Gambar 2.2 Contoh animasi 3D	11
Gambar 2.3 Tampilan Antarmuka <i>Software</i> Blender 3D	12
Gambar 2.4 Siklus Dari IMSDD	16
Gambar 2.5 Contoh <i>Stodyboard</i>	18
Gambar 2.6 Contoh <i>Use Case Diagram</i>	20
Gambar 2.7 Contoh <i>Activity Diagram</i>	21
Gambar 2.8 Contoh <i>Sequence Diagram</i>	21
Gambar 3.1 <i>Flowchart</i> Kerangka Berfikir	26
Gambar 3.2 <i>Storyboard</i> Kreangka Awal Tampilan Peta Interaktif.....	31
Gambar 3.3 <i>Navigational Structure</i> Perancangan Sistem	33
Gambar 4.1 Penggunaan <i>Scroll</i> dalam navigasi sistem	37
Gambar 4.2 Tabel <i>User Interface</i>	37
Gambar 4.3 <i>Sequence Diagram</i>	41
Gambar 4.4 <i>Class Diagram</i>	42
Gambar 4.5 <i>Use Case Diagram</i>	42
Gambar 4.6 Tampilan Awal Peta 3D Interaktif	43
Gambar 4.7 Tampilan pertama peta setelah scrolling	44
Gambar 4.8 <i>Scene 3</i> tampilan awal informasi	44
Gambar 4.9 Tampilan <i>Scene 3 Scroll Down</i>	45
Gambar 4.10 Tampilan <i>Scene 4 Scroll Down</i>	45
Gambar 4.11 Tampilan <i>Scene 4 Scroll Down</i>	46
Gambar 4.12 Tampilan akhir perjalanan tour peta 3D SMP Karya Pembangunan 2 Majalya	46
Gambar 4.12 <i>Navigational Structure</i>	47
Gambar 5.1 <i>Download Node.js</i>	91
Gambar 5.2 Langkah Instalasi Node.js	91
Gambar 5.3 Menentukan PATH Penyimpanan Node.js.....	92

Gambar 5.4 <i>Custom page setup</i>	92
Gambar 5.5 <i>Install Dependence</i>	93
Gambar 5.6 Instalasi Selesai	93
Gambar 5.7 Tampilan <i>folder, subfolder</i> beserta file aplikasi peta 3D	94
Gambar 5.8 Langkah <i>Commit and Push</i> menggunakan <i>Terminal</i>	95
Gambar 5.9 Membuat <i>repository</i>	95
Gambar 5.10 Nama <i>Repository</i>	96
Gambar 5.11 <i>Repository</i> telah dibuat	96
Gambar 5.12 Melakukan Perintah <i>commit</i> ke github	96
Gambar 5.13 <i>Commit</i> dan <i>Push</i> telah selesai	97
Gambar 5.14 Buat Project Baru	97
Gambar 5.15 Tambah Project Baru	97
Gambar 5.16 Pilih Skripsi-app-bokoko33	98
Gambar 5.17 <i>Configure Project</i>	98
Gambar 5.18 Proses <i>Deploying</i>	99
Gambar 5.19 <i>Deploying</i> berhasil	99
Gambar 5.20 Tampilan Awal	100
Gambar 5.21 Animasi Pertama	100
Gambar 5.22 Tampilan Peta	101
Gambar 5.23 Informasi Mengenai Ruang Guru	101
Gambar 5.24 Informasi Mengenai Ruang Tata Usaha	102
Gambar 5.25 Informasi tentang Mushola	102
Gambar 5.26 Grafik Peta 3D Mudah Digunakan	103
Gambar 5.27 Grafik tingkat pemahaman Scroll Navigasi	104
Gambar 5.28 Grafik tingkat kemenarikan Peta 3D SMP Karya Pembangunan 2 Majalaya	104
Gambar 5.29 Grafik Peta 3D akan membantu responden mengenal tata letak gedung dan ruangan	105
Gambar 5.30 Grafik Peta 3D dapat membantu dalam pencarian lokasi gedung atau ruangan	106

DAFTAR TABEL

Tabel 2.1 Ikhtisar	5
Tabel 4.1 <i>Analisis Software</i>	35
Tabel 4.2 <i>Activity Diagram</i>	41
Tabel 5.1 Spesifikasi Minimum Perangkat Keras	90
Tabel 5.2 Spesifikasi Minimum Perangkat Lunak	90
Tabel 5.3 Pengujian <i>Black Box</i>	107

BAB I

PENDAHULUAN

1.1. LATAR BELAKANG

Peta merupakan gambaran permukaan bumi yang ditampilkan pada suatu bidang datar dengan skala tertentu. Peta bisa disajikan dalam berbagai cara yang berbeda, mulai dari peta konvensional yang tercetak hingga peta digital yang tampil di layar komputer. Istilah peta berasal dari Bahasa Yunani *mapp* yang berarti taplak atau kain penutup meja. Secara umum peta memiliki beberapa fungsi, yaitu memperlihatkan posisi secara *vertical* atau *horizontal* dari suatu objek di permukaan bumi, memvisualisasikan ukuran maupun bentuk yang dikumpulkan dan menseleksi objek – objek tersebut.

Perkembangan teknologi saat ini dalam bidang pemetaan sudah sangat berkembang. Banyaknya pengembangan pemetaan dari peta konvensional ke peta digital, dalam bentuk 2 dimensi maupun 3 dimensi. Salah satu peta digital yang sudah ada saat ini diantaranya *google maps*, *here*, *waze*, dll.

SMP Karya Pembangunan 2 Majalaya sebagai salah satu sekolah menengah pertama yang ada di Indonesia berada di kabupaten Bandung dengan luas 1878 m². Dengan wilayah yang cukup luas dapat dipastikan SMP Karya Pembangunan 2 Majalaya memiliki infrastruktur berupa Gedung atau bangunan dengan jumlah yang banyak. Saat ini informasi mengenai kampus SMP Karya Pembangunan 2 Majalaya berupa peta sudah terdapat di berbagai media, akan tetapi peta tersebut hanya berupa peta master plan yang di ubah menjadi *word* dan berbentuk 2D maupun peta digital Google Maps.

Pengembangan peta tiga dimensi yang terperinci sangat diperlukan, hal ini ditujukan bahwa adanya pengembangan peta tiga dimensi SMP Karya Pembangunan 2 Majalaya. Terkait dengan UU RI No.14 tahun 2008 tentang keterbukaan informasi publik dimana peta kampus media informasi perlu di terbitkan kepada publik. Hal ini dapat diwujudkan dengan membangun sebuah peta tiga dimensi kampus SMP Karya Pembangunan 2 Majalaya dengan menggunakan *software* Blender dan menggunakan fungsi WebGL dengan dilengkapi library dari *Three.JS*

Blender 3D merupakan sebuah *software open-source* yang di peruntukan dalam hal membangun desain tiga dimensi, di dalam blender juga terdapat fitur pengkodean menggunakan Bahasa python, *game engine*, dan masih banyak lagi fitur yang terdapat pada *software* ini.

WebGL (*Web Graphics Library*) adalah teknologi web yang menyuguhkan akselerasi grafis 3D ke dalam browser tanpa memasang perangkat tambahan untuk API, WebGL biasanya dipanggil melalui API javascript dan penggunaanya selalu melibatkan elemen HTML5 (*<canvas>*)(codepolitan.com).

Maka pada penelitian ini penulis mengajukan penelitian skripsi dengan judul “Implementasi Peta 3 Dimensi Menggunakan Metode IMSDD (*Interactive Multimedia System Design And Development*) Dan WEBGL API Berbasis Web Studi Kasus Di SMP Karya Pembangunan 2 Majalaya”.

1.2. RUMUSAN MASALAH

Berdasarkan latar belakang yang telah di paparkan di atas, maka dapat dirumuskan permasalahan yang akan dibahas dalam proposal skripsi ini adalah “Bagaimana membangun peta tiga dimensi pada kampus SMP KP 2 ?, Majalaya menggunakan metode *Interactive Multimedia System Design and Development* dan WebGL API yang dilengkapi library dari *Three.js* berbasis *web* dapat menjadi panduan untuk mengenal tata letak bangunan kampus SMP Karya Pembangunan 2 Majalaya.

1.3. BATASAN MASALAH

Sebagai ruang lingkup penelitian skripsi ini, penulis mengambil batas cakupan pembahasan agar menjaga konsistensi tujuan dari perancangan aplikasi itu sendiri, sehingga masalah yang dihadapi tidak meluas dan pembahasan menjadi terarah. Batasan tersebut adalah:

1. Ruang lingkup dalam penelitian ini adalah kampus SMP Karya Pembangunan 2 Majalaya
2. Aplikasi di buat kedalam format .html dan hanya diterapkan ke dalam *web browser*

3. Aplikasi dirancang sebagai panduan untuk mengenalkan nama – nama Gedung dan ruangan yang ada di kampus SMP Karya Pembangunan 2 Majalaya
4. Menunjukkan gambaran 3D lokasi, bentuk, dan nama bangunan yang dipilih dalam bentuk tampilan *menu* utama dan *submenu*.

1.4. TUJUAN PENELITIAN

Tujuan penelitian ini adalah melakukan rancang bangun peta 3D kampus SMP Karya Pembangunan 2 Majalaya yang diimplementasikan kedalam *web* menggunakan blender dan *WebGl API* sebagai panduan dalam mengenal tata letak dan nama-nama Gedung serta ruangan yang ada di kampus SMP Karya Pembangunan 2 Majalaya.

Menghasilkan output html sebagai media menampilkan aplikasi yang bisa diakses menggunakan aplikasi browser seperti *chrome*, *firefox* dan masih banyak lagi.

Supaya siswa baru mampu melihat dan mendefinisikan tata letak gedung serta bangunan SMP Karya Pembangunan 2 Majalaya yang bisa di akses di rumah dan dimanapun, lebih mudah menghafal tata letak dan gedung tanpa harus melihat secara langsung.

Menunjukkan gambaran kecil bentuk bangunan serta letak bangunan SMP Karya Pembangunan 2 Majalaya.

1.5. METODOLOGI PENELITIAN

Metodologi yang digunakan dibagi menjadi 2 metode diantaranya Metode Pengumpulan Data dan Metode Pengembangan Sistem :

- a. Metode Pengumpulan data diantaranya : Studi Pustaka, Kuesioner, dan Wawancara
- b. Metode Pengembangan Sistem : metode pengembangan yang dipilih sesuai dengan sistem yang di buat adalah metode *Interactive Multimedia System Design and Development (IMSDD)* yang memiliki tahapan - tahapan diantaranya : *System Requirement, Design Consideration, Implementation, and Evaluation*.

1.6. SISTEMATIKA PENULISAN

Penulisan dalam skripsi ini disajikan dalam 5 bab dimana setiap bab mempunyai keterkaitan sebagai berikut :

BAB 1 Pendahuluan

Bab ini memuat latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, dan sistematika penulisan.

BAB 2 Tinjauan Pustaka

Merupakan cakupan teori-teori dan ulasan penelitian terdahulu yang menunjang dalam menyelesaikan penelitian yang akan dilaksanakan.

BAB 3 Metodologi Penelitian

Bab ini berisikan uraian langkah-langkah penyelesaian yang ditempuh meliputi kerangka pikir dan deskripsi.

BAB 4 Analisis dan Perancangan

Pada bagian ini membahas penelitian dan analisis yang dilaksanakan sesuai dengan metodologi penelitian.

BAB 5 Implementasi dan Pengujian

Menjelaskan tentang pengimplementasian aplikasi serta dilakukan tahap pengujian

BAB 6 Kesimpulan dan Saran

Berisi tentan kesimpulan yang telah di dapat dari dibuatnya aplikasi serta saran yang diterima dari pembuatan aplikasi

BAB II

TINJAUAN PUSTAKA

2.1. LANDASAN TEORI

Pada tahap ini, penulis mendapatkan referensi terkait dengan penelitian. Referensi didapatkan dari *e-book*. Informasi tersebut akan digunakan dalam penyusunan landasan teori, metodologi penelitian beserta cara membangun peta tiga dimensi kampus. Serta penulis mencari jurnal – jurnal penelitian terkait dengan tema yang di ambil untuk di jadikan perbandingan yang penulis lakukan. Hal tersebut dilakukan demi menghindari kesamaan penelitian yang telah dilakukan sebelumnya. Dengan melakukan hal ini, penulis juga dapat mengembangkan penelitian yang telah dilakukan.

Tabel 2.1 Tabel Ikhrisar

No	Judul	Masalah	Metode	Solusi
1	Visualisasi Kampus 3d Virtual Berbasis WebGL Menggunakan Three.js Studi Kasus: Gedung Teknik Informatika Sepuluh Nopember	Masyarakat yang ingin mencari informasi terkait Gedung Teknik Informatika Institut Teknologi Sepuluh Nopember dibatasi dengan hal – hal umum dan terbatas	Pengumpulan data observasi dan wawancara langsung untuk mendapatkan data akurat yang selanjutnya masuk tahap perancangan sistem.	Maka dibuatlah sebuah aplikasi yang bisa memudahkan pengguna mencari informasi tentang kampus Teknik informatika dalam bentuk Visual yang dilengkapi Navigasi interaktif dalam model 3D kampus Teknik informatika dengan daya jelajah pada game computer dengan <i>first-person view</i> , menampilkan informasi bagian dalam kampus menggunakan mode <i>free view</i> , <i>automati flights</i> yang berguna untuk rute

				terpendek dalam pencarian tempat dalam kampus.
2	Rancang Bangun Peta 3d Kampus 1 Uin Syarif Hidayatullah Jakarta Menggunakan Metode Imsdd (Interactive Multimedia System Design And Development) Dan Framework Blend4web Berbasis Web	Memiliki informasi yang minim tentang tata letak bangunan dan tempat sehingga sangat menyulitkan bagi mahasiswa baru dan umum untuk berkunjung.	Metode IMSDD (<i>Interactive Multimedia System Design And Development</i>)	Maka dari itu dibuat sebuah aplikasi berbasis web yang bisa memvisualisasikan bangunan dengan bentuk 3D menggunakan Blend4Web
3	Rancang Bangun Peta Interaktif Tiga Dimensi Monitoring Lahan (Monlah) Di Institut Teknologi Sepuluh Nopember	Karena luas lahan yang dimiliki oleh ITS terlalu luas, sehingga menyulitkan dalam menjalankan programnya, dari kekurangan tersebut maka diperlukan sebuah alat pemantauan kondisi lahan	Melakukan survey dan melakukan perancangan.	Cara membangun penggambaran peta lahan ITS dalam bentuk digital / website

Dari ketiga jurnal diatas penulis menyimpulkan beberapa kemungkinan, kelebihan serta kekurangan yang akan dihadapi sesuai dengan metode yang dipilih, berikut diantaranya :

1. Kemungkinan yang akan dihadapi dari ketiga simpulan diatas yaitu : mungkin semua pengguna mampu mengoperasikan sistem yang dibuat
2. Kelebihan yang dimiliki dari ketiga jurnal diatas yaitu sistem berbasis Web sehingga mampu diakses dari berbagai perangkat.

3. Kekurangan yang didapat dari tabel no 2 yang ditulis oleh ('ANGGER PURSAN PRATAMA-FST.pdf', no date) yaitu sistem yang dibuat harus memuat *play and stop* untuk menuju sistem. Metode ini sangat kurang efisien melihat dari kelengkapan sistem

Dari beberapa kemungkinan, kelebihan serta kekurangan penulis akan meningkatkan versi milik ('ANGGER PURSAN PRATAMA-FST.pdf', no date) dan menggunakan metode IMSDD sebagai sarana untuk mendapatkan visual serta desain yang begitu interaktif dan memanjakan mata, kelebihan yang akan di dapat pada versi selanjutnya yaitu hilangnya penggunaan *play and stop* untuk memudahkan pengguna tanpa harus menekan tombol *play and stop*.

2.2. DASAR TEORI

2.2.1. Informasi

Informasi adalah sekumpulan data atau fakta yang dikelola menjadi sesuatu yang bermanfaat bagi penerimanya. Informasi akan diproses terlebih dahulu agar penerima mudah untuk memahami informasi yang diberikan. Informasi yang akan di tampilkan kepada penerima biasanya sudah diolah menjadi bentuk yang bernilai atau bermakna.

Menurut (BODNAR, 2000) Pengertian informasi adalah sebuah data yang diolah sehingga dapat dijadikan dasar untuk mengambil keputusan yang tepat.

Menurut *M.Melino* (1990:331) Informasi merupakan data yang telah diproses dan diolah untuk tujuan tertentu. Adapun tujuan tersebut untuk menghasilkan sebuah keputusan.

1. Jenis-jenis informasi

Informasi dapat ditemukan dalam format dan bentuk apapun, baik media cetak maupun media *online*. Sebuah data dapat dikatakan sebagai informai ketika benar-benar berfungsi atau benar-benar digunakan. Ada beberapa jenis informasi sebagai berikut :

a. Informasi nyata

Informasi nyata atau faktual adalah informasi yang hanya berhubungan dengan fakta, biasanya jenis informasi ini jarang memberikan latar belakang yang mendalam tentang suatu topik tertentu.

b. Informasi analisis

Informasi analisis ialah sebuah informasi yang biasanya dihasilkan peneliti dalam studi tertentu.

c. Informasi subjektif

Informasi subjektif yang hanya dilihat dari satu sudut pandang, biasanya informasi ini berisi tentang pendapat atau argumentasi dari pihak tertentu.

d. Dan informasi objektif

Informasi objektif merupakan informasi yang dapat dipahami dari berbagai sudut pandang.

2. Fungsi informasi

Informasi memiliki beragam fungsi yang bermanfaat untuk kehidupan sehari-hari. Salah satunya informasi menyediakan peristiwa dan kondisi dalam masyarakat tertentu, menunjukan hubungan kekuasaan, serta memudahkan berbagai macam inovasi. Dengan begitu masyarakat umum bisa memperoleh informasi yang berkaitan dengan kebutuhan dan kepentingannya sebagai sumber pengetahuan baru.

2.2.2. Multimedia

1. Definisi Multimedia

Menurut *Steinmetz* (1995, p2) : Multimedia adalah gabungan dari seminimalnya sebuah media dimana validitas datanya tidak tergantung dari kondisi waktu, termasuk didalamnya teks dan grafik. Sedangkan yang dimaksud dengan media kontinu adalah sebuah media dimana validitas datanya tergantung dari kondisi waktu, termasuk di dalamnya suara dan video.

Multimedia diambil dari kata multi dan media. Multi berarti banyak dan media berarti media atau perantara. Media adalah gabungan dari beberapa unsur yaitu teks, grafik, suara, video dan animasi yang menghasilkan presentasi yang menakjubkan. Multimedia juga mempunyai komunikasi interaktif yang tinggi. Bagi pengguna computer multimedia dapat diartikan sebagai informasi computer yang dapat disajikan melalui audio atau video, teks, grafik, dan animasi. Disini dapat digambarkan bahwa multimedia adalah suatu kombinasi data atau media untuk menyampaikan suatu informasi sehingga informasi itu tersaji dengan lebih menarik.

2. Jenis dan Kategori Multimedia

Multimedia dapat dibagi menjadi dua jenis menurut (Munir, 2012) , yaitu:

- a. Berbentuk *online* yang terhubung oleh jaringan internet, dapat diakses tanpa memerlukan penyimpanan secara local
- b. Berbentuk *offline* yang tidak terhubung oleh jaringan internet, memiliki penyimpanan, alat *input* dan *output*, serta *Video Graphic Adapter*.

Sedangkan kategori – kategori dalam multimedia menurut (Munir, 2012), diantaranya:

- a. Multimedia linier, merupakan multimedia yang tidak dilengkapi dengan sebuah alat pengendali yang dapat digunakan pengguna dalam pengoperasiannya.
- b. Multimedia interaktif, merupakan multimedia yang dilengkapi oleh alat pengendali yang digunakan oleh pengguna dalam pengoperasiannya, seperti aplikasi permainan, dan lain-lain.

3. Elemen Multimedia

Elemen – elemen multimedia dapat merujuk pada pengoperasiannya, diantara lain (Munir, 2012):

- a. Multimedia non-temporal merupakan multimedia yang tidak tergantung pada waktu terdiri atas teks, grafika, dan gambar.

- b. Multimedia temporal merupakan multimedia yang tergantung pada waktu terdiri atas audio, video dan animasi.

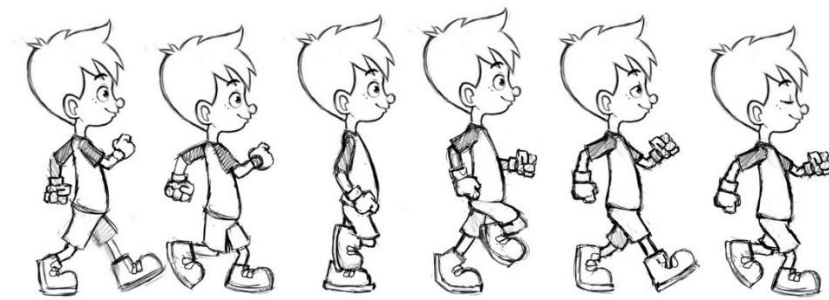
2.2.3. Animasi

Animasi menurut (Munir, 2012) adalah suatu kegiatan atau pekerjaan menghidupkan dan menggerakkan benda mati yang diberi dorongan atau gerakan supaya seperti menjadi hidup. Animasi dapat diartikan sebagai gambar yang memuat satu atau lebih objek yang seolah – olah hidup yang disebabkan oleh kumpulan gambar-gambar yang disusun beraturan dan bergantian ditampilkan.

1. Animasi 2D

Animasi dua dimensi biasa dikenal dengan nama *flat animation*, hanya memiliki ukuran Panjang (*x-axis*) dan lebar (*y-axis*). Animasi dua dimensi pada awal diciptakan dan kemunculannya pertama kali berupa film kartun, hal ini menyebabkan dikenalnya animasi dua dimensi sebagai animasi kartun. Nama kartun sendiri berasal dari kata *cartoon* yang berarti gambar yang lucu. Contoh animasi dua dimensi adalah *Tom & Jerry*, *Scooby-Doo* dan *Spongebob* (Munir, 2012)

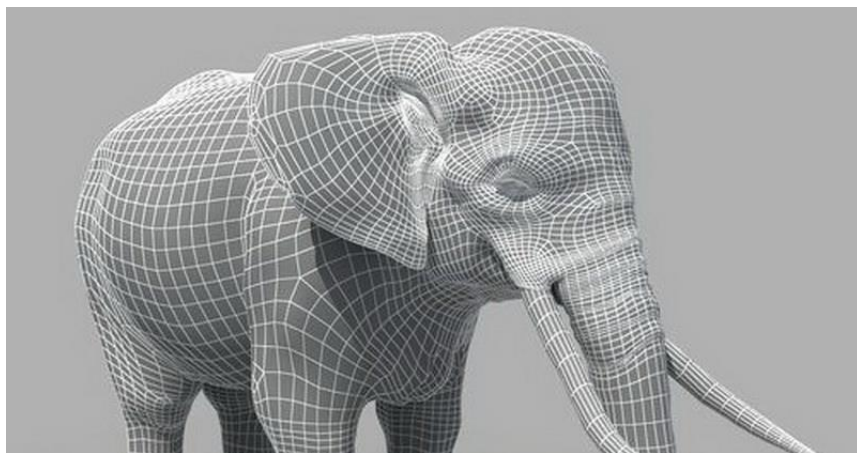
Awal pembuatan animasi dua dimensi dikerjakan menggunakan beberapa sketsa gambar yang Digambar satu persatu setiap gerakan dan di susun setiap sketsa gambar sedemikian rupa sehingga membentuk Gerakan. Orang yang mendesain dan membuat animasi disebut animator. Pembuatan animasi sekarang sudah menggunakan computer, hanya perlu membuat Gerakan antar *keyframe* dan tidak perlu membuat gambar satu persatu seperti pembuatan animasi konvensional (Munir, 2012)



Gambar 2.1 Contoh Sketsa gambar untuk animasi 2D
(Sumber: www.alawiahala.blogspot.com)

2. Animasi 3D

Animasi tiga dimensi merupakan perkembangan dari animasi dua dimensi. Dengan animasi tiga dimensi, karakter yang diperlihatkan semakin terasa hidup dan nyata mendekati wujud asli. Tiga dimensi adalah sebuah dimensi dimana memiliki ruang atau *volume* yang memiliki koordinat sumbu X, Y dan Z (Munir, 2012)



Gambar 2.2 Contoh animasi 3D
(Sumber: www.techno.okezone.com)

2.2.4. Blender 3D

Blender 3D merupakan sebuah perangkat lunak untuk membuat desain dan animasi 3D secara gratis dan *open source*. Dukungan terhadap semua pemodelan 3D, animasi, simulasi, *rigging*, *rendering*, *compositing* dan *motion tracking*, serta

dukungan terhadap *editing video* dan pembuatan *game* (Blender.org).



Gambar 2.3 Tampilan antarmuka *software* Blender 3D
(Sumber: www.blender.org)

Sedangkan menurut Allan (2008) dalam Risch H. Aswana (2018:11), Blender 3D merupakan sebuah perangkat lunak *open-source* yang digunakan untuk pemodelan, *rendering*, dan animasi dalam bentuk tiga dimensi. Kelebihan utama dari Blender adalah:

1. Bersifat *open source*
2. Bentuk *installer* blender lebih kecil ukurannya dibandingkan dengan *software* grafis 3D yang lain
3. Memiliki banyak *tools* didalamnya untuk membuat objek 3D seperti *modelling*, *UV Mapping*, *Texture Rigging*, *Sculpturing*, *Animation*, dan *Particles*
4. Dapat diimplementasikan ke dalam berbagai platform sistem operasi
5. Terdapat fitur yang dapat digunakan sebagai *game engine* dalam membuat sebuah *game* tanpa harus menggunakan program tambahan dan menggunakan Bahasa pemrograman Python

2.2.5. WebGL API

WebGL API (*Web Graphics Library*) adalah teknologi web yang menyuguhkan akselerasi grafis 3D ke dalam browser tanpa memasang perangkat lunak tambahan. Untuk API, WebGL biasanya dipanggil melalui API javascript dan penggunaannya selalu melibatkan elemen HTML5 *<canvas>* (codepolitan.com).

WebGL terintegrasi sepenuhnya kesemua standar web browser yang memungkinkan penggunaan percepatan GPU fisika dan pengulahan gambardan efek sebagai bagian dari kanvas halaman web. Elemen WebGL dapat dicampur dengan elemen HTML lainnya dan *composited* dengan bagian-bagian lain dari latar belakang halaman atau halaman. WebGL program terdiri dari kode kontrol yang ditulis dalam *JavaScript* dan kode shader yang dijalankan pada komputer *Graphics Processing Unit* (GPU).

Berawal dari percobaan yang dilakukan oleh *Vladimir Vukicevic* Ketika beradadi *Mozilla*. Vukicevic pertama kali mendemonstrasikan prototipe *Canvas* 3D pada tahun 2006. Dipenghujung tahun 2007, *Mozilla* dan *Opera* mulai membangunnya secara terpisah.

2.2.6. Three.js

Three.js adalah library *Java Script* untuk membuat game dan aplikasi 3D. *Three.JS* menggunakan WebGL yang mana merupakan API Java Script untuk merender grafik 2D dan 3D interaktif yang *native* browser tanpa memerlukan plugin browser tambahan.

2.2.7. Model Viewer

Model Viewer adalah komponen web open source yang dikembangkan oleh Google dan dikelola melalui GitHub. Model viewer bertujuan untuk menempatkan konten 3D di web dengan mudah dengan beberapa baris kode HTML. Pertama kali diperkenalkan dengan Chrome 72 pada juli 2019 dan memungkinkan pengguna untuk melihat 3d di browser dan perangkat seluler.

2.2.8. HTML

Hypertext Markup Language (HTML) adalah bahasa standar untuk dokumen yang dirancang untuk ditampilkan diperamban internet. dapat dilengkapi oleh CSS dan bahasa scripting seperti *JavaScript* dan *VBScript*.

HTML dapat menyematkan program yang ditulis dalam bahasa scripting seperti *JavaScript*, yang mempengaruhi perilaku dan konten halaman web. Dilengkapi dengan CSS mendefinisikan tampilan dan tata letak konten.

2.2.9. *Java Script*

JavaScript disingkat JS adalah bahasa pemrograman tingkat tinggi dan dinamis. *JavaScript* populer di internet dan dapat bekerja di sebagian besar penjelajah web populer seperti *Google Chrome*, *Internet Explorer (IE)*, *Mozilla*, *Firefox*, *Netscape*, *Opera* dan masih banyak lagi. Kode *JavaScript* dapat di sisipkan dalam halaman web menggunakan tag `<script>`.

2.2.10. CSS

Cascading Style Sheet (CSS) adalah bahasa yang digunakan untuk mengatur tampilan elemen yang tertulis dalam bahasa markup, seperti HTML. CSS berfungsi untuk memisahkan konten dari tampilan visualnya di situs.

CSS dibuat dan dikembangkan oleh W3C (*World Wide Web Consortium*) pada tahun 1996.

2.2.11. *Web Component*

Web Component adalah ruang yang mencakup halaman web seperti HTML, *JavaScript*, dan CSS yang telah bundle ke dalam paket yang dapat di sisipkan ke dalam elemen DOM.

2.2.12. DOM

Document Object Model (DOM) adalah sebuah antarmuka pemrograman (*Programing interface*) untuk HTML, XML dan SVG yang bersifat lintas platform dan language independent. Namun DOM diperlukan oleh *JavaScript* yang akan mengubah tampilan

sebuah situs web secara dinamis. Maka dari itu DOM adalah cara *JavaScript* melihat suatu halaman HTML.

2.2.13. Metode Pembangunan Multimedia

Metode pengembangan untuk pembuatan map tiga dimensi ini menggunakan tahapan IMSDD (*Interactive Multimedia System Design and Development*). Tahapan pada IMSDD menurut Dastbaz (2003) dalam Siti Nurul (2018:29-30) terdapat 4 tahap, diantaranya:

1. *System Requirement*

Pada tahapan ini hampir sama dengan tahap spesifikasi kebutuhan dan model dalam *waterfall*. Fungsi dari tahapan ini adalah:

- a. Untuk memberikan definisi pada system seperti membuat garis besar tujuan dan target dari system yang sedang dirancang.
- b. Untuk meyakinkan siapa pengguna yang akan menggunakan system tersebut dan jika terdapat persyaratan khusus lainnya yang perlu dipertimbangkan.
- c. Evaluasi pada perangkat lunak, perangkat keras dan *tools* yang dibutuhkan dan dipilih dengan tepat.
- d. Pertimbangan *platform* penyebaran yang tepat dibutuhkan oleh system

2. *Design Consideration*

Pada tahap ini bertujuan untuk menjelaskan secara terperinci pedoman-pedoman desain, yaitu:

a. Desain Metafora

Menentukan dan memilih model nyata yang akan dipergunakan sebagai solusi utama untuk desain *interface* system

b. Format dan Tipe Informasi

Menjelaskan tipe dari informasi yang diperlukan untuk diintegrasikan kedalam sebuah sistem

c. Struktur Navigasi

Menjelaskan strategi navigasi dengan jelas, yang terdiri dari fitur-fitur supaya terhindar dari masalah yang berhubungan dengan sistem

d. Sistem Kontrol

Menjelaskan tipe dan fitur kendali serta peralatan yang dibutuhkan sistem

3. *Implementation*

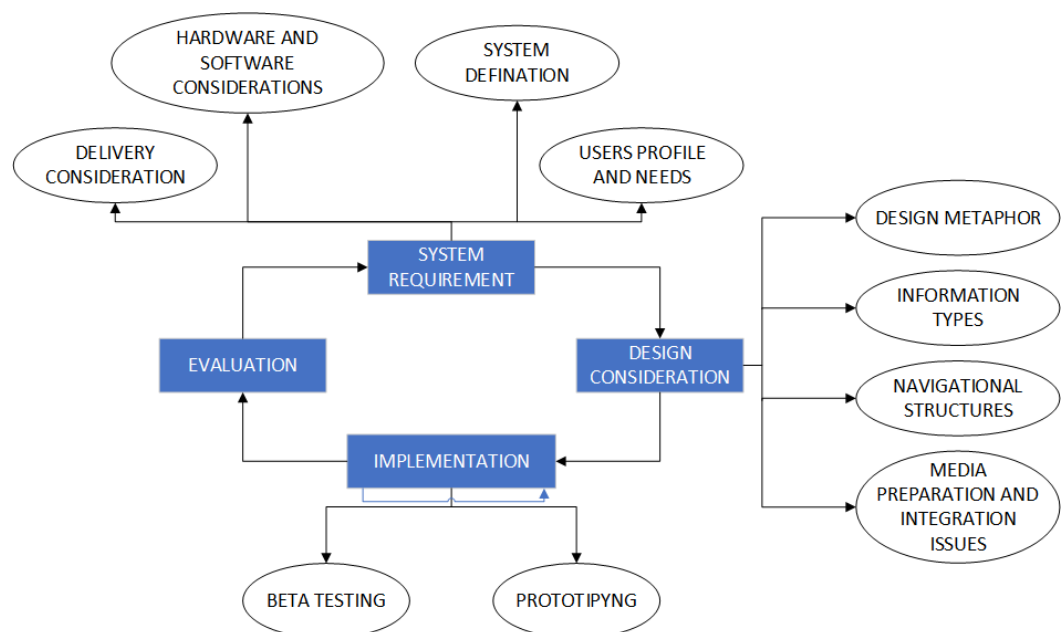
Setelah tahap pertimbangan desain, dilanjutkan kedalam tahapan implementasi system, tahapan ini terdiri dari:

a. Membuat *prototype* dari system

b. Melakukan *beta testing* pada *prototype* untuk menganalisa dan mencari masalah-masalah apa saja yang muncul pada control dan desain

4. *Evaluation*

Setelah tahapan implementasi selesai, dilanjutkan dengan tahapan evaluasi. Tahapan evaluasi menggunakan panduan yang telah dikemukakan ahli.



Gambar 2.4 Siklus dari IMSDD

(Sumber: Dastbaz, 2003)

2.2.14. *Storyboard*

Nenurut Dastbaz (2003) dalam Siti Nurul (2018:31) dan Angger Purnama (2019:16) *storyboard* adalah Teknik yang sudah digunakan selama bertahun - tahun dalam industri perfilman dengan menggunakan serangkaian gambar yang digunakan oleh sutradara dengan alat visual yang berfungsi dalam melihat bagaimana cerita dalam film berkembang adegan demi adegan. Selain itu, *storyboard* merupakan Teknik yang berguna untuk menggambarkan *user interface* pada sebuah multimedia interaktif.

Sedangkan *storyboard* menurut Vaughan (2011:300) merupakan gambar artistektural untuk proyek multimedia dalam bentuk satu kesatuan selama proses desain dan membantu menggambarkan arsitektur informasi.

1. Fungsi *Storyboard*

Storyboard sering disebut papan cerita. *Storyboard* berfungsi untuk menggambarkan alur cerita, mulai dari awal hingga akhir. Selain itu, *storyboard* juga berfungsi untuk merencanakan proses pengambilan gambar agar lebih terstruktur.

2. Manfaat *Storyboard*

Mengutip dari elemental media, *storyboard* memiliki 3 manfaat utama yaitu :

a. Mengoorganisir proses pengambilan gambar

Storyboard mempermudah proses pengambilan gambar, agar lebih efektif dan efisien. Selain itu, adanya *storyboard* juga membantu penentuan pengambilan angle gambar dan mengetahui bagan selanjutnya.

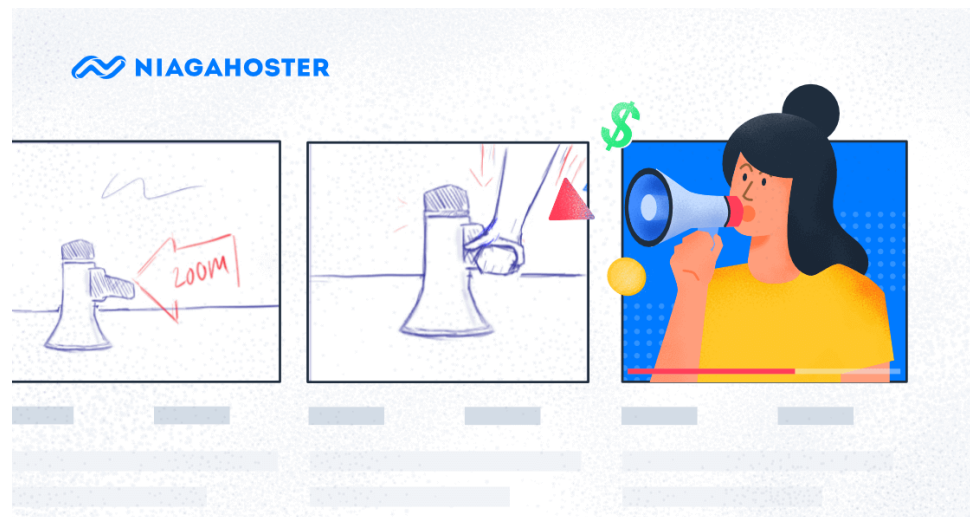
b. Memudahkan dalam membuat dan memahami alur cerita

Storyboard berisikan sketsa gambar berurutan sesuai naskah, sehingga mempermudah proses pembuatan pemahaman alur ceritanya. Sehingga seluruh orang

memiliki persepsi yang sama dan proses pengambilan gambar sesuai dengan yang diinginkan.

c. Dapat mengetahui kesalahan pada bagian awal

Pembuatan *storyboard* bisa memudahkan orang dalam mencari tahu kesalahan pada teks narasi, penggunaan media dan detail lainnya dalam *storyboard*. Hal ini juga dapat menghemat waktu, biaya, dan tenaga.



Gambar 2.5 Contoh *storyboard* (kiri) dan hasil antarmuka yang telah diimplementasikan (kanan)

(Sumber: www.niagahoster.com).

2.2.15. *State Transition Diagram* (STD)

State Transition Diagram menurut Hoffer pada Angger Purnama (2019) adalah suatu alat permodelan yang menggambarkan suatu state yang dihubungkan dengan *state* yang lainnya pada suatu diagram.

Pada dasarnya *State Transition Diagram* adalah sebuah diagram yang terdiri dari beberapa *state* dan transisi atau perpindahan antar *state*. Transisi *state* terdiri atas sebuah kondisi dan aksi. Kondisi disini merupakan suatu kejadian yang bisa diketahui oleh system. Aksi merupakan sebuah Tindakan yang dilakukan oleh system jika terdapat perubahan *state*.

2.2.16. *Beta Testing*

Beta Testing menurut Fine (2002:11) pada Angger Purnama (2019:17) adalah pendistribusian produk ke pasar targetnya, biasanya meminta target untuk turut berpartisipasi dalam pengujian produk sebelum peluncurannya. Pengujian beta dilakukan supaya dapat menemukan masalah yang tidak dapat ditemukan di dalam laboratorium uji. Beragamnya para peserta pengujian beta dapat mengeksploitasi keterbatasan pada produk dan menguji keseluruhan pengembangan pada produk.

2.2.17. UML

Unified Modelling Language (UML) yaitu suatu metode pemodelan secara visual untuk sarana perancangan sistem berorientasi objek, definisi UML yaitu suatu bahasa standar penulisan blue print pada visualisasi, perancangan dan juga pendokumentasian sistem software.

1. Tujuan UML

Tujuan dibuatnya UML dalam perancangan sebuah software sistem diantaranya:

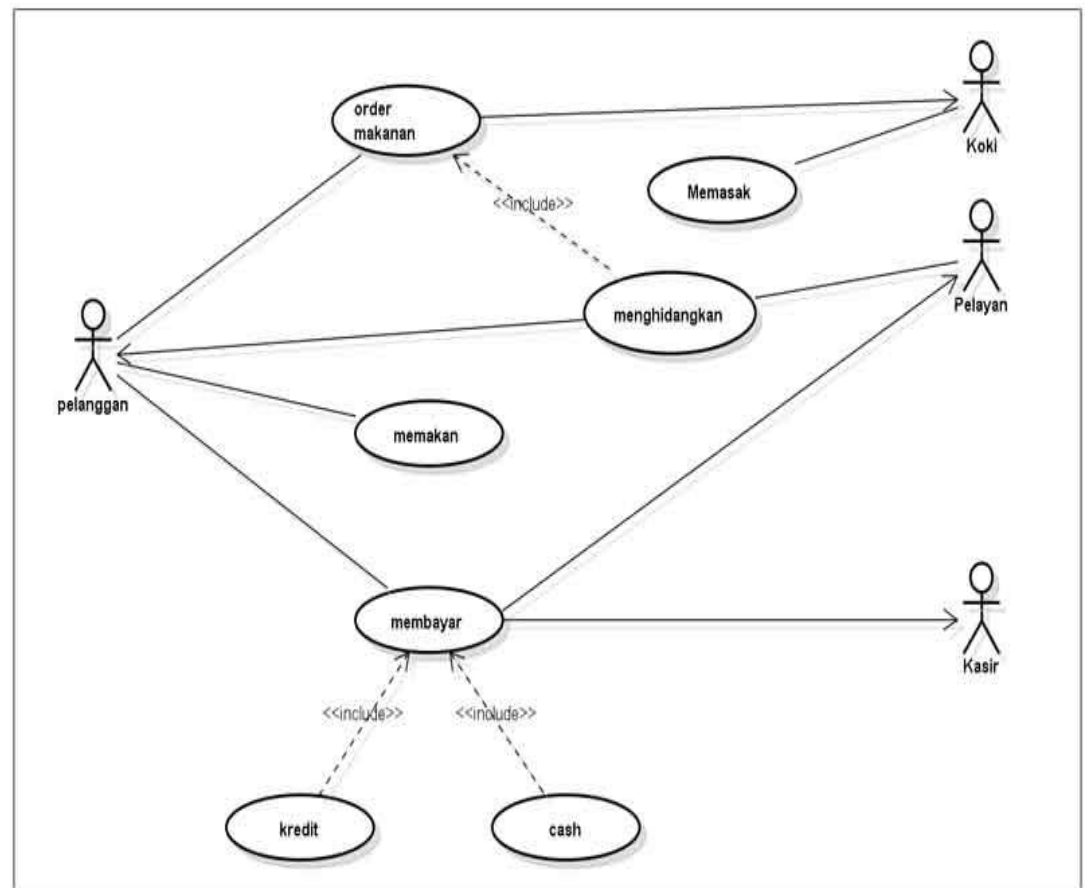
- a. Memberikan pemodelan visual kepada pengguna dari berbagai macam pemrograman maupun proses rekayasa
- b. Memberikan model terbaik
- c. Menyajikan model yang siap untuk digunakan, yaitu merupakan bahasa pemodelan visual yang ekspresif untuk mengembangkan sistem dan untuk saling menukar model secara mudah
- d. Berguna sebagai *blue print*
- e. Pemodelan sistem berorientasi objek
- f. Menciptakan bahasa pemodelan yang dapat dipergunakan oleh manusia dan mesin

2. Jenis-jenis UML

Jenis-jenis UML antara lain:

a. Use Case Diagram

Use case diagram yaitu salah satu jenis diagram pada UML yang menggambarkan interaksi sistem dan aktor, use case diagram juga dapat men-deskripsikan tipe interaksi antara pemakai dengan sistemnya.

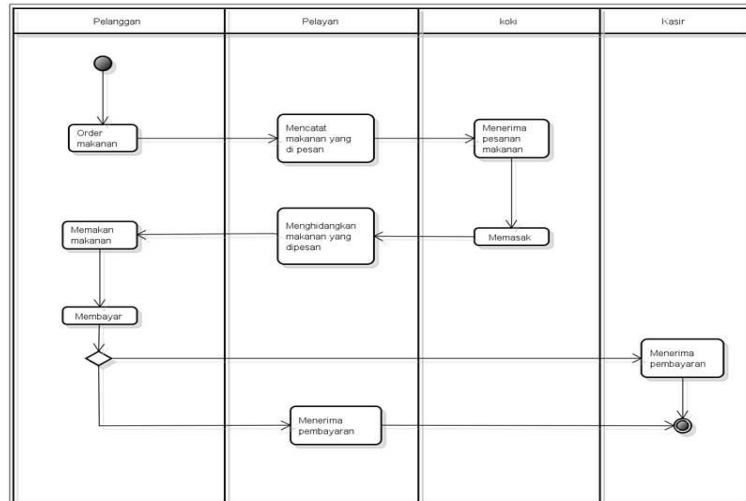


Gambar 2.6 Contoh *Use Case Diagram*

(Sumber: www.pengertianku.net)

b. Activity Diagram

Activity diagram atau diagram aktifitas yaitu salah satu jenis diagram pada UML yang dapat memodelkan proses-proses apa saja yang terjadi pada sistem.

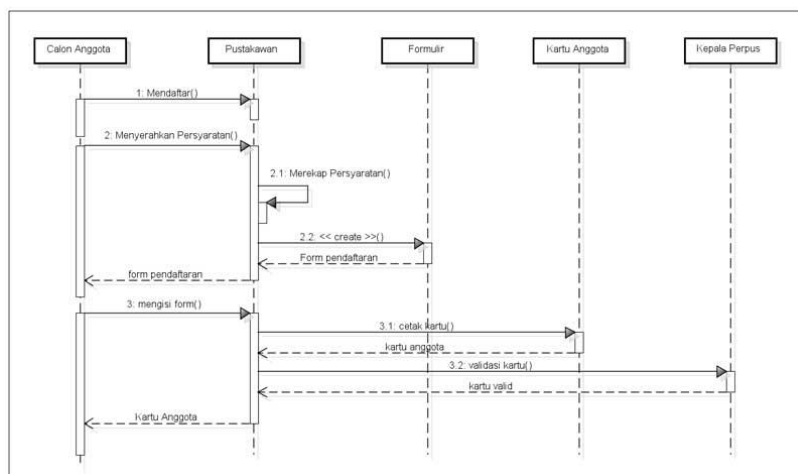


Gambar 2.7 Contoh Activity Diagram

(Sumber: www.pengertianku.net)

c. Sequence Diagram

Sequence diagram yaitu salah satu jenis diagram UML yang menjelaskan interaksi objek yang berdasarkan urutan waktu, sequence diagram juga dapat menggambarkan urutan atau tahapan yang harus dilakukan untuk dapat menghasilkan sesuatu seperti pada use case diagram.



Gambar 2.8 Contoh Sequence Diagram

(Sumber: www.pengertianku.net)

2.2.18. *Vercel App*

Vercel App (sebelumnya dikenal sebagai ZEIT) adalah platform cloud yang memungkinkan pengembang untuk meng-host situs web dan layanan web yang disebarakan secara instan, di skalakan secara otomatis, dan tidak memerlukan pengawasan.

Didirikan pada tahun 2015 oleh *Guillermo Rauch*,

VERCEL menawarkan antarmuka pengguna yang intuitif dengan konfigurasi minimal untuk hosting generator situs statis seperti *Gatsby* atau Hugo dan berbagai CMS Seperti *Contentful*, *prismic*, maupun *Wordpress*. Vercel juga merupakan perusahaan induk dari *framework* Next.js dan di lengkapi dengan fitur yang canggih.

2.2.19. *Vite.js*

Vite.js adalah alat pengembangan cepat untuk proyek web modern. Ini berfokus pada kecepatan dan kinerja dengan meningkatkan pengalaman pengembangan.

Vite menggunakan ES Browser asli untuk mengaktifkan dukungan untuk browser modern tanpa proses build.

Vite terdiri dari dua bagian utama :

1. *Dev Server*

Menyediakan dukungan untuk mengganti *Hot Module Replacement* (HMR) untuk memperbarui modul selama eksekusi aplikasi. Saat perubahan dibuat pada kode sumber aplikasi, hanya perubahan yang di diperbarui, bukan seluruh aplikasi yang dimuat ulang. Fitur ini membantu mempercepat waktu pengembangan.

2. *The Build Command*

The Build Command memungkinkan untuk menggabungkan kode yang dibuat dengan *rollup*, yang telah dikonfigurasi sebelumnya untuk menghasilkan aset statis yang sangat dioptimalkan untuk produksi.

2.2.20. glTF

glTF (*Graphics Language Transmission Format*) adalah format file standar untuk adegan dan model 3 dimensi, file glTF menggunakan salah satu dari kedua kemungkinan ekstensi file, antara lain :

1. glTF (JSON / ASCII)
2. glb (*Binary*).

.glTF dan .glb dapat mereferensikan sumber daya biner dan tekstur eksternal. Alternatifnya kedua format dapat mandiri dengan menyematkan buffer data biner secara langsung (sebagai *string* yang disandikan *base64* dalam .gltf atau sebagai *array byte* mentah dalam file .glb).

glTF adalah file *open-source* yang dikembangkan dan dikelola oleh *Group Khronos*, mendukung geometri model 3D, tampilan, hierarki grafik adegan, dan animasi.

2.2.21. *The Eight Golden Rules of Interface Design*

Dalam membangun sebuah sistem interaktif, perlu adanya aturan – aturan yang dapat digunakan sebagai panduan dalam hal mendesain antarmuka. Panduan atau prinsip-prinsip yang biasa digunakan dalam hal mendesain antarmuka sebuah system adalah *The Eight Golden Rules of Interface Design*. Delapan prinsip ini berawal dari pengalaman pengguna dan telah disempurnakan selama 20 tahun. Berikut ini merupakan prinsip-prinsip dari *golden rules* menurut *Shneidermann* dan *Plaisant* (2016:74).

1. *Strive for Consistency*

Salah satu kedelapan aturan yang sering dilanggar adalah *Strive for Consistency*. Aturan ini bisa jadi sulit diterapkan karena ada berbagai macam bentuk konsistensi. Penerapan desain antarmuka harus konsisten dalam segi tata letak, pewarnaan, kegunaan, maupun bentuk *font* serta terus digunakan sepanjang waktu dan tidak berubah-ubah.

2. *Cater to Universability*

Prinsip ini merupakan prinsip yang digunakan setelah mengenal dan memahami kebutuhan setiap pengguna. Kebutuhan dari setiap pengguna ada berbagai macam, hal ini dapat dibedakan berdasarkan keahlian, rentang usia, pengalaman pengguna dan sebagainya. Seperti memberikan fitur untuk pemula yang berisikan penjelasan dari tiap-tiap kegunaan yang ada di *interface*.

3. *Offer Informative Feedback*

Memberikan umpan balik informatif merupakan prinsip yang diterapkan pada setiap Tindakan pengguna. Pada Tindakan pengguna yang sering dan kecil bisa diterapkan respon yang sederhana, sedangkan pada Tindakan yang jarang dan besar maka respon yang diterapkan pada system harus substansial.

4. *Design Dialogs to Yeild Closure*

Prinsip ini merupakan prinsip yang diterapkan sebagai penunjuk adanya aturan saat pengguna melakukan Tindakan tertentu dengan adanya awal, tengah dan akhir.

5. *Prevent Errors*

Prinsip yang diterapkan untuk mencegah pengguna dalam melakukan kesalahan Tindakan, seperti penempatan menu-menu yang tidak penting dalam system tersebut. Jika pengguna melakukan kesalahan, pada desain antarmuka seharusnya memberikan peringatan dan instruksi yang sederhana untuk pemulihan.

6. *Permit Easy Reversal of Actions*

Prinsip ini diterapkan untuk mengurangi kecemasan pada pengguna dan harus memberikan opsi untuk mengembalikan pengguna ke posisi sebelumnya.

7. *Support Internal Locus of Control*

Menjadikan pengguna sebagai pemegang kendali penuh dalam system. Memastikan desain yang dibangun dapat memberikan kenyamanan pada pengguna dalam mengendalikan seluruh navigasi pada system tersebut.

8. *Reduce Short-term Memory Load*

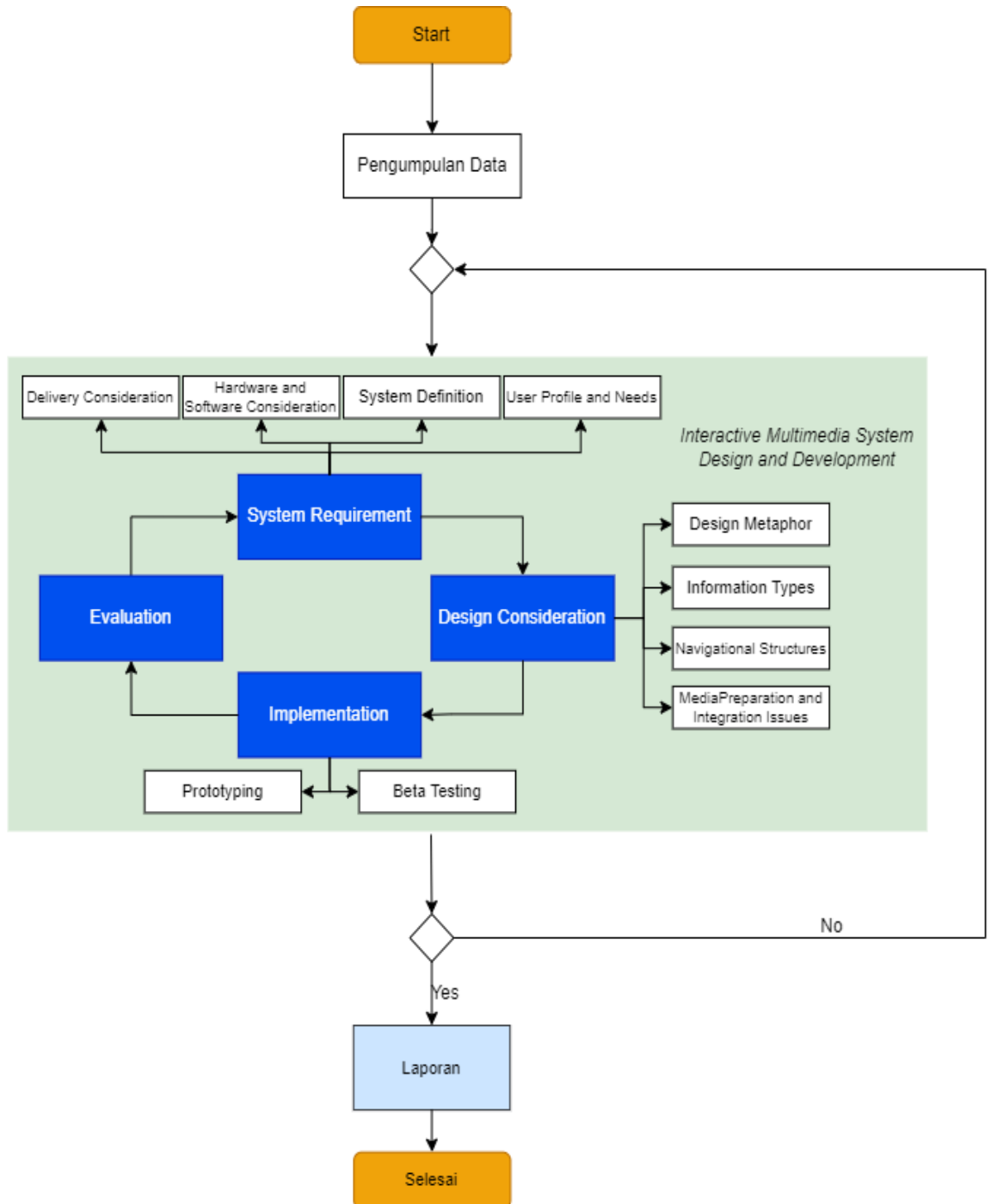
Keterbatasan proses informasi pada manusia dalam ingatan jangka pendek mengharuskan agar tampilan dibuat sesederhana mungkin dan menarik supaya mengurangi beban ingatan tersebut.

BAB III

METODOLOGI

3.1. KERANGKA PIKIR

Dalam melakukan penelitian penulis melakukan tahapan kegiatan sesuai dengan *flowchart* dibawah ini:



Gambar 3.1 *Flowchart* Kerangka Berfikir

3.2. DESKRIPSI

3.2.1. Pengumpulan Data

Dalam pengumpulan data ada beberapa cara yang di gunakan oleh peneliti, diantaranya :

1. Studi Literature

Dalam melakukan studi literature penulis menggunakan beberapa jurnal untuk di baca dan di telaah seperti jurnal dibawah ini :

- a. Di tulis oleh Munir dengan judul Multimedia Konsep dan Aplikasi dalam Pendidikan.
- b. Adapun jurnal yang di tulis oleh Najjar, Lawrence J. dengan judul *Journal of Educational Multimedia and Hypermedia*, v5 n2 1996.
- c. Adapun jurnal lain yang di tulis oleh Edward Angel, Dave Shreiner dengan judul *Interactive Computer Graphics with WebGL*.

Dan masih banyak lagi studi literature yang digunakan untuk berbagai macam referensi.

2. Observasi

Untuk mendapatkan data berupa informasi terkait penelitian yang telah di pilih sesuai tema dari penulis antara lain :

- a. melakukan pengukuran pada setiap sudut gedung untuk mendapatkan minimal capaian ukuran dan letak gedung SMP Karya Pembangunan 2 Majalaya.
- b. mendapatkan foto terkait gedung SMP Karya Pembangunan 2 Majalaya, dari memotret tampak depan, samping, hingga belakang.
- c. mencari berapa total luas lahan untuk melakukan conversi scala kedalam peta 3D.

3. Kuesioner

Dalam melakukan riset penelitian, penulis mengajukan 10 pertanyaan sederhana yang mudah di fahami bagi calon siswa SMP Karya Pembangunan 2 Majalaya. mencakup pada *before and after* sehingga menghasilkan total persentase (%) yang dicapai jika siswa – siswi baru menggunakan aplikasi 3D untuk menghafal tata letak atau denah tanpa harus melihat secara langsung.

Adapun metode yang digunakan untuk mengembangkan sistem yaitu menggunakan metode IMSDD (*Interactive Multimedia System and Development*). Metode ini merujuk pada elemen-elemen pengembangan sistem berbasis multimedia temporal.

3.2.2. System Requirement

Pada tahapan ini penulis melakukan analisis sistem informasi yang akan dirancang sehingga dapat menentukan kebutuhan sesuai dengan sistem yang akan dikembangkan. Berikut beberapa penggunaan perangkat keras serta perangkat lunak yang dibutuhkan diantaranya :

1. Delivery Consideration

Pada tahapan ini penulis mempertimbangkan sistem seperti apa yang akan di tampilkan atau di kirim kepada *user*. Sebagai contoh penulis menggunakan media web browser sebagai media pengirim informasi kepada *user* atau pengguna.

2. Hardware and Software Consideration

Spesifikasi penggunaan *hardware* dan *software* yang akan digunakan dalam menampilkan informasi terutama *device* yang digunakan oleh pengguna / *user* adalah diantaranya :

a. Hardware :

Laptop / PC : GPU NVIDIA GeForce 2xx atau lebih
/ Intel HD 4000 atau lebih

b. *Software Android:*

- a) Safari iOS didukung penuh pada 15 – 16, dan tidak didukung pada 3.2 – 14 Safari Ios
- b) Peramban Android didukung penuh pada 97 – 103, dan tidak didukung pada 2.3 – 4 Android
- c) Opera Mobile didukung penuh pada 64 – 64, dan tidak di dukung pada 10 – 12 Opera Mobile
- d) Chrome Mobile didukung penuh pada 97 – 103, dan tidak didukung pada 97 kebawah
- e) Firefox Mobile didukung penuh pada 95 – 101, dan tidak didukung dibawah 95

Adapun kebutuhan hardware dan software dalam penelitian , antara lain :

a. *Hardware*

Hardware atau perangkat keras yang dibutuhkan dalam pengembangan sistem antara lain :

- a) Laptop HP 14 *Notebook* PC Ram 6GB *Processor Intel® Core™ i3-4030U* CPU @ 1.90GHz (4 CPUs). Dengan menggunakan OS Windows 10
- b) *Graphic* NVIDIA GeForce 820M
- c) SSD 225 GB

b. *Software*

Software atau perangkat lunak yang di butuhkan dalam pembangunan sistem antara lain :

- a) Blender 3D v3.1.
- b) *Library Three.js*.
- c) *Text Editor* yang digunakan adalah *Visual Studio Code*.
- d) *Google Model Viewer Editor*.

3. *System Definition*

Pada tahapan ini untuk mendefnisikan sistem yang akan di gunakan untuk menampilkan peta 3 dimensi antara lain :

- a. Android
- b. iOS
- c. Windows
- d. Linux yang memiliki peramban web browser

4. *User Profile and Needs*

Langkah ini mengacu pada siapa yang membutuhkan informasi peta kampus SMP Karya Pembangunan 2 Majalaya, antara lain :

- a. Siswa
- b. Orang Tua / Wali
- c. Tamu

3.2.3. *Design Consideration*







Setelah melakukan analisis kebutuhan sistem, tahapan selanjutnya melakukan perancangan aplikasi sesuai dengan kebutuhan. Dalam melakukan perancangan desain penulis melibatkan beberapa rancangan kebutuhan diantaranya :

1. *Design Metaphor*

Metafora desain atau *Design Metaphor* digunakan dalam pembuatan aplikasi berbasis 3D supaya perancang aplikasi dapat menyimpulkan atau menyarankan rancangan dan desain yang akan di gunakan, berikut beberapa desain metafora yang dipilih dalam merancang aplikasi, antara lain :

a. *storyboard*

Storyboard digunakan untuk melihat alur tampilan sederhana dari frame ke frame yang lain, terutama dalam pemindahan camera pada scene, berikut adalah contoh gambar sederhana storyboard yang akan menampilkan peta 3 dimensi interaktif :

 <p>Selamat Datang</p>	<p>KETERANGAN PENGGUNA MELAKUKAN SCROLL UNTUK MELIHAT APA YANG AKAN DI TAMPILKAN</p>
	<p>SETELAH SCROLL MAKA AKAN TAMPIL PETA SEKOLAH DENGAN RASIO YANG JAUH</p>
	<p>SCROLL TERUS KEBAWAH SEHINGGA PETA MENDEKAT DAN INFORMASI AKAN MUNCUL</p>
	<p>MUNCUL INFORMASI TERKAIT GEDUNG DAN BANGUNAN</p>
	<p>MUNCUL INFORMASI TERKAIT GEDUNG DAN BANGUNAN</p>
	<p>MUNCUL INFORMASI TERKAIT GEDUNG DAN BANGUNAN</p>

Gambar 3.2 *Storyboard* Kerangka awal tampilan Peta
Interaktif

b. Teks

Teks berperan aktif dalam menampilkan informasi terkait keterangan yang akan di tampilkan pada peta

c. Gambar

Gambar merupakan komponen multimedia yang dapat menyampaikan informasi yang lebih menarik. Gambar sangat berguna untuk menyampaikan informasi yang tidak dapat dijelaskan dengan kata-kata.

2. *Information Types*

Dalam menyampaikan informasi kepada penerima, sistem yang akan dirancang dilengkapi dengan beberapa media, diantaranya :

a. Model 3D

Model 3 dimensi digunakan sebagai peran utama dalam pembuatan sistem, file yang digunakan dalam membuat model 3d adalah .gltf, terdapat 1.243 buah objek 3 dimensi dalam pembuatan model pada blender.

b. Teks

Teks berperan aktif dalam menampilkan informasi terkait keterangan yang akan ditampilkan pada peta.

c. Gambar

Gambar merupakan komponen multimedia yang dapat menyampaikan informasi yang lebih menarik. Gambar sangat berguna untuk menyampaikan informasi yang tidak dapat dijelaskan dengan kata-kata. Gambar yang dihasilkan komputer dibagi menjadi dua, diantaranya :

a) Animasi

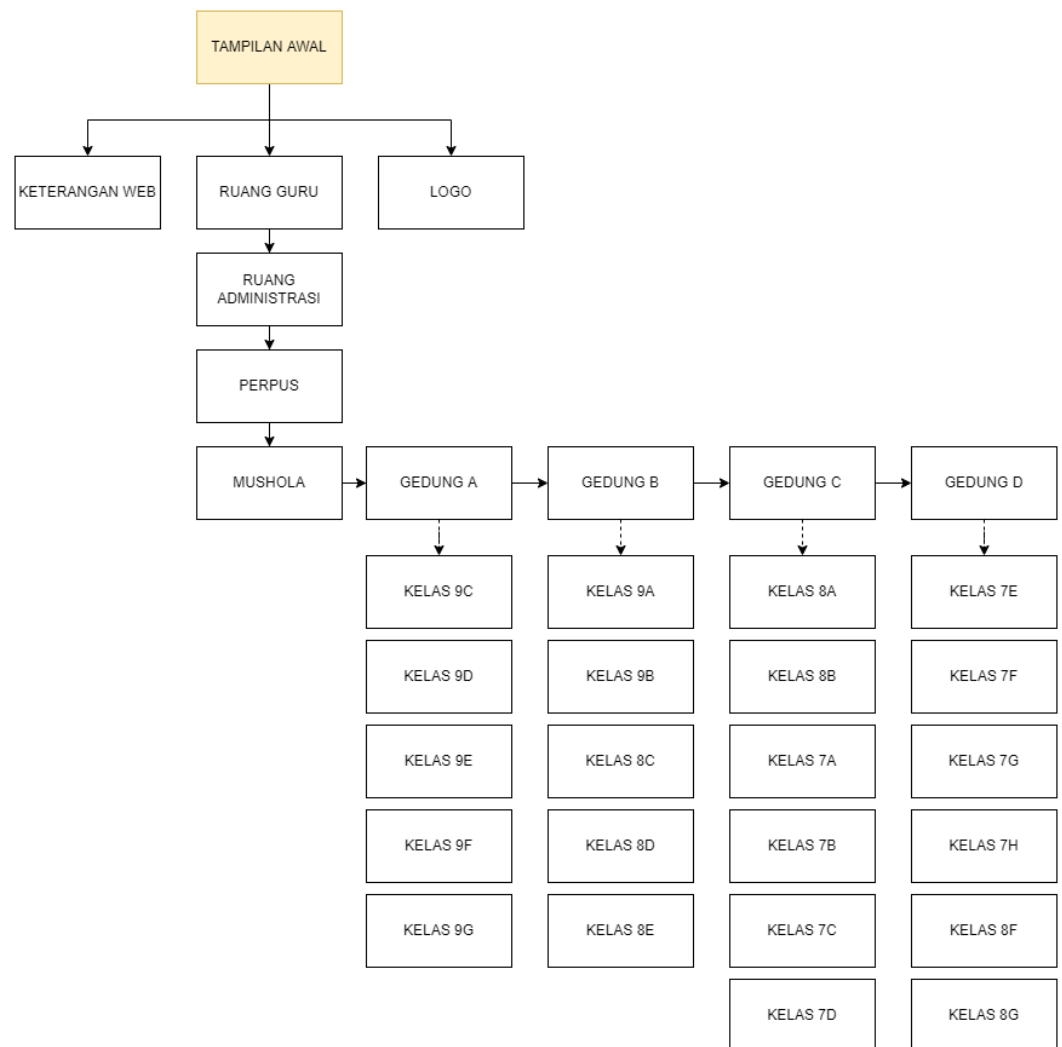
Animasi digunakan dalam menyampaikan informasi interaktif yang sangat membantu dalam menyampaikan informasi.

b) Video

Video digunakan dalam menyampaikan informasi interaktif karena kejelasan struktur penyampaian yang informatif menjadi media interaktif.

3. *Navigational Structures*

Perancangan struktur navigasi pada peta 3 Dimensi Kampus SMP Karya Pembangunan 2 Majalaya adalah sebagai berikut :



Gambar 3.3 *Navigational Structure* pada perancangan sistem peta 3D

4. *Media Preparation and Integration Issue*

Dalam melakukan persiapan dan integrasi media, penulis mendapatkan hal yang perlu di siapkan diantaranya menyediakan peralatan serta informasi yang akan disajikan.

3.2.4. Penerapan / Implementasi

Dalam tahapan ini penulis membuat kode program dengan menggunakan bahasa pemrograman HTML, *JavaScript*, dan CSS. Menggunakan teks editor Visual Studio Code. Serta melakukan uji prototype untuk mendapatkan dan mencari masalah atau *error* yang mungkin terjadi. Serta pada tahapan ini penulis melakukan beberapa pengujian beta version.

3.2.5. Evaluasi

Pada tahapan ini program Aplikasi Peta 3D SMP Karya Pembangunan 2 Majalaya Berbasis Web diuji untuk mendapatkan evaluasi pasca publish. Jika terjadi beberapa kemungkinan seperti perubahan desain karena keragaman pendapat dari berbagai isi dari koisioner atau kenyamanan pengguna.

3.2.6. Penyelesaian Laporan Penelitian

Setelah tahap pengembangan sistem aplikasi diselesaikan maka tahap selanjutnya melakukan penyelesaian laporan penelitian dengan melakukan bimbingan terhadap revisi yang dihasilkan dari diadakannya sidang UPS1 dan UPS2 sehingga menghasilkan laporan penelitian yang sesuai dengan kriteria dan standar dari pihak Program Studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Bale Bandung.

BAB IV

ANALISIS DAN PERANCANGAN

4.1. ANALISIS

Dalam melakukan analisis kebutuhan yang akan digunakan pada pembuatan aplikasi peta 3 dimensi ini ada beberapa langkah untuk mengidentifikasi kebutuhan serta permasalahan yang akan di hadapi, berikut diantaranya :

4.1.1. Analisis Masalah

Berdasarkan identifikasi masalah penelitian yang di ketahui yaitu belum adanya informasi letak gedung dan ruang yang ada di SMP Karya Pembangunan 2 Majalaya, sehingga terjadi kekeliruan dalam memasuki ruangan. Sebagian besar masalah yang dihadapi diantaranya adalah siswa baru, orang tua siswa, serta tamu. Maka dari itu dibuatnya sebuah aplikasi ini supaya bisa menjadi solusi dalam menampilkan informasi yang bisa di akses dimana saja.

4.1.2. Analisis Software

Pada penelitian yang dilakukan terdapat beberapa *software* yang digunakan sebagai alat penunjang pembuatan aplikasi. Adapun *software* yang dibutuhkan sebagai berikut :

Tabel 4.1 Analisis *Software*

Sistem Operasi	Windows 10 <i>Home Single Language</i> 64-bit (10.0, <i>Build</i> 19044)
Bahasa Pemrograman	Java Script
Aplikasi Penunjang	a) <i>Visual Studio Code</i> b) Blender 3D v3.2.2 c) Peramban Web (Google Chrome, Firefox, Opera, dll)

4.1.3. Analisis Pengguna

Dalam menentukan pengguna penulis mengajukan beberapa kriteria pengguna yang disarankan harus melihat informasi tata letak SMP Karya Pembangunan 2 Majalaya sebelum memasuki area kampus diantaranya :

1. Siswa Baru
2. Orang tua siswa
3. Tamu yang hendak memasuki kawasan SMP Karya Pembangunan 2 Majalaya.

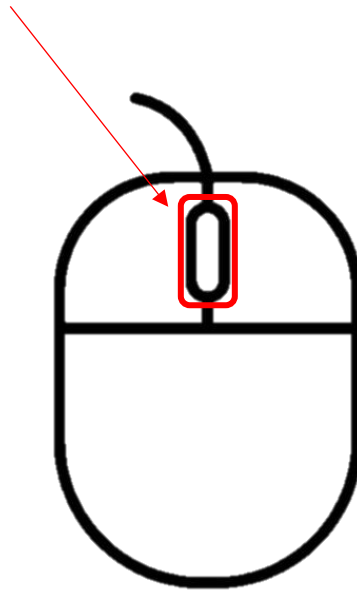
4.1.4. User Interface

User interface sangat berpengaruh terhadap minat siswa dalam menggunakannya. Siswa akan dimudahkan dalam penggunaan aplikasi serta konten yang efisien, tidak terlalu banyak tombol maupun navigasi yang menyulitkan para siswa, orang tua maupun tamu.

Banyak tombol yang membuat rasa takut serta waspada siswa terhadap sistem error maupun kerusakan perangkat keras membuat para siswa enggan menggunakan komputer maupun laptop, sehingga penulis membuat sebuah sistem informasi yang mudah digunakan serta interaktif.

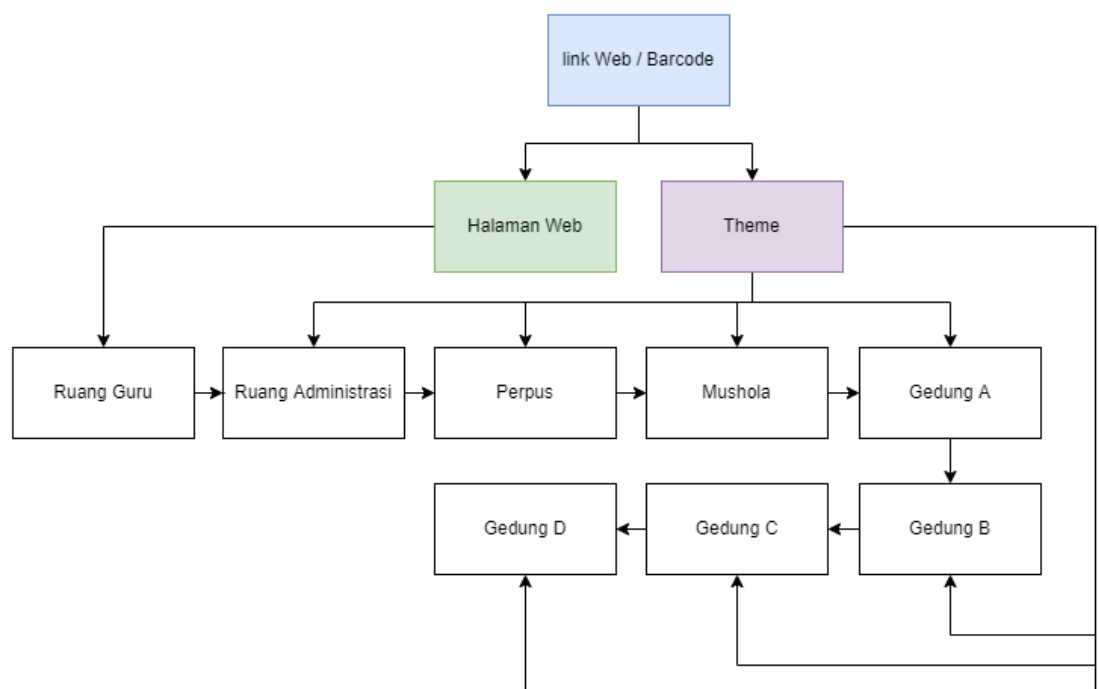
Penggunaan scroll mouse sebagai media navigasi yang mudah serta tidak menyulitkan para siswa terutama kepada siswa yang masih awam terhadap penggunaan komputer maupun laptop.

Adapun siswa yang telah terbiasa menggunakan android sebagai pijakan mencari ilmu aplikasi peta tiga dimensi bisa diakses menggunakan *android device* yang telah dilengkapi dengan aplikasi chrome sebagai media untuk membuka aplikasi.



Gambar 4.1 Penggunaan *Scroll* dalam navigasi sistem

Navigasi penggunaan *scroll mouse* sesuai yang telah di tentukan pada bab 3 yaitu metodologi antara lain sebagai berikut :



Gambar 4.2 Tabel User Interface

Sesuai dengan bagan di atas siswa, orang tua, tamu bisa mengakses halaman web termasuk semua fitur yang bersangkutan

tanpa memerlukan login maupun validasi data, theme menerangkan aplikasi bisa merubah tampilan web sesuai dengan kebutuhan dan keinginan.

4.1.5. Fitur – Fitur

Dalam aplikasi yang akan dibuat terdapat beberapa fitur yang dapat dijalankan. fitur yang dilengkapi pada aplikasi ini adalah :

1. Smoothing Scrolling

Pengguliran halus adalah fitur yang diterapkan di sebagian besar browser untuk memungkinkan pengguliran yang lebih baik pada halaman web dengan berbagai ukuran.

2. Dark Mode Button

Dark Mode atau biasa di sebut sebagai mode hitam, gelap, malam, desain web mode gelap adalah istilah yang digunakan untuk menggambarkan tata letak apapun dengan skema warna terang ke gelap yang lebih disukai banyak pengguna digunakan di lingkungan kurang cahaya. Sehingga bisa menjadi daya tarik tersendiri kepada para siswa maupun orang tua dan tamu.

4.1.6. Analisis Data

dalam perancangan sistem informasi ini dibutuhkan data kuesioner siswa serta wawancara yang di langsungkan kepada orang tua / wali, serta guru. Data ini ditentukan setelah melakukan riset dari jurnal yang sudah di cantumkan.

4.2. PERANCANGAN

Pada tahapan ini penulis menggunakan metode IMSDD yang memiliki 4 tahap diantaranya : 1) *System Requirement*, 2) *Consideration Design*, 3) *Implementation*, 3) *Evaluation*

4.2.1. System Requirement

Pada tahap ini, pemberian definisi seperti membuat tujuan dan sasaran dari sistem yang sedang dibangun. Pada pengembangan peta tiga dimensi SMP Karya Pembangunan 2 Majalaya ini, penulis merencanakan kebutuhan sistem diantaranya:

a. *System Definition*

Definisi sistem yang akan dibangun adalah sebuah media peta tiga dimensi interaktif SMP Karya Pembangunan 2 Majalaya untuk calon siswa/I, orang tua, dan tamu dengan menggunakan library three.js dalam pengembangannya dan tahapan evaluasi menggunakan *eight golden rules of interface design*. Peta tiga dimensi ini dapat diakses secara online multi-platform, akan tetapi penulis memfokuskan penggunaan *desktop-based* yaitu menggunakan perangkat komputer atau laptop. Peta tiga dimensi ini berisi lokasi bangunan atau gedung dan lingkukan kampus SMP Karya Pembangunan 2 Majalaya.

b. *Hardware and Software Consideration*

Kebutuhan *hardware*, *software* dan *tools* merupakan kebutuhan perangkat keras dan perangkat lunak dalam membangun aplikasi ini. berikut diantaranya:

a) *Hardware*

- Laptop HP Notebook PC Ram 6GB Processor Intel® Core™ i3-4030U CPU @ 1.90GHz (4 CPUs). Dengan menggunakan OS Windows 10
- Graphic NVIDIA GeForce 820M
- SSD 225GB
- Mouse

b) *Software and Tools*

- Google Chrome for Windows Version 104.0.5112.102 (Official Build) (64-bit)
- Blender 3D v.3.2.2
- Balsamiq Mockup Version: 4.5.5 Editor Version: 5.3.1
- Microsoft Office Word 2019
- Microsoft Office Visio 2016
- <https://app.diagrams.net/> Apps Flow Diagram Online.
- Library Three.js
- Text Editor Visual Studio Code

- *Google Model Viewer Editor Online Version*
- Node JS
- Vercel App
- Vite JS

c. User Profile and Needs

Target pengguna pada peta 3 dimensi ini ialah siswa/i, orang tua atau wali, serta masyarakat umum dan tamu yang berkunjung ke kampus SMP Karya Pembangunan 2 Majalaya. untuk dapat mengakses peta tiga dimensi ini, pengguna bisa membukanya melalui *barcode* dan *link* yang telah terpasang di majalah dinding. Menggunakan platform android, iOS, Linux yang memiliki peramban web maupun komputer / PC dan laptop.

d. Delivery Consideration

Yang dibutuhkan untuk mengakses peta tiga dimensi ini bisa di semua platform yang memiliki peramban web untuk mengaksesnya. Dalam penelitian ini fokus pengembangannya pada perangkat desktop atau PC/Laptop, dalam pengembangan ini hanya dibuat untuk dimensi layar yang lebih besar seperti komputer atau laptop.

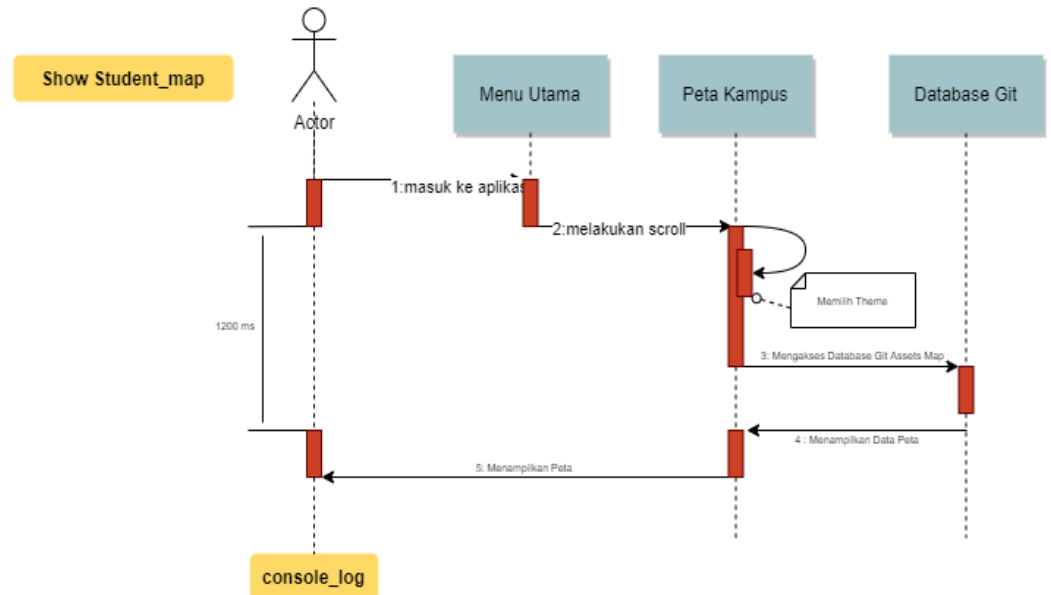
4.2.2. Consideration Design

Pada tahap ini menjelaskan secara terperinci pedoman-pedoman desain, tahapannya diantara lain sebagai berikut:

a. Design Metaphor

Design Metaphor yang digunakan penulis dalam peta tiga dimensi ini adalah smart touring, dimana pengguna memiliki kebebasan untuk mengakses dan melakukan perjalanan di setiap scrolling pada device, memberikan informasi serta animasi yang akan di tampilkan pada setiap scrolling. Terdapat beberapa tahapan dalam pembuatan desain metaphor berikut diantaranya:

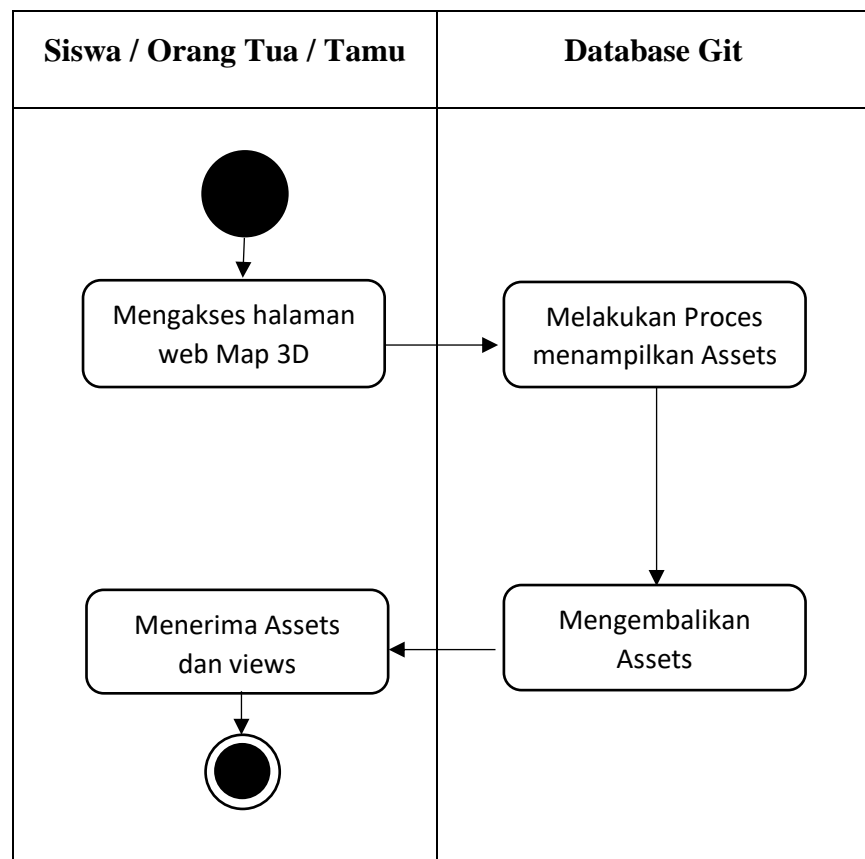
a) Sequence Diagram



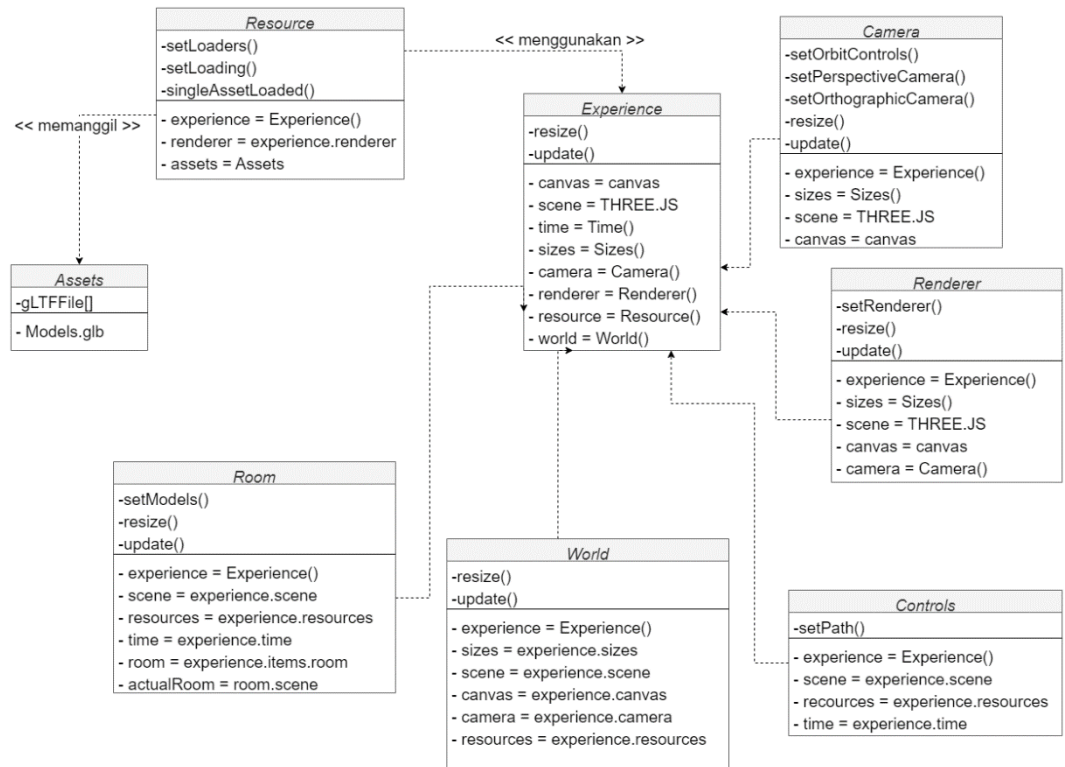
Gambar 4.3 Sequence Diagram

b) Activity Diagram

Tabel 4.2 Tabel Activity Diagram

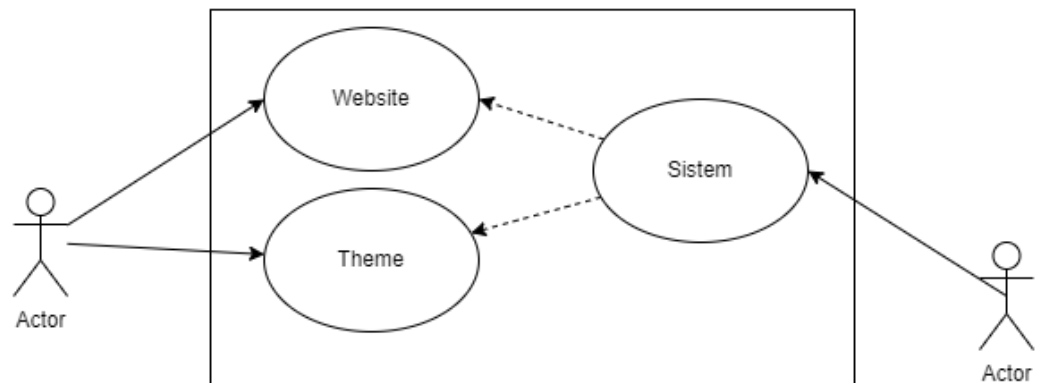


c) Diagram Class



Gambar 4.4 Class Diagram

d) Use Case Diagram



Gambar 4.5 Use Case Diagram

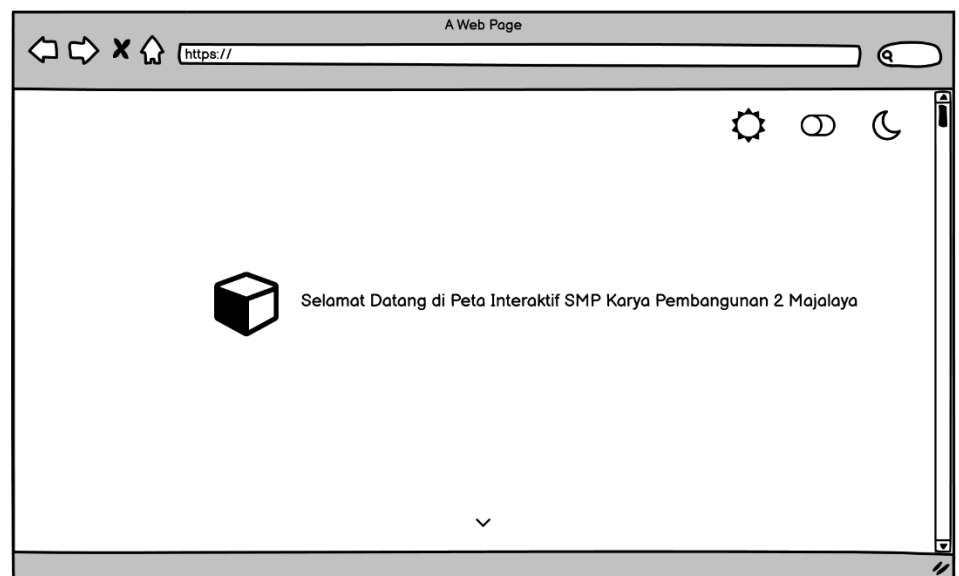
Use Case Diagram berperan penting dalam melihat alur penggunaan sebuah sistem, terdapat 2 aktor yang saling terkait dalam mengakses sistem, tetapi hanya seorang admin yang mampu mengakses penuh sistem dalam pemeliharaannya.

e) Storyboard

Sebelum melakukan implementasi logic pembuatan *code* menggunakan *javascript* pada peta tiga dimensi, penulis terlebih dahulu menggambarkan antarmuka yang akan dibuat untuk mempermudah penerangan awal aplikasi yang akan dibangun supaya dapat memberikan gambaran terhadap aplikasinya nanti. Tujuannya adalah sebagai acuan dalam pembuatan aplikasi.

a. Scene 1

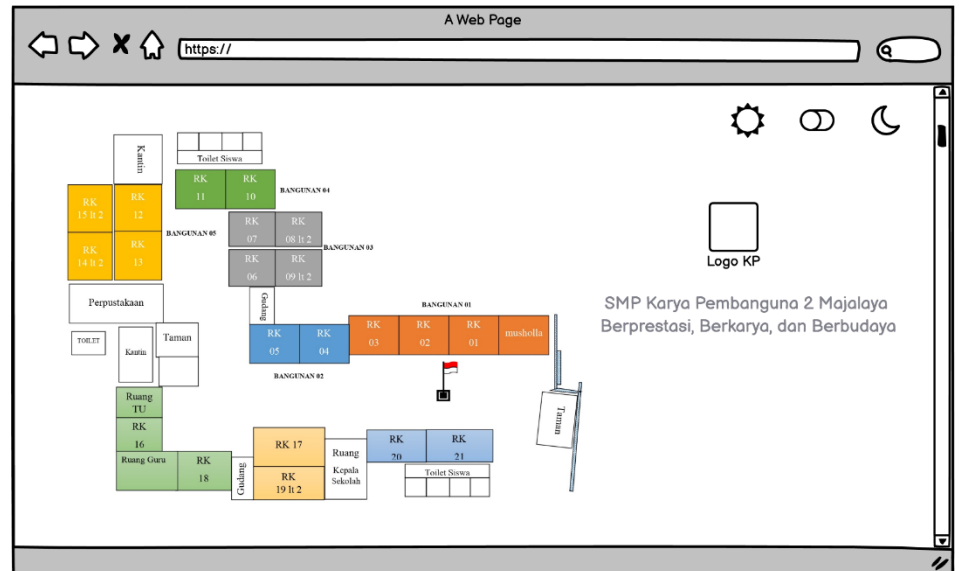
Pada *scene 1* ini, halaman utama akan muncul. Pada halaman ini terdapat tampilan “Selamat Datang di Peta Interaktif Kampus SMP Karya Pembangunan 2 Majalaya” serta terdapat sebuah box yang akan memulai perjalanan peta 3D.



Gambar 4.6 Tampilan awal Peta 3D Interaktif SMP Karya
Pembangunan 2 Majalaya

b. Scene 2

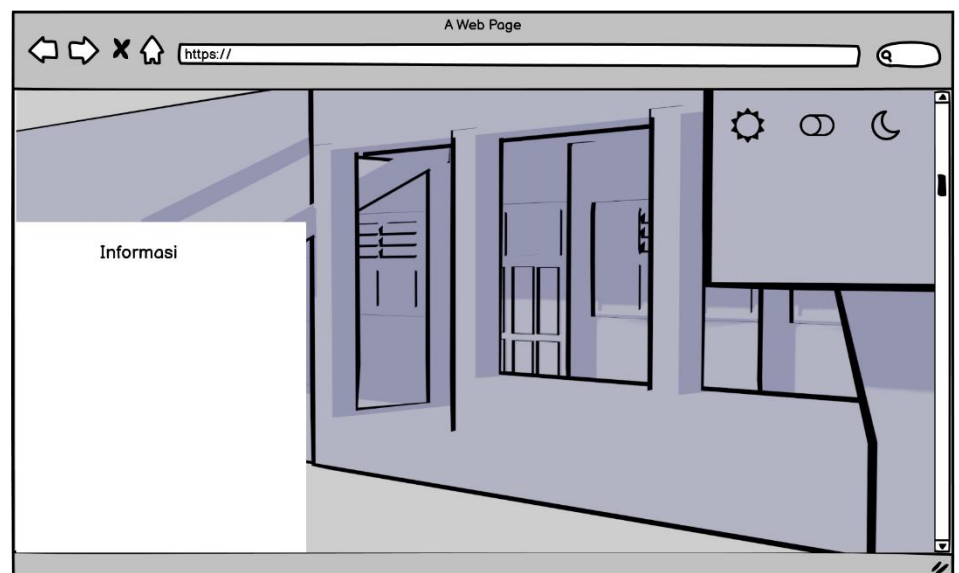
Pada *scene 2* halaman kedua yaitu memperlihatkan peta 3D yang di *shoot* dari atas, *camera location* ($x = 0$; $y = 0$; $z = 5$), *rotation* ($x = 0^\circ$, $y = 0^\circ$, $z = 0^\circ$).



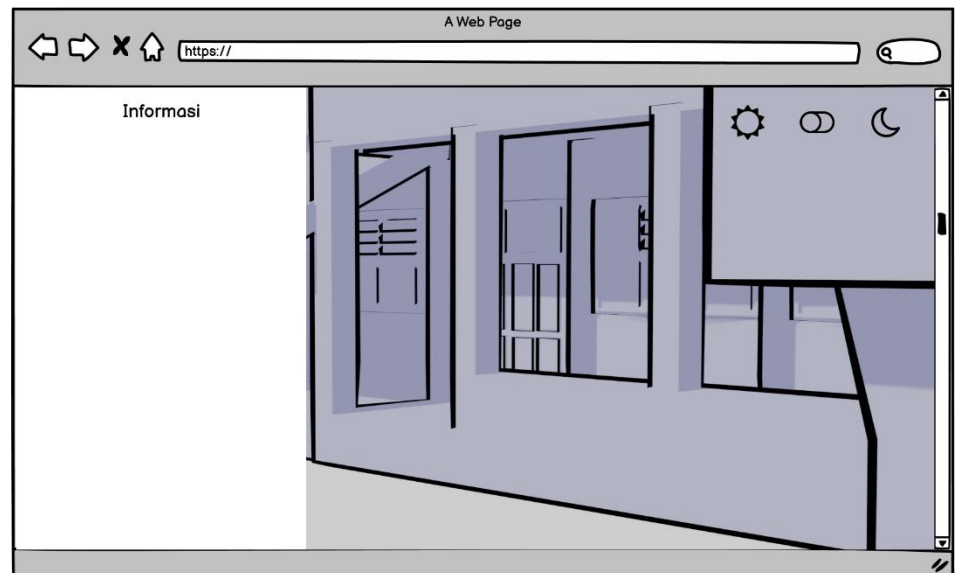
Gambar 4.7 tampilan pertama peta setelah *scroll* dilakukan

c. *Scene 3*

Scene 3 menampilkan gedung pertama yaitu Ruang Guru sebagai awal *trouring*, serta mulai menampilkan informasi seputar ruang guru yang terdapat pada layout sebelah kiri



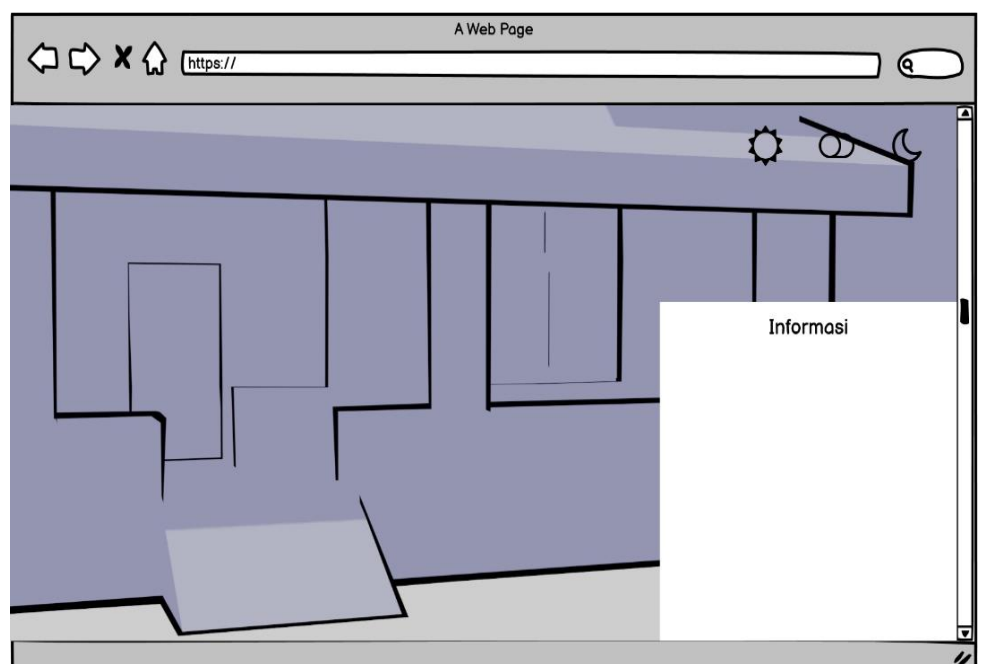
Gambar 4.8 *scene 3* tampilan awal informasi



Gambar 4.9 tampilan *scene 3 scrollldown*

d. *Scene 4*

Pada *scene 5* arah kamera di rubah kembali pada titik yang di tentukan sesuai dengan *curve*. Perubahan ini mengarah pada gedung tata usaha atau biasa di sebut sebagai staff administrasi



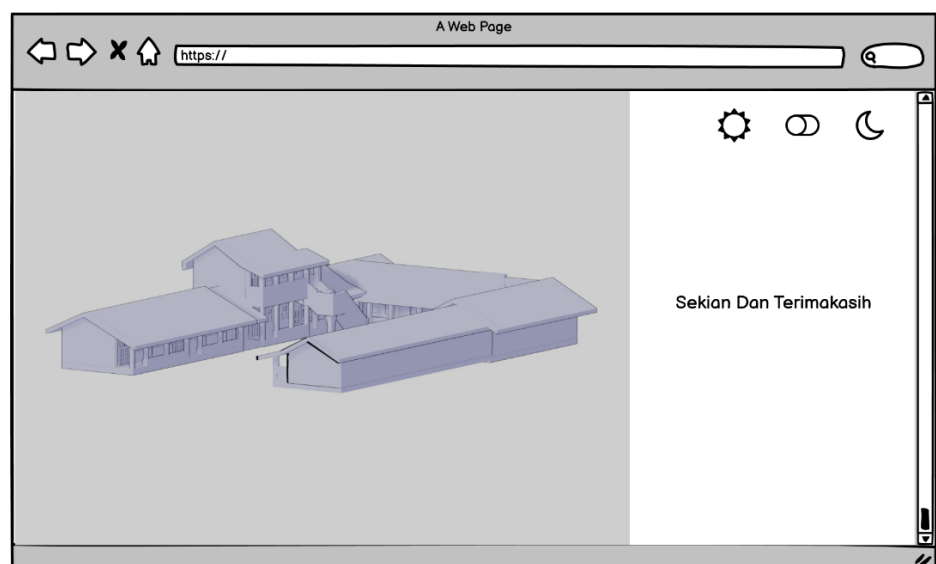
Gambar 4.10 tampilan *scene 4 scrollldown*



Gambar 4.11 tampilan *scene 4 scrolldown*

Kegiatan *tour* akan terus berlangsung seiring dengan *scroll-down* dan *scroll-up* pengguna.

e. *Scene 5*



Gambar 4.12 tampilan akhir perjalanan *tour* Peta 3D SMP Karya
Pembangunan 2 Majalaya

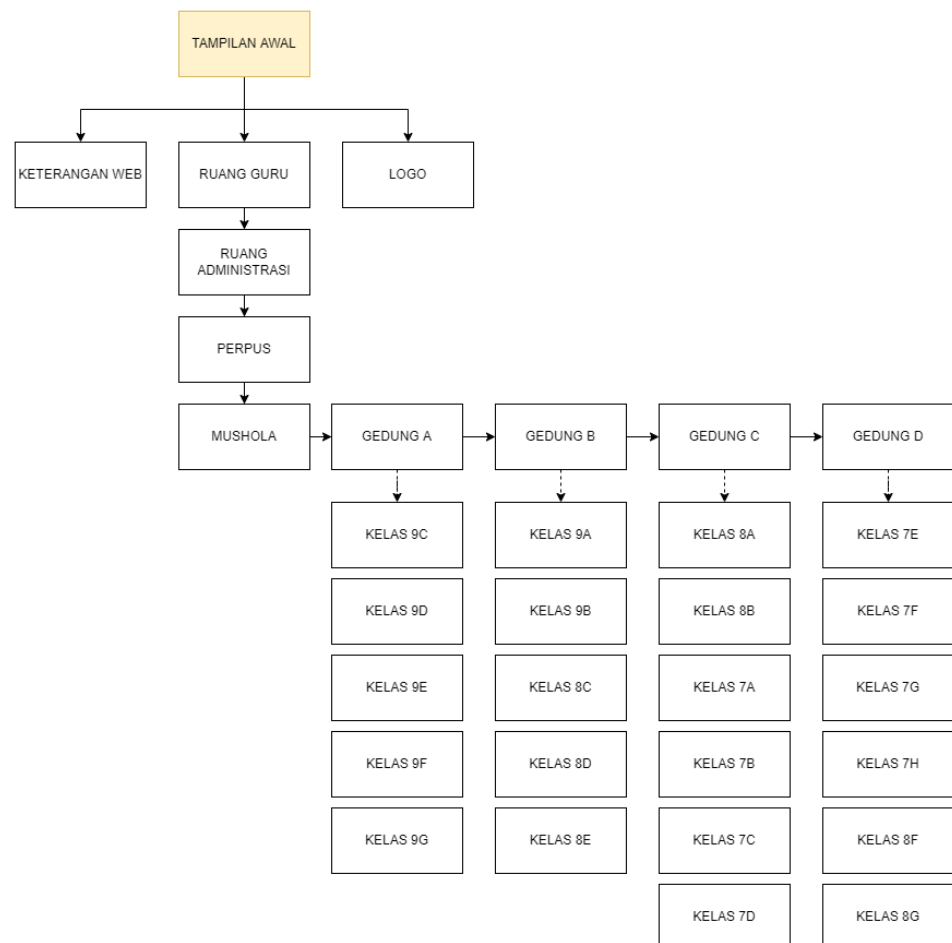
b. Information Types

Berikut ini adalah rincian lengkap dari aset informasi yang digunakan penulis dalam pengembangan peta tiga dimensi berdasarkan bentuk format dan tipe informasi:

- a) Teks: terdapat informasi yang harus ditampilkan dalam scrool touring seperti keterangan gedung, dan nama nama ruangan.
- b) Model 3D: model tiga dimensi yang digunakan dalam sistem ini ada 8 objek yang masing masing memiliki format gLTF.

c. Navigational Structures

Perancangan struktur navigasi untuk peta 3D kampus SMP Karya Pembangunan 2 Majalaya ialah sebagai berikut:



Gambar 4.12 *Navigational Structure* pada perancangan sistem peta 3D

Gambar 4.3 merupakan rancangan struktur navigasi alur perjalanan pada aplikasi peta 3D Kampus SMP Karya Pembangunan 2 Majalaya yang akan dibuat. Pada saat pertama kali membuka aplikasi peta tiga dimensi, terdapat halaman pertama yaitu peta awal dari sudut *camera location* ($x = 0$; $y = 0$; $z = 5$), *rotation* ($x = 0^\circ$, $y = 0^\circ$, $z = 0^\circ$), pada tampilan pertama ini penggunaan *mouse wheel (scroll)* akan mempengaruhi *position* serta *rotation* pada *camera*.

d. Media Preparation and Integration Issue

Kontrol yang digunakan berupa sebuah mouse yang dilengkapi dengan scroll maupun kontrol sentuhan pada layar sentuh. Scroll ini bisa langsung di gunakan untuk melakukan perjalanan touring.

BAB V

IMPLEMENTASI DAN PENGUJIAN

5.1. IMPLEMENTASI

5.1.1. Listing Program

1. *Utility Programs Folder*

- a. *Assets.js assets* memiliki peran penting dalam menampilkan *resources*

```
export default [  
  {  
    name: "room",  
    type: "glbModel",  
    path: "/models/Models1.glb",  
  },  
  {  
    name: "windows",  
    type: "videoTexture",  
    path: "/textures/hihi.mp4",  
  },  
];
```

b. *Resources.js*

```
import * as THREE from "three";  
import { EventEmitter } from "events";  
import { GLTFLoader } from  
"three/examples/jsm/loaders/GLTFLoader.js";
```



```

import { DRACOLoader } from
"three/examples/jsm/loaders/DRACOLoader.js";
import Experience from "../Experience.js";

export default class Resources extends EventEmitter {
  constructor(Assets) {
    super();
    this.experience = new Experience();
    this.renderer = this.experience.renderer;

    this.assets = Assets;

    this.items = {};
    this.queue = this.assets.length;
    this.loaded = 0;

    this.setLoaders();
    this.startLoading();
  }

  setLoaders() {
    this.loaders = {};
    this.loaders.gltfLoader = new GLTFLoader();
    this.loaders.dracoLoader = new DRACOLoader();
    this.loaders.dracoLoader.setDecoderPath("/draco/");
    this.loaders.gltfLoader.setDRACOLoader(this.loaders.dracoLoa
der);
  }

  startLoading() {
    for (const asset of this.assets) {
      if (asset.type === "glbModel") {

```

```

        this.loaders.gltfLoader.load(asset.path, (file) => {
            this.singleAssetLoaded(asset, file);
        });
    } else if (asset.type === "videoTexture") {
        this.video = {};
        this.videoTexture = {};
        this.video[asset.name] = document.createElement("video");
        this.video[asset.name].src = asset.path;
        this.video[asset.name].muted = true;
        this.video[asset.name].playInline = true;
        this.video[asset.name].autoplay = true;
        this.video[asset.name].loop = true;
        this.video[asset.name].play();

        this.videoTexture[asset.name] = new THREE.VideoTexture(
            this.video[asset.name]
        );
        this.videoTexture[asset.name].flipY = true;
        this.videoTexture[asset.name].minFilter =
THREE.NearestFilter;
        this.videoTexture[asset.name].magFilter =
THREE.NearestFilter;
        this.videoTexture[asset.name].generateMipmaps = false;
        this.videoTexture[asset.name].encoding =
THREE.sRGBEncoding;

        this.singleAssetLoaded(asset, this.videoTexture[asset.name]);
    }
}
}

singleAssetLoaded(asset, file) {
    this.items[asset.name] = file;
}

```

```

this.loaded++;

console.log("Asset is Loading");
if (this.loaded === this.queue) {
  console.log("All Asset well done");
  this.emit("ready");
}
}
}

```

- c. *Sizes.js* file ini berperan dalam mengatur ukuran layar secara otomatis, sehingga jika layar desktop mengecil tampilan *canvas* akan menyesuaikan layar

```

import { EventEmitter } from "events";

export default class Sizes extends EventEmitter {
  constructor() {
    super();
    this.width = window.innerWidth;
    this.height = window.innerHeight;
    this.aspect = this.width / this.height;
    this.pixelRatio = Math.min(window.devicePixelRatio, 2);
    this.frustrum = 5;
    if (this.width < 968) {
      this.device = "mobile";
    } else {
      this.device = "desktop";
    }
  }
}

```

```

window.addEventListener("resize", () => {
  this.width = window.innerWidth;
  this.height = window.innerHeight;
  this.aspect = this.width / this.height;
  this.pixelRatio = Math.min(window.devicePixelRatio, 2);
  this.emit("resize");

  if (this.width < 968 && this.device !== "mobile") {
    this.device = "mobile";
    this.emit("switchdevice", this.device);
  } else if (this.width >= 968 && this.device !== "desktop") {
    this.device = "desktop";
    this.emit("switchdevice", this.device);
  }
});
}
}

```

d. Time.js

```

import { EventEmitter } from "events";

export default class Time extends EventEmitter {
  constructor() {
    super();
    this.start = Date.now();
    this.current = this.start;
    this.elapsed = 0;
    this.delta = 16;

    this.update();
  }
}

```

```

update() {
  const currentTime = Date.now();
  this.delta = currentTime - this.current;
  this.current = currentTime;
  this.elapsed = this.current - this.start;

  this.emit("update");
  window.requestAnimationFrame(() => this.update());
}
}

```

e. ConvertDisvToSpan.js

```

export default function (element) {
  element.style.overflow = "hidden";
  element.innerHTML = element.innerText
    .split("")
    .map((char) => {
      if (char === " ") {
        return `<span>${char}</span>`;
      }
      return `<span class="animatedis">${char}</span>`;
    })
    .join("");
  return element;
}

```

2. World Programs Folder

a. World.js

```

import * as THREE from "three";
import Experience from "../Experience.js";

```

```

import Room from "./Room.js";
import Controls from "./Controls.js";
import Environment from "./Environment.js";
import Floor from "./Floor.js";
import { EventEmitter } from "events";

export default class World extends EventEmitter {
  constructor() {
    super();
    this.experience = new Experience();
    this.sizes = this.experience.sizes;
    this.scene = this.experience.scene;
    this.canvas = this.experience.canvas;
    this.camera = this.experience.camera;
    this.resources = this.experience.resources;
    this.theme = this.experience.theme;

    this.resources.on("ready", () => {
      this.environment = new Environment();
      this.floor = new Floor();
      this.room = new Room();
      this.emit("worldready");
      this.controls = new Controls();
    });
    this.theme.on("switch", (theme) => {
      this.switchTheme(theme);
    });
  }

  switchTheme(theme) {
    if (this.environment) {
      this.environment.switchTheme(theme);
    }
  }
}

```

```

    }
  }

  resize() {}

  update() {
    if (this.room) {
      this.room.update();
    }
    if (this.controls) {
      this.controls.update();
    }
  }
}

```

b. Room.js

```

import * as THREE from "three";
import Experience from "../Experience.js";
import GSAP from "gsap";
import { RectAreaLightHelper } from
"three/examples/jsm/helpers/RectAreaLightHelper.js";

export default class Room {
  constructor() {
    this.experience = new Experience();
    this.scene = this.experience.scene;
    this.resources = this.experience.resources;
    this.time = this.experience.time;
    this.room = this.resources.items.room;
    this.actualRoom = this.room.scene;
    this.roomChildren = {};
  }
}

```

```

this.lerp = {
  current: 0,
  target: 0,
  ease: 0.1,
};

```

```

this.setModel();
this.onMouseMove();
// this.setAnimation();
}

```

```

setModel() {
  this.actualRoom.children.forEach((child) => {
    child.castShadow = true;
    child.receiveShadow = true;

    if (child instanceof THREE.Group) {
      child.children.forEach((groupchild) => {
        groupchild.castShadow = true;
        groupchild.receiveShadow = true;
      });
    }

    if (child.name === "Glass1") {
      child.material = new THREE.MeshPhysicalMaterial();
      child.material.roughness = 0;
      child.material.color.set(0x279fdd);
      child.material.ior = 3;
      child.material.transmission = 1;
      child.material.opacity = 1;
    }
  });
}

```



```

if (child.name === "Cube046") {
  child.material = new THREE.MeshBasicMaterial({
    map: this.resources.items.windows,
  });
}

child.scale.set(0, 0, 0);
if (child.name === "Box") {
  // child.scale.set(1, 1, 1);
  child.position.set(0, 0, 0);
}

this.roomChildren[child.name.toLowerCase()] = child;
});

const width = 2.5;
const height = 0.2;
const intensity = 1;
const rectLight = new THREE.RectAreaLight(
  0xffffff,
  intensity,
  width,
  height
);
rectLight.position.set(-10.6235, 0.3, 8.06713);
rectLight.rotation.x = -Math.PI / 2;
rectLight.rotation.z = Math.PI / 3.31;
this.actualRoom.add(rectLight);

this.roomChildren["reclight"] = rectLight;

```

```

// const rectLightHelper = new RectAreaLightHelper(rectLight);
// rectLight.add(rectLightHelper);

this.scene.add(this.actualRoom);
this.actualRoom.scale.set(0.2, 0.2, 0.2);
this.actualRoom.rotateY = 3;
}

// setAnimation() {
//   this.mixer = new THREE.AnimationMixer(this.actualRoom);
//   this.swin = this.mixer.clipAction(this.room.animations[0]);
//   this.swin.play();
// }

onMouseMove() {
  window.addEventListener("mousemove", (e) => {
    this.rotation =
      ((e.clientX - window.innerWidth / 2) * 2) /
window.innerWidth;
    this.lerp.target = this.rotation * 0.02;
  });
}

resize() {}

update() {
  this.lerp.current = GSAP.utils.interpolate(
    this.lerp.current,
    this.lerp.target,
    this.lerp.ease
  );
  this.actualRoom.rotation.y = this.lerp.current;
}

```

```
}

```

c. Environment.js

```
import * as THREE from "three";
import Experience from "../Experience.js";
import GSAP from "gsap";
import GUI from "lil-gui";

export default class Environment {
  constructor() {
    this.experience = new Experience();
    this.scene = this.experience.scene;
    this.resources = this.experience.resources;
    this.obj = {
      colorObj: { r: 0, g: 0, b: 0 },
      intensity: 3,
    };

    this.setSunlight();
  }

  setSunlight() {
    this.sunLight = new THREE.DirectionalLight("#ffffff", 3);
    this.sunLight.castShadow = true;
    this.sunLight.shadow.camera.far = 20;
    this.sunLight.shadow.mapSize.set(1024, 1024);
    this.sunLight.shadow.normalBias = 0.05;
    // const helper = new
    THREE.CameraHelper(this.sunLight.shadow.camera);
    // this.scene.add(helper);
    this.sunLight.position.set(-2.5, 3, 3);
  }
}
```

```
this.scene.add(this.sunLight);

this.ambientLight = new THREE.AmbientLight("#ffffff", 1);
this.scene.add(this.ambientLight);
}

switchTheme(theme) {
  if (theme === "dark") {
    console.log(this.sunLight);
    GSAP.to(this.sunLight.color, {
      b: 0.4666666666666667,
      g: 0.21568627450980393,
      r: 0.23137254901960785,
    });
    GSAP.to(this.ambientLight.color, {
      b: 0.4666666666666667,
      g: 0.21568627450980393,
      r: 0.23137254901960785,
    });
    GSAP.to(this.sunLight, {
      intensity: 0.78,
    });
    GSAP.to(this.ambientLight, {
      intensity: 0.78,
    });
  } else {
    GSAP.to(this.sunLight.color, {
      r: 255 / 255,
      g: 255 / 255,
      b: 255 / 255,
    });
    GSAP.to(this.ambientLight.color, {
```

```

        r: 255 / 255,
        g: 255 / 255,
        b: 255 / 255,
    });
    GSAP.to(this.sunLight, {
        intensity: 1,
    });
    GSAP.to(this.ambientLight, {
        intensity: 1,
    });
    }
}

resize() {}

update() {}
}

```

d. Floor.js

```

import * as THREE from "three";
import Experience from "../Experience.js";

export default class Floor {
    constructor() {
        this.experience = new Experience();
        this.scene = this.experience.scene;

        this.setFloor();
        this.setCircles();
    }
}

```

```
setFloor() {  
  this.geometry = new THREE.PlaneGeometry(100, 100);  
  this.material = new THREE.MeshStandardMaterial({  
    color: 0xffffff,  
    side: THREE.BackSide,  
  });  
  this.plane = new THREE.Mesh(this.geometry, this.material);  
  this.plane.rotation.x = Math.PI / 2;  
  this.plane.position.y = -0.5;  
  this.plane.receiveShadow = true;  
  this.scene.add(this.plane);  
}  
  
setCircles() {  
  const geometry = new THREE.CircleGeometry(5, 64);  
  const material = new THREE.MeshStandardMaterial({ color:  
0x718cdd });  
  const material1 = new THREE.MeshStandardMaterial({ color:  
0xf37338 });  
  const material2 = new THREE.MeshStandardMaterial({ color:  
0x0cda73 });  
  this.circleFirst = new THREE.Mesh(geometry, material);  
  this.circleSecond = new THREE.Mesh(geometry, material1);  
  this.circleThird = new THREE.Mesh(geometry, material2);  
  
  this.circleFirst.position.y = -0.498;  
  this.circleSecond.position.y = -0.466;  
  this.circleThird.position.y = -0.3411;  
  
  this.circleFirst.scale.set(0, 0, 0);  
  this.circleSecond.scale.set(0, 0, 0);
```

```

this.circleThird.scale.set(0, 0, 0);

this.circleFirst.rotation.x =
  this.circleSecond.rotation.x =
  this.circleThird.rotation.x =
    -Math.PI / 2;
this.circleFirst.receiveShadow =
  this.circleSecond.receiveShadow =
  this.circleThird.receiveShadow =
    true;

this.scene.add(this.circleFirst);
this.scene.add(this.circleSecond);
this.scene.add(this.circleThird);
}

resize() {}

update() {}
}

```

e. Controls.js

```

import Experience from "../Experience.js";
import GSAP from "gsap";
import { ScrollTrigger } from "gsap/ScrollTrigger.js";
import ASScroll from "@ashthornton/asscroll";

export default class Controls {
  constructor() {
    this.experience = new Experience();
    this.scene = this.experience.scene;
  }
}

```

```

this.sizes = this.experience.sizes;
this.resources = this.experience.resources;
this.time = this.experience.time;
this.camera = this.experience.camera;
this.room = this.experience.world.room.actualRoom;
this.room.children.forEach((child) => {
  if (child.type === "RectAreaLight") {
    this.rectLight = child;
  }
});
this.circleFirst = this.experience.world.floor.circleFirst;
this.circleSecond = this.experience.world.floor.circleSecond;
this.circleThird = this.experience.world.floor.circleThird;
GSAP.registerPlugin(ScrollTrigger);

document.querySelector(".page").style.overflow = "visible";

this.setScrollTrigger();
}

setScrollTrigger() {
  // First Section -----
  this.firstMoveTimeline = new GSAP.timeline({
    scrollTrigger: {
      trigger: ".first-move",
      start: "top top",
      end: "bottom bottom",
      scrub: 0.6,
      invalidateOnRefresh: true,
    },
  });
  this.firstMoveTimeline.to(this.room.position, {

```



```

x: () => {
    return this.sizes.width * 0.003;
},
});

// Second Section -----
this.secondMoveTimeline = new GSAP.timeline({
    scrollTrigger: {
        trigger: ".second-move",
        start: "top top",
        end: "bottom bottom",
        scrub: 0.6,
        invalidateOnRefresh: true,
    },
})
.to(
    this.room.position,
    {
        x: () => {
            return -this.sizes.width * 0.002;
        },
        z: () => {
            return this.sizes.height * 0.002;
        },
    },
    "same"
)
.to(
    this.room.scale,
    {
        x: 0.3,
        y: 0.3,

```

```

        z: 0.3,
    },
    "same"
)
.to(
    this.rectLight,
    {
        width: 0.5 * 4,
        heigth: 0.2 * 4,
    },
    "same"
);

// Third Section -----
this.thirdMoveTimeline = new GSAP.timeline({
    scrollTrigger: {
        trigger: ".third-move",
        start: "top top",
        end: "bottom bottom",
        scrub: 0.6,
        invalidateOnRefresh: true,
    },
}).to(this.camera.orthographicCamera.position, {
    y: 0,
    x: -4.5,
});

this.sections = document.querySelectorAll(".section");
this.sections.forEach((section) => {
    if (section.classList.contains("right")) {
        GSAP.to(section, {
            borderTopLeftRadius: 10,

```

```

scrollTrigger: {
  trigger: section,
  start: "top bottom",
  end: "top top",
  scrub: 0.6,
},
});
GSAP.to(section, {
  borderBottomLeftRadius: 700,
  scrollTrigger: {
    trigger: section,
    start: "bottom bottom",
    end: "bottom top",
    scrub: 0.6,
  },
});
} else {
  GSAP.to(section, {
    borderTopRightRadius: 10,
    scrollTrigger: {
      trigger: section,
      start: "top bottom",
      end: "top top",
      scrub: 0.6,
    },
  });
  GSAP.to(section, {
    borderBottomRightRadius: 700,
    scrollTrigger: {
      trigger: section,
      start: "bottom bottom",
      end: "bottom top",

```

```

        scrub: 0.6,
    },
});

}

});

/* Circle Animations */
/*First Circle */
this.firstMoveTimeline = new GSAP.timeline({
    scrollTrigger: {
        trigger: ".first-move",
        start: "top top",
        end: "bottom bottom",
        scrub: 0.6,
        invalidateOnRefresh: true,
    },
}).to(this.circleFirst.scale, {
    x: 5,
    y: 5,
    z: 5,
});

// Second Circle -----
this.secondMoveTimeline = new GSAP.timeline({
    scrollTrigger: {
        trigger: ".second-move",
        start: "top top",
        end: "bottom bottom",
        scrub: 0.6,
        invalidateOnRefresh: true,
    },
});

```

```

.to(this.circleSecond.scale, {
  x: 5,
  y: 5,
  z: 5,
})
.to(
  this.room.position,
  {
    y: 0.3,
  },
  "same"
);

// Third Circle -----
this.thirdMoveTimeline = new GSAP.timeline({
  scrollTrigger: {
    trigger: ".third-move",
    start: "top top",
    end: "bottom bottom",
    scrub: 0.6,
    invalidateOnRefresh: true,
  },
}).to(this.circleThird.scale, {
  x: 5,
  y: 5,
  z: 5,
});
}

resize() {}

update() {}

```

```
}

```

3. Experience Programs Folder

a. Camera.js

```
import * as THREE from "three";
import Experience from "../Experience";
import { OrbitControls } from
"three/examples/jsm/controls/OrbitControls";

export default class Camera {
  constructor() {
    this.experience = new Experience();
    this.sizes = this.experience.sizes;
    this.scene = this.experience.scene;
    this.canvas = this.experience.canvas;

    this.createPerspectiveCamera();
    this.createOrthographicCamera();
    this.setOrbitControls();
  }

  createPerspectiveCamera() {
    this.perspectiveCamera = new THREE.PerspectiveCamera(
      35,
      this.sizes.aspect,
      0.1,
      1000
    );
    this.scene.add(this.perspectiveCamera);
  }
}
```

```

createOrthographicCamera() {
  this.orthographicCamera = new THREE.OrthographicCamera(
    (-this.sizes.aspect * this.sizes.frustum) / 2,
    (this.sizes.aspect * this.sizes.frustum) / 2,
    this.sizes.frustum / 2,
    -this.sizes.frustum / 2,
    -50,
    50
  );

  this.orthographicCamera.position.y = 0.45;
  this.orthographicCamera.position.z = 1.2913;
  this.orthographicCamera.position.x = 1;
  this.orthographicCamera.rotation.x = -Math.PI / 10;
  this.scene.add(this.orthographicCamera);

  // this.helper = new
  THREE.CameraHelper(this.orthographicCamera);
  // this.scene.add(this.helper);

  // const size = 20;
  // const division = 20;
  // const gridHelper = new THREE.GridHelper(size, division);
  // this.scene.add(gridHelper);

  // const axesHelper = new THREE.AxesHelper(10);
  // this.scene.add(axesHelper);
}

setOrbitControls() {

```

```

    this.controls = new OrbitControls(this.perspectiveCamera,
this.canvas);

    this.controls.enableDamping = true;
    this.controls.enableZoom = true;
}

resize() {
    // update Perspective Camera on Resize
    this.perspectiveCamera.aspect = this.sizes.aspect;
    this.perspectiveCamera.updateProjectionMatrix();

    // update Orthographic Camera on Resize
    this.orthographicCamera.left =
        (-this.sizes.aspect * this.sizes.frustrum) / 2;
    this.orthographicCamera.right =
        (this.sizes.aspect * this.sizes.frustrum) / 2;
    this.orthographicCamera.top = this.sizes.frustrum / 2;
    this.orthographicCamera.bottom = -this.sizes.frustrum / 2;
    this.orthographicCamera.updateProjectionMatrix();
}

update() {
    // console.log(this.perspectiveCamera.position);
    this.controls.update();
    // this.helper.matrixWorldNeedsUpdate = true;
    // this.helper.update();
    // this.helper.position.copy(this.orthographicCamera.position);
    // this.helper.position.copy(this.orthographicCamera.position);
}
}

```


b. Experience.js

```

import * as THREE from "three";
import Camera from "../Camera.js";
import Theme from "../Theme.js";
import Renderer from "../Renderer.js";
import Preloader from "../Preloader.js";
import Sizes from "../Utils/Sizes.js";
import Time from "../Utils/Time.js";
import Resources from "../Utils/Resources.js";
import Assets from "../Utils/Assets.js";
import World from "../World/World.js";
import Controls from "../World/Controls.js";

export default class Experience {
  static instance;
  constructor(canvas) {
    if (Experience.instance) {
      return Experience.instance;
    }
    Experience.instance = this;
    this.canvas = canvas;
    this.scene = new THREE.Scene();
    this.time = new Time();
    this.sizes = new Sizes();
    this.camera = new Camera();
    this.renderer = new Renderer();
    this.resources = new Resources(Assets);
    this.theme = new Theme();
    this.world = new World();
    this.preloader = new Preloader();
    this.preloader.on("enablecontrols", () => {
      this.controls = new Controls();
    });
  }
}

```

```

});

this.sizes.on("resize", () => {
  this.resize();
});

this.time.on("update", () => {
  this.update();
});
}

resize() {
  this.camera.resize();
  this.world.resize();
  this.renderer.resize();
}

update() {
  this.preloader.update();
  this.camera.update();
  this.world.update();
  this.renderer.update();
}
}

```

c. Renderer.js

```

import * as THREE from "three";
import Experience from "../Experience.js";

export default class Renderer {
  constructor() {
    this.experience = new Experience();
  }
}

```

```
this.sizes = this.experience.sizes;
this.scene = this.experience.scene;
this.canvas = this.experience.canvas;
this.camera = this.experience.camera;

this.setRenderer();
}

setRenderer() {
  this.renderer = new THREE.WebGLRenderer({
    canvas: this.canvas,
    antialias: true,
  });

  this.renderer.physicallyCorrectLights = true;
  this.renderer.outputEncoding = THREE.sRGBEncoding;
  this.renderer.toneMapping = THREE.CineonToneMapping;
  this.renderer.toneMappingExposure = 1.75;
  this.renderer.shadowMap.enabled = true;
  this.renderer.shadowMap.type = THREE.PCFSoftShadowMap;
  this.renderer.setSize(this.sizes.width, this.sizes.height);
  this.renderer.setPixelRatio(this.sizes.pixelRatio);
}

resize() {
  this.renderer.setSize(this.sizes.width, this.sizes.height);
  this.renderer.setPixelRatio(this.sizes.pixelRatio);
}

update() {
  // this.renderer.setViewport(0, 0, this.sizes.width,
  this.sizes.height);
```

```

this.renderer.render(this.scene, this.camera.orthographicCamera);

// second screen
// this.renderer.setScissorTest(true);
// this.renderer.setViewport(
//   this.sizes.width - this.sizes.width / 3,
//   this.sizes.height - this.sizes.height / 3,
//   this.sizes.width / 3,
//   this.sizes.height / 3
// );

// this.renderer.setScissor(
//   this.sizes.width - this.sizes.width / 3,
//   this.sizes.height - this.sizes.height / 3,
//   this.sizes.width / 3,
//   this.sizes.height / 3
// );
// this.renderer.render(this.scene,
this.camera.perspectiveCamera);

// this.renderer.setScissorTest(false);
}
}

```

d. Preloader.js

```

import { EventEmitter } from "events";
import Experience from "../Experience";
import GSAP from "gsap";
import convert from "../Utils/convertDisvToSpans";

export default class Preloader extends EventEmitter {

```

```

constructor() {
  super();
  this.experience = new Experience();
  this.scene = this.experience.scene;
  this.sizes = this.experience.sizes;
  this.resources = this.experience.resources;
  this.camera = this.experience.camera;
  this.world = this.experience.world;
  this.device = this.sizes.device;

  this.sizes.on("switchdevice", (device) => {
    this.device = device;
  });

  this.world.on("worldready", () => {
    this.setAssets();
    this.playIntro();
  });
}

setAssets() {
  convert(document.querySelector(".intro-text"));
  convert(document.querySelector(".hero-main-title"));
  convert(document.querySelector(".hero-main-description"));
  convert(document.querySelector(".hero-second-subheading"));
  convert(document.querySelector(".second-sub"));
  this.room = this.experience.world.room.actualRoom;
  this.roomChildren = this.experience.world.room.roomChildren;
  console.log(this.roomChildren);
}

firstIntro() {
  return new Promise((resolve) => {

```

```

this.timeline = new GSAP.timeline();
this.timeline.to("preloader", {
  opacity: 0,
  delay: 1,
  onComplete: () => {
    document.querySelector(".preloader").classList.add("hidden"
);
  },
});
if (this.device === "desktop") {
  this.timeline
    .to(this.roomChildren.box.scale, {
      x: 1.4,
      y: 1.4,
      z: 1.4,
      ease: "back.out(2.5)",
      duration: 0.7,
    })
    .to(this.room.position, {
      x: -1,
      ease: "power1.out",
      duration: 0.7,
    });
} else {
  this.timeline
    .to(this.roomChildren.box.scale, {
      x: 1.4,
      y: 1.4,
      z: 1.4,
      ease: "back.out(2.5)",
      duration: 0.7,
    })

```

```
.to(this.room.position, {
  z: -1,
  ease: "power1.out",
  duration: 0.7,
  onComplete: resolve,
});
}
this.timeline
.to(".intro-text .animatedis", {
  yPercent: -100,
  stagger: 0.07,
  ease: "back.out(1.7)",
  onComplete: resolve,
})
.to(
  ".arrow-svg-wrapper",
  {
    opacity: 1,
  },
  "same"
)
.to(
  ".toggle-bar",
  {
    opacity: 1,
    onComplete: resolve,
  },
  "same"
);
});
}
```

```
secondIntro() {  
  return new Promise((resolve) => {  
    this.secondTimeline = new GSAP.timeline();  
    if (this.device === "desktop") {  
      this.secondTimeline  
        .to(".intro-text .animatedis", {  
          yPercent: 100,  
          stagger: 0.05,  
          ease: "back.in(1.7)",  
        })  
        .to(  
          ".arrow-svg-wrapper",  
          {  
            opacity: 0,  
          },  
          "fadeout"  
        )  
        .to(  
          this.room.position,  
          {  
            x: 0,  
            y: 0,  
            z: 0,  
            ease: "power1.out",  
          },  
          "same"  
        )  
        .to(  
          this.roomChildren.box.rotation,  
          {  
            y: 2 * Math.PI + Math.PI / 4,  
          },  
        )  
    }  
  })  
}
```



```

"same"
)
.to(this.roomChildren.box.scale, {
  x: 5,
  y: 5,
  z: 5,
})
.to(
  ".hero-main-title .animatedis",
  {
    yPercent: -100,
    stagger: 0.07,
    ease: "back.out(1.7)",
  },
  "same"
)
.to(
  ".hero-main-description .animatedis",
  {
    yPercent: -100,
    stagger: 0.07,
    ease: "back.out(1.7)",
  },
  "same"
)
.to(
  ".hero-second-subheading .animatedis",
  {
    yPercent: -100,
    stagger: 0.07,
    ease: "back.out(1.7)",
  },

```

```

"same"
)
.to(
  ".second-sub .animatedis",
  {
    yPercent: -100,
    stagger: 0.07,
    ease: "back.out(1.7)",
  },
  "same"
)
.to(this.camera.orthographicCamera.position, { y: 0.5 })
.to(
  this.roomChildren.box.position,
  {
    x: 0.6387,
    y: 5,
    z: 1.3243,
  },
  "same"
)
.set(this.roomChildren.box.position, {
  y: 0.8,
  ease: "back.out",
})
.to(this.roomChildren.box.scale, {
  x: 0,
  y: 0,
  z: 0,
  ease: "back.out",
})
.to(this.roomChildren.gedunga.scale, {

```

```
x: 6,  
y: 1,  
z: 2,  
ease: "back.out(2.2)",  
duration: 0.5,  
}))  
.to(this.roomChildren.gedunga1.scale, {  
  x: 1,  
  y: 1,  
  z: 1,  
  ease: "back.out(2.2)",  
  duration: 0.5,  
}))  
.to(this.roomChildren.gedunga2.scale, {  
  x: 1,  
  y: 1,  
  z: 1,  
  ease: "back.out(2.2)",  
  duration: 0.5,  
}))  
.to(this.roomChildren.gedunga3.scale, {  
  x: 1,  
  y: 1,  
  z: 1,  
  ease: "back.out(2.2)",  
  duration: 0.5,  
}))  
.to(this.roomChildren.gedunga4.scale, {  
  x: 1,  
  y: 1,  
  z: 1,  
  ease: "back.out(2.2)",
```

```
        duration: 0.5,
    ))
    .to(this.roomChildren.rg.scale, {
        x: 1,
        y: 1,
        z: 1,
        ease: "back.out(2.2)",
        duration: 0.5,
    ))
    .to(this.roomChildren.rks.scale, {
        x: 1,
        y: 1,
        z: 1,
        ease: "back.out(2.2)",
        duration: 0.5,
    ))
    .to(this.roomChildren.tu.scale, {
        x: 1,
        y: 1,
        z: 1,
        ease: "back.out(2.2)",
        duration: 0.5,
    ))
    .to(this.roomChildren.uks.scale, {
        x: 2,
        y: 1,
        z: 2,
        ease: "back.out(2.2)",
        duration: 0.5,
    ))
    .to(this.roomChildren.warung.scale, {
        x: 1,
```

```
        y: 1,
        z: 1,
        ease: "back.out(2.2)",
        duration: 0.5,
    });
} else {
    this.secondTimeline.to(this.room.position, {
        x: 0,
        y: 0,
        z: 0,
        ease: "power1.out",
    });
}
});
}

onScroll(e) {
    if (e.deltaY > 0) {
        console.log("addedEvent");
        this.removeEventListeners();
        this.playSecondIntro();
    }
}

onTouch(e) {
    this.initialY = e.touches[0].clientY;
}

onTouchMove(e) {
    let currentY = e.touches[0].clientY;
    let difference = this.initialY - currentY;
    if (difference > 0) {
```

```
        console.log("swipe UPs");
        this.removeEventListeners();
        this.playSecondIntro();
    }
    this.initialY = null;
}

removeEventListeners() {
    window.removeEventListener("wheel", this.scrollOnceEvent);
    window.removeEventListener("touchstart", this.touchStart);
    window.removeEventListener("touchmove", this.touchMove);
}

async playIntro() {
    await this.firstIntro();
    this.moveFlag = true;
    this.scrollOnceEvent = this.onScroll.bind(this);
    this.touchStart = this.onTouch.bind(this);
    this.touchMove = this.onTouchMove.bind(this);
    window.addEventListener("wheel", this.scrollOnceEvent);
    window.addEventListener("touchstart", this.touchStart);
    window.addEventListener("touchmove", this.touchMove);
}

async playSecondIntro() {
    this.moveFlag = false;
    this.scaleFlag = true;
    await this.secondIntro();
    this.scaleFlag = false;
    this.emit("enablecontrols");
}
```

```

move() {
  if (this.device === "desktop") {
    this.room.position.set(-1, 0, 0);
  } else {
    this.room.position.set(0, 0, -1);
  }
}

scale() {
  if (this.device === "desktop") {
    this.room.scale.set(0.2, 0.2, 0.2);
  } else {
    this.room.scale.set(0.07, 0.07, 0.07);
  }
}

update() {
  if (this.moveFlag) {
    this.move();
  }
  if (this.scaleFlag) {
    this.scale();
  }
}
}

```

e. Theme.js

```

import { EventEmitter } from "events";

export default class Theme extends EventEmitter {

```

```

constructor() {
  super();

  this.theme = "light";
  this.toggleButton = document.querySelector(".toggle-button");

  this.toggleCircle = document.querySelector(".toggle-circle");

  this.setEventListener();
}

setEventListener() {
  this.toggleButton.addEventListener("click", () => {
    this.toggleCircle.classList.toggle("slide");
    this.theme = this.theme === "light" ? "dark" : "light";
    document.body.classList.toggle("dark-theme");
    document.body.classList.toggle("light-theme");
    // console.log(this.theme);
    this.emit("switch", this.theme);
  });
}
}

```

5.1.2. Implementasi Sistem

Tempat implementasikannya aplikasi peta tiga dimensi ini antara lain dilakukan di kampus SMP Karya Pembangunan 2 Majalaya, dengan detail sebagai berikut:

a. Waktu dan Tempat

Lokasi : SMP Karya Pembangunan 2 Majalaya

Alamat : Jl.Wangisagara – Majalaya
 Tempat : Gedung A Kelas 9C
 Waktu : 14.00 WIB s/d selesai.

b. Subjek

Siswa : 25 Orang
 Orang Tua : 5 Orang
 Guru : 35 Orang
 Tamu : -

5.1.3. Spesifikasi Sistem

1. Perangkat Keras

Perangkat keras yang digunakan sebagai berikut :

Tabel 5.1 Spesifikasi Minimum Perangkat Keras dalam menjalankan Sistem

No	Perangkat Keras	Spesifikasi Minimum	Spesifikasi yang digunakan
1	<i>Processor</i>	Intel HD 4000	Intel® Core™ i3-4030U 1.90GHz Or Higher
2	GPU	GPU NVIDIA <i>GeForce 2xx</i>	GPU NVIDIA GeForce 820M Or Higher
3	<i>Internet</i>	<i>Internet Support</i>	4G Or Higher

2. Perangkat Lunak

Tabel 5.2 Spesifikasi Minimum Perangkat Lunak dalam menjalankan Sistem

No	Nama Perangkat Lunak	Spesifikasi Minimum
----	----------------------	---------------------

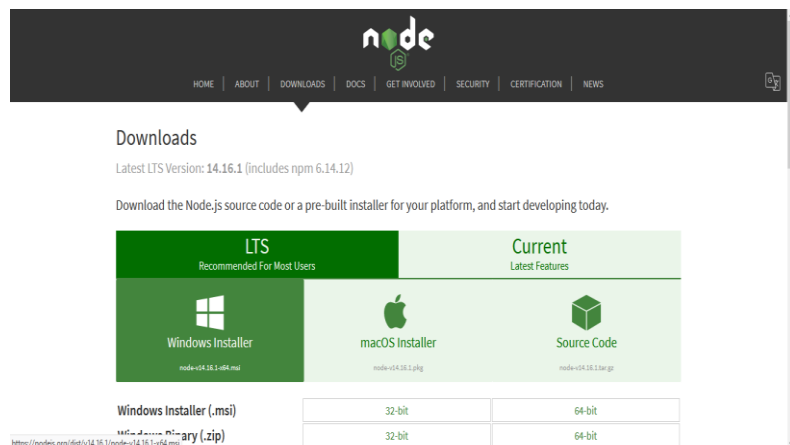
1	Sistem Operasi	Windows 7 or Higher
2	Web Browser	Chrome v.54 Firefox v.51

3. Instalasi Sistem

Berikut adalah langkah-langkah instalasi aplikasi Peta 3D Kampus SMP Karya Pembangunan 2 Majalaya.

Dalam menginstal sistem, terdapat beberapa langkah yang harus di lakukan berikut langkah yang perlu di lakukan:

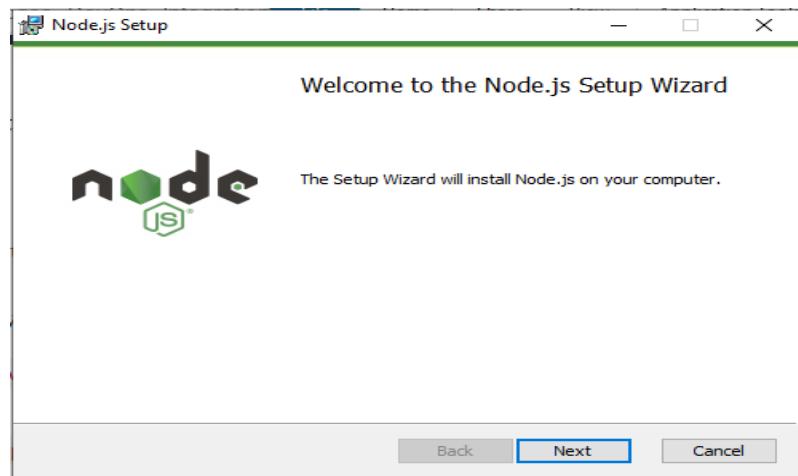
- a) Instalasi node.js harus terinstall terlebih dahulu pertama buka Official page Node.js untuk mendownload installer untuk windows lalu pilih opsi “*Windows Installer*”



Gambar 5.1 Download Node.js

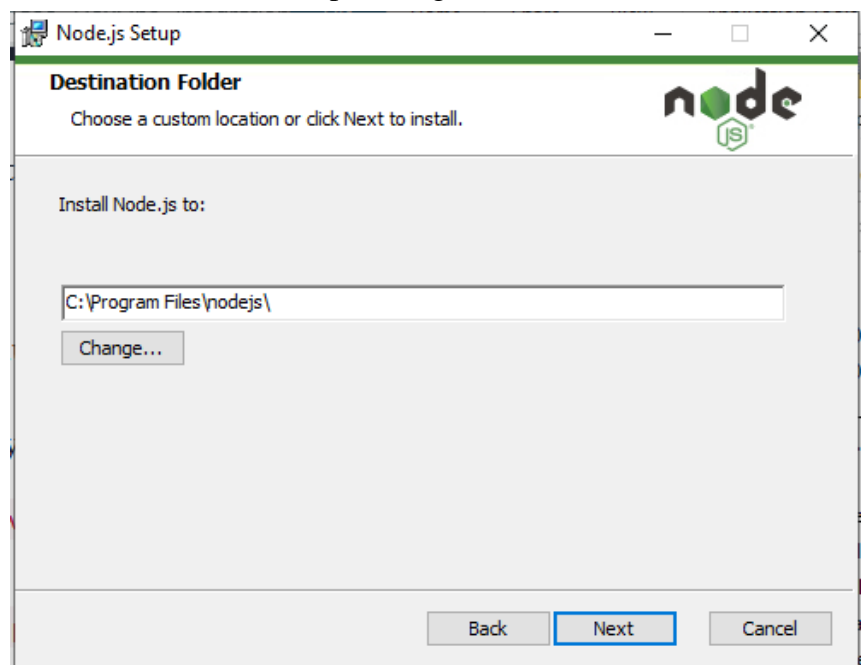
- Download sesuai dengan bit laptop/pc masing masing

- Jalankan *installer* Node.msi yang telah di download lalu *accept license agreement*:



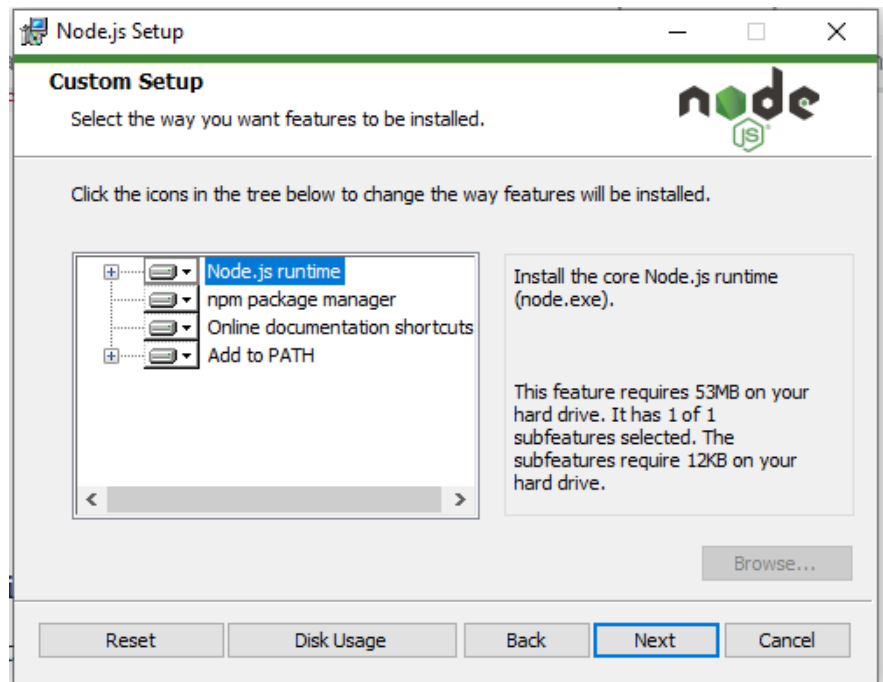
Gambar 5.2 Langkah Instalasi Node.js

- Tentukan folder tempat menginstall



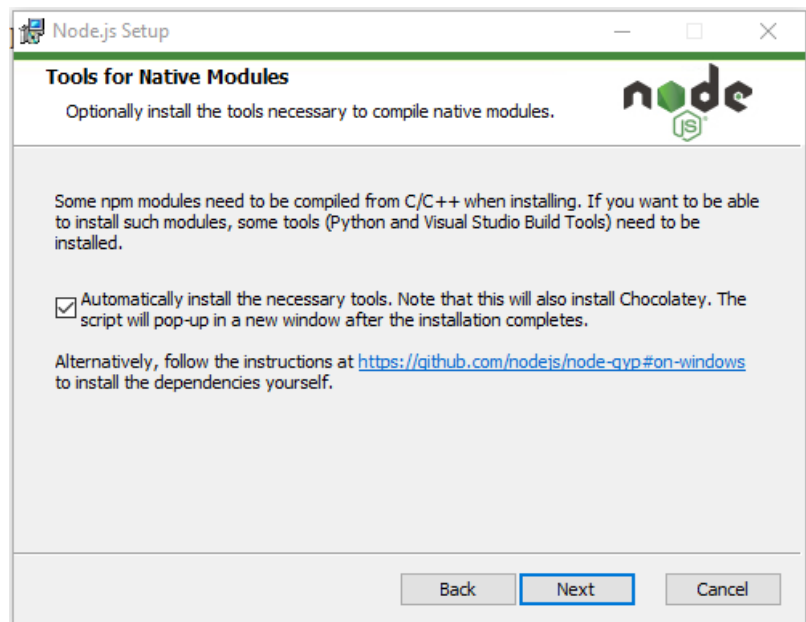
Gambar 5.3 Menentukan PATH penyimpanan Node.js

- Pada custom setup, klik next



Gambar 5.4 Custom page setup

- *Install dependencies*, dengan mencentang “*Automatically Install the Necessary Tools*”.

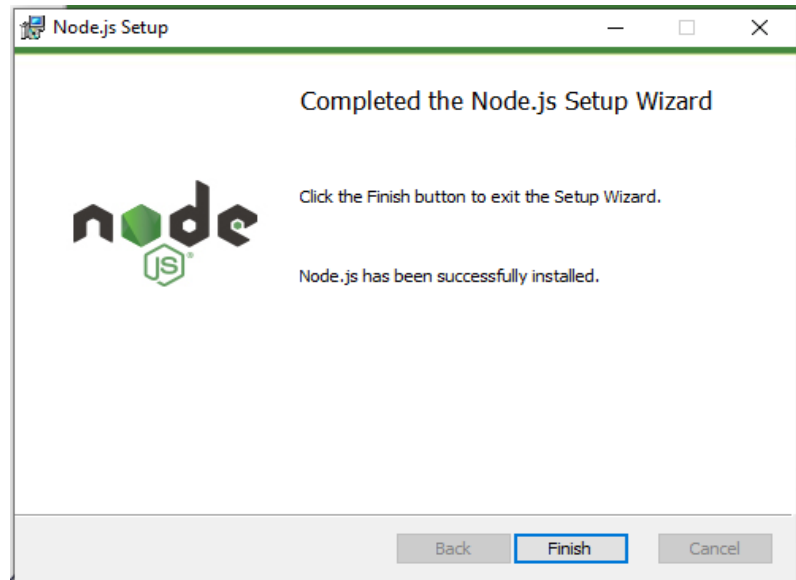


Gambar 5.5 Install Dependencee

- Klik “*Install*” untuk Memulai

Catatan : Instalasi akan membutuhkan *privilage* sebagai *Administrator*, sehingga klik “yes” apabila muncul opsi meminta izin Admin untuk melakukan instalasi.

- Instalasi Selesai

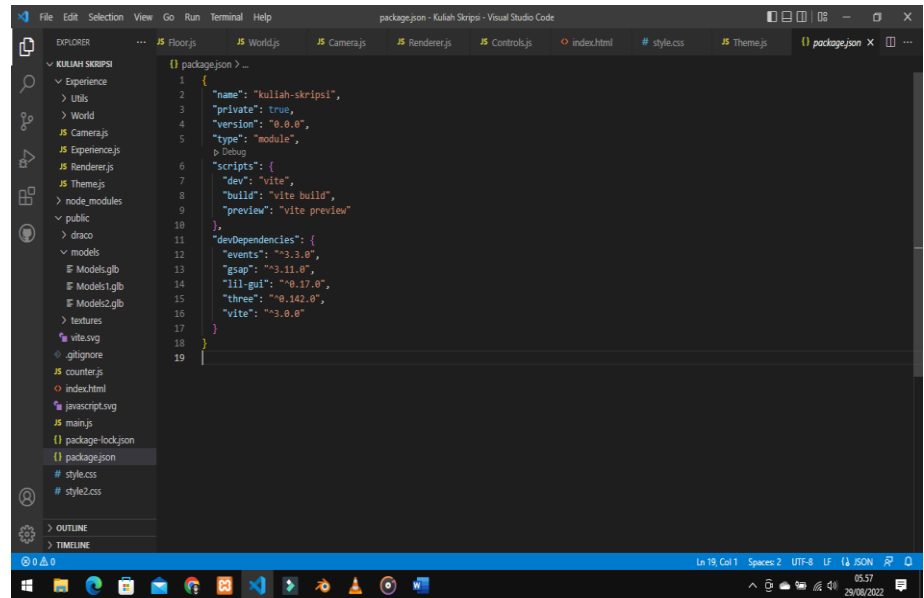


Gambar 5.6 Instalasi Selesai

- Setelah instalasi selesai buka sistem *folder* beserta *subfolder* menggunakan *visual studio code* (*Opsional* sesuai penggunaan text editor)

b) *Push* aplikasi ke dalam *Git Repo*

Setelah Node.js telah terinstall, buka editor text buka folder aplikasi peta 3D yang telah di download menggunakan editor text. Sehingga folder – subfolder beserta file terlihat dengan jelas



Gambar 5.7 Tampilan *folder*, *subfolder* beserta file aplikasi peta 3D

Jika tampilan telah sesuai dengan gambar diatas buka terminal lalu ketikan perintah seperti dibawah:

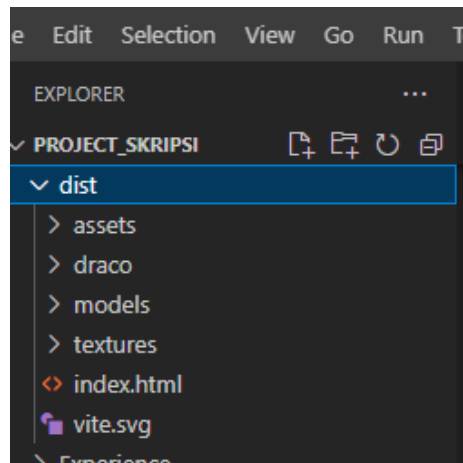
```
>> npm run build
```

Perintah diatas ditujukan untuk membuat *folder /dist*. di dalam folder terdapat chace aplikasi yang telah di buat. Serta dapat meng-hash sumber kode asli.

```
>> npm run review
```

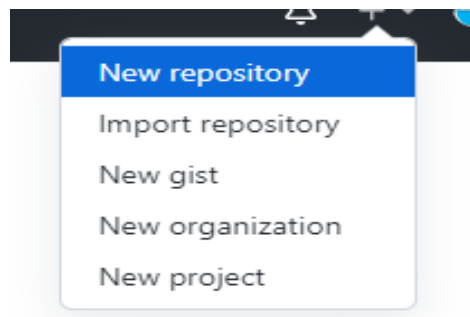
Setelah melakukan *run build* untuk membuat cache langkah selanjutnya menampilkan aplikasi dari chache secara review, berguna untuk menampilkan aplikasi secara local.

Setelah terdapat folder `/dist` langkah selanjutnya push project kedalam github menggunakan Terminal



Gambar 5.8 Langkah *commit and push* menggunakan *Terminal*

Buat *repository* baru di *github*



Gambar 5.9 Membuat *repository* baru

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner * taufiqjamil47 / Repository name * skripsi-app-bokoko333 ✓

Great repository names are short and memorable. Need inspiration? How about [automatic-octo-train?](#)

Description (optional)

☒ Public
Anyone on the internet can see this repository. You choose who can commit.

☐ Private
You choose who can see and commit to this repository.

Initialize this repository with:
Skip this step if you're importing an existing repository.

☐ Add a README file
This is where you can write a long description for your project. [Learn more.](#)

Add .gitignore
Choose which files not to track from a list of templates. [Learn more.](#)

.gitignore template: None

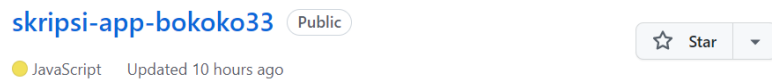
Choose a license
A license tells others what they can and can't do with your code. [Learn more.](#)

License: None

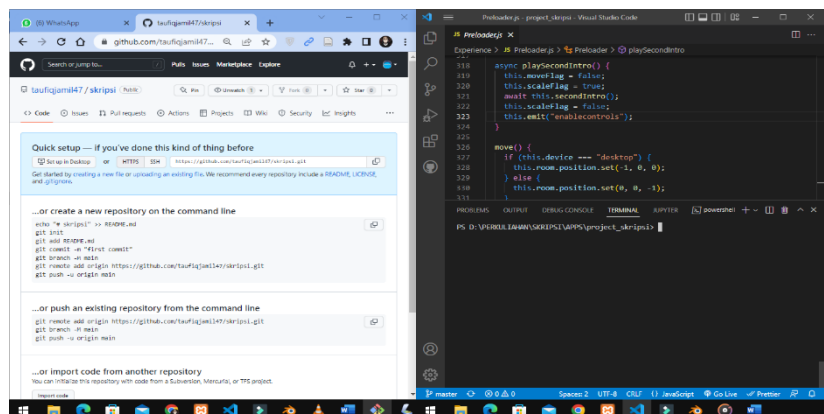
① You are creating a public repository in your personal account.

[Create repository](#)

Gambar 5.10 Nama *Repository*

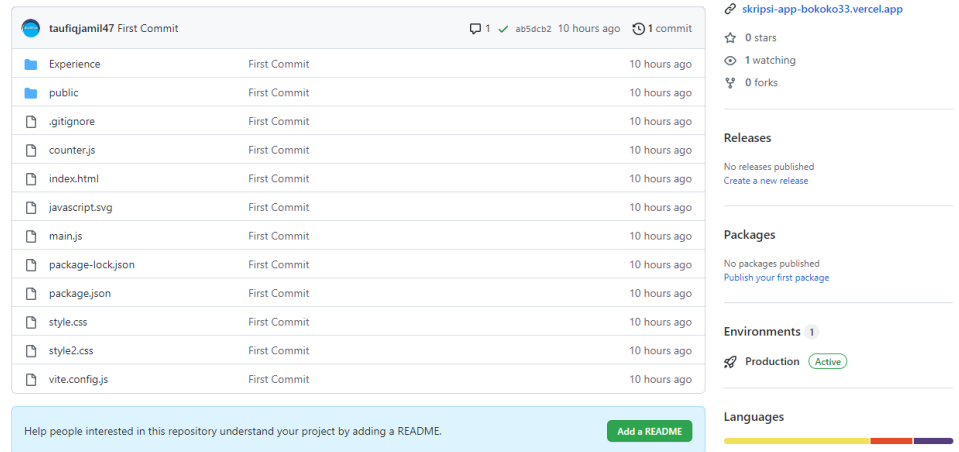


Gambar 5.11 *Repository* telah terbuat



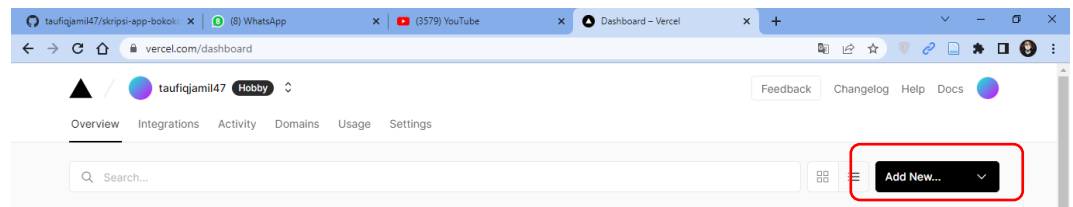
Gambar 5.12 Melakukan perintah *commit* ke github

Ikuti langkah langkah seperti gambar di sebelah kiri untuk melakukan *push* dan *commit* langkah pada gambar sebelah kiri menjelaskan semua *point* yang di butuhkan.

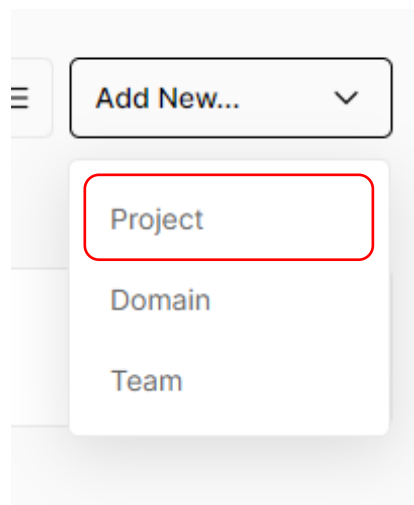


Gambar 5.13 *Commit* dan *Push* telah selesai

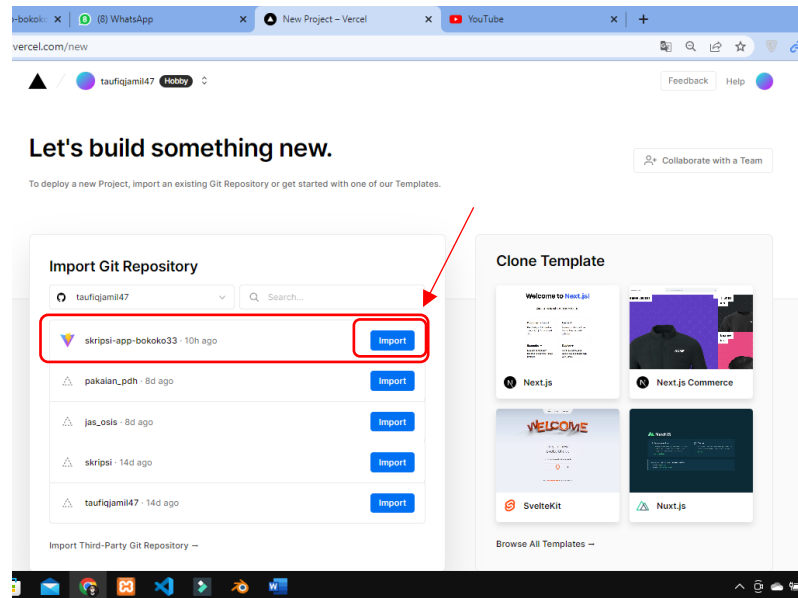
c) Commit Sistem Kedalam *Vercel app*



Gambar 5.14 Buat *Project* Baru

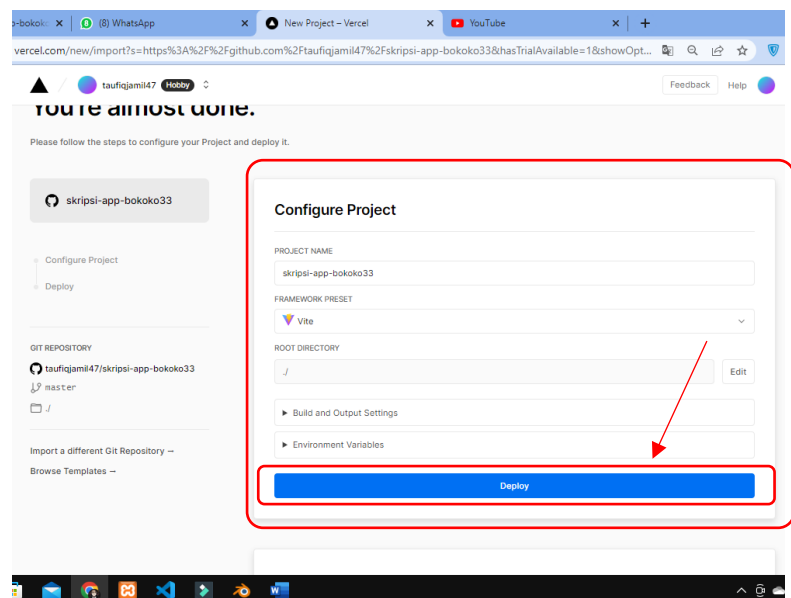


Gambar 5.15 Tambah *Project* baru

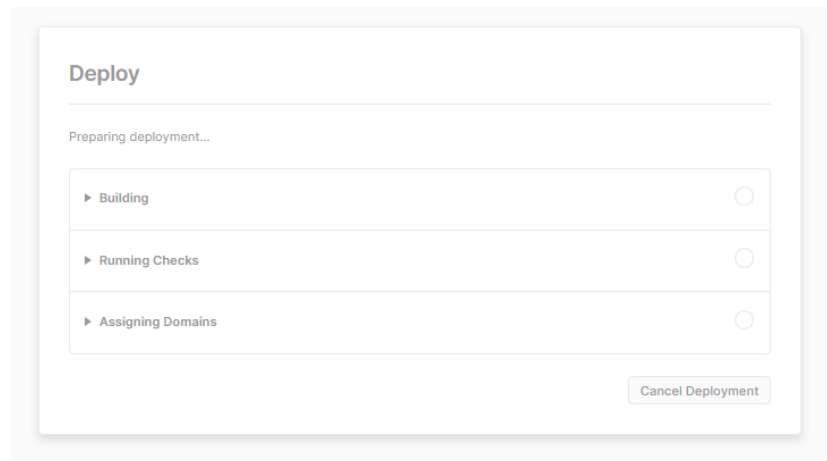


Gambar 5.16 Pilih Skripsi-app-bokoko33

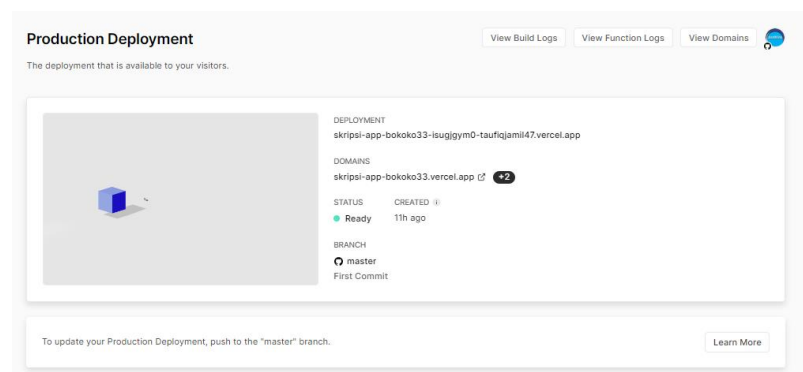
Pilih import Git Repository sesuai nama yang telah dibuat pada Git Repository sebelumnya, lalu tekan “import”

Gambar 5.17 *Configure Project*

Setelah tombol klik import selesai maka akan muncul tampilan diatas, jika konfigurasi telah selesai tekan tombol “deploy”, maka akan muncul tampilan seperti gambar dibawah, proses sedang berlangsung.

Gambar 5.18 Proses *deploying*

Tunggu hingga proses deploy selesai, jika telah selesai maka akan muncul tampilan seperti gambar dibawah.



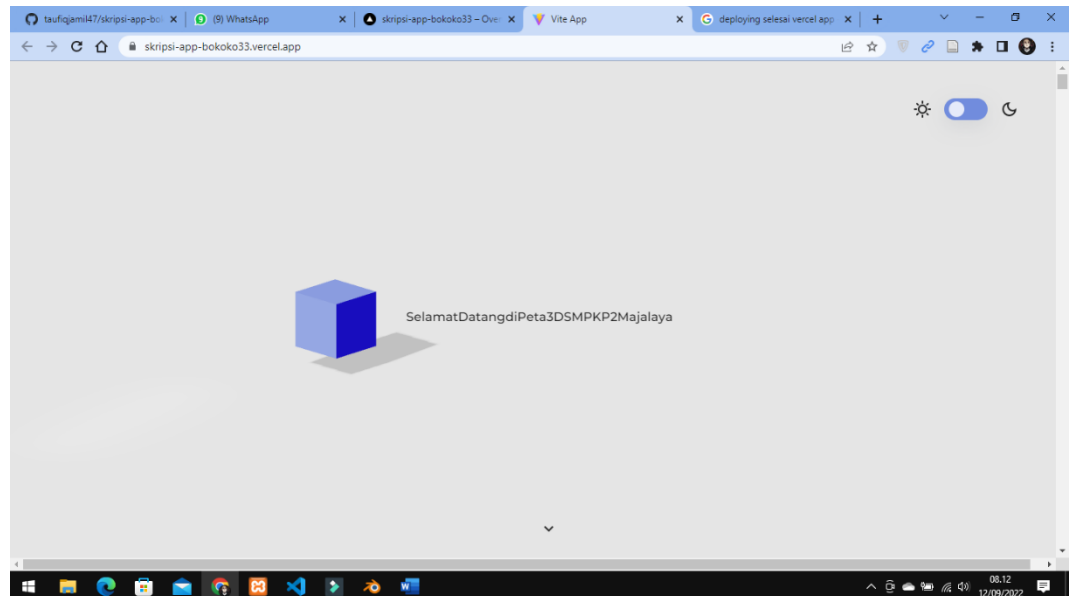
Gambar 5.19 Deploying berhasil

4. Menjalankan Sistem

a) Prototyping

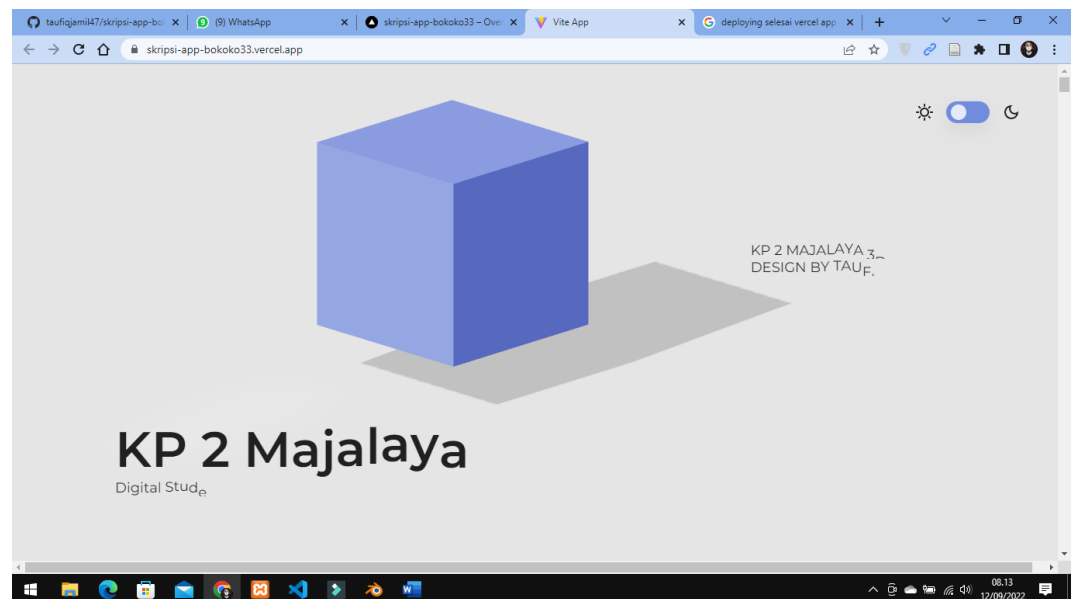
Prototype peta 3D Kampus SMP Karya Pembangunan 2 Majalaya ini akan dipakai dalam pengujian beta untuk mendapatkan data tentang apakah ada kesalahan fungsionalitas pada peta 3D yang dibuat supaya dapat diperbaiki dan kemudian dapat dilanjutkan ketahap evaluasi. Berikut adalah hasil dari penggunaan peta 3D kampus SMP Karya Pembangunan 2 Majalaya

menggunakan komputer untuk menjelaskan hasil dari prototype ini.



Gambar 5.20 Tampilan awal

Pada gambar 5.20 *scene* yang pertama kali muncul setelah pengguna membuka peta 3D SMP Karya Pembangunan 2 Majalaya menampilkan box 3D yang dibuat menggunakan blender

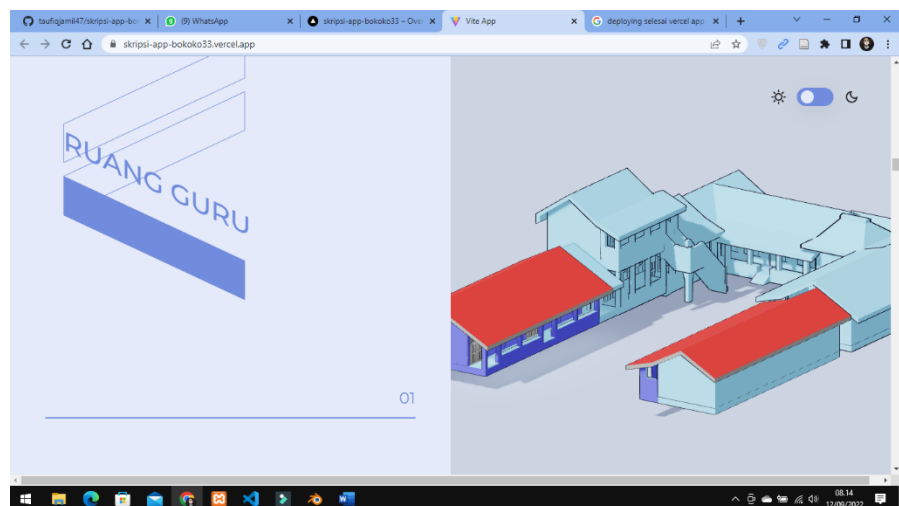


Gambar 5.21 Animasi pertama

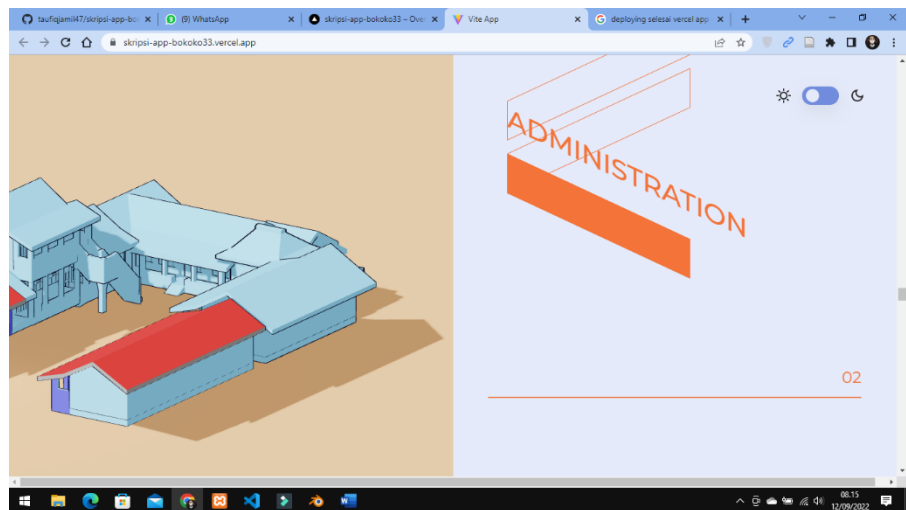


Gambar 5.22 Tampilan Peta

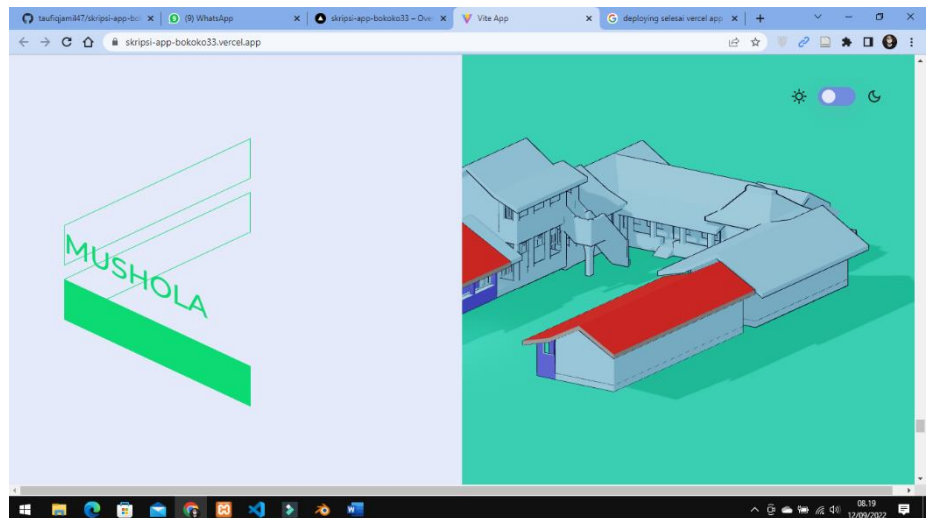
Pada gambar 5.22 tampilan peta mulai diperlihatkan setelah box animasi telah selesai ditampilkan. Pengguna akan terus melakukan scroll sampai informasi yang lain di tampilkan. Sesuai pada tabel navigational structure aplikasi akan terus menampilkan gedung beserta informasi terkait Kampus SMP Karya Pembangunan 2 Majalaya



Gambar 5.23 Informasi mengenai Ruang Guru



Gambar 5.24 Informasi mengenai Ruang Tata Usaha



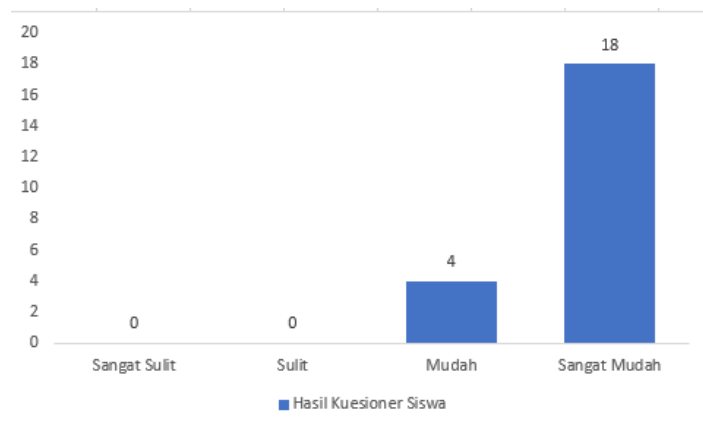
Gambar 5.25 Informasi tentang Mushola

b) Beta Testing

Pada tahap Beta testing dilakukan oleh 25 siswa/I SMP Karya Pembangunan 2 Majalaya. untuk melakukan pengujian beta terhadap prototype peta 3D ini. fokus dalam pengujian ini adalah pada kebutuhan fungsionalitas dari aplikasi, yaitu kesesuaian output dan input yang didapat oleh sistem. Pengujian antarmuka peta 3D dengan mencoba semua fungsionalitas dari menu utama, scrolling dan theme. Pengujia ini bertujuan untuk mengetahui tingkat kepuasan dan pendapat pengguna pada peta 3D ini.

Hasil beta testing bahwa secara fungsionalitas peta 3D SMP Karya Pembangunan 2 Majalaya telah memenuhi kebutuhan sistem yang diperlukan. Dari kuesioner yang di sebarakan kepada responden, hasil kuesioner pada setiap pertanyaan dapat dilihat pada grafik dibawah ini:

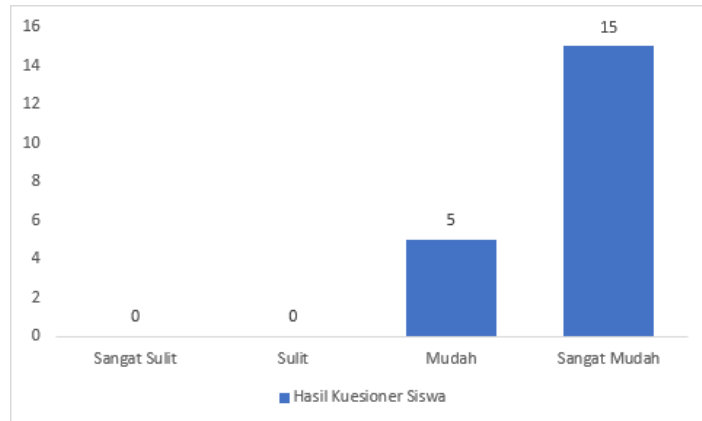
1. Apakah Peta 3D SMP Karya Pembangunan 2 Majalaya mudah digunakan?



Gambar 5.26 Grafik Peta 3D mudah digunakan

Pada grafik yang ditampilkan dalam gambar 5.26. dilakukan perhitungan skala likert dengan keterangan “Sangat Mudah”=4, “Mudah”=3, “Sulit”=2, “Sangat Sulit”=1. Hasil yang didapatkan sebesar 4 orang menunjukkan bahwa responden menyatakan Peta 3D ini mudah untuk digunakan dan 18 orang menyatakan sangat mudah digunakan.

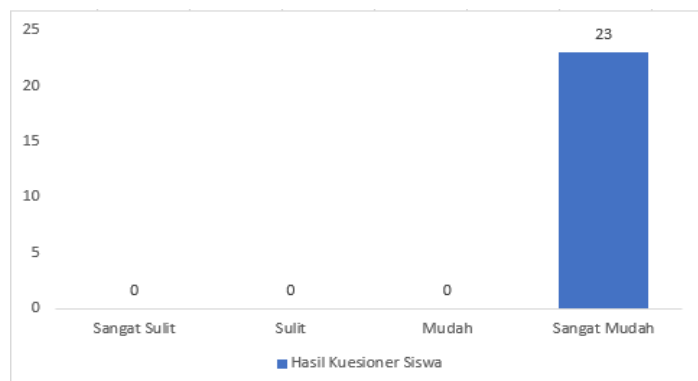
2. Apakah Navigasi pada scroll menu sangat sulit untuk di kendalikan?



Gambar 5.27 Grafik tingkat pemahaman Scroll Navigasi

Pada grafik yang ditampilkan dalam gambar 5.27. dilakukan perhitungan menggunakan skala likert dengan keterangan “Sangat Mudah”=4, “Mudah”=3, “Sulit”=2, “Sangat Sulit”=1. Hasil yang didapatkan sebesar 5 Orang menunjukkan bahwa responden menyatakan kontrol navigasi Scroll mudah dipahami dan 15 orang menyatakan sangat mudah dipahami

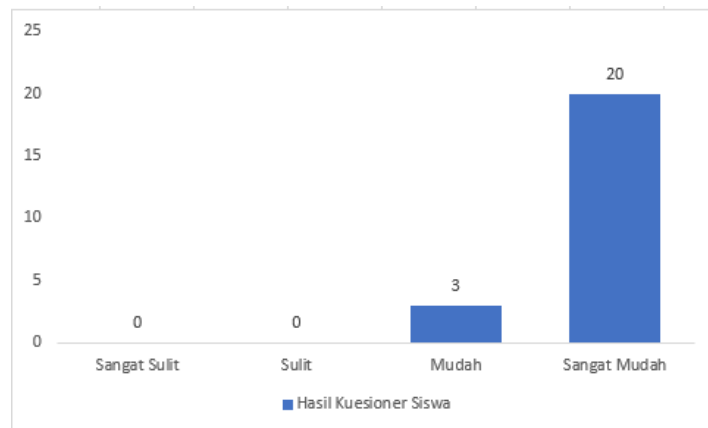
3. Apakah peta 3D kampus SMP Karya Pembangunan 2 Majalaya ini menarik untuk digunakan dibandingkan dengan master plan 2D SMP Karya Pembangunan 2 Majalaya?



Gambar 5.28 Grafik tingkat kemenarikan Peta 3D SMP Karya Pembangunan 2 Majalaya

Pada grafik yang ditampilkan dalam gambar 5.28 dilakukan perhitungan menggunakan skala likert dengan keterangan “Sangat Menarik”=4, “Menarik”=3, “Tidak Menarik”=2, “Sangat Tidak Menarik”=1. Hasil yang didapatkan sebesar 23 orang (100%) menunjukkan bahwa responden menyatakan peta 3D ini sangat menarik.

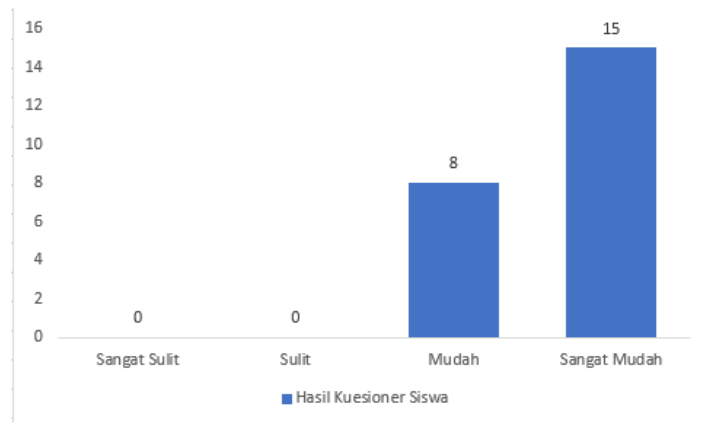
4. Apakah dengan Peta 3D SMP Karya Pembangunan 2 Majalaya ini dapat membantu anda mengenal tata letak gedung dan ruangan yang ada di kampus SMP Karya Pembangunan 2 Majalaya ?



Gambar 5.29 Grafik Peta 3D akan membantu responden mengenal tata letak gedung dan ruangan

Pada grafik yang ditampilkan dalam gambar 5.29. dilakukan perhitungan menggunakan skala likert dengan keterangan “Sangat Membantu”=4, “Membantu”=3, “Tidak Membantu”=2, “Sangat Tidak Membantu”=1. Hasil yang didapatkan sebesar 3 orang menunjukkan bahwa responden menyatakan Peta 3D ini membantu dan 20 orang menyatakan sangat membantu dalam mengenal tata letak gedung dan ruangan yang ada di kampus SMP Karya Pembangunan 2 Majalaya.

5. Apakah anda merasa peta 3D kampus SMP Karya Pembangunan 2 Majalaya ini dapat membantu dalam pencarian lokasi gedung atau ruangan yang ingin di tuju?



Gambar 5.30 Grafik Peta 3D dapat membantu dalam pencarian lokasi gedung atau ruangan

Pada grafik yang ditampilkan dalam gambar 5.30 dilakukan perhitungan menggunakan skala likert dengan keterangan “Sangat Membantu”=4, “Membantu”=3, “Tidak Membantu”=2, “Sangat Tidak Membantu”=1. Hasil yang didapatkan sebesar 8 orang menunjukkan bahwa responden menyatakan Peta 3D ini dapat membantu dan 15 Orang menyatakan sangat membantu dalam pencarian lokasi gedung atau ruangan yang ada di kampus SMP Karya Pembangunan 2 Majalaya

5.2. PENGUJIAN

Setelah proses pengembangan web peta 3D selesai, proses selanjutnya adalah proses pengujian web. Pengujian sistem informasi menggunakan *black box testing*, *black box testing* merupakan pengujian yang dilakukan hanya mengamati hasil eksekusi melalui data uji dan memeriksa fungsi dari perangkat lunak pengujian *black box*.

Tabel 5.3 Pengujian *Black Box*

Aktifitas Pengujian	Realisasi yang diharapkan	Hasil Pengujian	Kesimpulan
Akses	Mengakses menggunakan link yang disediakan	Masuk menu Web	[x] Diterima [] Ditolak
Halaman Web	Muncul tampilan awal aplikasi	Toggle bar yang disertai animasi muncul	[x] Diterima [] Ditolak
Theme	Menggunakan Theme untuk memindahkan dari light mode menjadi dark mode	Menggunakan tombol toggle bar	[x] Diterima [] Ditolak
Scroll	Scrolling	Menggunakan Scroll pada mouse	[x] Diterima [] Ditolak

Dalam pengujian black box adala salah satu aksi yang belum bisa di realisasikan pada animasi yang begitu menarik ketika user menggunakan scroll animasi per scene akan ditampilkan. Oleh sebab itu penambahan materi ini akan dikembangkan oleh seorang admin di sekolah berdasarkan kebutuhan dan capaian animasi yang mampu di buat.

5.3. EVALUASI

Pada tahapan evaluasi ini, sistem yang sudah dibuat dan diuji berdasarkan pada *The Eight Golden Rules of Interface Design* yang dikemukakan oleh Ben Schneiderman. Berikut merupakan pembahasan dari tiap tiap prinsip *The Eight Golden Rules of Interface Design* yang diterapkan pada Peta 3D Kampus SMP Karya Pembangunan 2 Majalaya.

1. Strive for Consistency

Maksud yang terdapat pada prinsip ini ialah konsistensi dalam aplikasi seperti istilah-istilah yang identik harus digunakan dalam layar.

2. Cater to Universallibility

Maksud yang terdapat pada prinsip ini ialah menjelaskan perbedaan antara pengguna yang telah ahli dengan pengguna yang masih baru

3. Offer Informative Feedback

Maksud yang terdapat pada prinsip ini ialah setiap aksi yang digunakan oleh pengguna harus ada umpan balik sistem

4. Design Dialogs to Yield Closure

Maksud yang terdapat pada prinsip ini ialah adanya urutan aksi yang dilakukan pengguna mulai dari awal, tengah, dan akhir.

5. Prevent Errors

Maksud yang terdapat pada prinsip ini ialah mencoba mengarahkan pengguna untuk terhindar dari melakukan kesalahan.

6. Permit Easy Reversal of Actions

Maksud yang terdapat pada prinsip ini ialah sebisa mungkin terdapat pembalikan aksi dalam sistem supaya memberikan kemudahan pada pengguna dalam menjelajah.

7. Support Internal Locus of Controls

Maksud yang terdapat pada prinsip ini ialah memberikan kewenangan pengguna sebagai pemegang kendali dalam sistem

8. Reduce Short-term Memory Load

Maksud yang terdapat pada prinsip ini ialah mengurangi ingatan jangka pendek, hal ini dikarenakan bahwa rata-rata manusia memiliki ingatan yang pendek, maka dari itu desain yang dibuat sesederhana mungkin dan menarik untuk mengurangi beban ingatan tersebut.

BAB VI

KESIMPULAN DAN SARAN

6.1. KESIMPULAN

Dari hasil penelitian dan pembahasan yang telah dilaksanakan, maka dapat diambil kesimpulan bahwa:

1. Peta diperlukan dalam pencarian denah maupun lokasi karena jika tanpa peta orang bisa salah dalam mengambil arah, sehingga dibuatlah informasi peta baik 2D maupun 3D.
2. Peta 3D Kampus SMP Karya Pembangunan 2 Majalaya dikembangkan menggunakan metode *Interactive Multimedia System Design and Developmnt* yang terdiri atas *Requirement System, Consideration Design, Implementation*, dan *Evaluation*.
3. Peta 3D Kampus SMP Karya Pembangunan 2 Majalaya menggunakan *Library Three.js* dan *Blender 3D* dalam mendesain bentuk 3D serta penerapan *logic* pemrograman navigasi.
4. Implementasi peta 3D berbasis web dirancang supaya mampu memberikan gambaran kecil tentang keadaan dan tata letak atau lokasi di SMP Karya Pembangunan 2 Majalaya, yang akan membuat siswa baru, orang tua, maupun tamu bisa mengetahui bentuk dan rupa gedung yang hendak di kunjunginya.
5. Mempercepat proses menghafal tanpa harus terjun kelapangan.
6. Sistem yang dibangun memiliki tampilan yang menarik sehingga memanjakan mata.

6.2. SARAN

Adapun sarang yang penulis sampaikan adalah sebagai berikut:

1. Penelitian ini diharuskan memberikan efek yang baik terhadap daya ingat siswa terhadap lingkungannya, meskipun hanya melihat dari platform
2. Harus memiliki daya minat yang tinggi kepada siswa agar siswa mampu mengimplementasikannya di kemudian hari.

3. Menambahkan fitur animasi yang semakin banyak sehingga membuat daya tarik siswa semakin meningkat
4. Menambahkan kampus-kampus Yayasan Karya Pembangunan.

DAFTAR PUSTAKA

- Ananraytama, N.T., Safriadi, N. & Pratiwi, H.S., 2018, 'Penerapan Fitur 3D Maps pada Aplikasi Virtual Tour sebagai Media Promosi Wisata Qubu Resort', *Jurnal Sistem dan Teknologi Informasi (JUSTIN)*, 6(3), 135.
- Burhan, F., 2019, 'PENGEMBANGAN MEDIA PEMBELAJARAN PEMBELAJARAN KE DALAM BENTUK MULTIMEDIA INTERAKTIF MENGGUNAKAN METODE IMSDD (Studi Kasus: Pembelajaran Matematika Materi Menghitung Luas Bangun Datar Kelas III Sekolah Dasar di SDN No.64 Kota Timur)', *Skripsi*, 1(532413042).
- Devega, M., 2020, 'PETA 3D UNIVERSITAS LANCANG KUNING', 2(1), 13.
- Dirksen, J., 2013, *Learning Three.js: the JavaScript 3D Library for WebGL*, 1st. publ, Packt Publishing, Birmingham.
- Dwijayanti, I., 2019, 'Penerapan Teknologi Webgl Pada Sistem E-Mebel Menggunakan Metode Uml-Based', 10.
- Kraus, N.C., 2019, 'Navigation Structures', in C.W. Finkl & C. Makowski (eds.), *Encyclopedia of Coastal Science*, Encyclopedia of Earth Sciences Series., pp. 1242–1243, Springer International Publishing, Cham.
- Mardasatria, A., 2015, 'Visualisasi Kampus 3D Virtual Berbasis WebGL menggunakan three.js, Studi Kasus: Gedung Teknik Informatika Institut Teknologi Sepuluh Nopember', 68.
- Panggabean, R.F., Safriadi, N. & Nasution, H., no date, 'RANCANG BANGUN PETA VIRTUAL 3D KAMPUS UNTAN DENGAN FITUR PANORAMA 3600 X 1800', 6.
- Pradiptojati, D., Samop, F., Sani, N.A. & Hakim, J.A.R., 2014, 'Rancang Bangun Peta Virtual 3D Jurusan Teknik Informatika Institut Teknologi Sepuluh Nopember dengan Unity3D Engine', 3(2), 6.
- Pranata, R.E., Samopa, F. & Nisafani, A.S., 2018, 'Rancang Bangun Peta Interaktif Tiga Dimensi Monitoring Lahan (Monlah) di Institut Teknologi Sepuluh Nopember', *Jurnal Teknik ITS*, 7(1), 155–159.
- Rizqiyah, M., 2014, 'JURUSAN TEKNIK INFORMATIKA FAKULTAS SAINS DAN TEKNOLOGI UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM MALANG', 146.
- Sari, J.P. & Erlansari, A., 2014, 'RANCANG BANGUN APLIKASI LAYANAN BERBASIS LOKASI DENGAN PENERAPAN AUGMENTED REALITY MENGGUNAKAN METODE MARKERLESS BERBASIS ANDROID', 2, 11.
- Y. Rudi Kriswanto, 2020, 'PENERAPAN ARSITEKTUR INFORMASI PADA DIGITAL LIBRARY', *Jurnal Pustaka Budaya*, 7(2), 75–90.

Lampiran - 1

Kuesioner Siswa

Kuesioner sebelum melihat tata letak bangunan / gedung SMP Karya Pembangunan 2 Majalaya

A. Pertanyaan Lokasi, Ingat, pertanyaan ini tidak mengandung benar maupun salah !												
1	Dari 1 sampai 5 seberapa tahukah kamu tata letak bangunan / gedung seperti kelas dan berbagai ruangan di SMP KP 2 Majalaya ?	<table border="1"> <tr> <td colspan="3">Tidak Tahu</td> <td colspan="2">Sangat Tahu</td> </tr> <tr> <td>1</td> <td>2</td> <td>3</td> <td>4</td> <td>5</td> </tr> </table>	Tidak Tahu			Sangat Tahu		1	2	3	4	5
Tidak Tahu			Sangat Tahu									
1	2	3	4	5								
2	Dari 1 sampai 5 seberapa tahukah kamu kelas yang ada di SMP KP 2 Majalaya?	<table border="1"> <tr> <td colspan="3">Tidak Tahu</td> <td colspan="2">Sangat Tahu</td> </tr> <tr> <td>1</td> <td>2</td> <td>3</td> <td>4</td> <td>5</td> </tr> </table>	Tidak Tahu			Sangat Tahu		1	2	3	4	5
Tidak Tahu			Sangat Tahu									
1	2	3	4	5								
B. Pertanyaan Bangunan, Ingat, pertanyaan ini tidak mengandung benar maupun salah !												
1	Dari 1 sampai 5 Seberapa mudahkan menurutmu tata letak gedung SMP KP 2 Majalaya ?	<table border="1"> <tr> <td colspan="3">Sulit</td> <td colspan="2">Sangat Mudah</td> </tr> <tr> <td>1</td> <td>2</td> <td>3</td> <td>4</td> <td>5</td> </tr> </table>	Sulit			Sangat Mudah		1	2	3	4	5
Sulit			Sangat Mudah									
1	2	3	4	5								
2	Dari 1 sampai 5 apakah kamu mampu menghafal tata letak bangunan SMP KP 2 Majalaya jika tidak dilihat secara langsung?	<table border="1"> <tr> <td colspan="3">Tidak Mampu</td> <td colspan="2">Sangat Mampu</td> </tr> <tr> <td>1</td> <td>2</td> <td>3</td> <td>4</td> <td>5</td> </tr> </table>	Tidak Mampu			Sangat Mampu		1	2	3	4	5
Tidak Mampu			Sangat Mampu									
1	2	3	4	5								

3	Dari 1 sampai 5 apakah kamu mengetahui Tempat administrasi ?	<div> <div>Tidak Tahu</div> <div>Sangat Tahu</div> <div>1</div> <div>2</div> <div>3</div> <div>4</div> <div>5</div> </div>
4	Dari 1 sampai 5 apakah kamu tahu Ruang Guru?	<div> <div>Tidak Tahu</div> <div>Sangat Tahu</div> <div>1</div> <div>2</div> <div>3</div> <div>4</div> <div>5</div> </div>
5	Dari 1 sampai 5 apakah kamu pernah memasuki Ruang Administrasi?	<div> <div>Tidak Pernah</div> <div>Lebih dari sekali</div> <div>1</div> <div>2</div> <div>3</div> <div>4</div> <div>5</div> </div>
6	Dari 1 sampai 5 apakah kamu pernah mengunjungi kelas selain kelas sendiri?	<div> <div>Tidak Pernah</div> <div>Lebih dari sekali</div> <div>1</div> <div>2</div> <div>3</div> <div>4</div> <div>5</div> </div>
7	Dari 1 Sampai 5 Mampukah kamu menghafal beberapa gedung dan bangunan seperti kelas, Ruang Administrasi, Ruang Guru tanpa melihat secara langsung ?	<div> <div>Tidak Mampu</div> <div>Sangat Mampu</div> <div>1</div> <div>2</div> <div>3</div> <div>4</div> <div>5</div> </div>
8	Dari 1 Sampai 5 Seberapa seringkah kamu melihat dan memasuki bangunan di SMP KP 2 Majalaya ?	<div> <div>Tidak Pernah</div> <div>Lebih dari sekali</div> <div>1</div> <div>2</div> <div>3</div> <div>4</div> <div>5</div> </div>

Kuesioner setelah melihat tata letak bangunan / gedung SMP Karya Pembangunan 2 Majalaya menggunakan implementasi objek 3 dimensi berbasis Web

A. Pertanyaan Lokasi, Ingat, pertanyaan ini tidak mengandung benar maupun salah !												
1	Dari 1 sampai 5 seberapa tahukah kamu tata letak bangunan / gedung seperti kelas dan berbagai ruangan di SMP KP 2 Majalaya ?	<table border="1"> <tr> <td colspan="2">Tidak Tahu</td> <td colspan="3">Sangat Tahu</td> </tr> <tr> <td>1</td> <td>2</td> <td>3</td> <td>4</td> <td>5</td> </tr> </table>	Tidak Tahu		Sangat Tahu			1	2	3	4	5
Tidak Tahu		Sangat Tahu										
1	2	3	4	5								
2	Dari 1 sampai 5 seberapa tahukah kamu kelas yang ada di SMP KP 2 Majalaya?	<table border="1"> <tr> <td colspan="2">Tidak Tahu</td> <td colspan="3">Sangat Tahu</td> </tr> <tr> <td>1</td> <td>2</td> <td>3</td> <td>4</td> <td>5</td> </tr> </table>	Tidak Tahu		Sangat Tahu			1	2	3	4	5
Tidak Tahu		Sangat Tahu										
1	2	3	4	5								
B. Pertanyaan Bangunan, Ingat, pertanyaan ini tidak mengandung benar maupun salah !												
1	Dari 1 sampai 5 Seberapa mudahkan menurutmu tata letak gedung SMP KP 2 Majalaya ?	<table border="1"> <tr> <td colspan="2">Sulit</td> <td colspan="3">Sangat Mudah</td> </tr> <tr> <td>1</td> <td>2</td> <td>3</td> <td>4</td> <td>5</td> </tr> </table>	Sulit		Sangat Mudah			1	2	3	4	5
Sulit		Sangat Mudah										
1	2	3	4	5								
2	Dari 1 sampai 5 apakah kamu mampu menghafal tata letak bangunan SMP KP 2 Majalaya jika tidak dilihat secara langsung?	<table border="1"> <tr> <td colspan="2">Tidak Mampu</td> <td colspan="3">Sangat Mampu</td> </tr> <tr> <td>1</td> <td>2</td> <td>3</td> <td>4</td> <td>5</td> </tr> </table>	Tidak Mampu		Sangat Mampu			1	2	3	4	5
Tidak Mampu		Sangat Mampu										
1	2	3	4	5								

3	Dari 1 sampai 5 apakah kamu mengetahui Tempat administrasi ?	<div> <div>Tidak Tahu</div> <div>Sangat Tahu</div> <div>1</div> <div>2</div> <div>3</div> <div>4</div> <div>5</div> </div>
4	Dari 1 sampai 5 apakah kamu tahu Ruang Guru?	<div> <div>Tidak Tahu</div> <div>Sangat Tahu</div> <div>1</div> <div>2</div> <div>3</div> <div>4</div> <div>5</div> </div>
5	Dari 1 sampai 5 apakah kamu pernah memasuki Ruang Administrasi?	<div> <div>Tidak Pernah</div> <div>Lebih dari sekali</div> <div>1</div> <div>2</div> <div>3</div> <div>4</div> <div>5</div> </div>
6	Dari 1 sampai 5 apakah kamu pernah mengunjungi kelas selain kelas sendiri?	<div> <div>Tidak Pernah</div> <div>Lebih dari sekali</div> <div>1</div> <div>2</div> <div>3</div> <div>4</div> <div>5</div> </div>
7	Dari 1 Sampai 5 Mampukah kamu menghafal beberapa gedung dan bangunan seperti kelas, Ruang Administrasi, Ruang Guru tanpa melihat secara langsung ?	<div> <div>Tidak Mampu</div> <div>Sangat Mampu</div> <div>1</div> <div>2</div> <div>3</div> <div>4</div> <div>5</div> </div>
8	Dari 1 Sampai 5 Seberapa seringkah kamu melihat dan memasuki bangunan di SMP KP 2 Majalaya ?	<div> <div>Tidak Pernah</div> <div>Lebih dari sekali</div> <div>1</div> <div>2</div> <div>3</div> <div>4</div> <div>5</div> </div>

Lampiran - 2

Daftar Pertanyaan Wawancara Pendahuluan

1. Berapakah jumlah lantai di bangunan ini? dan difungsikan sebagai apa?
2. Apakah Gedung SMP Karya Pembangunan 2 Majalaya ini memiliki manajemen pengelolaan bangunan gedung?
3. Apakah gedung SMP Karya Pembangunan 2 Majalaya ini memiliki standar pemeliharaan bangunan (SOP)?
4. Apakah pengelola memiliki program kerja pemeliharaan bangunan gedung ?
5. Bagaimana pihak pengelola memonitoring program kerja pemeliharaan gedung SMP Karya Pembangunan 2 Majalaya?
6. Dimanakah tata letak bangunan yang di gunakan sebagai sarana Administrasi?
7. Dimanakah tata letak serta bangunan yang digunakan untuk Ruang Guru?
8. Dimanakah tata letak serta bangunan yang digunakan untuk Mushola?

Jawaban

1. Terdapat 2 lantai bangunan di SMP KP 2 Majalaya, tetapi masih ada bangunan yang hanya memiliki 1 lantai
2. Punya, seorang operator memiliki struktur manajemen pengelolaan bangunan.
3. Tidak terlalu di utamakan, tetapi dalam melakukan pemeliharaan selalu ada anggaran tiap pertahunnya.
4. Ada, setiap setahun sekali bangunan akan melakukan cat ulang dan penataan kembali.
5. Dalam memonitoring biasanya mereka akan berkomunikasi dengan pihak sekolah secara daring, tetapi biasanya akan ada yang datang survey ke lokasi pembangunan.
6. Terletak tepat di depan kedua pohon mangga.
7. Terletak tepat di bawah kelas.
8. Terletak tepat setelah pintu masuk dibuka

LEMBAR PENGESAHAN LEMBAGA

IMPLEMENTASI PETA 3 DIMENSI MENGGUNAKAN METODE IMSDD (*INTERACTIVE MULTIMEDIA SYSTEM DESIGN AND DEVELOPMENT*) DAN *WEBGL API* BERBASIS *WEB*

(Studi Kasus Di Smp Karya Pembangunan 2 Majalaya)

Disusun oleh:

TAUFIQ JAMIL HANAFI

NPM. 301180015

Telah diterima dan disetujui untuk memenuhi persyaratan mencapai gelar

SARJANA KOMPUTER

Pada

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS BALE BANDUNG**

Baleendah, Agustus 2022

Disetujui Oleh

Mengetahui,

Ketua Program Studi

Peneliti

Yusuf Muharam, S.Kom, M.Kom

NIK. 04104820003

Taufiq Jamil Hanafi

NPM. 301180015

Mengetahui,

Kepala SMP KP 2 Majalaya

Imas Ratna Sumirat, S.Pd

DAFTAR RIWAYAT HIDUP

Keterangan Diri :

Nama	: Taufiq Jamil Hanafi
NPM	: 301180015
Jurusan	: Teknik Informatika
Semester	: VIII (Delapan)
Tahun Ajaran	: 2018/2019
Jenis Kelamin	: Laki – laki
Agama	: Islam
Tempat/Tanggal Lahir	: Bandung, 27 Juli 1999
Kebangsaan/Suku	: Indonesia/Sunda
Alamat	: Kp Carik Rt 03 / Rw 07, Desa. Padamulya Kec.Majalaya, Kab.Bandung
Riwayat Pendidikan	: SD (2002 – 2012) SMP (2012 – 2014) SMK (2014 – 2017)
Riwayat Pekerjaan	: Staff Administrasi SMP KP 2 Majalaya (2017 - Saat ini)
Riwayat Organisasi	: Pramuka, Osis, Paskibra



Demikian daftar riwayat hidup ini dibuat dengan sebenar-benarnya.

Bandung, Agustus 2022
Yang Membuat,

Taufiq Jamil Hanafi
NPM. 301180015