



TUGAS AKHIR

**RANCANGAN DAN PEMBUATAN WEBSITE PSYCHIC
SEBAGAI SARANA TES PSIKOLOGI BERDASARKAN
METODE RIASEC OLEH Dr. JOHN HOLLAND
 MENGGUNAKAN LARAVEL 8**

Bidang keahlian :
PROGRAMMING

Pembimbing :
**1. AHMAD MUHIBUL FIRDAUS, M.Pd.
2. IQBAL ZAKARIYAH, M.T.**

SIDQIANA AZZAHRA
NIM : 2022.11.1056

PROGRAM STUDI INTENSIFIKASI TIK (PRODISTIK)
MA ZAINUL HASAN 1 GENGGONG
PAJARAKAN PROBOLINGGO
Bekerja sama dengan
UNIVERSITAS NEGERI MALANG
JAWA TIMUR
2025



TUGAS AKHIR

**RANCANGAN DAN PEMBUATAN WEBSITE PSYCHIC
SEBAGAI SARANA TES PSIKOLOGI BERDASARKAN
METODE RIASEC OLEH Dr. JOHN HOLLAND
 MENGGUNAKAN LARAVEL 8**

Bidang keahlian :
PROGRAMMING

Pembimbing :

1. AHMAD MUHIBUL FIRDAUS, M.Pd.
2. IQBAL ZAKARIYAH, M.T.

SIDQIANA AZZAHRA
NIM : 2022.11.1056

**PROGRAM STUDI INTENSIFIKASI TIK (PRODISTIK)
MA ZAINUL HASAN 1 GENGGONG
PAJARAKAN PROBOLINGGO**
Bekerja sama dengan
**UNIVERSITAS NEGERI MALANG
JAWA TIMUR
2025**

LEMBAR PERSETUJUAN

**RANCANGAN DAN PEMBUATAN WEBSITE PSYCHIC SEBAGAI
SARANA TES PSIKOLOGI BERDASARKAN METODE RIASEC OLEH
Dr. JOHN HOLLAND MENGGUNAKAN LARAVEL 8**

Laporan Tugas Akhir

Telah diperiksa dan disetujui untuk diuji :

Genggong, 19 Desember 2024

Pembimbing I

AHMAD MUHIBUL FIRDAUS, M.Pd.

Genggong, 19 Desember 2024

Pembimbing II

IQBAL ZAKARIYAH, M.T.

dan telah dipertahankan di depan penguji pada :

Genggong, 21 Desember 2024

(Penguji I)

(Penguji II)

LEMBAR PENGESAHAN

**RANCANGAN DAN PEMBUATAN WEBSITE PSYCHIC SEBAGAI
SARANA TES PSIKOLOGI BERDASARKAN METODE RIASEC OLEH
Dr. JOHN HOLLAND MENGGUNAKAN LARAVEL 8**

Laporan Tugas Akhir

Diajukan sebagai syarat untuk memperoleh sertifikat
Program Studi Intensifikasi Teknologi Informasi Dan Komunikasi
(PRODISTIK)
MA ZAINUL HASAN 1 GENGGONG
Bekerja sama dengan
UNIVERSITAS NEGERI MALANG

Oleh :

SIDQIANA AZZAHRA

NIM : 2022.11.1056

Disahkan di :

Genggong, 15 Januari 2025, oleh :

Ketua	Kepala
PRODISTIK MA Zainul Hasan 1	MA Zainul Hasan 1

<u>AHMAD MUHIBUL F., M.Pd.</u>	<u>NUN AHSAN MALIKI, S.Sy.M.Pd.</u>
NIP :	NIP :

MOTTO

Jangan pernah takut untuk mencoba hal – hal yang baik. Sebab perjalanan seribu mill pasti dimulai dari satu langkah. Tak perlu takut untuk jatuh. Maksimalkan doa dan ikhtiar, apapun hasilnya, itulah yang terbaik dari Tuhanmu.

وَعَسَىٰ أَن تَكُرْ هُوَ شَيْئًا وَهُوَ خَيْرٌ لَكُمْ وَعَسَىٰ أَن تُحِبُّوْ شَيْئًا وَهُوَ شَرٌ لَكُمْ وَاللَّهُ يَعْلَمُ وَأَنْتُمْ لَا تَعْلَمُونَ

“Boleh jadi kau membenci sesuatu, padahal itu baik bagimu, dan boleh jadi (pula) kau menyukai sesuatu, padahal ia amat buruk bagimu, Allah Maha Mengetahui, sedangkan kamu tidak mengetahui”. (QS. Al-Baqarah 2:216)

Ketika gagal menghampirimu, jangan pernah berhenti berjuang. Teruslah bangkit untuk maju, niscaya Tuhan mengangkat derajatmu.

وَلَا تَهُنُوا وَلَا تَحْزِنُوا وَأَنْتُمُ الْأَعْلَوْنَ إِنْ كُنْتُمْ مُؤْمِنِينَ

“Janganlah kamu merasa lemah, dan jangan (pula) bersedih hati, sebab derajatmu tinggi jika kamu orang yang beriman”. (QS. Ali-Imran 3:139)

PERSEMBAHAN

Tugas akhir dengan judul “*Rancangan dan Pembuatan Website ‘Psychic’ Sebagai Sarana Tes Psikologi Berdasarkan Metode RIASEC Oleh Dr. John Holland Menggunakan Laravel 8*” dipersembahkan untuk memenuhi syarat mendapatkan sertifikat PRODISTIK (Program Studi Intensifikasi Teknologi Informasi Dan Komunikasi) MA Zainul Hasan 1 Genggong.

Azzahra, S. 2024, “Rancangan dan Pembuatan Website ‘Psychic’ Sebagai Sarana Tes Psikologi Berdasarkan Metode RIASEC Oleh Dr. John Holland Menggunakan Laravel 8”, Tugas Akhir, Program Studi Intensifikasi Teknologi Informasi dan Komunikasi (PRODISTIK) Madrasah Aliyah Zainul Hasan 1 Genggong Probolinggo,
Pembimbing I : AHMAD MUHIBUL FIRDAUS, M.Pd.
Pembimbing II: IQBAL ZAKARIYAH, M.T.

ABSTRAK

Kehadiran teknologi dan internet seakan tidak terpisahkan dari gaya hidup sehari – hari. Hampir semua kalangan menggunakan internet untuk mendapatkan informasi. Hal ini mendorong generasi bangsa Indonesia untuk mendalami pengetahuan di bidang teknologi, salah satunya dengan mengembangkan inovasi melalui pembuatan website. Beberapa perangkat lunak yang dibutuhkan adalah *Laravel, HTML, CSS, JavaScript, dan MySQL*.

Website digunakan sebagai penerjemah kode *HTML* ke dalam bentuk visual berupa teks, gambar, video, bahkan suara. Informasi yang disampaikan melalui website bersifat bebas, namun lebih menarik jika berkaitan dengan permasalahan yang sering terjadi di dunia nyata. Misalnya, dengan membuat website *Psychic* berisi tes psikologi yang mengarahkan para pelajar SMA dalam memilih jurusan di perguruan tinggi sesuai *passion* masing – masing. Metode yang digunakan adalah *RIASEC* yang dikemukakan oleh Dr. John Holland.

Proses ini dimulai dengan membuat *wireframe* sebagai rancangan mengenai tampilan yang akan dibuat. Dilanjutkan membuat kode program berdasarkan *wireframe* yang telah dibuat. Sebelum website ini dipublikasikan, penulis perlu mengadakan uji coba terhadap beberapa pengguna untuk memastikan apakah kode program yang dibuat berjalan sebagaimana mestinya.

Kata Kunci : Website, *Psychic*, *Wireframe*, *RIASEC*, Dr. John Holland, *Laravel*

Azzahra, S. 2024, "*Design and Creation of the 'Psychic' Website as a Psychological Testing Tool Based on the RIASEC Method by Dr. John Holland Using Laravel 8*", Final Project, Information and Communication Technology Intensification Study Program (PRODISTIK) Madrasah Aliyah Zainul Hasan 1 Genggong Probolinggo,

Pembimbing I : AHMAD MUHIBUL FIRDAUS, M.Pd.

Pembimbing II : IQBAL ZAKARIYAH, M.T.

ABSTRACT

The presence of technology and the internet seems inseparable from everyday lifestyle. Almost all people use the internet to get information. This encourages the Indonesian generation to deepen their knowledge in the field of technology, one of which is by developing innovation through creating websites. Some of the software needed is Laravel, HTML, CSS, JavaScript, PHP, and *MySQL*.

Websites are used as translators of HTML code into visual form in the form of text, images, videos, and even sound. The information conveyed via the website is free, but more interesting if it relates to problems that often occur in the real world. For example, by creating a Psychic website containing psychological tests that guide high school students in choosing college majors according to their respective passions. The method used is RIASEC proposed by Dr. John Holland.

This process begins by creating a wireframe as a design for the display that will be created. Continue to create program code based on the wireframe that has been created. Before this website is published, the author needs to conduct trials on several users to ensure whether the program code created works as it should.

Kata Kunci : Website, *Psychic*, Wireframe, *RIASEC*, Dr. John Holland, Laravel

KATA PENGANTAR

Puji Syukur kehadirat Allah Subhanahu Wa Ta’ala Yang Maha Pengasih dan Penyayang, hingga tugas akhir ini dapat diselesaikan sesuai harapan. Begitu pula, sholawat dan salam semoga senantiasa dilimpahkan kepada Baginda Rasulullah Shalallahu ‘Alaihi Wa Sallam, pembawa cahaya kebenaran dan keimanan.

Sebelum seseorang terjun di bidang web developer, terdapat beberapa modal dasar yang harus dikuasai. Mulai dari *HTML*, *CSS*, hingga pemrograman ttingkat lanjut seperti *Laravel* dan sebagainya. Dalam laporan ini, penulis menyajikan pengenalan dasar dasar pemrograman tersebut sebelum menuju langkah – langkah pembuatan website. Namun, untuk memahami proses pembuatan website tidak cukup jika terpacu pada laporan ini saja. Pembaca perlu menelusuri modul pembelajaran lain atau bahkan video tutorial yang berkaitan.

Tentu saja, proses pembuatan serta penyusunan tugas akhir ini tidak akan berjalan dengan semestinya tanpa dukungan dari berbagai pihak. Maka dari itu, penulis hendak menyampaikan banyak terima kasih kepada:

1. Nun Ahsan Maliki, S.Sy, M.Pd. selaku Kepala Madrasah Aliyah Zainul Hasan 1 Genggong
2. Tim Universitas Negeri Malang (UM) selaku mitra kerjasama PRODISTIK MA Zainul Hasan 1 Genggong
3. Ustaz Ahmad Muhibul Firdaus, M.Pd. sebagai Ketua PRODISTIK MA Zainul Hasan 1 Genggong sekaligus dosen pembimbing yang begitu banyak menyalurkan ilmu dan motivasi hingga dapat menyelesaikan tugas akhir ini.
4. Ustaz Iqbal Zakariyah, M.T. selaku pengajar jurusan programming sekaligus dosen pembimbing yang banyak menuangkan ilmu serta ide pada masa pembelajaran berlangsung hingga proses pembuatan tugas akhir ini.
5. Ustaz Yanto, S.Pd.I,M.Si. selaku pengajar jurusan programming. Begitu banyak motivasi yang beliau salurkan serta pengorbanan, mulai dari tenaga, pikiran, waktu, dan sebagainya dikerahkan demi pembentukan karakter yang disiplin dan mahir di bidangnya.

6. Kedua orangtua saya, Bapak Rahmad Hartono dan Ibu Erva Kurnia Ulfa yang senantiasa memberikan doa dan dukungan terbaik, mendidik sepenuh hati, serta mengorbankan banyak hal demi kebaikan dan kesuksesan buah hatinya di masa depan.
7. Ustazah Najwan Nada, S.Sos. sebagai guru BK yang telah meluangkan waktu di tengah – tengah kesibukan untuk memberikan tanggapan terbaiknya mengenai tema tugas akhir ini sekaligus korektor pada soal psikologi yang digunakan di website ini
8. Teman – teman MA Zainul Hasan 1, baik kelas programming, kelas XII IPA 1, bahkan teman seperjuangan yang banyak mengukir sejarah dan lika liku yang bermakna, terutama dalam proses belajar pemrograman.
9. Teman – teman G7 dan G8 yang banyak memberi dukungan.

Penulis

Sidqiana Azzahra

DAFTAR ISI

LEMBAR PERSETUJUAN	i
LEMBAR PENGESAHAN	ii
MOTTO	iii
PERSEMBAHAN.....	iv
ABSTRAK.....	v
ABSTRACT	vi
KATA PENGANTAR	vii
DAFTAR ISI	ix
DAFTAR TABEL.....	xii
DAFTAR GAMBAR.....	xiii
BAB I PENDAHULUAN	1
1.1 Latar Belakang Masalah.....	1
1.2 Rumusan Masalah.....	3
1.3 Tujuan Penulisan.....	3
1.4 Manfaat Penulisan.....	3
1.5 Batasan Masalah	4
BAB II KAJIAN TEORI	6
2.1 Internet	6
2.2 Website	6
2.2.1 Pengertian	6
2.2.2 Jenis – Jenis Website	7
2.2.3 Unsur - Unsur Website.....	8
2.2.4 Modal Dasar Membuat Website	9
2.2.4 Visual Studio Code	10
2.4. Psikologis	11
2.3.1 Pengertian	11
2.3.2 <i>PSYCHIC</i>	12
2.3.3 Tes Psikologi.....	14
2.4. Metode <i>RIASEC</i>	15
2.4.1 Dr. John Holland	15

2.4.2 Realistic.....	16
2.4.3 Investigative	17
2.4.4 Artistic.....	17
2.4.5 Social.....	18
2.4.6 Enterprising.....	18
2.5.7 Conventional	18
2.5 Laravel	19
2.6 HTML.....	20
2.7 CSS	22
2.8 JavaScript.....	24
2.9 MySQL.....	24
BAB III PERENCANAAN DAN PERANCANGAN	27
3.1 Perencanaan.....	27
3.1.1 Bahan yang dibutuhkan.....	27
3.1.2 Hardware	27
3.1.3 Software	28
3.1.4 Wireframe	30
3.1.5 Jadwal Penulisan	39
3.1.6 Flowchart Produk.....	39
3.1.7 Flowchart Laporan	41
3.1.8 Flowchart Pengguna.....	42
3.2 Perancangan	43
3.2.1 Download dan Instalasi XAMPP	43
3.2.2 Download dan Instalasi Composer	45
3.2.3 Instalasi Laravel	47
3.2.4 Konfigurasi Database pada .env.....	49
3.2.5 Pembuatan Tabel Database	50
3.2.6 Pemasukan Kolom Database ke Folder App/Models	52

3.2.7 Publikasi File	54
3.2.8 Pembuatan Navigasi.....	55
3.2.9 Pembuatan Halaman Website	61
3.2.10 Rancangan Kode Perhitungan Skor	154
3.2.11 Ekspor Hasil ke PDF.....	160
3.2.12 Pengaturan Route	163
BAB IV UJI COBA DAN HASIL.....	1666
4.1 Uji Coba	1666
4.2. Hasil	1688
BAB V PENUTUP	18686
5.1 Kesimpulan	18686
5.6 Kritik dan Saran	1866
DAFTAR PUSTAKA	
BIOGRAFI PENULIS	

DAFTAR TABEL

Tabel 3.1.5.1 Jadwal Penulisan	39
--------------------------------------	----

DAFTAR GAMBAR

Gambar 2.3.2.1. Logo Psychic	12
Gambar 2.4.1.1. Dr. John Holland	15
Gambar 3.1.2.1. Laptop Lenovo	27
Gambar 3.1.2.2. Modem 4G	27
Gambar 3.1.2.3. Visual Studio Code	28
Gambar 3.1.2.4. MySQL dan phpMyAdmin	28
Gambar 3.1.2.5. Google Chrome	28
Gambar 3.1.2.6. Figma	29
Gambar 3.1.2.7. Microsoft Word	29
Gambar 3.1.2.8. Corel Draw 2019	29
Gambar 3.1.4.1. Wireframe Home	30
Gambar 3.1.4.2. UI/UX Home	30
Gambar 3.1.4.3. Wireframe Login	31
Gambar 3.1.4.4. UI/UX Login	31
Gambar 3.1.4.5. Wireframe Register	31
Gambar 3.1.4.6. UI/UX Register	31
Gambar 3.1.4.7. Wireframe Major	31
Gambar 3.1.4.8. UI/UX Major	31
Gambar 3.1.4.9. Wireframe Show Major	32
Gambar 3.1.4.10. UI/UX Show Major	32
Gambar 3.1.4.11. Wireframe Campus	32

Gambar 3.1.4.12. UI/UX Campus	32
Gambar 3.1.4.13. Wireframe Show Campus	33
Gambar 3.1.4.14. UI/UX Show Campus	33
Gambar 3.1.4.15. Wireframe Psych Test	34
Gambar 3.1.4.16. UI/UX Psych Test	34
Gambar 3.1.4.17. Wireframe Start Exam	34
Gambar 3.1.4.18. UI/UX Start Exam.....	34
Gambar 3.1.4.19. Wireframe Select Radio Button	35
Gambar 3.1.4.20. UI/UX Select Radio Button	35
Gambar 3.1.4.21. Wireframe Test Result	35
Gambar 3.1.4.22. UI/UX Test Result	35
Gambar 3.1.4.23. Wireframe History	36
Gambar 3.1.4.24. UI/UX History.....	36
Gambar 3.1.4.25. Wireframe Feedback	36
Gambar 3.1.4.26. UI/UX Feedback	36
Gambar 3.1.4.27. Wireframe RIASEC	37
Gambar 3.1.4.28. UI/UX RIASEC	37
Gambar 3.1.4.29. Wireframe Detailed Type.....	38
Gambar 3.1.4.30. UI/UX Detailed Type.....	38
Gambar 3.2.1.1. Download XAMPP	43
Gambar 3.2.1.2. Instalasi XAMPP.....	43
Gambar 3.2.1.3. Select Component XAMPP	44

Gambar 3.2.1.4. Folder Instalasi XAMPP	44
Gambar 3.2.1.5. Bitnami for XAMPP	45
Gambar 3.2.1.6. Start XAMPP	45
Gambar 3.2.2.1. Instalasi Composer.....	46
Gambar 3.2.2.2. Folder Instalasi Composer	46
Gambar 3.2.2.3. Ready to Install Composer.....	47
Gambar 3.2.2.4. Finishing Install Composer	47
Gambar 3.2.3.1. Buka Composer.....	48
Gambar 3.2.3.2. Instalasi Laravel	48
Gambar 3.2.3.3. Finishing Install Laravel	49
Gambar 3.2.4.1. Open Visual Studio Code.....	49
Gambar 3.2.4.2. Open Folder.....	50
Gambar 3.2.4.3. Pengaturan Root	50
Gambar 3.2.5.1. Penelusuran Database.....	51
Gambar 3.2.5.2. Create Database.....	51
Gambar 3.2.5.3. Create Tabel	52
Gambar 3.2.5.4. Simpan Tabel	52
Gambar 3.2.6.1. Buat file Model	53
Gambar 3.2.7.1. Copy file ke public	54
Gambar 3.2.7.2. Paste file di public	55
Gambar 3.2.8.1. New File Navigation	55
Gambar 3.2.8.1. Create Navigasi	61

Gambar 4.1.1. Tampilan Awal Layar	166
Gambar 4.1.2. Aktifkan XAMPP.....	167
Gambar 4.1.3. Buka Terminal.....	167
Gambar 4.1.4. Menjalankan Terminal	168
Gambar 4.1.5. Ketik ID Address	168
Gambar 4.2.1. Hasil Halaman Home Dekstop.....	169
Gambar 4.2.2. Tampilan Halaman Home Android	169
Gambar 4.2.3. Tampilan Halaman Login Dekstop	170
Gambar 4.2.4. Tampilan Halaman Login Android	170
Gambar 4.2.5. Tampilan Halaman Register Dekstop	171
Gambar 4.2.6. Tampilan Halaman Register Android	171
Gambar 4.2.7. Tampilan Halaman Profile Dekstop.....	172
Gambar 4.2.8. Tampilan Halaman Profile Android.....	172
Gambar 4.1.9. Tampilan Profile Update Dekstop.....	173
Gambar 4.2.10. Tampilan Profile Update Android.....	173
Gambar 4.2.11. Tampilan Halaman Campus Index Dekstop	174
Gambar 4.2.12. Tampilan Halaman Campus Index Android	174
Gambar 4.2.13. Tampilan Halaman Campus Show Dekstop	175
Gambar 4.2.14. Tampilan Halaman Campus Show Android	175
Gambar 4.2.15. Tampilan Halaman Jurusan Index Dekstop	176
Gambar 4.2.16. Tampilan Halaman Jurusan Index Android	176
Gambar 4.2.17. Tampilan Halaman Jurusan Show Dekstop	177

Gambar 4.2.18. Tampilan Halaman Jurusan Show Android	177
Gambar 4.2.19. Tampilan Halaman Contact Dekstop	178
Gambar 4.2.20. Tampilan Halaman Contact Android	178
Gambar 4.2.21. Tampilan Halaman Psych Test Dekstop	179
Gambar 4.2.22. Tampilan Halaman Psych Test Android	179
Gambar 4.2.23. Tampilan Halaman Tes Uji Dekstop.....	180
Gambar 4.2.24. Tampilan Halaman Tes Uji Android.....	180
Gambar 4.2.25. Tampilan Halaman History Dekstop.....	181
Gambar 4.2.26. Tampilan Halaman History Android.....	181
Gambar 4.2.27. Tampilan Footer Dekstop.....	182
Gambar 4.2.28. Tampilan Footer Android.....	182
Gambar 4.2.29. Contoh Data Jawaban User	183
Gambar 4.2.30. Contoh Data Skor.....	184
Gambar 4.2.31. Hasil Perhitungan Skor Secara Koding.....	185

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Kehadiran teknologi masa kini seakan tidak terlepas dari gaya hidup sehari-hari. Penggunaannya meliputi hampir semua kalangan, baik pelajar, pejabat, mahasiswa, pegawai, hingga masyarakat tingkat rendah pun membutuhkan manfaat dari teknologi. Kebutuhan semacam ini yang mendorong munculnya berbagai pemikiran dalam mengembangkan inovasi dari masa ke masa. Dimulai sejak ARPA (*Advanced Research Project Agency*)¹ mengembangkan inovasinya di tahun 1958 dengan menciptakan teknologi komputer. Pada tahun - tahun berikutnya, berbagai perkembangan melaju begitu pesat. Hal tersebut ditunjukkan dengan hadirnya penemuan berupa internet.

Kehadiran komputer tanpa internet bagi mobil tanpa bensin. Baik di kampus, sekolah, apalagi kantor pusat maupun daerah menggunakan internet sebagai media untuk mendapatkan informasi. Bahkan di sekolah, sudah diharuskan adanya internet untuk melangsungkan proses belajar mengajar dan evaluasi yang dilakukan secara online. Maka, sudah seharusnya pengetahuan mengenai internet dipelajari oleh semua kalangan di Indonesia, terutama pelajar. Apalagi di masa depan, Indonesia memasuki era MEA² yang memungkinkan orang – orang dari negara lain masuk ke wilayah Indonesia. Tentu hal ini merupakan peluang sekaigus tantangan bagi generasi bangsa. Orang yang mempelajari atau mengelola perkembangan teknologi pun tidak selamanya bergelar sarjana komputer. Bahkan mengawalinya dengan sekadar coba – coba jika ditekuni lambat laun juga pasti bisa menguasainya.

Bagi pemula yang baru saja terjun di bidang ini, mungkin belum memahami tentang website. Website merupakan salah satu perangkat lunak yang digunakan sebagai penerjemah kode HTML ke dalam bentuk visual sesuai teks, gambar,

¹ Departemen Pertahanan Amerika Serikat

² Masyarakat Ekonomi ASEAN

animasi, video, bahkan suara. Sementara kode program untuk mendesain suatu website disebut bahasa pemrograman. Sama halnya teknologi, bahasa pemrograman pun mengalami banyak perkembangan dan menghasilkan berbagai macam bahasa. Mulai dari Javascript, PHP, Python, C++, serta masih banyak lagi.

Informasi yang hendak disampaikan melalui website bersifat bebas. Namun, website akan lebih banyak diminati jika tema yang digunakan sesuai dengan permasalahan yang sering terjadi pada kehidupan nyata. Contoh, banyak pelajar SMA yang belum menemukan jati diri masing – masing. Belum memiliki cita – cita, atau bahkan cita – cita yang diimpikan tidak sesuai dengan *passion* yang dimiliki. Hasilnya, pemilihan jurusan tidak didasari oleh landasan yang tepat. Sesuatu yang dilakukan tanpa didasari minat yang kuat akan mati di tengah jalan. Hal ini dapat memicu turunnya kualitas dan produktivitas pendidikan Indonesia serta meningkatnya angka pengangguran. Menurut data Badan Pusat Statistik (BPS), TPT atau presentase jumlah pengangguran terhadap jumlah angkatan kerja di Indonesia mencapai 5,32 persen atau 7,86 juta pengangguran dari total 147,71 juta angkatan kerja³. Berdasarkan fakta tersebut, dibutuhkan adanya kesadaran masing – masing individu untuk menelusuri minat dan bakat yang dimiliki. Misalnya dengan mengadakan konsultasi kepada para ahli (psikolog atau konselor) agar menyediakan tes psikologis yang bertujuan memetakan minat bakat seseorang sesuai tipe kepribadian masing – masing. Salah satunya dengan menerapkan metode RIASEC⁴ yang dikemukakan oleh Dr. John Holland.

Dr. John Holland merupakan salah satu psikolog asal Amerika dan Profesor Emeritus Sosiologi di Universitas Johns Hopkins yang menciptakan model pengembangan karier, *Holland Occupational Themes*, yang dikenal sebagai *Holland Codes*. Melalui metode ini, seseorang akan dikelompokkan menuju 6 tipe kepribadian untuk memahami arah kecenderungan potensi yang dimiliki.

³ Nikita Rosa, *Angka Pengangguran di Indonesia Tinggi, Kemenko PMK Ungkap Penyebabnya*, detik.com Diakses Pada Sabtu, 14 September 2024

⁴ *Realistic, Investigative, Artistic, Social, Enterprising, dan Conventional*

Hanya saja, tidak semua orang memiliki kesempatan dalam menjalankan metode ini. Berbagai kendala mungkin terjadi, baik dari segi jarak, waktu, maupun tidak adanya tenaga ahli yang mampu menangani kondisi ini. Dari situlah peran teknologi dan komunikasi mulai dibutuhkan. Dalam proses pembuatan website, seorang programmer wajib menguasai dasar HTML⁵ dan CSS⁶. Bahkan, untuk menjadikan website menjadi lebih dinamis dan interaktif, programmer perlu mempelajari salah satu bahasa pemrograman seperti Javascript, PHP, atau lainnya. Kini pun tersedia *framework*⁷ PHP berupa *Laravel* yang menyediakan berbagai komponen yang siap pakai untuk memudahkan pengembangan aplikasi website.

Oleh karenanya, penulis membuat “Rancangan dan Pembuatan Website Psychic Sebagai Sarana Tes Psikologi Berdasarkan Metode RIASEC Oleh Dr. John Holland Menggunakan Laravel 8” supaya pengguna dapat memanfaatkan website ini di mana saja dan kapan saja.

1.2 Rumusan Masalah

1. Bagaimana proses “Rancangan dan Pembuatan Website *Psychic* Sebagai Sarana Tes Psikologi Berdasarkan Metode *RIASEC* Oleh Dr. John Holland Menggunakan Laravel 8”?

1.3 Tujuan Penulisan

Memahami proses “Rancangan dan Pembuatan Website *Psychic* Sebagai Sarana Tes Psikologi Berdasarkan Metode *RIASEC* Oleh Dr. John Holland Menggunakan Laravel 8”.

1.4 Manfaat Penulisan

Manfaat yang diharapkan dalam penulisan “Perancangan dan Pembuatan Website ‘PSYCHIC’ sebagai Sarana Tes Psikologi Berdasarkan Metode *RIASEC* Menggunakan *Laravel*, *HTML*, *CSS*, *Javascript*, dan *MySQL*” antara lain :

⁵ Hypertext Markup Language

⁶ Cascading Style Sheet

⁷ Framework adalah kerangka kerja atau struktur dasar yang digunakan oleh pengembang untuk membangun dan mengelola aplikasi. Framework menyediakan komponen, aturan, dan alat-alat yang telah siap pakai, sehingga pengembang tidak perlu membangun semuanya dari awal.

1. Membantu para pelajar khususnya tingkat SMA dalam menentukan jurusan yang sesuai dengan tipe kepribadian dan minat bakat
2. Memudahkan guru BK dalam mengarahkan siswa – siswinya menuju perguruan tinggi yang sesuai
3. Sebagai upaya meningkatkan kualitas pendidikan santri dan alumni MA Zainul Hasan 1 Genggong
4. Mendalami pengetahuan dan kreativitas dalam berpikir secara logika dan komputasional
5. Berbagi ilmu mengenai proses Perancangan dan Pembuatan Website ‘PSYCHIC’ Sebagai Sarana Tes Psikologi Berdasarkan Metode RIASEC Oleh Dr. John Holland Menggunakan Laravel 8
6. Sebagai salah satu syarat mendapatkan sertifikat Program Studi Intensifikasi TIK (PRODISTIK) MA Zainul Hasan 1 Genggong.

1.5 Batasan Masalah

Penulis memberikan batasan – batasan masalah dalam pembahasan sehingga tugas akhir ini sesuai dan terarah. Berikut batasan – batasan yang diberikan :

1. Website Psychic dibuat menggunakan *Laravel, HTML, CSS, Javascript, dan MySQL*.
2. Sasaran yang dituju hanya pelajar setingkat SMA yang ingin melanjutkan studi ke perguruan tinggi.
3. Hanya berisi 120 soal tes psikologi yang ditetapkan untuk mengarahkan pengguna ke beberapa jurusan yang sesuai.
4. Terdapat beberapa fitur penunjang yang memudahkan pengguna dalam menelusuri informasi terkait kampus dan jurusan di Indonesia.
5. Website ini belum mencapai tahap sempurna, sehingga perlu adanya pengembangan di kemudian hari.
6. Metode yang digunakan adalah RIASEC yang dikemukakan oleh Dr. John Holland.
7. Desain logo *Psychic* dibuat menggunakan aplikasi CorelDraw 2019.

8. *Wireframe* (Sketsa tampilan) website ini dibuat menggunakan Figma.
9. Pengguna website membutuhkan jaringan internet.

BAB II

KAJIAN TEORI

2.1 Internet

Internet merupakan singkatan dari *Interconnected Network* yang berfungsi sebagai jaringan komunikasi global yang menghubungkan jaringan suatu komputer dengan komputer lain di seluruh dunia. Menurut KBBI, internet adalah jaringan komunikasi elektronik yang menghubungkan jaringan komputer yang terorganisasi di seluruh dunia melalui telepon atau satelit⁸.

Perkembangan internet dimulai pada tahun 1958 sejak ARPA dari Amerika Serikat membentuk sebuah jaringan internet pertama yang kemudian diberi nama ARPANET. Penemuan inilah yang menjadi cikal bakal pemikiran ide inovatif dalam pengembangannya. Jika awal penemuan internet hanya berbasis teks, sejak tahun 1990 layanan sejenis tampilan berbasis grafis atau biasa dikenal dengan *www* (*World Wide Web*) mulai dikembangkan. Kemudian dilanjutkan pada tahun 1993 interNIC didirikan untuk menjalankan layanan pendaftaran domain, hingga muncul aturan nama domain dan Alamat IP berupa simbol numerik dan format tertentu sebagai pengenal system komputer.

2.2 Website

Website adalah salah satu media yang paling sering untuk diakses dan digunakan dalam mencari berbagai informasi dan sarana komunikasi. Faktor utamanya terjadi akibat perkembangan internet dari tahun ke tahun mengalami peningkatan dari sisi pengguna aktifnya. Berikut penjelasan mengenai pengertian, jenis – jenis, serta hal – hal yang harus disiapkan sebelum mempelajari cara membuat website.

2.2.1 Pengertian

Website adalah sekumpulan halaman atau lokasi berisi informasi yang disimpan dan disajikan melalui internet oleh siapapun secara online.

⁸ CNBC Indonesia, *Mengenal Apa itu Internet, Sejarah Perkembangan dan Manfaatnya*. 21 September 2022. Diakses pada Minggu, 15 September 2024.

Biasanya untuk tampilan awal dapat diakses melalui halaman utama (*homepage*) menggunakan browser dengan menuliskan URL yang tepat. Di dalam sebuah homepage, juga memuat beberapa halaman web turunan yang saling terhubung satu dengan yang lain⁹.

Website pertama kali di dunia dicetuskan oleh ilmuan asal Inggris, Tim Berners Lee pada akhir 1980 dalam projek *www (World Wide Web)*. Situs ini resmi diluncurkan secara online pada 6 Agustus 1991 dengan URL <http://info.cern.ch>.

2.2.2 Jenis – Jenis Website

Berdasarkan pengembangannya, jenis website dibedakan menjadi tiga, antara lain website statis, website dinamis, dan website interaktif.

- Website statis merupakan website yang memiliki tampilan yang tetap dan tidak banyak mengalami perubahan. Biasanya untuk perubahannya sendiri hanya terletak pada tampilan desain halaman web saja, terkait konten tidak mengalami perubahan yang besar. Contoh dari web statis yaitu yang menampilkan profil perusahaan atau organisasi.
- Website dinamis adalah website yang mengalami perubahan secara terus menerus sesuai dengan kebutuhan dan relevansi dari bisnis dan perkembangan zaman. Website dinamis memiliki tampilan yang lebih interaktif, dan menyediakan fitur kolom komentar, dan chatting. Contoh dari situs web ini adalah blog, situs berita online, *e-commerce*, sistem informasi, dan lain sebagainya.
- Website interaktif adalah website yang dirancang untuk dapat saling berinteraksi antar penggunanya. Jenis situs ini biasanya tergolong ke dalam platform media sosial seperti Facebook, Twitter, Instagram, dan platform *social media* yang lain.

⁹ Muhammad Robith Adani, *Pengertian Website, Fungsi, Manfaat, dan Cara Membuatnya*, Sekawan Media. Diakses pada Minggu, 15 September 2024

2.2.3 Unsur - Unsur Website

Berikut penjelasan mengenai lima unsur yang melekat pada website. Tanpa unsur tersebut situs web tidak akan bisa ditemukan oleh pengguna.

- **Domain** merupakan alamat sebuah website yang tercantum dalam sebuah tautan. Dahulu, untuk mengunjungi situs web pengguna wajib mengetahui alamat IP terlebih dahulu yang ditandai dengan deretan angka. Namun, hal ini sangat sulit diingat. Sehingga, muncullah penamaan alamat website semacam <http://psychic.com> atau sebagainya. Dalam pembuatan domain, sebaiknya tidak terlalu panjang agar mudah diingat oleh pengguna. Penulisan domain yang menarik dapat meningkatkan keinginan seseorang untuk mengunjungi website terkait.
- **Hosting** adalah server yang berfungsi menyimpan segala bentuk *database* maupun file website sehingga dapat dikelola dan diakses melalui internet. Hosting dapat dianalogikan sebagai memori berisi informasi semacam gambar, video, teks, bahkan audio. Salah satu contoh server penyedia hosting website adalah MySQL.
- **Konten** merupakan isi yang memuat informasi di dalam website. Dengan adanya konten, pengguna dapat mengetahui informasi yang ada di website tersebut. Sebagai contoh, website *Psychic* ditujukan sebagai tes minat bakat dan informasi seputar perguruan tinggi di Indonesia. Jadi, segala penjelasan mengenai perguruan tinggi itulah yang disebut dengan konten. Mulai dari lokasi kampus, deskripsi, dan sebagainya.
- **Bahasa Pemrograman** adalah sebuah instruksi standar untuk memerintah komputer agar menjalankan fungsi tertentu. Bahasa pemrograman ini merupakan suatu himpunan dari aturan sintaksis dan semantik yang dipakai untuk mengolah kode program. Bahasa ini memungkinkan programmer dapat menentukan secara persis dan sistematis, data mana yang akan diolah, bagaimana data diteruskan,

dan langkah apa yang harus diambil dalam berbagai situasi¹⁰. Dengan kata lain, programmer menuangkan pemikiran logikanya atau algoritma sesuai *syntax* yang ditentukan pada masing – masing bahasa. Beberapa Bahasa pemrograman yang cukup populer dan sering digunakan seperti JavaScript, PHP, C++, Python, Pascal, dan lain – lain.

- **Tampilan** dapat diumpamakan sebagai *style* yang digunakan dalam konten website. Informasi dalam sebuah website memang penting, namun tampilan yang menarik juga tidak kalah penting dan diimpikan oleh sebagian besar pengguna. Tampilan ini dapat diatur melalui CSS. Tampilan yang elegan akan menambah daya tarik pengguna selama mengunjungi website tersebut.

2.2.4 Modal Dasar Membuat Website

Seorang yang ahli dalam dunia teknologi khususnya web developer harus menguasai beberapa hal yang berkaitan dengan proses pembuatannya, yaitu:

- Menguasai salah satu system operasi, misalnya *windows* atau *linux*. Bedanya, sistem operasi *linux* lebih lanjut keahliannya, sehingga otomatis lebih terjaga pula keamanan datanya.
- Mengetahui beberapa istilah penting yang berkaitan dengan internet, seperti *web browser*, *web server*, *hosting*, *domain*, dan lain sebagainya.
- Menguasai salah satu Bahasa pemrograman komputer, khususnya Bahasa yang sering digunakan pada website seperti *PHP*, *JavaScript*, *ASP*, atau lainnya.
- Menguasai salah satu aplikasi *database*, misalnya *MySQL*, *Laravel*, *Ms.Access*, *Oracle*, atau yang lainnya. Bagian ini akan dibutuhkan jika situs web yang akan dibuat bersifat dinamis.
- Memahami salah satu aplikasi desain grafis dengan tujuan mempermudah programmer untuk membuat rancangan website

¹⁰ Wikipedia, *Bahasa Pemrograman*. Diakses pada Minggu, 15 September 2024

dalam bentuk sketsa, seperti *CorelDraw*, *AdobePhotoshop*, *MacromediaFireworks*, *Figma*, atau semacamnya.

- Menguasai salah satu aplikasi editor web yang digunakan untuk menulis script Bahasa pemrograman. Aplikasi editor secara umum dibedakan menjadi dua, yakni *hand code* dan *What you see is what you get (WYSIWYG)*. Contoh aplikasi *hand code* adalah *Notepad* dan *Visual Studio Code* karena editor ini tidak bisa secara langsung menampilkan wujud nyatanya. Sedangkan editor (*WYSIWYG*) contohnya *Adobe Dreamweaver*¹¹ yang dapat langsung melihat hasilnya.

2.2.4 Visual Studio Code

Visual Studio Code, yang merupakan aplikasi *code editor* untuk membantu proses pengembangan sebuah aplikasi. Software ini dikembangkan oleh perusahaan teknologi raksasa ternama, Microsoft. Meskipun begitu, VSCODE mendukung untuk dioperasikan pada perangkat selain Windows, seperti Linux dan Mac OS.

Dengan menggunakan *code editor* satu ini, kamu dapat membuat maupun mengedit *source code* dengan berbagai bahasa pemrograman, seperti Node.Js, JavaScript, dan sebagainya. VSCODE bahkan juga mendukung bahasa lain seperti Python, Java, PHP, dan .NET. Hal ini memungkinkan sebab VSCODE mempunyai ekosistem luas dengan banyak extension.

Fitur utama dari Visual Code Studio mencakup berbagai alat dan fitur basic editing. Dalam fitur ini, kamu bisa menemukan berbagai tools seperti Multiple Selection, Column Selection, hingga Keyboard Shortcuts. Selain tools yang sudah disebutkan, VSCODE juga dilengkapi dengan fitur Auto Save serta Hot Exit.

Kedua fitur ini memungkinkan kamu untuk mengedit tanpa takut lupa menyimpan perubahan terakhir pada file. Pekerjaanmu di VSCODE akan

¹¹ Editor web yang menyediakan fitur di mana pengguna bisa mendesain layout secara visual tanpa menulis code, namun hasilnya diterjemahkan menjadi kode di latar belakang.

tersimpan otomatis tanpa perlu kamu simpan secara manual. Dengan begitu, kamu tidak perlu khawatir jika perangkat komputer atau laptop mengalami gangguan di tengah-tengah proses editing.

2.4. Psikologis

Psikologi adalah suatu kondisi yang amat berkaitan dengan Kesehatan mental manusia. Kondisi psikologi meliputi sumber kendali diri, pola pikir, keyakinan, dan orientasi tujuan. Dengan kata lain, psikologi merupakan landasan kepribadian pada masing – masing individu.

2.3.1 Pengertian

Berikut beberapa pengertian mengenai psikologi menurut para ahli:

- **Plato dan Aristoteles**

psikologi adalah ilmu pengetahuan yang mempelajari tentang hakikat jiwa dan mempelajari proses dari awal hingga akhir¹².

- **Wilhem Wundt**

psikologi adalah ilmu yang tidak hanya mempelajari tentang hakikat jiwa saja, tetapi juga mempelajari tentang pengalaman-pengalaman yang dirasakan dan yang timbul. Nah, bisa jadi kamu juga merasakan hal yang sama. Misalnya kamu menciptakan perasaan cinta, munculnya rasa benci dan semacamnya. Atau bentuk perasaan lain yang ditimbulkan, seperti keinginan, perasaan panca indra dan pikiran-pikiran¹⁸.

- **Woodworth dan Marquis**

Psikologi sebagai ilmu yang mempelajari aktivitas individu sejak dalam kandungan hingga akhir hayat seseorang. Tidak berhenti di sini saja, woodworth dan Marquis juga akan mengulas tentang aktivitas yang ada dalam hubungan dengan alam disekeliling juga. Jadi ilmu psikologi tidak hanya mempelajari

¹² Deepublish Store, *14 Pengertian Psikologi Menurut Para Ahli di Dunia*, Diakses pada Senin, 16 September 2024

sepenuhnya tentang manusia saja, tetapi sekeliling termasuk alam semesta pun termasuk di sini¹⁸.

- **Knight**

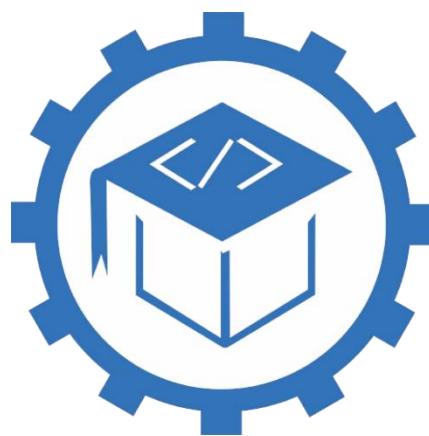
psikologi sebagai ilmu yang mendalam pengalaman dan tingkah laku manusia secara tersistematis. Baik mendalam secara normal hingga secara abnormal.

- **Dr. Singgih Dirgagunarsa**

psikologi sebagai ilmu yang mempelajari tingkah laku manusia.

2.3.2. PSYCHIC

PSYCHIC merupakan singkatan dari *Psychic*¹³, *Search Your Choice College*. Secara sederhana, website *PSYCHIC* diartikan sebagai sarana tes psikologi yang bertujuan mengarahkan para pengguna sesuai kepribadian masing-masing serta beberapa rekomendasi jurusan yang tepat. Website ini juga menyediakan informasi singkat mengenai kampus dan jurusan di Indonesia, sehingga dapat menentukan kampus maupun jurusan pilihannya. Di samping itu, terdapat ‘logo *Psychic*’ yang dirancang menggunakan aplikasi CorelDraw 2019. Berikut gambar beserta filosofinya.



Gambar 2.3.2.1. Logo Psychic

¹³ Dalam Bahasa Inggris, *Psychic* memiliki arti yang sama dengan psikologi

- **Warna Biru** memancarkan sinar ketenangan dan lingkungan yang nyaman ketika belajar, menciptakan suasana kondusif dalam konsentrasi berpikir. Hal ini sesuai dengan website *PSYCHIC* yang menunjang perkembangan di bidang Pendidikan. Di samping itu, dalam dunia teknologi warna biru juga sering digunakan sebagai *platform e-learning*. Warna ini memberi kesan modernitas dan inovasi dalam kemajuan teknologi.
- **Roga Gerigi** menunjukkan adanya mekanisme atau proses dalam mengolah jawaban berdasarkan tes psikologi hingga menghasilkan hasil yang akurat dalam menentukan tipe kepribadian seseorang. Dalam konteks tes RIASEC, lambang ini juga bisa mengisyaratkan penggalian potensi dari sistem psikologi kepribadian dalam membantu peserta tes menemukan jurusan yang sesuai. Ini menunjukkan bahwa ada mekanisme atau struktur yang mendukung dalam membangun kesuksesan di masa depan.
- **Topi Toga** melambangkan target pencapaian akademik dalam dunia Pendidikan sebagai harapan utama pembuatan website *PSYCHIC* dengan memilih jurusan yang tepat sesuai tipe kepribadian masing – masing. Topi toga juga memiliki nilai tradisional yang dalam, mewakili sejarah dan warisan pendidikan yang dihormati. Ini menjadi bagian dari seremonial yang menegaskan pentingnya pendidikan dalam kehidupan seseorang.
- **Simbol “</>”** sebagai tanda penutup tag HTML yang menjadi materi dasar bagi para developer. Dengan kata lain, simbol tersebut menunjukkan bahwa pembuatan website ini dikembangkan oleh kelas programming yang diadakan program PRODISTIK MA Zainul Hasan 1 Genggong.
- **Huruf P** merupakan inisial huruf pertama dari website ini, yaitu *PSYCHIC* sekaligus huruf pertama dari kata *Programming*.
- **Huruf Psi** dalam Bahasa Yunani memiliki arti jiwa, pikiran, atau kehidupan mental. Maksudnya, website ini merupakan sarana tes

psikologi bertujuan memberikan rekomendasi jurusan sesuai pola pikir atau minat yang berbeda pada setiap pengguna.

2.3.3 Tes Psikologi

Tes psikologi adalah prosedur yang digunakan untuk mengukur dan mengevaluasi berbagai aspek psikologi seseorang, seperti kognisi, emosi, kepribadian, sikap, perilaku, dan fungsi mental lainnya. Tujuan dari tes ini adalah untuk mendapatkan informasi tentang karakteristik dan potensi psikologis seseorang sehingga mereka dapat lebih memahami diri mereka lebih baik¹⁴.

Tes psikologi digunakan dalam berbagai konteks, seperti konseling, penilaian klinis, pendidikan, seleksi karyawan, dan penelitian. Tes ini dirancang dengan standar ilmiah yang ketat untuk memastikan bahwa hasilnya benar, valid, dan akurat. Beberapa jenis tes psikologi sebagai berikut:

- **Tes Kepribadian** untuk mengevaluasi ssifat kepribadian seseorang
- **Tes Psikopatologi** untuk menentukan apakah ada gangguan jiwa atau kesehatan mental
- **Tes Kecerdasan Emosional** untuk mengevaluasi kemampuan seseorang dalam mengidentifikasi, memahami, dan mengelola emosional
- **Tes Minat Bakat** untuk menentukan kecenderungan potensi dan minat seseorang dalam menentukan jurusan atau karir yang sesuai dengan kepribadian
- **Tes Kemampuan Sosial** untuk mengevaluasi kemampuan dan keterampilan seseorang seperti kemampuan berempati atau komunikasi
- **Tes Proyektif** untuk mengungkapkan aspek tak sadar dari pikiran dan kepribadian seseorang melalui penggunaan gambar atau stimulan lainnya.

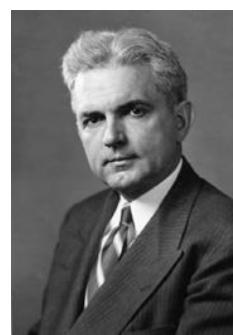
¹⁴ Sampoerna Academy, *Mengenal 10 Jenis Tes Psikologi, Bukan Cuma Tes Kepribadian!*, 19 Agustus 2023. Diakses pada Senin, 16 September 2024

- **Tes Intelektual (IQ)** untuk penilaian tingkat kemampuan kognitif seseorang yang mencakup kemampuan verbal, numerik, dan spasial.
- **Tes Perilaku dan Sikap** untuk mengevaluasi sikap dan perilaku individu dalam situasi tertentu
- **Tes Kesehatan Mental** menangani tes secara keseluruhan, termasuk tingkat depresi dan kecemasan.
- **Tes Stres dan Koping** menangani bagaimana seseorang menangani stress dan menghadapi masalah kehidupan

2.4. Metode RIASEC

RIASEC merupakan tes psikologi minat bakat yang dikemukaan oleh Dr. John Holland, seorang psikolog Amerika dan Profesor Emeritus Sosiologi di Universitas Johns Hopkins yang menciptakan model pengembangan karier, *Holland Occupational Themes*, yang dikenal sebagai *Holland Codes*. Melalui metode ini, seseorang akan dikelompokkan menuju 6 tipe kepribadian untuk memahami arah kecenderungan potensi yang dimiliki. Dengan demikian, nilai produktivitas akan lebih efektif jika orang tersebut lebih menitikberatkan pengembangan dan pendalaman terhadap minat dan bakat yang sesuai. *RIASEC* yang merupakan singkatan dari *Realistic, Investigative, Artistic, Social, Enterprising, dan Conventional*.

2.4.1 Dr. John Holland



Gambar 2.4.1.1.Dr. John Holland

Dr. John Holland merupakan seorang psikolog terkenal kelahiran Obama pada tahun 1919. Penemuan teorinya mengenai Tes Psikologi RIASEC amat berpengaruh di berbagai belahan dunia, khususnya pada abad ke 20. Sejak kecil, ia

dibesarkan di dalam hati- tanah Amerika, bersama dengan dua saudara laki-laki dan saudara perempuan, dipandu oleh sekolah ibu guru dan ayah eksekutif periklanan, yang keduanya “memiliki ditandai minat intelektual “American Psychological Association, 1995. Dia tinggal di rumah untuk kuliah, menghadiri Universitas Kota Omaha atau University of Omaha, jurusan psikologi, matematika, dan Perancis dan lulus pada tahun 1942. Seperti banyak pria muda di tahun 1940-an, dia masuk Angkatan Darat setelah kuliah. Mungkin karena ia mengambil jurusan psikologi, Swasta Karier militer Belanda termasuk posisi sebagai pewawancara klasifikasi, penguji tes, asisten psikologis, dan *Wechsler Intelligence Test administrator*. Pada tahun 1966, Holland berpendapat bahwa masing - masing orang yang berada di lingkungan realistik, intelektual, artistik, sosial, atau konvensional mayoritas memiliki tipe kepribadian yang sama. Orang yang memiliki minat berbeda dan bekerja pada lingkungan yang berbeda adalah orang yang memiliki sejarah hidup yang berbeda pula. Sehingga, pilihan karir ialah suatu ekspresi atau suatu perluasan kepribadian dalam dunia kerja yang diikuti oleh identifikasi berikutnya dengan stereotipe pekerjaan yang spesifik. Perbandingan antara diri dengan persepsi terhadap suatu pekerjaan dan penerimaan atau penolakan adalah penentu utama dalam pilihan karir. Keseuaian antara tinjauan diri seseorang dengan penetapan pemilihan pekerjaan ialah berhubungan dengan model gaya pribadi. Hal ini memicu Holland untuk mengembangkan pikirannya dalam mengklasifikasi tipe kepribadian seseorang menjadi enam bagian sebagai berikut.

2.4.2 Realistic

Seseorang dengan kepribadian *realistic* umumnya menyukai bentuk pekerjaan bersifat praktis dan langsung, senang melakukan aktivitas yang melibatkan mesin, benda, hewan, dan yang melibatkan gerak serta kekuatan. Orang pada tipe ini sebaiknya diarahkan untuk mengeksplor aktivitas dengan pendekatan STEAM agar dapat menciptakan sebuah produk.

Berdasarkan tes RIASEC, bidang yang cocok untuk tipe kepribadian *realistic* yaitu pertanian, arsitektur, seni, kesehatan, pariwisata, pilot hukum, industri, dan sains.

2.4.3 Investigative

Seseorang pada kategori *investigative* cenderung menyukai ilmu pengetahuan. Mereka sangat suka mencari fakta dan jawaban dari suatu permasalahan. Cenderung kritis dan memiliki rasa ingin tahu yang tinggi tentang hal yang abstrak, senang belajar, dan memecahkan masalah yang berhubungan dengan matematika atau sains.

Seseorang yang *investigative* memiliki kemampuan *problem solving*, analisis, dan penalaran yang cenderung lebih tinggi, sehingga cocok untuk diarahkan pada aktivitas membaca buku atau menonton film bertema sains atau misteri, melakukan eksperimen sains, mengeksplor kegiatan *mind sports* seperti catur dan *bridge* untuk dapat mendorong bakat dan minatnya

Berdasarkan tes RIASEC, bidang yang cocok untuk tipe kepribadian *Investigative* yaitu sebagai peneliti, *software engineer* atau *programmer*, ilmuwan, dan akademisi.

2.4.4 Artistic

Seseorang pada tipe *artistic* cenderung menyukai proses berimajinasi, memikirkan ide-ide yang kreatif, inovatif, beda dari yang lain, dan bisa melihat sesuatu dari berbagai perspektif. Mereka mengeksplorasi rasa ingin tahu mereka dengan cara yang sangat ekspresif menggunakan berbagai media dalam kegiatannya dan lebih menyukai kegiatan yang berkaitan dengan seni. Sebaiknya diarahkan untuk mengeksplorasi aktivitas yang berkaitan dengan seni, seperti:

1. Melukis
2. Menyanyi atau menari
3. Mengeksplorasi berbagai alat musik
4. Membuat kerajinan tangan
5. Mengeksplor *digital design*

Berdasarkan tes RIASEC, bidang yang cocok untuk tipe kepribadian *Artistic* yaitu *designer*, penulis, aktor, dan musisi.

2.4.5 Social

Seseorang pada kategori *social* cenderung senang berinteraksi dengan orang-orang di sekitarnya, suka melakukan sesuatu yang membantu orang lain, seperti mengajar, merawat, memberikan pertolongan, dan memberikan informasi. Mereka cenderung memiliki kepribadian yang peduli, bertanggung jawab, dan senang berkolaborasi untuk mengerjakan sesuatu, lebih senang melakukan aktivitas yang melibatkan orang lain.

Berdasarkan tes RIASEC, bidang yang cocok untuk tipe kepribadian *Social* yaitu pendidik, perawat, konselor, dan pekerja sosial.

2.4.6 Enterprising

Seseorang pada kategori ini terlihat memiliki jiwa pemimpin dan pandai memengaruhi pola pikir orang lain, senang berbicara dan memberikan ide-idenya, mudah bergaul dan pandai berkomunikasi secara menarik, memiliki sikap yang energik, percaya diri, dan optimis.

Berdasarkan tes RIASEC, bidang yang cocok untuk tipe kepribadian *Enterprising* yaitu *Business* dan *management, sales, marketing*, dan politisi.

2.5.7 Conventional

Seseorang pada kategori ini sangat terstruktur, patuh pada aturan, dan sangat suka mengorganisir segala sesuatu, berorientasi pada detail dan *well-organized*, lebih mudah mengerjakan aktivitas yang membutuhkan ketelitian tinggi daripada anak lainnya, menyukai angka yang tertulis secara sistematis dan teratur. Sebaiknya diarahkan pada aktivitas Belajar dengan metode mencatat, melakukan aktivitas yang melibatkan angka, dan mengeksplor kode-kode pemrograman.

Berdasarkan tes RIASEC, bidang yang cocok untuk tipe kepribadian *Conventional* yaitu *Akuntan, Administration, Programmer* dan *Banker*.

2.5 Laravel

Laravel merupakan suatu *platform* PHP yang digunakan sebagai kerangka kerja dalam membangun suatu website. Pada proses pengembangan aplikasi web, terdapat beberapa alat bantu yang disebut *framework*. Salah satu dari banyaknya *framework* PHP yang populer ialah *Laravel*. *Laravel* dirilis oleh Taylor Otwell pada tahun 2011 sebagai proyek pribadi dengan tujuan menyempurnakan *framework* sebelumnya. Laravel termasuk *framework* PHP opensource yang mudah dipahami, hal ini mengikuti pola desain model-view-controller. Laravel bertujuan untuk membuat proses pengembangan lebih menyenangkan bagi pengembang tanpa mengorbankan fungsionalitas aplikasi. Laravel memiliki serangkaian fitur yang bervariasi dan dapat meningkatkan kecepatan pengembangan suatu website. Fitur yang disediakan antara lain:

- ***MVC Architecture***

Fitur ini digunakan sebagai fitur yang paling signifikan untuk membuat halaman website jadi lebih menarik dan efisien. Fitur bawaan ini juga dapat meningkatkan kemanan skalabilitas aplikasi, memungkinkan proses penggunaan dan kinerja yang lancar.

- ***Dynamic Templates***

Melalui fitur ini, programmer dapat memanfaatkan template dinamis untuk merancang tata letak lebih memuaskan, mengintegrasikan CSS dengan menggabungkan kerangka kerja JavaScript yang kuat. Dengan fitur ini, programmer dapat menghasilkan halaman web dan aplikasi web secara konsisten supaya memberikan hasil sesuai yang diinginkan.

- ***Object Relation Mapping (ORM)***

Fitur ini memiliki fungsi sederhana namun penting dalam pengimplementasian kode aktif PHP. ORM mendorong programmer untuk menulis query database dengan cepat dan mudah menggunakan *syntax* PHP. Laravel melakukan ORM lebih cepat dan lancar dibandingkan kerangka PHP lainnya. Hal ini memudahkan bahwa fitur ini juga sangat menguntungkan karena waktu pengembangan yang lebih cepat.

- ***Modular Library***

Laravel menggabungkan perpustakaan berorientasi objek untuk kerangka ini dan tidak ditemukan dalam kerangka PHP lainnya. Fitur ini menawarkan berbagai fitur dan fungsi yang mudah digunakan saat mengembangkan halaman website. Jumlah pengguna aktif, pemalsuan permintaan lintas situs, enkripsi dan pengaturan ulang kata sandi adalah semua fungsi yang dapat diperiksa kembali.

- ***Data Migration***

Fitur bawaan Laravel berikut ini dapat dimanfaatkan untuk mengubah struktur database secara efektif menggunakan kode PHP, dikarenakan migrasi data dari satu program ke program lainnya merupakan persyaratan umum bagi seorang developer. Dengan fitur ini, programmer dapat melakukan migrasi ulang database. Berkait pembuat skema Laravel. Seluruh proses migrasi dilakukan secara otomatis yang memungkinkan programmer untuk memigrasikan data tanpa kesulitan atau kekhawatiran akan kehilangan data.

- ***Integrated Tools***

Dengan menggunakan fitur ini, programmer dapat menghindari proses pemrograman yang memakan banyak waktu. Fitur ini bertujuan menyederhanakan tugas-tugas sulit, seperti memelihara beberapa sistem database. Melalui *command line*, pengembang juga dapat membuat file MVC dan mengelola konfigurasi asset yang diinginkan.

- ***Unit Testing***

Laravel memiliki fitur testing dan mampu menjalankan beberapa pengujian unit, menjamin bahwa semua update diimplementasikan dengan benar tanpa mengubah kode, Unit PHP dan pengujian otomatis didukung oleh Laravel¹⁵.

2.6 HTML

HTML (Hypertext Markup Language) merupakan bahasa markup standar untuk membuat interkoneksi data atau dokumen dalam bentuk *hypertext* secara

¹⁵ @admin_official, *Laravel: Definisi, Fitur, Manfaat, Cara Kerja, Keunggulan, dan Kekurangan*. Telkom University Jakarta. Diakses pada Senin, 16 September 2024

elektronik. Dokumen HTML disimpan dalam format teks regular dan mengandung *tag – tag* yang memerintah *web browser* untuk menampilkan perintah dengan spesifik. Susunan *HTML* terdiri dari *heading*, *main atau article*, dan *footer*. Walaupun susunannya seperti *coding*, *HTML* tidak termasuk dalam bahasa pemrograman, melainkan Bahasa markup standar sebagai penyusun kerangka pada halaman website. Hal itu disebabkan *HTML* tidak mampu memberikan fungsi yang dinamis.

Sama halnya dengan internet, *HTML* juga awal ditemukan oleh Tim Berners-Lee, seorang ahli fisika di Lembaga Penelitian CERN yang berlokasi di Swiss. Versi pertamanya di tahun 1991 *HTML* hanya memiliki 18 *tag*. Namun, tidak berhenti begitu saja. Setiap kali memperbarui versi *HTML*, selalu ada *tag* ataupun *attribute* yang ditambahkan.

Spesifikasi *HTML* dikelola dan dikembangkan oleh *World Wide Web Consortium (W3C)*. Berdasarkan *HTML Element Reference* milik *Mozilla Developer Network*, saat ini ada 140 tag HTML, meskipun sebagianya sudah tidak lagi didukung oleh beberapa versi terbaru browser. Ekstensi file HTML adalah **.html** atau **.htm**, yang bisa dilihat dengan menggunakan *web browser* apa pun (seperti *Google Chrome*, *Safari*, atau *Mozilla Firefox*). Browser tersebut membaca file *HTML* dan menerjemahkan konten sehingga pengguna internet bisa melihat dan membacanya.

Biasanya, rata-rata situs web menyertakan sejumlah halaman *HTML* yang berbeda-beda. Contohnya, halaman beranda diberi nama *home.html*, sedang halaman ujian diberi nama *tesuji.html*.

Masing-masing halaman terdiri dari serangkaian *tags* (bisa disebut juga *elements*), yang tersusun untuk membentuk sebuah halaman website. Tag tersebut membuat hierarki yang menyusun konten hingga menjadi bagian, paragraf, heading, dan *block* konten lainnya.

Sebagian besar elemen bahasa markup ini memiliki tag pembuka dan penutup yang menggunakan syntax **<tag></tag>**.

Berikut kelebihan HTML:

- Digunakan secara luas dan memiliki banyak sumber serta komunitas yang besar.
- Dijalankan secara alami di setiap web browser.
- Mudah dipelajari.
- Open-source dan sepenuhnya gratis.
- Rapi dan konsisten.
- Menjadi standar resmi web, dikelola oleh (W3C).
- Integrasi mudah dengan bahasa *backend*, seperti PHP dan Node.js.

Kemudian, kekurangan HTML adalah:

- Umumnya digunakan untuk halaman web statis. Untuk fitur dinamis, Anda bisa menggunakan JavaScript atau bahasa backend, seperti PHP.
- Tidak bisa menjalankan *logic*. Sehingga, semua halaman web harus dibuat terpisah meskipun menggunakan elemen yang sama, seperti header dan footer.
- Fitur baru tidak bisa digunakan dengan cepat di sebagian browser.

2.7 CSS

CSS adalah singkatan dari *Cascading Style Sheet* yang berfungsi mengendalikan tampilan pada halaman website. Ibarat sebuah bangunan, *CSS* berperan sebagai cat warna yang menambah daya tarik para pengunjung. Menurut Agusriandi, *CSS* merupakan suatu teknologi yang yang dimanfaatkan untuk mempermudah *interface* sebuah website¹⁶.

¹⁶ Agusriandi, *Dasar – Dasar Penguasaan Pemrograman Web*, CV Budi Utama, Hal 12

CSS dibuat dan dikembangkan oleh W3C¹⁷ pada tahun 1996 untuk alasan yang sederhana. Dulu *HTML* tidak dilengkapi dengan *tags* yang berfungsi membedakan format halaman.

Tags, seperti , diperkenalkan di *HTML* versi 3.2, dan ketika itu menyebabkan banyak masalah bagi *developer*. Website memiliki berbagai *font*, warna *background*, dan *style*. Maka untuk menulis kembali kode memerlukan proses yang sangat panjang dan sulit. Oleh sebab itu, W3C membuat *CSS* untuk menyelesaikan masalah ini.

HTML dan *CSS* memiliki keterikatan yang erat. *HTML* adalah bahasa markup (fondasi situs) dan *CSS* memperbaiki style (untuk semua aspek yang terkait pada file *HTML* dengan tampilan website), maka kedua bahasa pemrograman ini harus berjalan beriringan.

Sebelum menggunakan *CSS*, semua *stylizing* harus disertakan ke dalam markup *HTML*. Semua wajib mendeskripsikan semua *background*, warna *font*, *alignment*, dan lain-lain secara terpisah.

Sejak adanya *CSS*, programmer dapat mengatur tampilan semua aspek pada file yang berbeda, lalu menentukan *style*, kemudian mengintegrasikan file *CSS* di atas markup *HTML*. Jadi, markup *HTML* bisa lebih mudah di-*Maintain*. Selain itu, tidak perlu mendeskripsikan tampilan dari masing-masing elemen secara berulang-ulang. Kode yang digunakan kini lebih singkat, dan *error* dapat diminimalisir.

CSS menggunakan bahasa Inggris sederhana berbasis syntax yang dilengkapi dengan sekumpulan *rule* yang mengaturnya. Satu yang menjadi pertanyaannya sekarang, Struktur *syntax* *CSS* cukup sederhana. Struktur ini memiliki *selector* dan *declaration block*. Pilih elemen yang diinginkan, kemudian deklarasi yang harus dilakukan terhadap elemen tersebut. *Selector* mengarah ke elemen *HTML* yang ingin Anda ubah tampilannya., sementara *Declaration block* memuat satu atau lebih banyak deklarasi yang dipisahkan dengan tanda titik koma. Setiap deklarasi menyertakan nama dan value dari properti *CSS*, yang

¹⁷ World Wide Consortium

dipisahkan dengan tanda koma. Umumnya deklarasi *CSS* diakhiri dengan tanda titik koma, sedangkan declaration block dikelilingi oleh tanda kurung kurawal.

2.8. JavaScript

JavaScript berbeda dengan Bahasa pemrograman *Java*. Jika Bahasa pemrograman *Java* dikaitkan dengan sistem operasi android, beda halnya dengan *JavaScript*. *JavaScript* adalah Bahasa *script* yang ditempelkan pada kode *HTML* dan diproses di sisi *klien*. Dengan adanya *JavaScript*, kemampuan *HTML* menjadi semakin luas. Sebagai contoh, dengan menggunakan *JavaScript* dimungkinkan untuk memvalidasi *value* dari formular sebelum formular dikirim ke *server*.

Dimulai pada tahun 1995, *JavaScript* diciptakan oleh Brendan Eich di *Netscape Communications*. Awalnya, *JavaScript* hanya digunakan untuk keperluan internal dengan *web browser* perusahaan tersebut, *Netscape Navigator*.

Dulunya, *JavaScript* bernama *LiveScript*, yang kemudian diubah menjadi *JavaScript* agar bisa menjadi “teman” bagi bahasa pemrograman *Java* milik mitra mereka, Sun Microsystems. Kami akan menjelaskan perbedaan *Java* dan *JavaScript* nanti.

JavaScript terus berkembang bersama browser web baru seperti **Mozilla Firefox** dan **Google Chrome**. Bahkan, saat ini sedang dikembangkan mesin *JavaScript* modern pertama, yaitu **V8**, yang bertugas untuk mengompilasi bytecode menjadi kode mesin asli.

2.9 MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data yang dimiliki dan disponsori oleh sebuah Perusahaan Komersial Swedia MySQLAB, dimana memegang hak cipta hampir atas semua kode sumbernya. Perusahaan ini didirikan oleh David Axmark, Allan Larsson, dan Michael ‘Monty’ Widenius¹⁸. *PHP* dan *MySQL* merupakan dua teknologi yang sering digunakan untuk membuat aplikasi berbasis website. Jika *PHP* dipakai untuk mengatur *logic*, misalnya

¹⁸ Wikipedia Ensiklopedia Bebas, *MySQL* diakses pada Senin, 16 September 2024

interaksi antara pengguna dengan *database*, sedangkan *MySQL* berfungsi sebagai server penyimpanan data.

Berawal pada tahun 1994, versi pertama *MySQL* mulai dimunculkan. Awalnya hanya diciptakan untuk kepentingan pribadi dari mSQL¹⁹ berdasarkan Bahasa tingkat rendah ISAM²⁰, yang dianggap terlalu lambat dan tidak fleksibel oleh para kreatornya. Hingga menuju masa selanjutnya, perkembangan terus berlanjut dengan rincian sebagai berikut:

- 23 Mei 1995 : Rilis internal pertama
- Akhir 1996 : pengembangan versi 3.19 dari www.tcx.se
- Januari 1997 : pengembangan versi 3.20
- Tahun 1998 : Rilis versi *Windows* untuk *Windows 95* dan *Network*, pengembangan versi 3.21 sampai versi 3.22 alpha, beta
- Tahun 2000 : pengembangan versi 3.23 beta
- Tahun 2002 : pengembangan versi 4.0 beta
- Tahun 2004 : pengembangan versi 4.1 beta
- Maret 2005 : pengembangan versi 5.0, kursor, prosedur tersimpan, pemicu, tampilan, transaksi XA
- Tahun 2008 : *Sun Microsystem* mengakui sisi *MySQLAB*, pengembangan versi 5.1, penjadwal acara, partisi, API plugin, replikasi berbasis baris, tabel log server.
- Desember 2010 : *MySQL* versi 5.5 tersedia secara umum dengan berbagai fitur antara lain: Mesin penyimpanan default adalah InnoDB, yang mendukung transaksi dan batasan integritas referensial, Peningkatan subsistem I/O InnoDB, dan beberapa lainnya.
- 22 Mei 2009 : MySQL Server 6.0.11-alpha diumumkan sebagai rilis terakhir dari lini 6.0. Pengembangan MySQL Server di masa mendatang menggunakan Model Rilis Baru. Fitur-fitur yang dikembangkan untuk 6.0 sedang dimasukkan ke dalam rilis mendatang.

¹⁹ Mini SQL adalah sistem manajemen basis data ringan dari Hughes Technologies

²⁰ Indexed Sequential Access Method (ISAM) adalah metode untuk membuat, memelihara, dan memanipulasi berkas data komputer sehingga rekaman dapat diambil secara berurutan atau acak dengan satu atau beberapa kunci

- Februari 2013 : Fitur-fitur baru termasuk peningkatan kinerja untuk pengoptimal kueri , throughput transaksional yang lebih tinggi di *InnoDB*, API memakai gaya *NoSQL* baru, peningkatan partisi untuk kueri dan pengelolaan tabel yang sangat besar, tipe kolom *TIMESTAMP* yang menyimpan milidetik dengan benar, peningkatan replikasi, dan pemantauan kinerja yang lebih baik dengan memperluas data yang tersedia melalui *PERFORMANCE_SCHEMA*. Mesin penyimpanan InnoDB juga menyertakan dukungan untuk pencarian teks lengkap dan peningkatan kinerja komit grup.
- Agustus 2015 : MySQL mendukung tipe data JSON asli yang ditentukan oleh RFC 7159
- April 2018 : pengembangan versi 8.0
- Tahun 2019 : *MySQL* dinyatakan sebagai DBMS dari peringkat *DB-Engines*²¹

²¹ Gelmann, Matthias (3 Januari 2020). *MySQL adalah DBMS Tahun 2019* . db-engines.com . DB-Engines . Diakses tanggal 7 Januari 2020 .

BAB III

PERENCANAAN DAN PERANCANGAN

3.1 Perencanaan

3.1.1 Bahan yang dibutuhkan

Sebelum melakukan penelitian penulis melakukan hal – hal di bawah ini:

- a. Memahami dan menggali informasi mengenai metode RIASEC
- b. Membuat logo website
- c. Menyiapkan foto dan video yang berkaitan
- d. Mengumpulkan informasi kampus dan jurusan di Indonesia
- e. Menginstall Aplikasi Visual Studio Code
- f. Menginstall Database MySQL dan Laravel

3.1.2 Hardware

Hardware atau perangkat keras yang dibutuhkan dalam pembuatan tugas akhir ini sebagai berikut:

- Laptop Lenovo



Gambar 3.1.2.1. Laptop Lenovo

- Modem 4G



Gambar 3.1.2.2. Modem 4G

3.1.3 Software

Selain *hardware*, penulis juga membutuhkan *software* atau perangkat lunak sebagai berikut :

- Visual Studio Code



Gambar 3.1.2.3. Visual Studio Code

- Database
- MySQL dan phpMyAdmin



Gambar 3.1.2.4. MySQL dan phpMyAdmin

- Google Chrome



Gambar 3.1.2.5. Google Chrome

- Figma



Gambar 3.1.2.6. Figma

- Microsoft Word

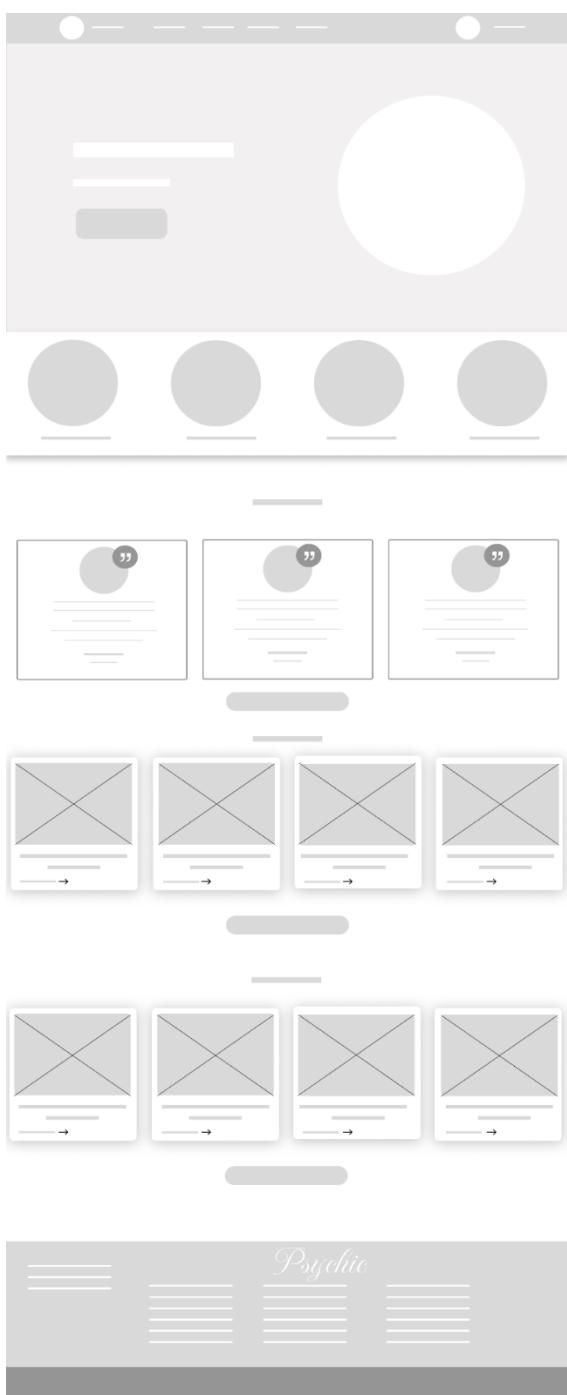


Gambar 3.1.2.7. Microsoft Word

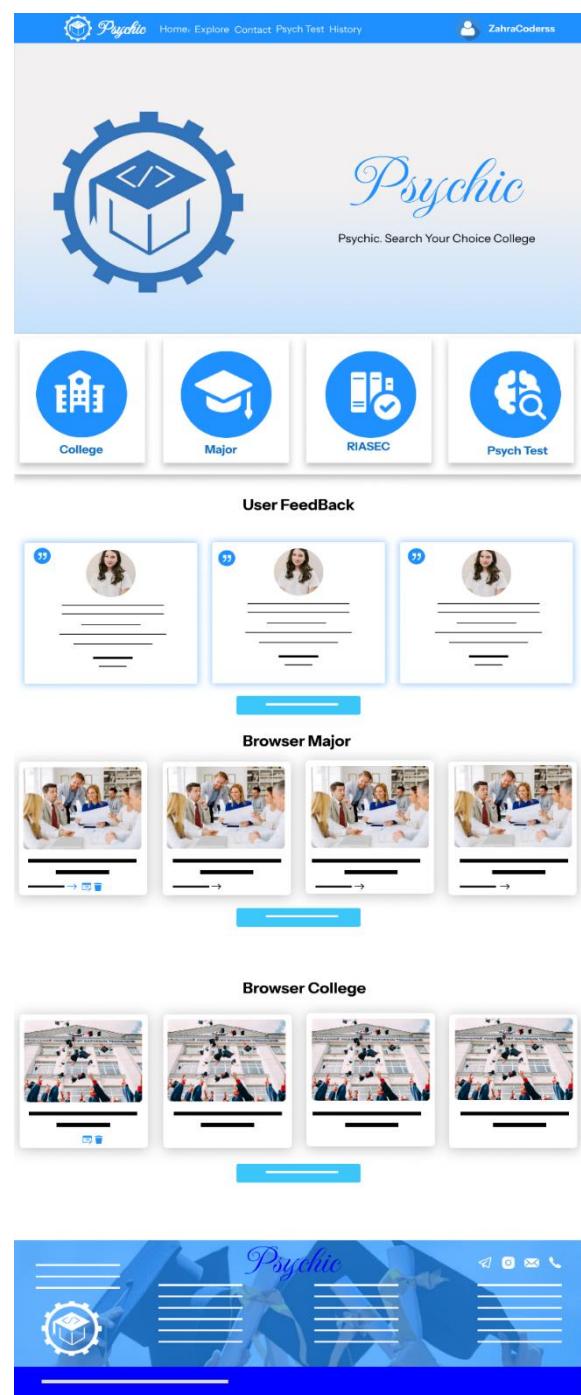


Gambar 3.1.2.8. Corel Draw 2019

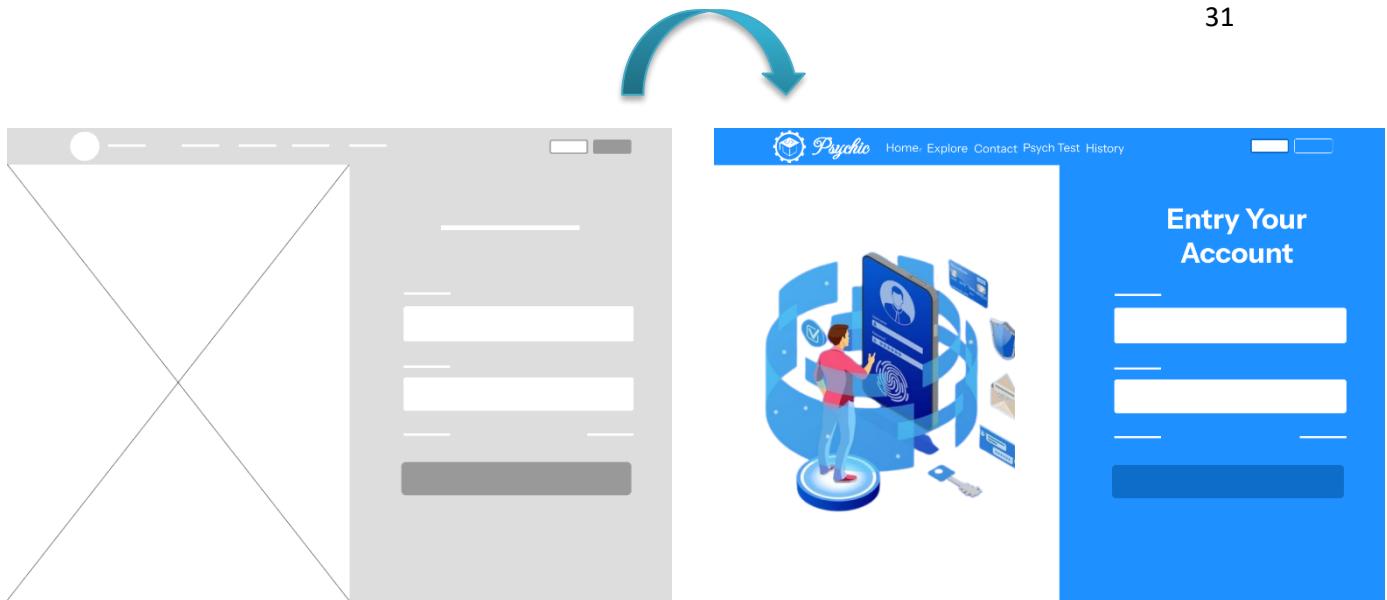
3.1.4 Wireframe



Gambar 3.1.4.1. Wireframe Home

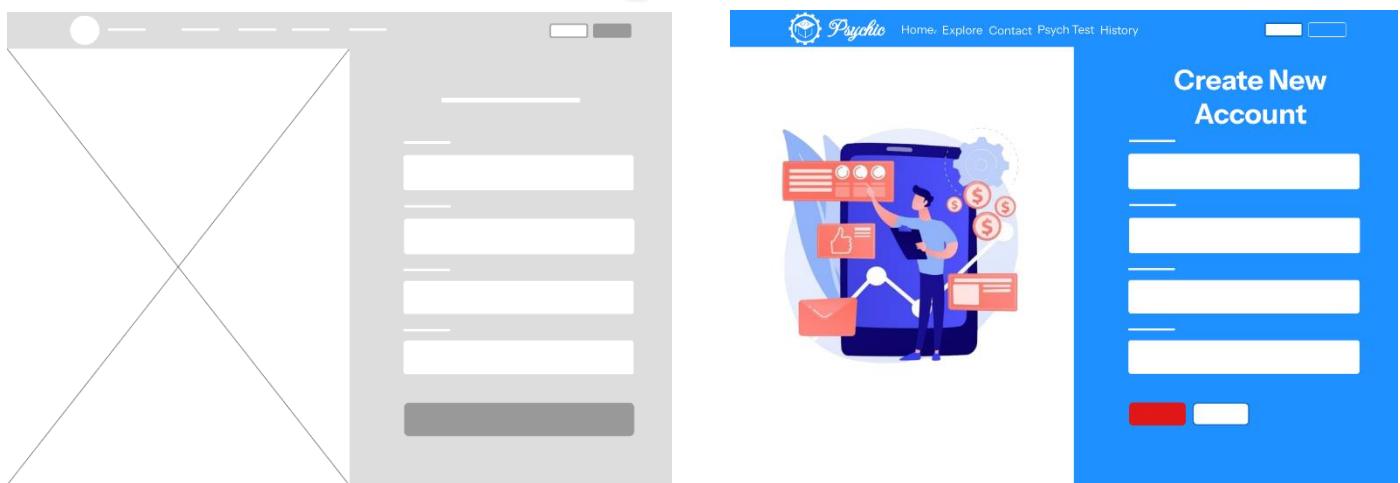


Gambar 3.1.4.2. UI/UX Home



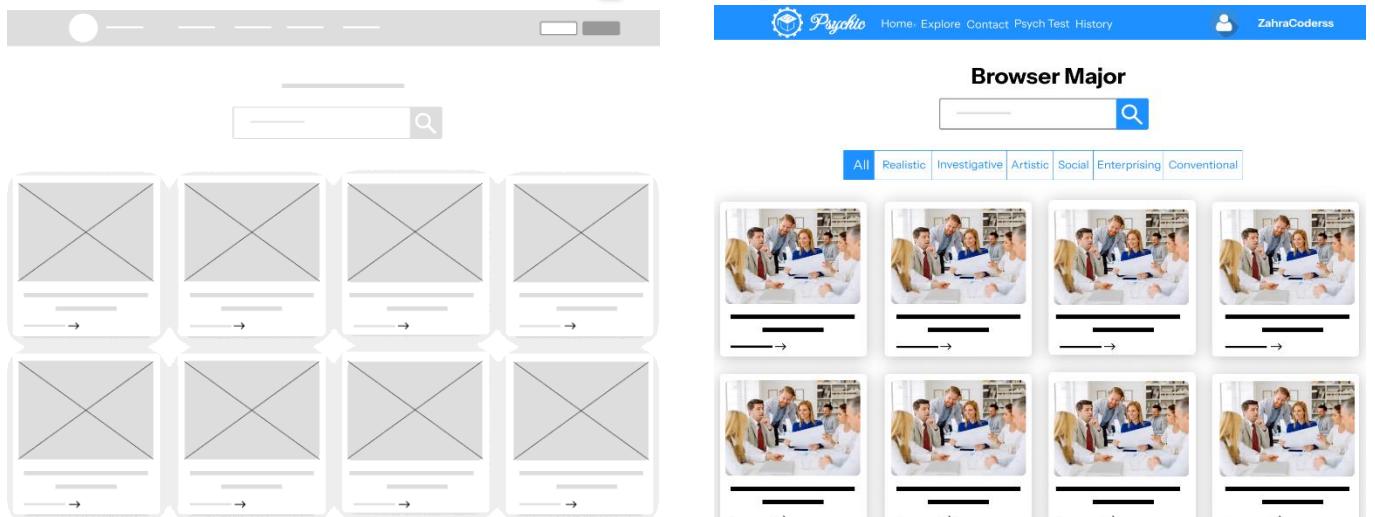
Gambar 3.1.4.3. Wireframe Login

Gambar 3.1.4.4. UI/UX Login



Gambar 3.1.4.5. Wireframe Register

Gambar 3.1.4.6. UI/UX Register



Gambar 3.1.4.7. Wireframe Major

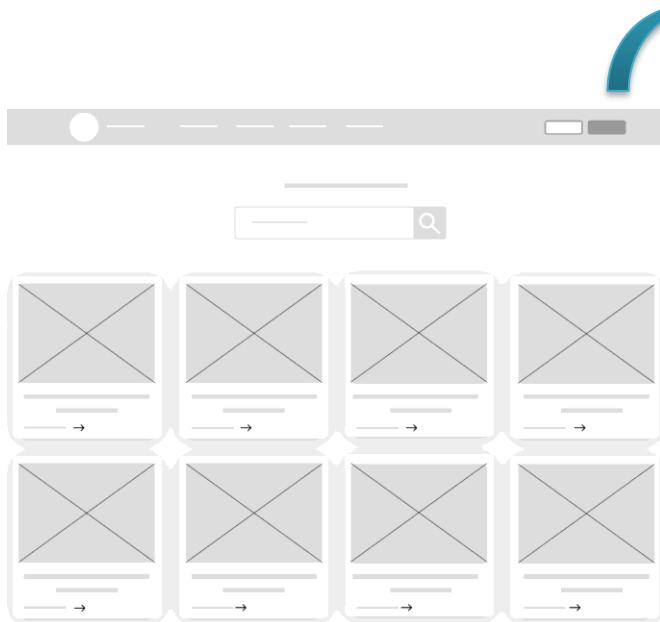
Gambar 3.1.4.8. UI/UX Major



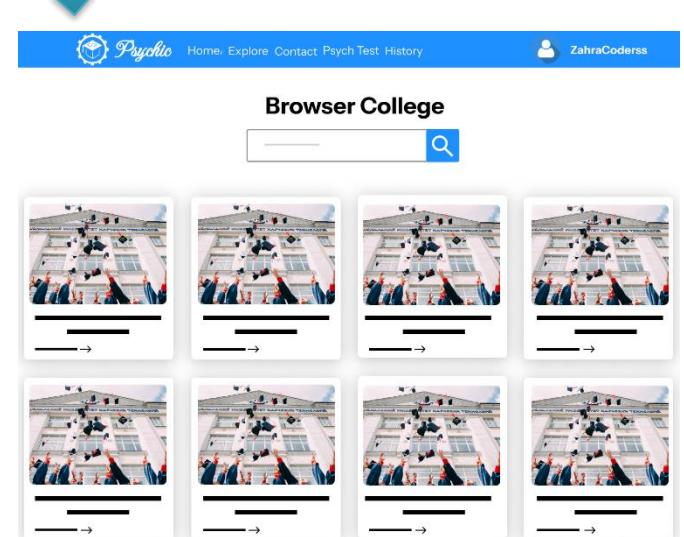
Gambar 3.1.4.9. Wireframe Show Major



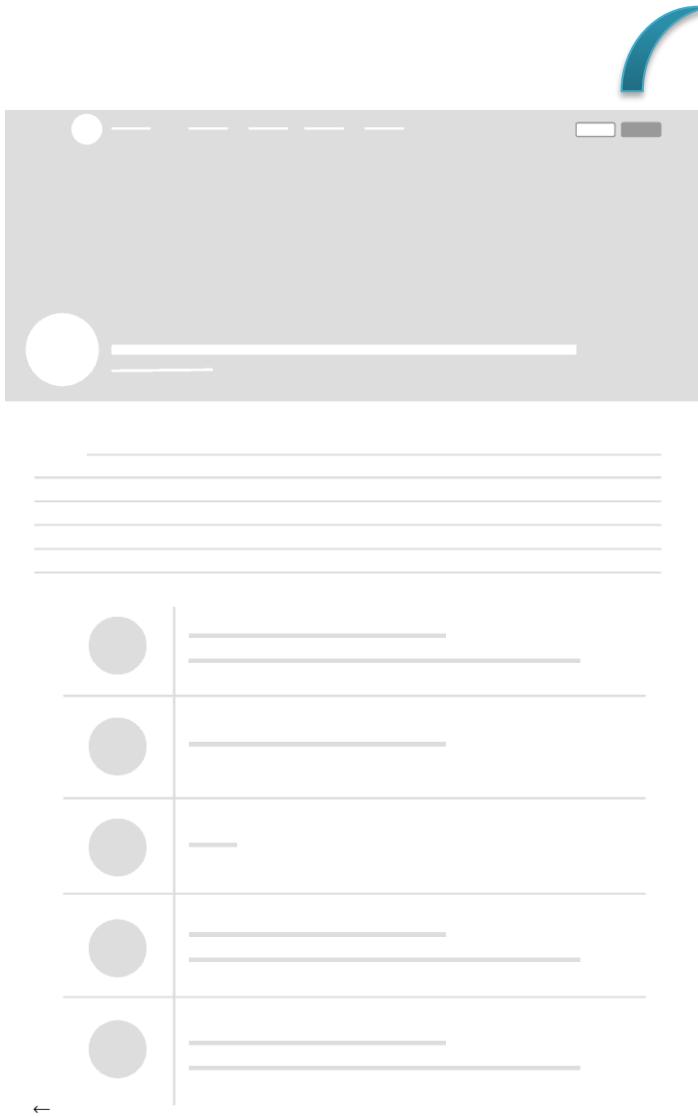
Gambar 3.1.4.10. UI/UX Show Major



Gambar 3.1.4.11. Wireframe Campus

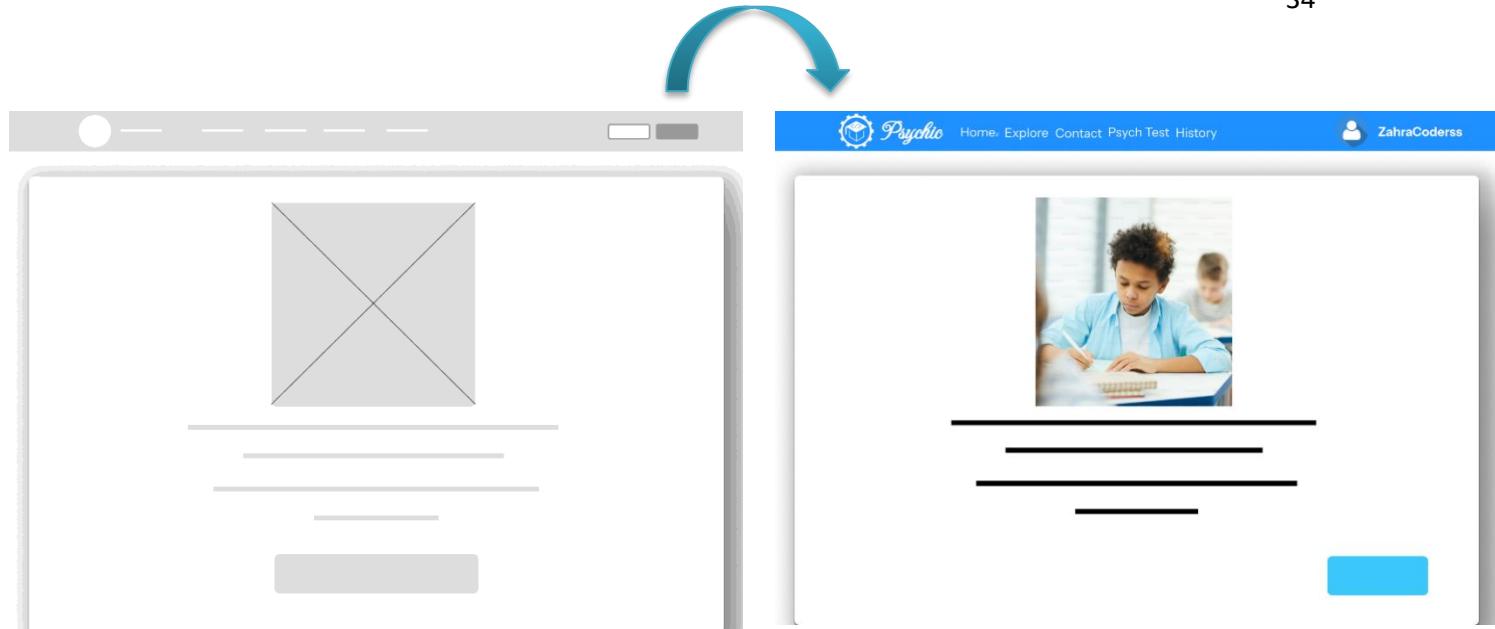


Gambar 3.1.4.12. UI/UX Campus



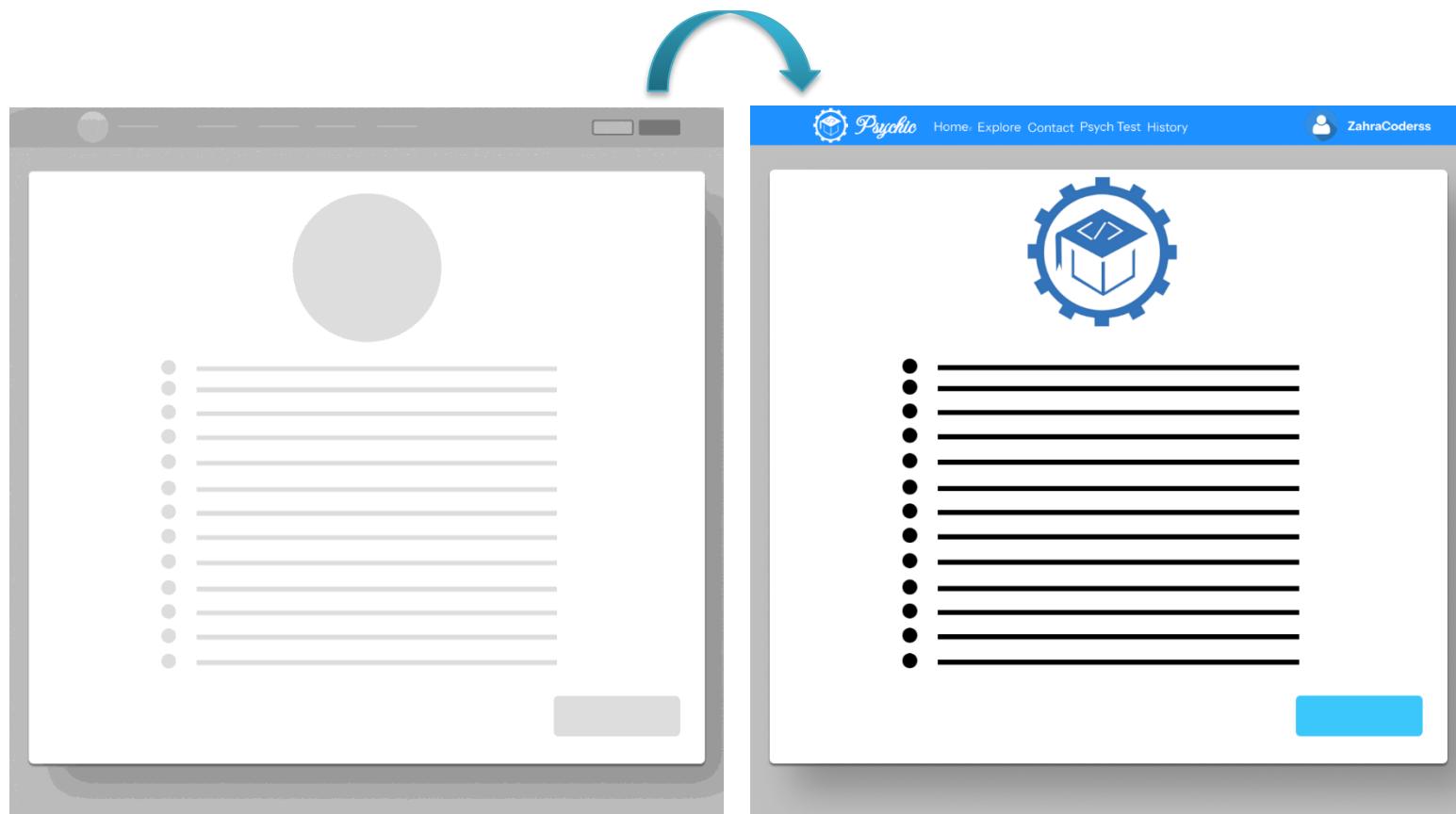
Gambar 3.1.4.13. Wireframe Show Campus

Gambar 3.1.4.14. UI/UX Show Campus



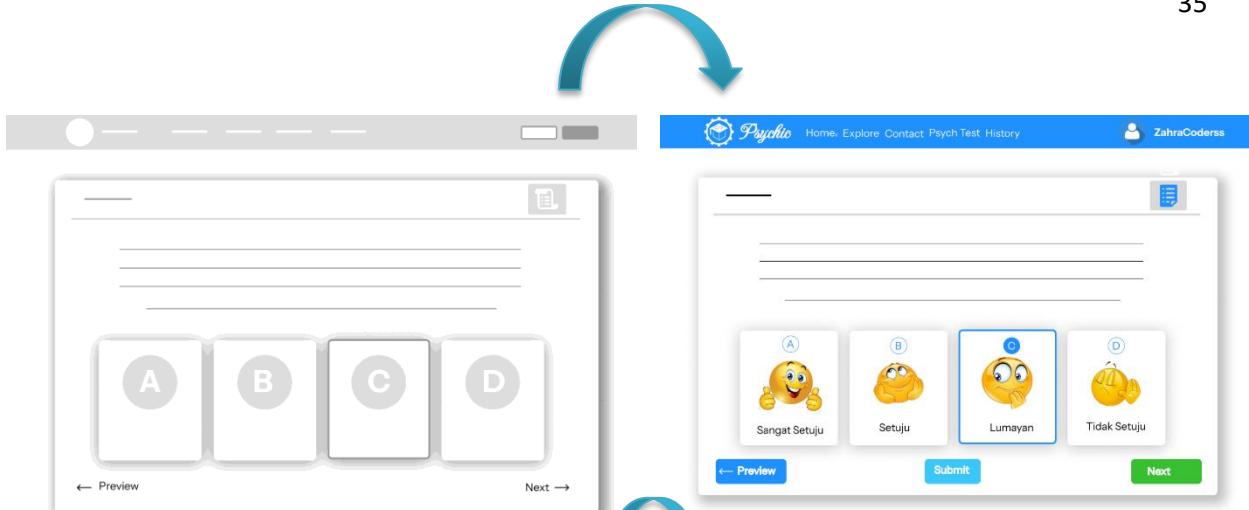
Gambar 3.1.4.15. Wireframe Psych Test

Gambar 3.1.4.16. UI/UX Psych Test



Gambar 3.1.4.17. Wireframe Start Exam

Gambar 3.1.4.18. UI/UX Start Exam



Hai, Nama...!

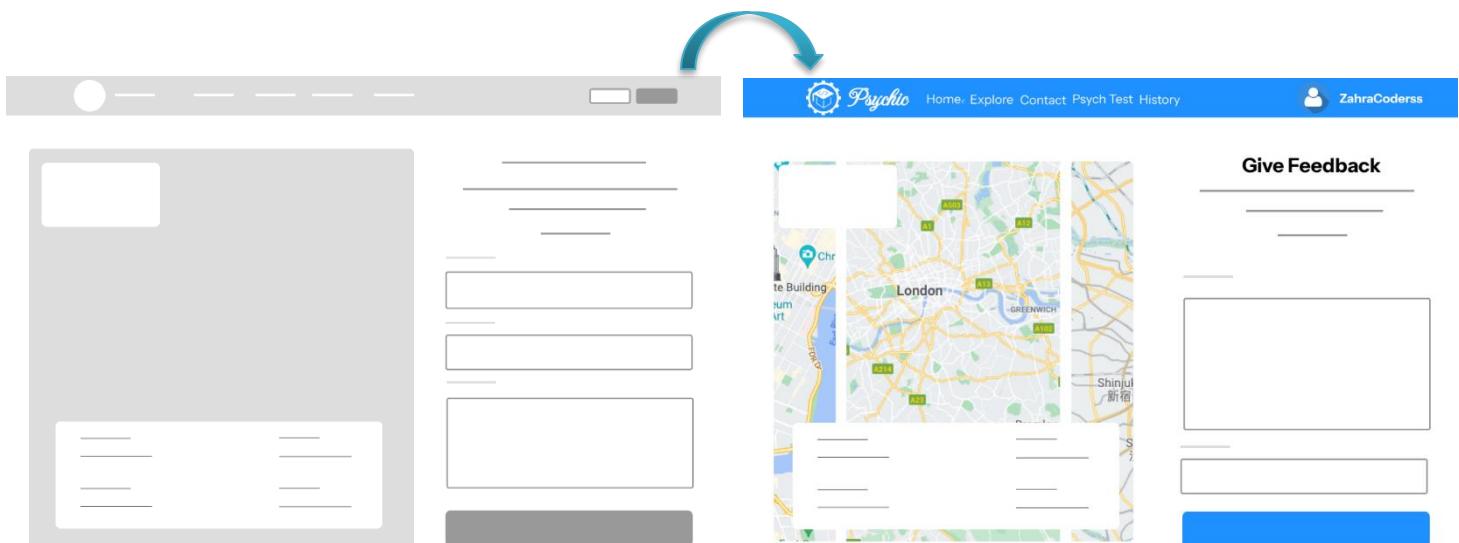
Gambar 3.1.4.21. Wireframe Test

Gambar 3.1.4.22. UI/UX Test Result



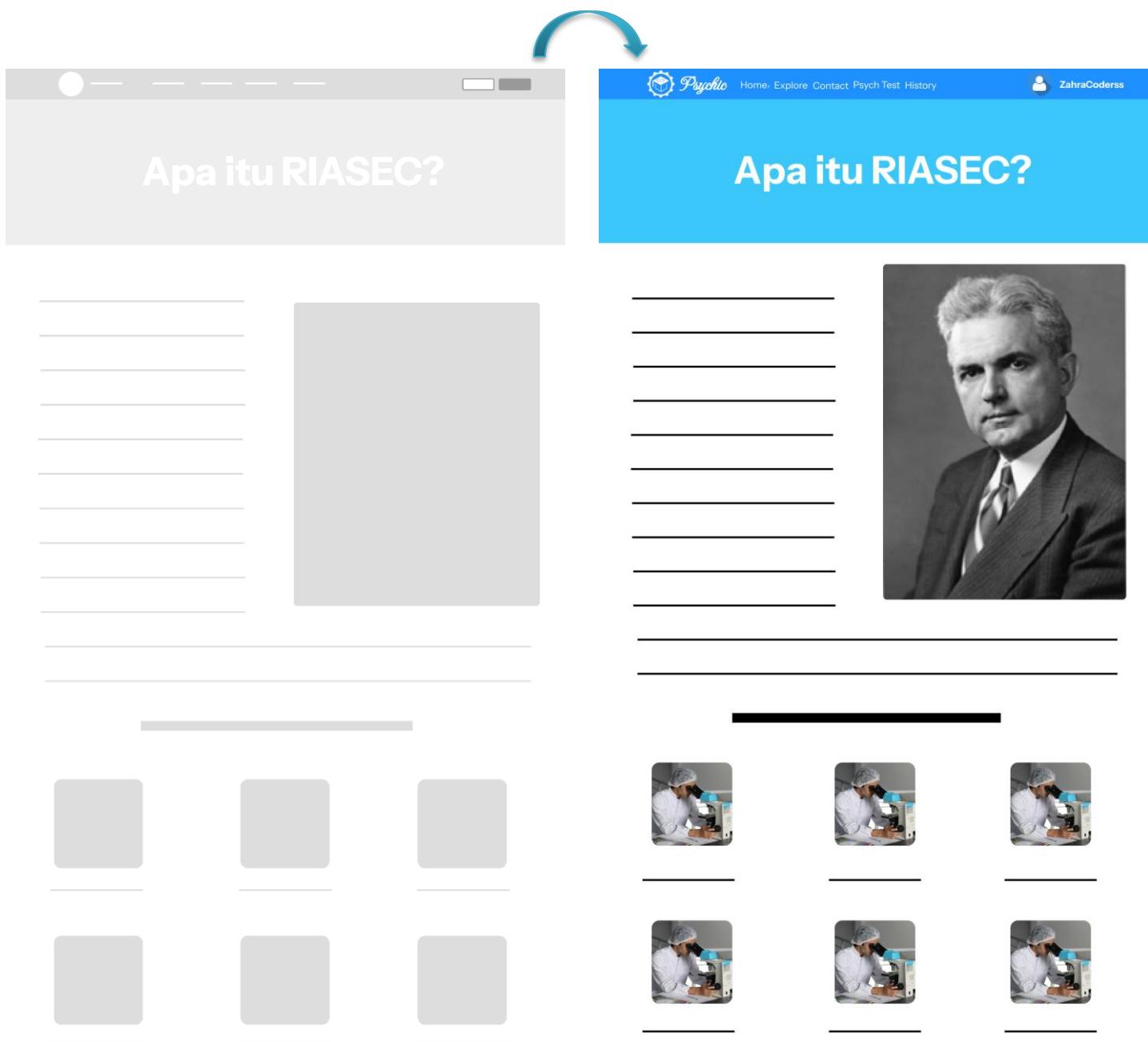
Gambar 3.1.4.23. Wireframe History

Gambar 3.1.4.24. UI/UX History



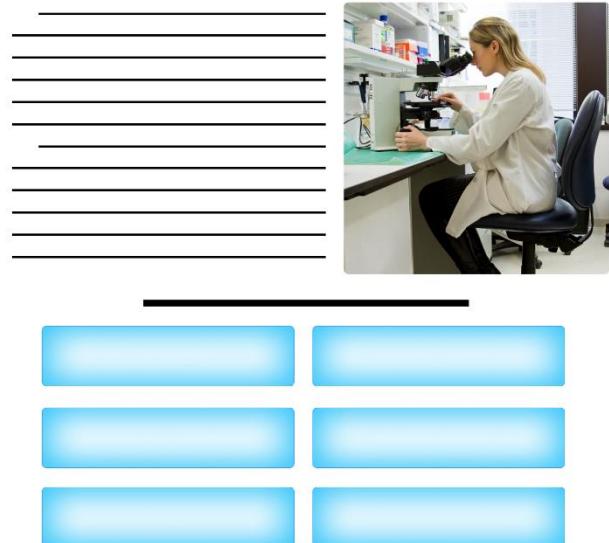
Gambar 3.1.4.25. Wireframe Feedback

Gambar 3.1.4.26. UI/UX Feedback



Gambar 3.1.4.27. Wireframe RIASEC

Gambar 3.1.4.28. UI/UX RIASEC



Gambar 3.1.4.29. Wireframe Detailed Type

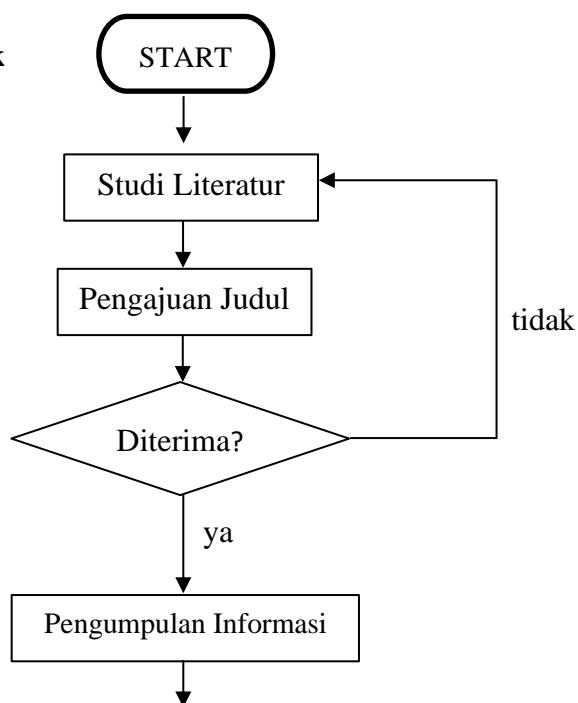
Gambar 3.1.4.30. UI/UX Detailed Type

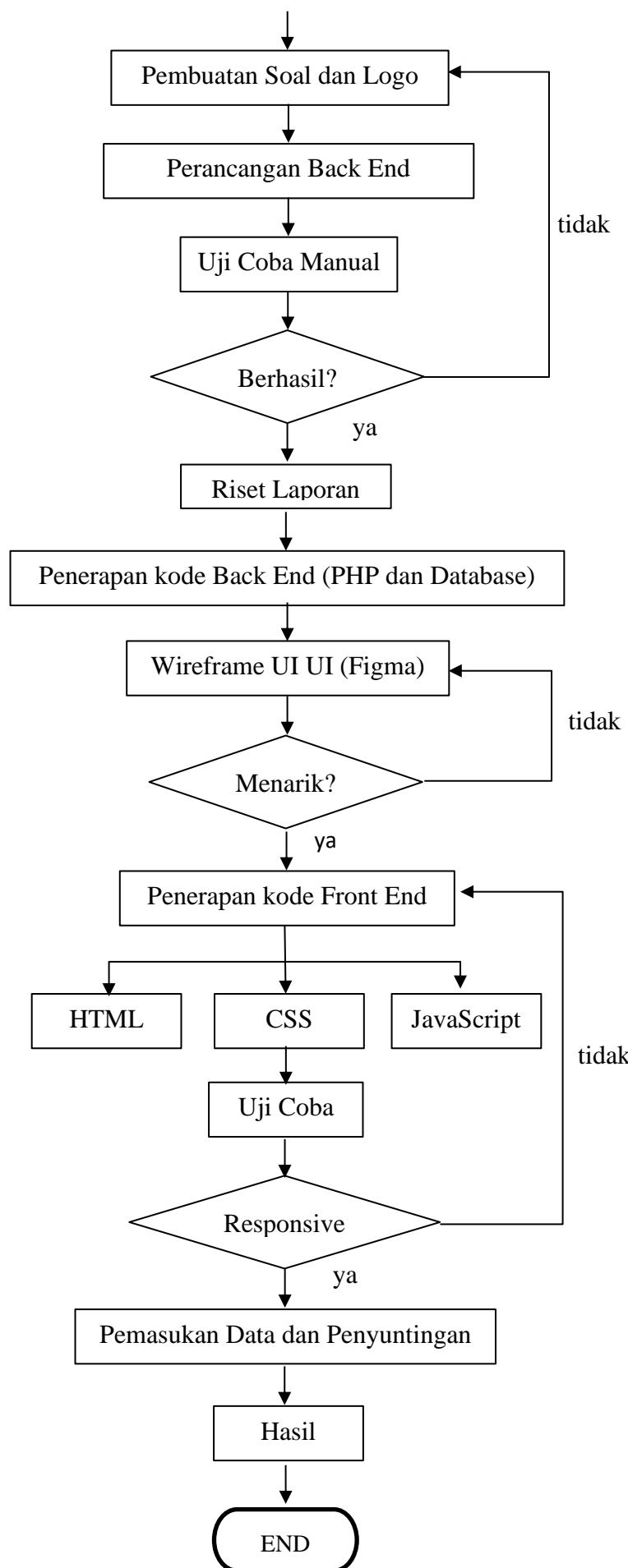
3.1.5 Jadwal Penulisan

No	Aktivitas	Mar	Apr	Mei	Jun	Jul	Agu	Sep	Okt	Nov	Des
1	Studi Literatur										
2	Pengajuan Judul										
3	Pengumpulan Informasi										
4	Pembuatan Logo										
5	Riset Soal										
6	Pembuatan Soal										
7	Rancangan Back End										
8	Wireframe UI / UX										
9	Riset Laporan										
10	Penerapan kode										
11	Pembuatan Laporan										
12	Pemasukan data										
13	Penyuntingan dan uji coba										
14	Finish										

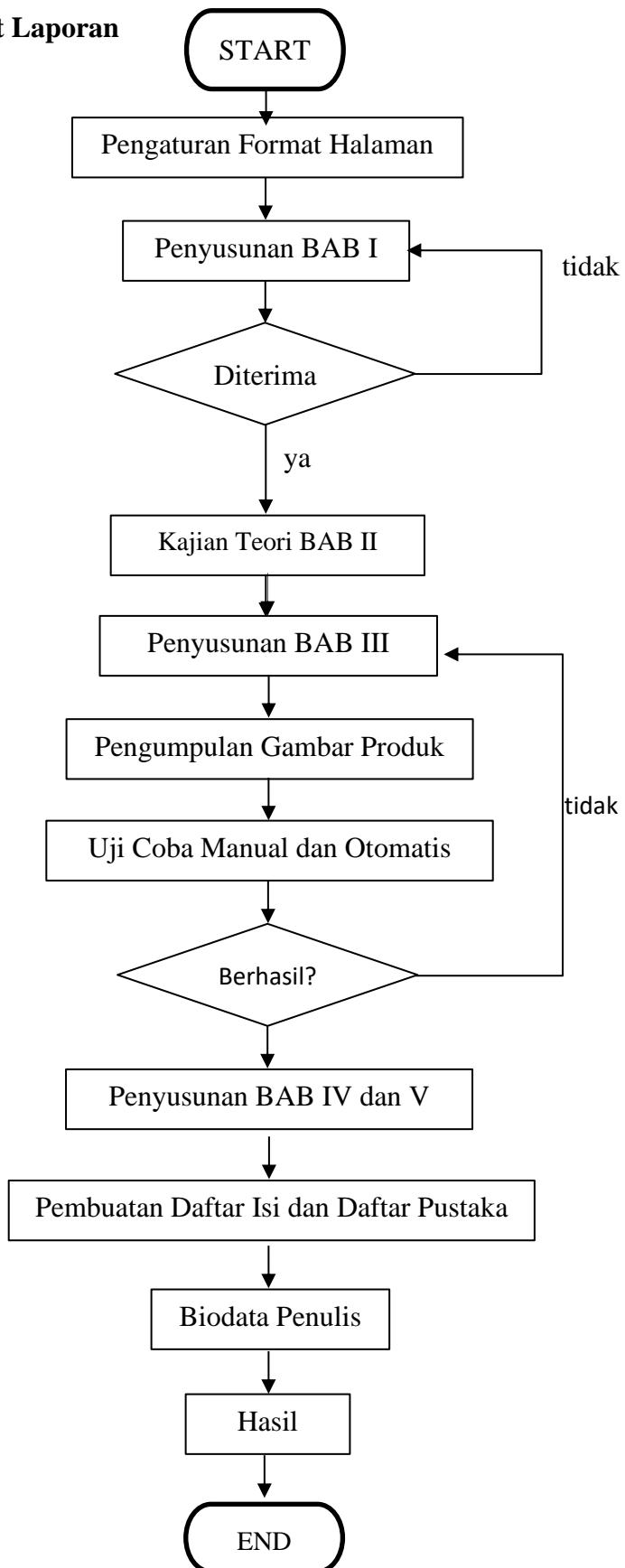
Tabel 3.1.5.1. Jadwal Penulisan

3.1.6 Flowchart Produk

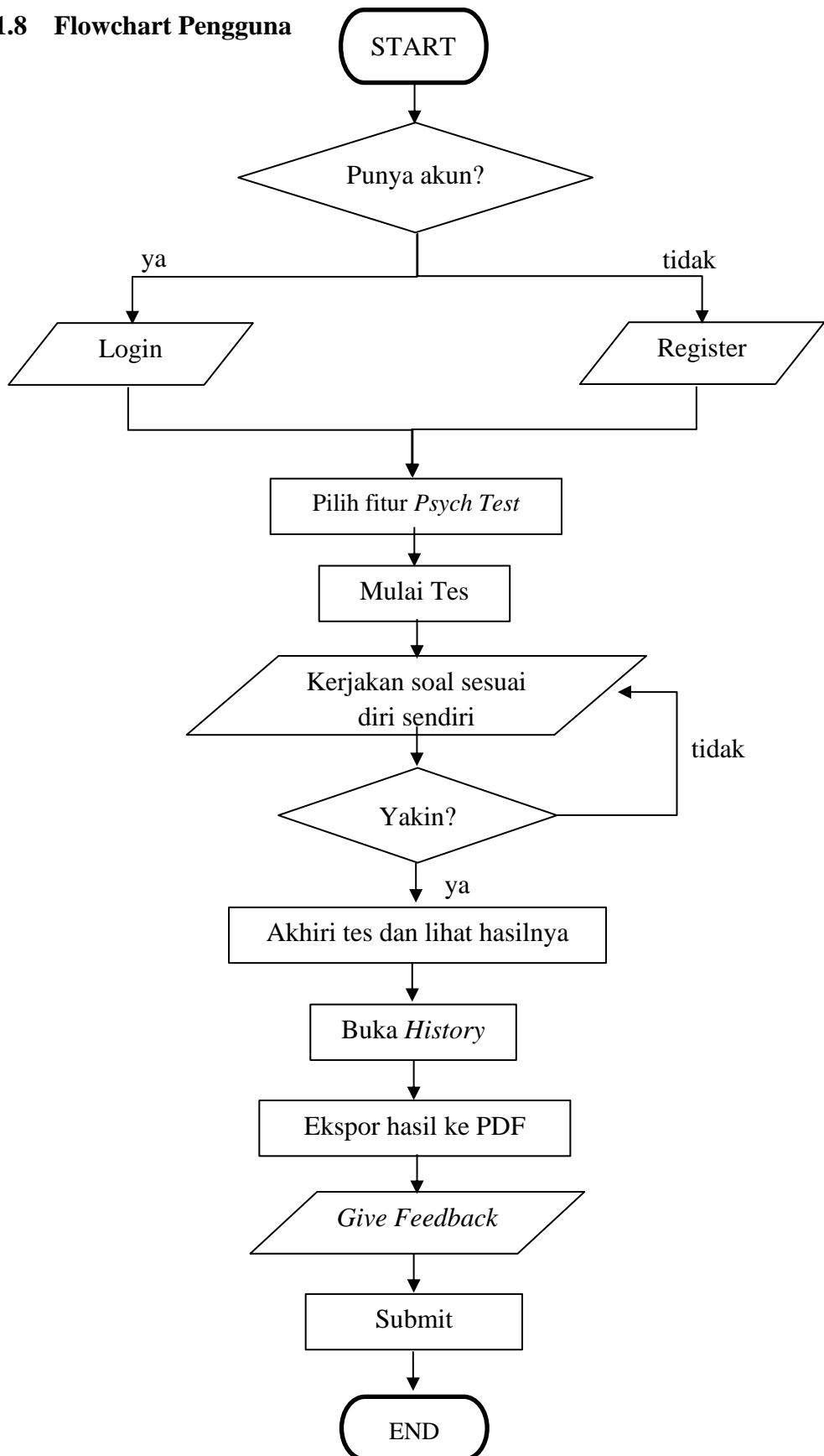




3.1.7 Flowchart Laporan



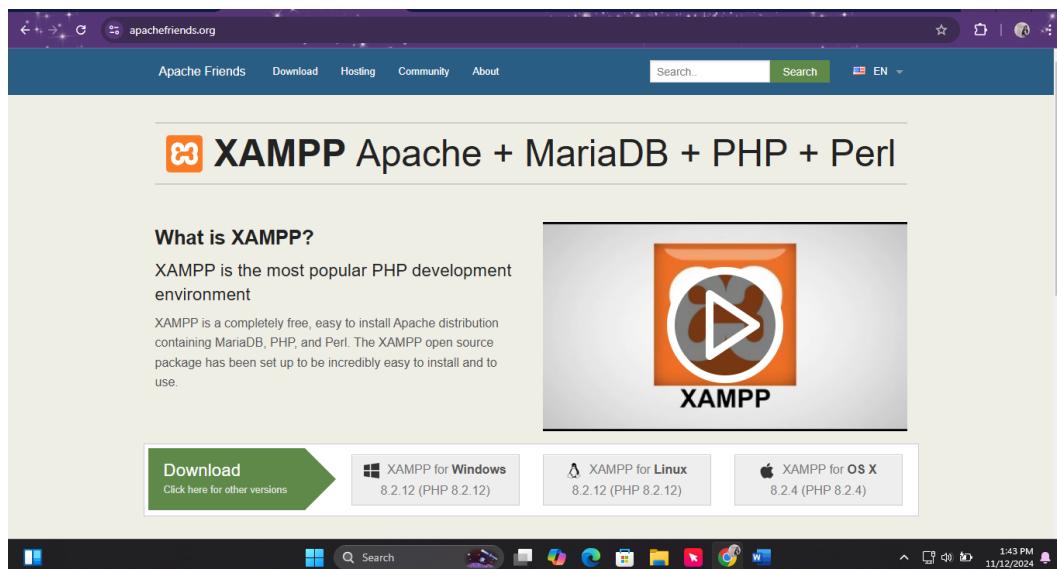
3.1.8 Flowchart Pengguna



3.2 Perancangan

3.2.1 Download dan Instalasi XAMPP

- Buka alamat link <https://apachefriends.org>, tekan *Download*



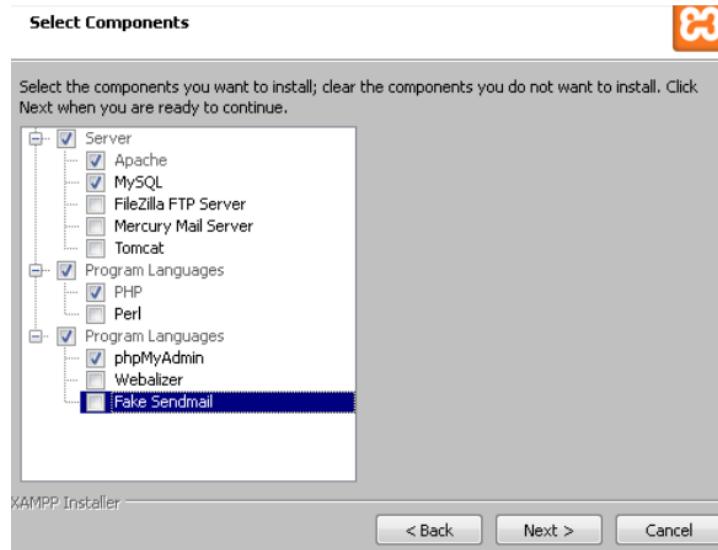
Gambar 3.2.1.1. Download XAMPP

- Lanjutkan menuju proses instalasi. Jika muncul pesan *Apakah Anda Yakin?*, tekan *Yes*. Berikutnya akan muncul tampilan seperti ini, tekan *Next*.



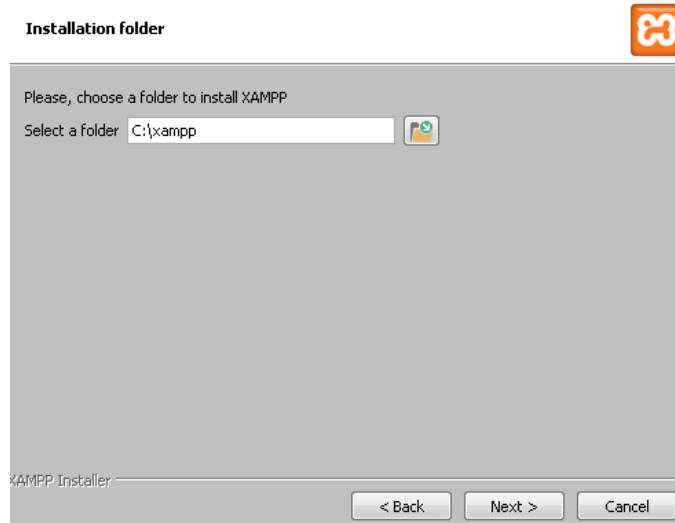
Gambar 3.2.1.2. Instalasi XAMPP

- Pada Tampilan *Select Components*, centang bagian *MySQL* dan *phpMyAdmin*, tekan *Next*.



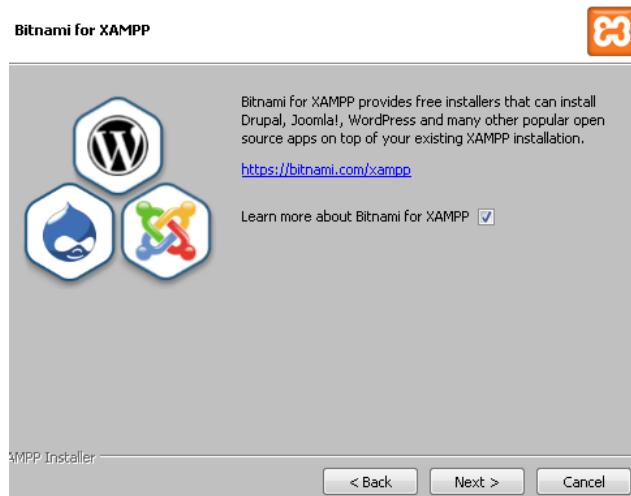
Gambar 3.2.1.3. Select Component XAMPP

- Isi kolom input sesuai folder tujuan penyimpanan XAMPP. Disarankan menyimpannya di data *C:\xampp*.



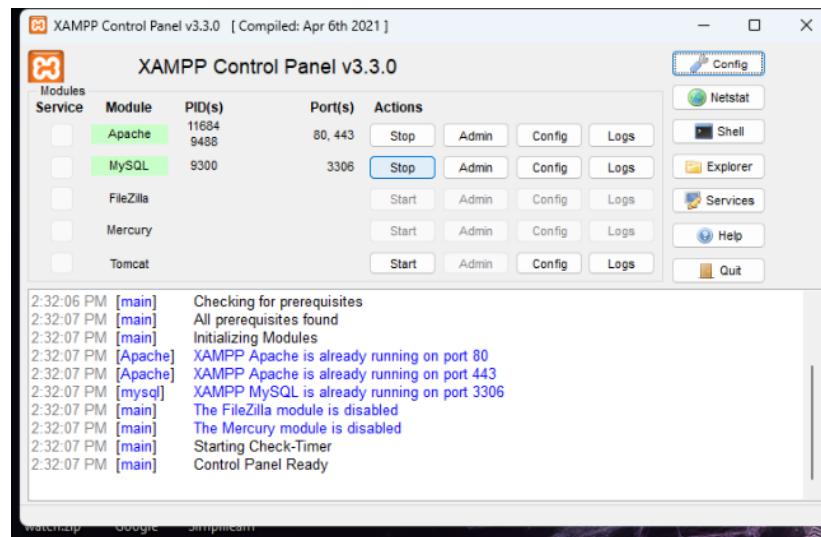
Gambar 3.2.1.4. Folder Instalasi XAMPP

- Selanjutnya, centang jika ingin menginstall Bitnami untuk XAMPP. Kemudian, silakan tekan tombol *Next* untuk melanjutkan proses instalasi. Jika sudah, tekan *Finish*.



Gambar 3.2.1.5. Bitnami for XAMPP

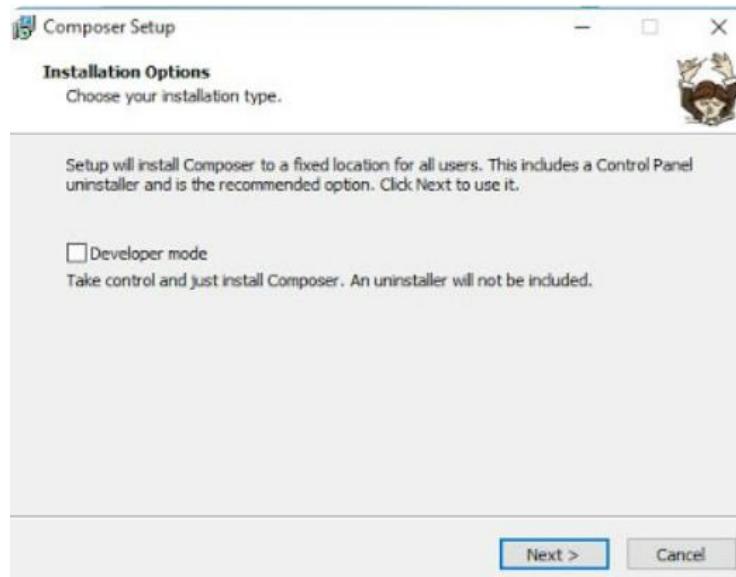
- Untuk menjalankan XAMPP, aktifkan *Apache* dan *MySQL* dengan menekan tombol *Start*.



Gambar 3.2.1.6. Start XAMPP

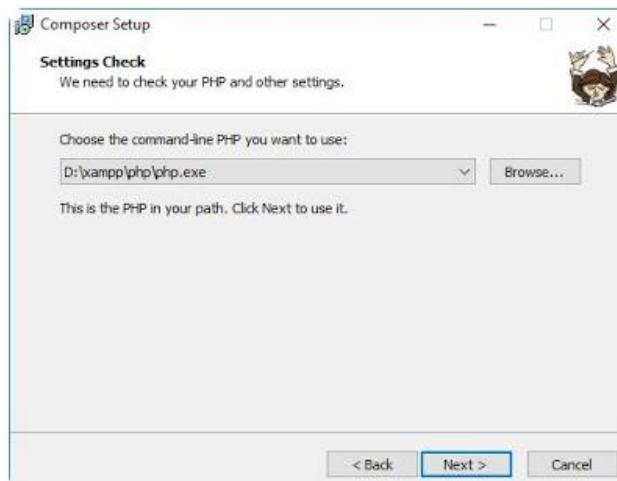
3.2.2 Download dan Instalasi Composer

- Setelah melakukan instalasi XAMPP, download *Composer* melalui situs resmi <https://getcomposer.org/download/>. Lanjutkan dengan klik dua kali *Composer-Setup.exe*, lalu tekan tombol *Next*.



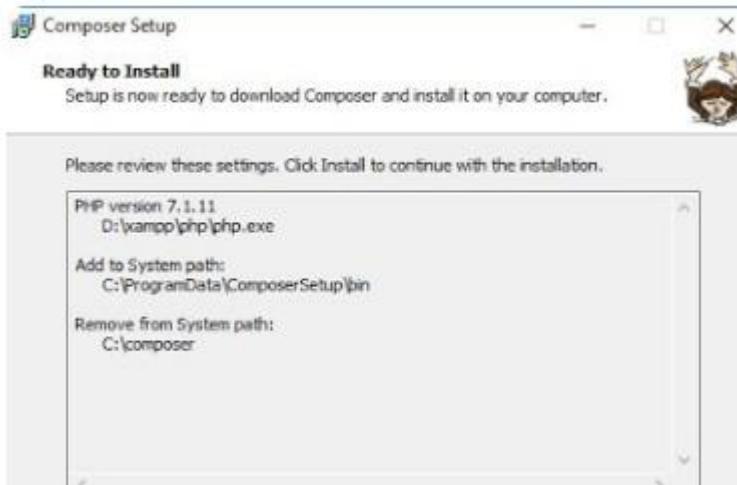
Gambar 3.2.2.1. Instalasi Composer

- Sama halnya dengan XAMPP, Composer juga meminta user mengisi file lokasi PHP. Tekan *Browser* untuk mencari, kemudian *Next*.



Gambar 3.2.2.2. Folder Instalasi Composer

- Jika muncul halaman *proxy setting*, langsung tekan *Next* hingga muncul *Ready to Install*. Klik *Install*



Gambar 3.2.2.3. Ready to Install Composer

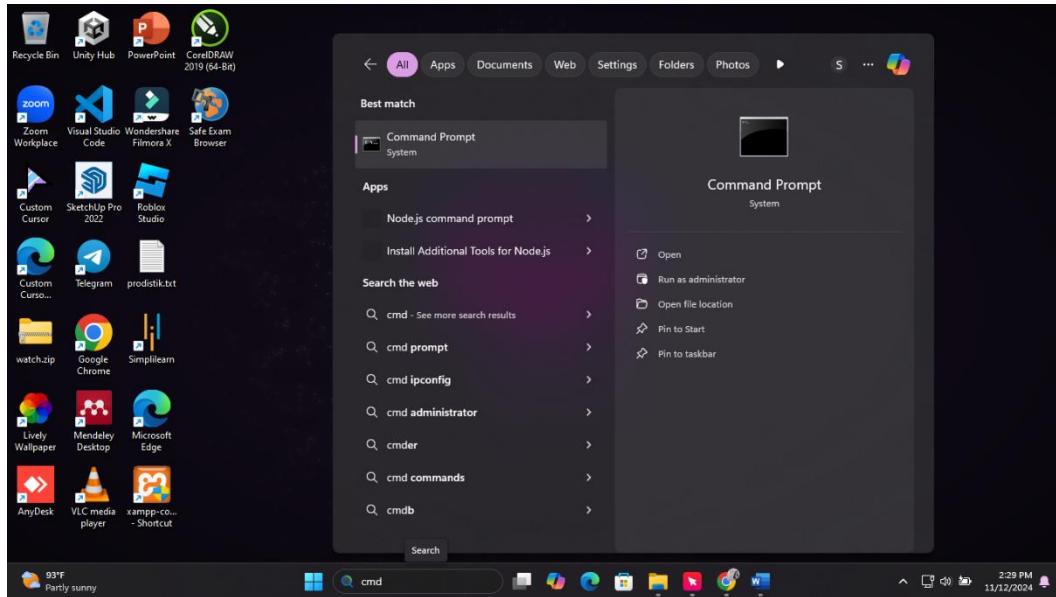
- Tunggu beberapa saat, tekan lagi *Install*. Jika selesai, tekan tombol *Finish*.



Gambar 3.2.2.4. Finishing Install Composer

3.2.3 Instalasi Laravel

- Instalasi Laravel dapat dilakukan melalui cmd. Setelah Instalasi Composer, buka Start, buka aplikasi Composer.



Gambar 3.2.3.1. Buka Composer

- Ketik *Cd C:/Laragon/www/*, *Enter* untuk masuk ke folder Laragon, kemudian ketik *composer create-project laravel/tugasakhir*, *Enter*. Halaman cmd akan muncul tampilan seperti di bawah ini.

```

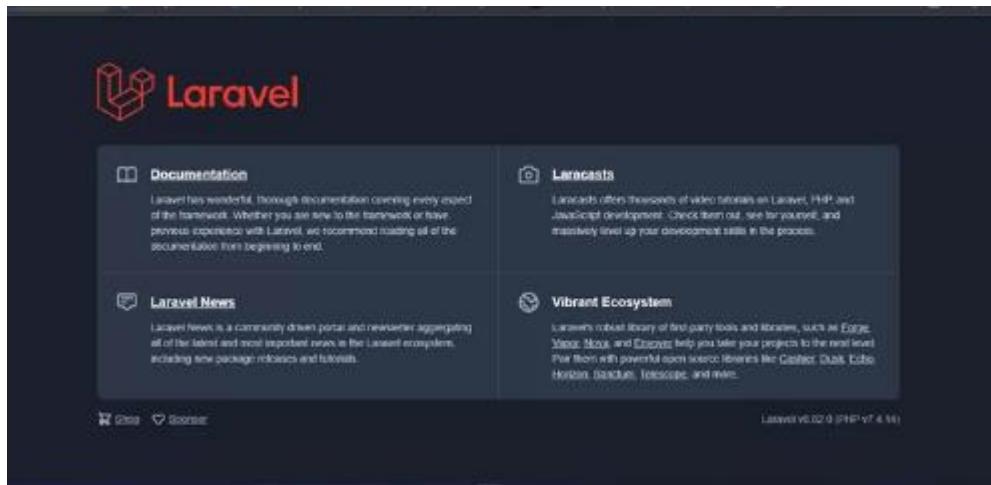
Command Prompt
- Installing sebastian/code-unit-reverse-lookup (3.0.3): Extracting archive
- Installing phpunit/php-code-coverage (9.2.10): Extracting archive
- Installing doctrine/instantiator (1.4.6): Extracting archive
- Installing phpspc/prophesyc (v1.15.0): Extracting archive
- Installing phar-io/version (3.1.0): Extracting archive
- Installing phar-io/manifest (2.0.3): Extracting archive
- Installing phar-io/deep-copy (1.10.2): Extracting archive
- Installing phpsuit/phpsuit (9.5.13): Extracting archive
[No package suggestions were added by new dependencies, use 'composer suggest' to see details.
Package swiftmailer/swiftmailer is abandoned, you should avoid using it. Use symfony/mailer instead.]
Generating optimized autoload files
> Illuminate\Foundation\ComposerScripts::postAutoloadDump
> @php artisan package:discover --ansi
Discovered Package: nesbot/carbon
Discovered Package: laravel/ignition
Discovered Package: fruitcake/laravel-cors
Discovered Package: laravel/sail
Discovered Package: laravel/sanctum
Discovered Package: laravel/tinker
Discovered Package: munozaduro/collision
Package manifest generated successfully.
77 packages you are using are looking for funding.
Use the 'composer fund' command to find out more!
> @php artisan vendor:publish --tag=laravel-assets --ansi --force
No publishable resources for tag [laravel-assets].
Publishing complete.
> @php artisan key:generate --ansi
Application key set successfully.

C:\laragon\www>

```

Gambar 3.2.3.2. Instalasi Laravel

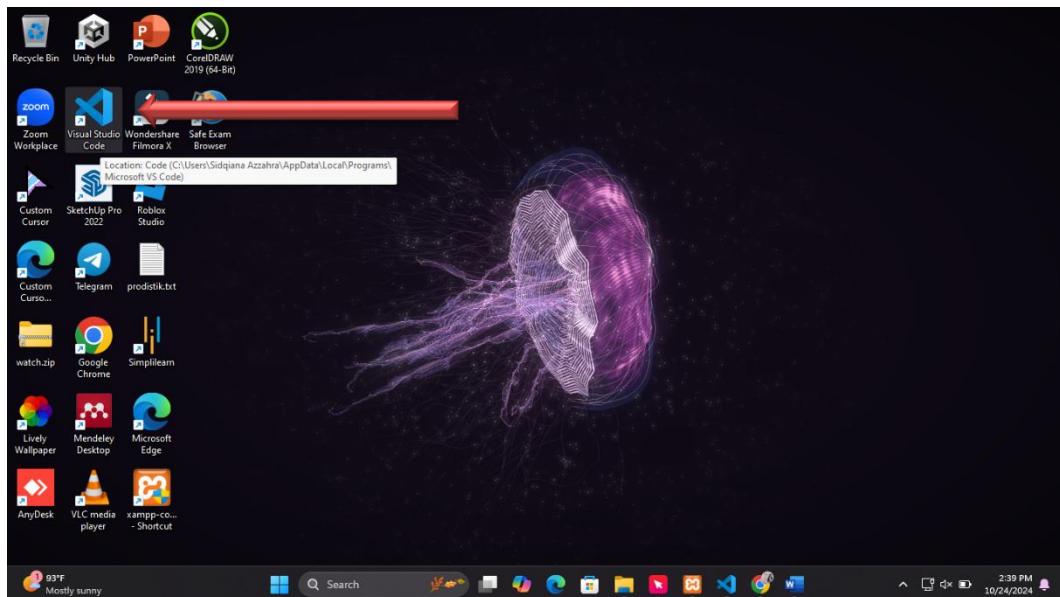
- Setelah proses instalasi, aktifkan laravel dengan perintah *php artisan serve*. Selain melalui cmd, perintah ini juga bisa dijalankan melalui terminal pada aplikasi *Visual Studio Code*. Nanti akan muncul tampilan seperti ini.



Gambar 3.2.3.3. Finishing Install Laravel

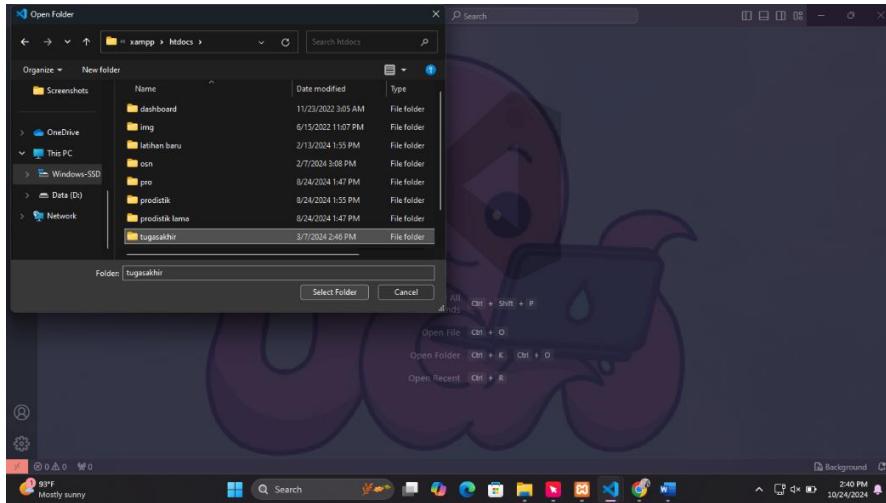
3.2.4 Konfigurasi Database pada .env

- Pada tampilan awal layar, buka aplikasi *Visual Studio Code*.



Gambar 3.2.4.1. Open Visual Studio Code

- Klik *File* pada pojok kiri atas, *Open Folder* atau dengan cara lain tekan Ctrl+K Ctrl+O, kemudian pilih folder *tugasakhir* yang telah disiapkan sebelumnya, tekan *Select Folder*.



Gambar 3.2.4.2. Open Folder

- Buka file `.env`, lalu ketik nama folder `tugasakhir` pada `DB_DATABASE` seperti berikut ini.

```

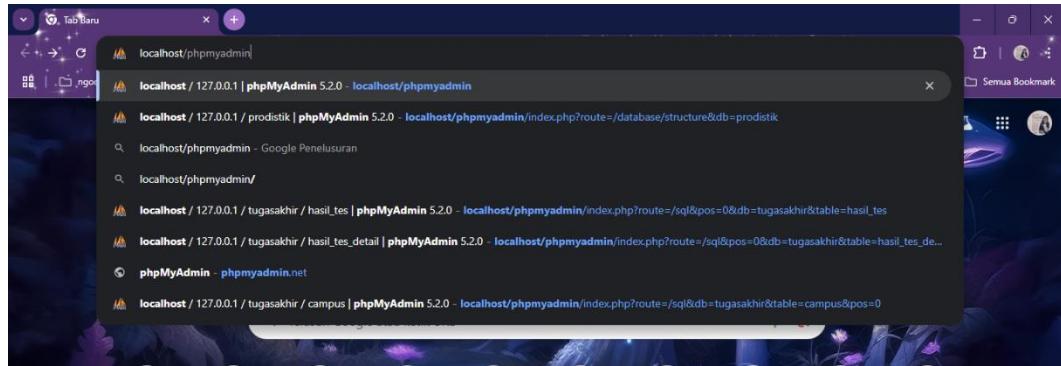
File Edit Selection View Go Run ... ← → Untitled (Workspace)
EXPLORER tugasakhir > .env
1 APP_NAME=Laravel
2 APP_ENV=local
3 APP_KEY=base64:qfh8sXIAntu2N6NUYKuboArUkptUtrdH7x5M8PgZHM-
4 APP_DEBUG=true
5 APP_URL=http://localhost
6
7 LOG_CHANNEL=stack
8 LOG_DEPRECATIONS_CHANNEL=null
9 LOG_LEVEL=debug
10
11 DB_CONNECTION=mysql
12 DB_HOST=127.0.0.1
13 DB_PORT=3306
14 DB_DATABASE=tugasakhir
15 DB_USERNAME=root
16 DB_PASSWORD=
17
18 BROADCAST_DRIVER=log
19 CACHE_DRIVER=file
20 FILESYSTEM_DRIVER=local
21 QUEUE_CONNECTION=sync
22 SESSION_DRIVER=file
23 SESSION_LIFETIME=120
24
25 MEMCACHED_HOST=127.0.0.1
26
27 REDIS_HOST=127.0.0.1
28 REDIS_PASSWORD=null
29 REDIS_PORT=6379
30
31 MAIL_MAILER=smtp
32 MAIL_HOST=mailhog

```

Gambar 3.2.4.3. Pengaturan Root

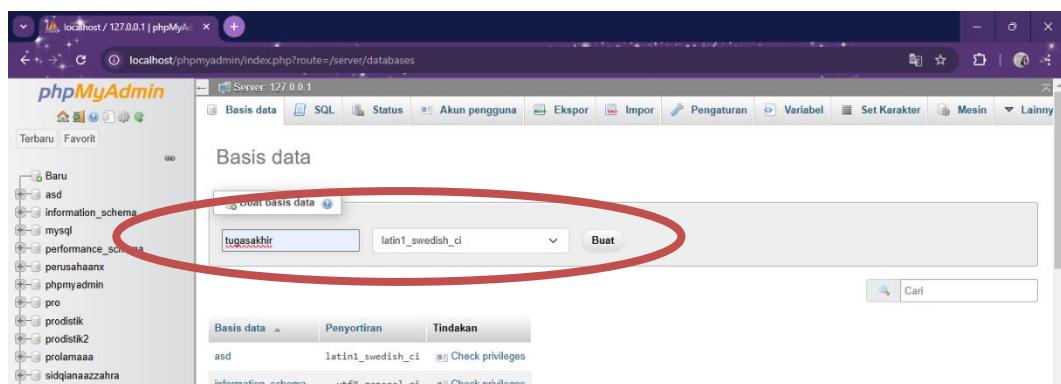
3.2.5 Pembuatan Tabel Database

- Buka Google Chrome, kemudian ketik `http://localhost/phpmyadmin` pada alamat ID. Tekan enter untuk melakukan penelusuran.



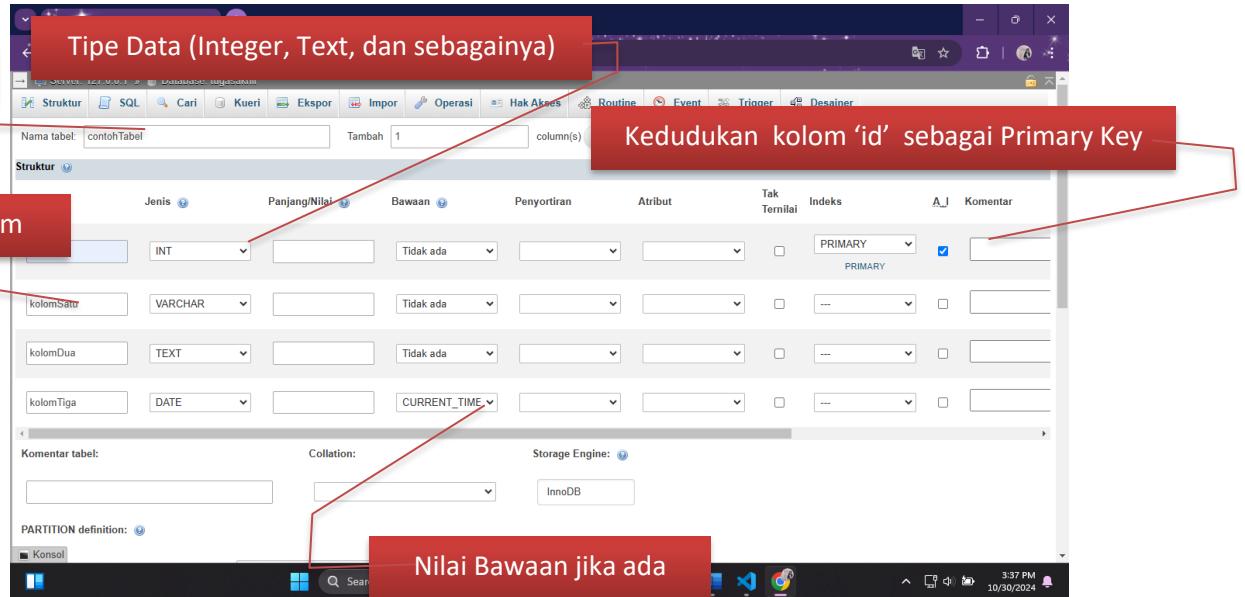
Gambar 3.2.5.1. Penelusuran Database

- Setelah muncul tampilan seperti ini, buat database baru bernama ‘tugasakhir’ dengan menekan tombol *Baru* atau *New*, lalu isi nama basis data pada kolom berikut (tidak diperkenankan menggunakan spasi atau huruf kapital) dan tekan *Buat* atau *Create*.



Gambar 3.2.5.2. Create Database

- Buat tabel yang diperlukan. Dalam website ini, tabel yang diperlukan antara lain tabel *campus*, *courses*, *hasil_tes*, *hasil_tes_detail*, *jawaban*, *jurusan*, *soal*, *testimoni*, *tipe*, dan *users*. Sama dengan sebelumnya, tabel dibuat dengan menekan tombol *Buat* atau *Create*, kemudian isi form di bawah ini. Dalam penamaan tabel dan kolom tidak disarankan menggunakan spasi, melainkan boleh menggunakan tanda ‘_’.



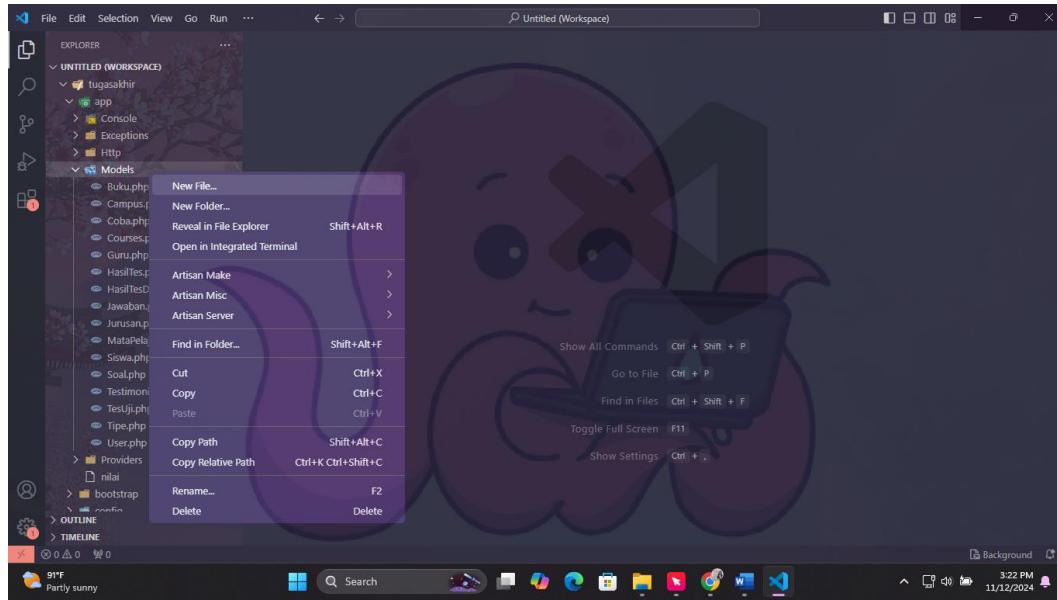
Gambar 3.2.5.3. Create Tabel



Gambar 3.2.5.4. Simpan Tabel

3.2.6 Pemasukan Kolom Database ke Folder App/Models

- Buka kembali aplikasi *Visual Studio Code*, buka folder app\Models, klik kanan, kemudian pilih *New File*



Gambar 3.2.6.1. Buat file Model

- Ketik kode di bawah ini, sesuaikan dengan nama kolom pada masing – masing tabel. Berikut merupakan contoh file model *Jurusan.php* untuk tabel *jurusans*.

```

• <?php
•
• namespace App\Models;
• use Illuminate\Database\Eloquent\Model;
• class Jurusan extends Model
• {
•     protected $table = 'jurusan'; // nama tabel
•     protected $primaryKey = 'id'; // kolom sebagai Primary Key
•     protected $fillable = [ // isi dengan nama kolom
•         'id',
•         'created_at',
•         'updated_at',
•         'title',
•         'content',
•         'files',
•         'activated',
•         'tipeSatu',
•         'tipeDua',
•         'tipeTiga'
•     ];

```

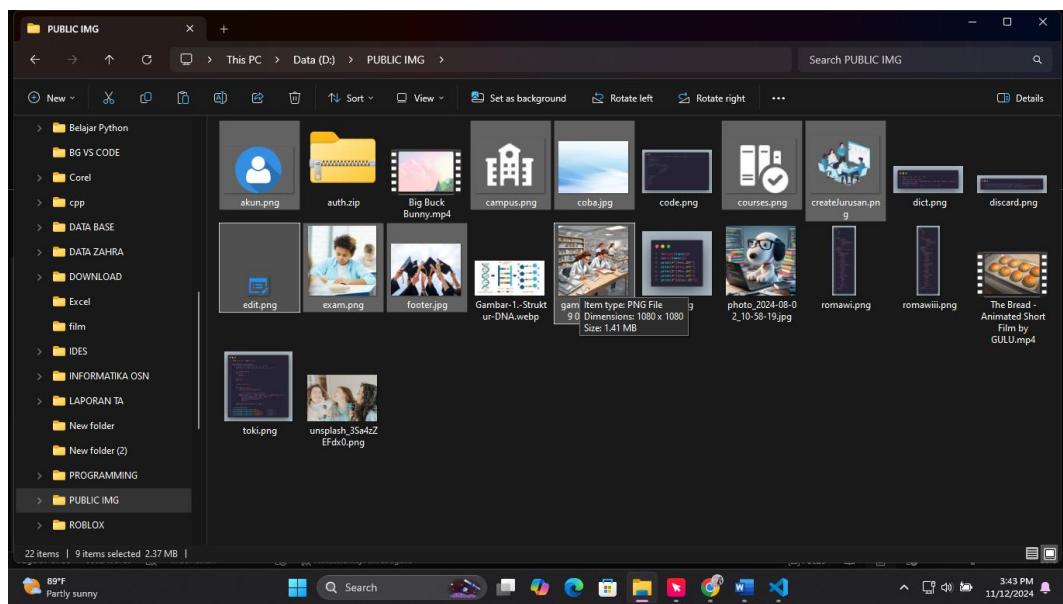
-];
- }

- Lakukan hal yang sama untuk tabel *campus*, *courses*, *hasil_tes*, *hasil_tes_detail*, *jawaban*, *soal*, *testimoni*, *tipe*, dan *users*.

3.2.7 Publikasi File

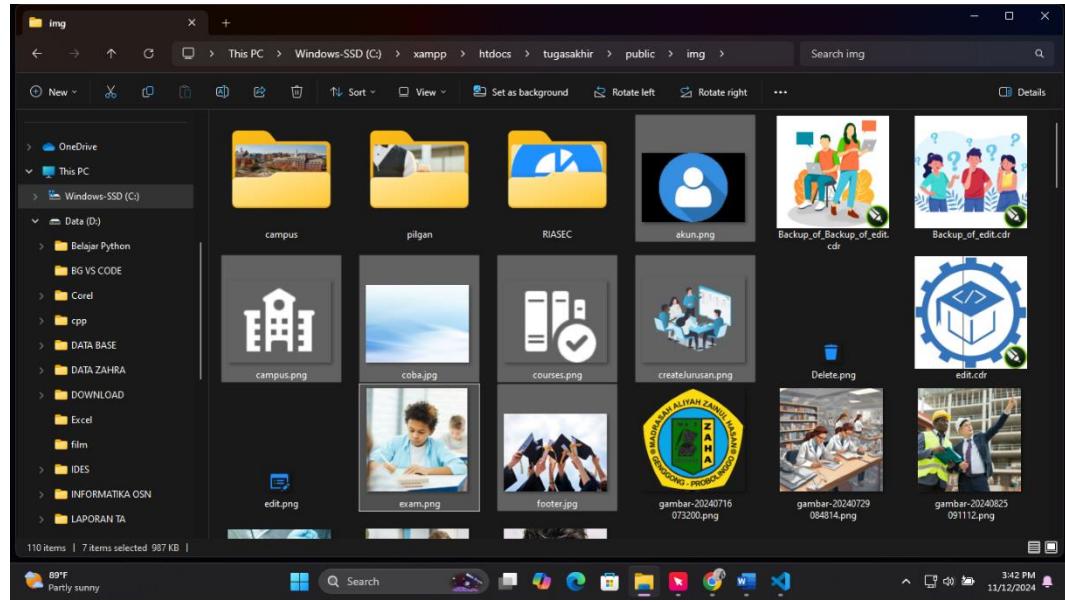
Dalam pembuatan website, pasti memerlukan adanya publikasi file berupa gambar, video, dan sebagainya. Sehingga, penyimpanan file tersebut juga tidak boleh sembarangan. Laravel telah menyediakan folder *public* yang dapat diisi dengan file gambar, video, bahkan *CSS*. Pindahkan file yang diperlukan dengan langkah – langkah berikut.

- Buka folder sesuai penyimpanan file, pilih file mana saja yang diperlukan dalam pembuatan website. Kemudian, copy dengan menekan *Ctrl + C*.



Gambar 3.2.7.1. Copy file ke public

- Pindahkan pada folder *public/img*. Tekan *Ctrl + V* atau klik kanan, *paste*.

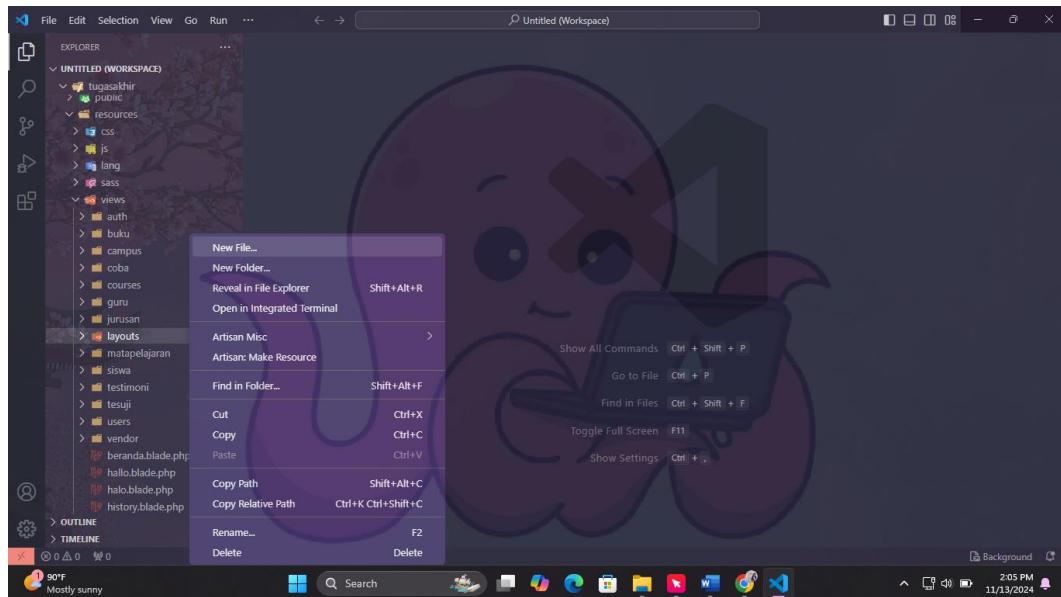


Gambar 3.2.7.2. Paste file di public

3.2.8 Pembuatan Navigasi

Sebelum membuat halaman website, perlu membuat navigasi terlebih dahulu. Tentukan fitur apa saja yang akan disediakan. Berikut tata cara pembuatannya.

- Buka Aplikasi *Visual Studio Code*, klik kanan pada folder resources/views/layouts, pilih *New File* untuk membuat file baru. Beri nama *app.blade.php* pada file tersebut.



Gambar 3.2.8.1. New File Navigation

- Buat Tag <html> sebagaimana berikut.

```
• <!doctype html>
• <html lang="{!! str_replace('_', '-', app()->getLocale()) !!}">
• </html>
```

- Ketik kode di bawah ini sebagai elemen *head*, letakkan di antara tag pembuka <html> dan tag penutup </html>.

```
• <head>
•   <meta charset="utf-8">
•   <meta name="viewport" content="width=device-width, initial-scale=1">
•
•   <!-- CSRF Token -->
•   <meta name="csrf-token" content="{!! csrf_token() !!}">
•
•   <title>{!! config('app.name', 'Laravel') !!}</title>
•
•   <!-- Fonts -->
•   <link rel="dns-prefetch" href="//fonts.gstatic.com">
•   <link href="https://fonts.googleapis.com/css?family=Nunito" rel="stylesheet">
•
•   <!-- Styles -->
•
•   <link href="{!! asset('vendor/bootstrap/css/bootstrap.min.css') !!}" rel="stylesheet">
•
•   </head>
•   <script type="module"
•          src="https://cdn.jsdelivr.net/npm/ldrs/dist/auto/metronome.js"></script>
•
```

- Kemudian, buat elemen <body> berisi kode html tentang fitur apa saja yang akan dimunculkan. Pastikan kode ini tetap berada di dalam elemen tag *html*.

```
• <body>
•   <div class=" sticky-top w-100 h-2">
```

- ```
<div class="container">
 <nav class="navbar navbar-expand-md" style="" aria-label="Fourth navbar example">
 <div class="container-fluid">

 Psychic
 <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarsExample04" aria-controls="navbarsExample04" aria-expanded="false" aria-label="Toggle navigation">

 </button>
 <div class="collapse navbar-collapse" id="navbarsExample04">
 <ul class="navbar-nav me-auto mb-2 mb-md-0">
 <li class="nav-item">
 Home

 <li class="nav-item dropdown">
 Explore
 <ul class="dropdown-menu">
 RIASEC
 College
 Major

 <li class="nav-item">
 Contact

 <li class="nav-item">
```

```

• Psych Test
•
• @if (auth()->check())
• <li class="nav-item">
• <a id="menu" class="nav-link" href="{{ route('history') }}"
• >History
•
• @endif
•
•
•
• @if (auth()->check())
•
• <div class="dropdown-center" style="text-decoration: none">
• <a id="akun" class="nav-link dropdown-toggle fw-bold text-light"
• href="#" data-bs-toggle="dropdown" aria-expanded="false" >{{ auth()->user()->name }}
• <ul class="dropdown-menu">
•
• <a href="{{ route('users.index') }}" id="out" class="dropdown-item"
• style="margin-left: 5%;">Profile
• Logout
•
• </div>
•
• @else
• Login

```

```

• Register
•
• @endif
•
• </div>
•
• </div>
•
• </nav>
•
• </div>
•
•
• <main class="py-0">
• @yield('content')
• </main>
•
•
• <script src="{{ asset('vendor/bootstrap/js/bootstrap.bundle.min.js') }}"></script>
• </body>

```

- Agar tampilan lebih menarik, tambahkan CSS dengan membuat tag `<style></style>` di dalam elemen `head`.

```

• <style>
•
• @import url('https://fonts.cdnfonts.com/css/instrument-sans'); // import font
•
•
• @media only screen and (max-width: 767px){
•
•
• #menu:active{
• color:rgb(255, 255, 255);
• border-radius:15%;
• background-color:#1E90FF;
• box-shadow: 5px 5px 5px rgb(36, 59, 237);
• transition-duration:0.2s;
• }
•
• #akun{
• width:10% ;
• height:auto
• }
•
•
• }

```

```
• #menu:hover{
• border-bottom: 3px white solid;
• font-weight: bold;
• }
• @media only screen and (min-width: 768px){
• #akun{
• margin-left:20%;
• width:5%;height:auto
• }
• }
• }
• }
• #menu{
• color:rgb(255, 255, 255)
• }
• *{
• font-family: 'Instrument Sans', sans-serif;
• }
• #out:hover{
• color: #1E90FF;
• box-shadow: inset 0px 0px 10px 2px rgba(99, 209, 225, 0.4);
• transition-duration: 0.2s;
• }
• .login{
• border-radius:10%;margin-left:8%;text-decoration:none;padding:0.5em;border:
• 0.1em #FFFFFF solid;width:10%;text-align:center;color:#fdfefe;
• }
• .login:hover{
• background-color: rgb(255, 255, 255);
• transition-duration:0.3s;
• color:#1E90FF;;
• }
• .register{
```

```

• border-radius:5%;margin-left:5%;text-
decoration:none;padding:0.5em;width:10%,text-
align:center;color:#1E90FF;;background-color:rgb(255,255,255);
}
• .register:hover{
background-color:#1E90FF;;
color: rgb(255, 255, 255);
}
• transition-duration:0.3s;
• border-radius: 4% ;
• border: 2px #FFFFFF solid
}
• .sticky-top{
background-color: #1E90FF;
}
•
•
• </style>
```

- Hasilnya akan muncul seperti ini.



Gambar 3.2.8.1. Create Navigasi

### 3.2.9 Pembuatan Halaman Website

- Dalam pembuatan halaman website, awali setiap halaman dengan memberikan kode berikut.
- ```

• @extends('layouts.app')
• @section('content')
• // isi dengan kode html
• @endsection
```
- Kemudian, isi kode html sesuai halaman yang akan ditampilkan. Misal, pada halaman *home.blade.php*. Buat elemen *head*.

```

• <head>
• <meta name="viewport" content="width=device-width, initial-scale=1">
• <link rel="stylesheet" href="css/footer.css">
• <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/aos@2.3.4/dist-aos.css" />
• <!-- --- INI TULISAN BERJALAN --- -->
```

- <script src="https://unpkg.com/typed.js@2.1.0/dist/typed.umd.js">
- </script>
- <!-- --- INITIVIKASI --- -->
- <script src="https://unpkg.com/sweetalert/dist/sweetalert.min.js">
- </script>
- </head>
- Selanjutnya, buat elemen <header></header> setelah tag penutup </head>
- <header>
- <div id="myCarousel" class="carousel slide mb-6" data-bs-ride="carousel" data-bs-interval="10000">
- <div class="carousel-indicators">
- <button type="button" data-bs-target="#myCarousel" data-bs-slide-to="0" class="active" aria-label="Geser 1" aria-current="true"></button>
- <button type="button" data-bs-target="#myCarousel" data-bs-slide-to="1" aria-label="Geser 2"></button>
- <button type="button" data-bs-target="#myCarousel" data-bs-slide-to="2" aria-label="Geser 3"></button>
- <button type="button" data-bs-target="#myCarousel" data-bs-slide-to="3" aria-label="Geser 4"></button>
- <button type="button" data-bs-target="#myCarousel" data-bs-slide-to="4" aria-label="Geser 5"></button>
- <button type="button" data-bs-target="#myCarousel" data-bs-slide-to="5" aria-label="Geser 6"></button>
- </div>
-
- <div class="carousel-inner" style="background-image: url(./img/coba.jpg);background-size:cover;background-repeat:no-repeat;" >
- <div class="carousel-item active" >
-
- <div class="container">
- <div class="carousel-caption text-end">
- <h1 id="head" class="text-primary text-end" style="font-family: Vivaldi;">Psychic
- <p id="bodi" class="text-dark text-end fw-bold">Psychic. </p>
- </div>

- `</div>
- `</div>
- `<div style="max-height: 30em" class="carousel-item">
- `
- `</div>
- `<div style="max-height: 30em" class="carousel-item">
- `
- `</div>
- `<div style="max-height: 30em" class="carousel-item">
- `
- `</div>
- `<div style="max-height: 30em" class="carousel-item">
- `
- `</div>
- `<div style="max-height: 30em" class="carousel-item">
- `
- `</div>
- `</div>
- `<button class="carousel-control-prev" type="button" data-bs-target="#myCarousel" data-bs-slide="prev">
- `
- `Sebelumnya
- `</button>
- `<button class="carousel-control-next" type="button" data-bs-target="#myCarousel" data-bs-slide="next">
- `
- `Berikutnya
- `</button>
- `</div>
- `</div>
- `</header>
- Untuk memunculkan isi halaman *home*, tambahkan kode berikut. Biasanya, halaman beranda sebuah website berisi informasi umum terkait website tersebut.
- `<div class="d-flex flex-wrap rounded mt-5">

- [!\[\]\(img/campus.png\)

College](/campus)
- [!\[\]\(img/jurusang.png\)

Major](/jurusan)
- [!\[\]\(img/courses.png\)

RIASEC](/courses)
- [!\[\]\(img/ujiptensi.png\)

Psych Test](/tesiji)

- </div>
- <h2 data-aos="flip-left" class="text-center fw-bold mt-5 text-dark">User Feedback</h2>
- <div id="div" class="gap-3 p-3 overflow-x-auto d-flex" >
- @if(\$tes)
- @foreach (\$tes as \$row)
- <div data-aos="fade-down" class="rounded mt-5 p-3" id="feed">
-
-
- <canvas id="canvas" style="display:none;"></canvas>
- <div data-aos="fade-up" data-aos-easing="linear" data-aos-duration="500" class="overflow-y-auto text-center m-auto mt-4" style="height:31%;width:80%;">
- <q>{!!\$row->content!!}</q>
- </div>
- <div class="person">
- <h4 class="fw-bold text-center mt-3">{!!\$row->title!!}</h4>
- <p class="text-center text-secondary">{!!\$row->Alamat!!}</p>
- </div>
- </div>
- @endforeach
- @endif
-
- </div>
-
- <form action="{{ route('testimoni.create') }}">
- <button data-aos="zoom-out" class="custom-btn btn-4">Give Feedback</button>
- </form>
-
- {{-- <div class="bg-danger" data-aos="fade-right" data-aos-offset="300" data-aos-easing="ease-in-sine" class="categories_left">HAAALOOOOO</div> --}}
- <h2 data-aos="flip-left" class="text-center fw-bold mt-5">Browse Majors</h2>
- <div id="respon" style="display:grid;position:relative">
- @if (\$rows)

```

●      @foreach ($rows as $row)
●      @if($row->id <5)
●          <div data-aos="zoom-in" class="card-group mt-5 rounded-2 shadow-lg"
●              style="padding:5%;background-color:white;">
●              <div class="card-body">
●                  
●                  {{-- <div>{!! $row->activated == 1 ? '<span style="border-bottom-right-radius: 20px;padding:5px 10px;margin:0" class="badge text-bg-success">public</span>' : '<span style="border-bottom-right-radius: 20px;padding:5px 10px;margin:0" class="badge text-bg-danger">private</span>' !!}</div> --}}
●                  <h3 class="fw-bold mt-3" style="font-family:'Lato',sans-serif;">{{$row->title}}</h3>
●                  </div>
●                  <div id="tombol" class="card-footer btn-group rounded-5 py-1 d-flex gap-5">
●                      <div>
●                          <a href="{{ route('jurusan.show', $row->id) }}">Learn More &rarr;</a>
●                      </div>
●                      @if (auth()->check())
●                          @if(auth()->user()->email == 'sidqianaazzahra1@gmail.com')
●                              <div class="d-flex">
●                                  <a href="{{ route('jurusan.edit', $row->id) }}"></a>
●                                  <form method="post" action="{{ route('jurusan.destroy', $row->id) }}"
●                                      id="hapusForm">
●                                      @csrf
●                                      @method('DELETE')
●                                      <button type="submit" style="border: none"></button>
●                                  </form>
●                              </div>
●                          @endif
●                      @endif
●                  </div>

```

```

●      </div>
●      @endif
●      @endforeach
●      @endif
●      </div>
●      <br>
●
●      <form action="jurusan">
●          <button data-aos="zoom-out" class="custom-btn btn-4">Show All</button>
●      </form>
●      <h2 data-aos="flip-left" class="text-center fw-bold mt-5">Browse College</h2>
●      <div id="respon" style="display:grid;position:relative">
●          @if ($campus)
●              @foreach ($campus as $row)
●                  @if($row->id <5)
●                      <div data-aos="zoom-in" class="card-group mt-5 rounded-2 shadow-lg"
●                          style="padding:5%;background-color:white;">
●                          <div class="card-body">
●                              
●                              {{-- <div>{!! $row->activated==1
●
●                  ? '<span style="border-bottom-right-radius: 20px;padding:5px
●                      10px;margin:0" class="badge text-bg-success">public</span>' :
●                  '<span style="border-bottom-right-radius: 20px;padding:5px
●                      10px;margin:0" class="badge text-bg-danger">private</span>' }}--}}
●
●                  <h3 class="fw-bold mt-3" style="font-family:'Lato',sans-serif;">{$row-
●                      >title}</h3>
●                  </div>
●                  <div id="tombol" class="card-footer btn-group rounded-5 py-1 d-flex gap-5" >
●                      <div>
●                          <a href="{{ route('campus.show',$row->id) }}">Learn More &rarr;</a>
●                      </div>
●                      @if (auth()->check())
●                          @if(auth()->user()->email == 'sidqianaazzahra1@gmail.com')
●                          <div class="d-flex">

```

- id) }">
- <form method="post" action="{ route('campus.destroy', \$row->id) }" id="hapusForm">
- @csrf
- @method('DELETE')
- <button type="submit" style="border: none"></button>
- </form>
- </div>
- @endif
- @endif
- </div>
- </div>
- @endif
- @endforeach
- @endif
- </div>
-

-
- <form action="campus">
- <button data-aos="zoom-out" class="custom-btn btn-4">Show All</button>
- </form>

- Akhiri halaman *beranda* dengan <footer></footer> sebagai berikut. *Copy* elemen footer ini di setiap halaman yang telah dibuat.

- <footer class="mt-5">
- <div style="height:auto;background-color:rgba(76, 153, 253, 0.8)">
- <h1 style="text-align:center;font-family: Vivaldi;font-size:8rem;color:blue">Psychic</h1>
- <div class="d-flex justify-content-around flex-wrap">
- <div class="ml-auto p-3" id="foot">
- <h3>Ingin sukses? jangan salah jurusan. Temukan potensimu di website ini...!</h3>
-
- </div>

```

●      <div class="ml-auto p-3" id="foot">
●          <h3 class="fw-bold"><a class="text-decoration-none fw-bold"
● style="color:blue" href="#myCarousel">Fitur</a></h3>
●
●          <ul>
●              <li>Home</li>
●              <li>Explore RIASEC</li>
●              <li>Explore Major</li>
●              <li>Explore College</li>
●              <li>Contact</li>
●              <li>Psych Test</li>
●          </ul>
●      </div>
●      <div class="ml-auto p-3" id="foot">
●          <h3 class="fw-bold"><a class="text-decoration-none fw-bold"
● style="color:blue" href="{{ route('courses.index') }}">RIASEC</a></h3>
●
●          <ul>
●              <li>Dr. John Holland</li>
●              <li>Realistic</li>
●              <li>Investigative</li>
●              <li>Artistic</li>
●              <li>Social</li>
●              <li>Enterprising</li>
●              <li>Conventional</li>
●          </ul>
●      </div>
●      <div class="ml-auto p-3" id="foot">
●          <h3 class="fw-bold"><a class="text-decoration-none fw-bold"
● style="color:blue" href="{{ route('testimoni.create') }}">Contact Us</a></h3>
●
●          <div class="d-flex flex-wrap mb-3">
●              <a href="mailto:azzahrasidqiana22@gmail.com" class="my-auto m-3
● fw-bold text-dark d-flex align-items-center" style="flex-shrink: 1; min-width: 0;">
●                  
●                  <span style="word-break: break-all; margin-left:
● 5px;">azzahrasidqiana22@gmail.com</span>
●          </a>

```

```

●          </div>
●          <div class="d-flex flex-wrap mb-3">
●              <a href="https://www.instagram.com/sidqianaazzahra" class="my-auto
m-3 fw-bold text-dark d-flex align-items-center" style="flex-shrink: 1; min-width:
0;">
●                  
●                  <span style="word-break: break-all; margin-left:
5px;">sidqianaazzahra</span>
●              </a>
●          </div>
●
●          <div class="d-flex flex-wrap mb-3">
●              <a href="https://web.telegram.org/a/#1101791976" class="my-auto m-3
fw-bold text-dark d-flex align-items-center" style="flex-shrink: 1; min-width: 0;">
●                  
●                  <span style="word-break: break-all; margin-left:
5px;">ErvaKurnia</span>
●              </a>
●          </div>
●
●          <div class="d-flex flex-wrap mb-3">
●              <a
href="https://wa.me/6282142635182?text=Halo,%20apa%20ada%20yang%20bisa20
%20saya20%20bantu?20%20senang20%20berbicara20%dengan%20Anda" class="my-auto m-
3 fw-bold text-dark d-flex align-items-center" style="flex-shrink: 1; min-width: 0;">
●                  
●                  <span style="word-break: break-all; margin-left: 5px;">+62 821-
4263-5182</span>
●              </a>
●          </div>
●          </div>
●      </div>
●      <div style="background-color:blue" class="d-flex mt-5 p-3">

```

- © 2024 Psychic Test. All Rights Reserved.</footer>
- </div>
- </div>
- </footer>
- Tambahkan style CSS untuk footer dengan membuat file *footer.css* di dalam folder *public/css*.

```

• footer{
•     background-image: url(http://localhost:8/img/footer.jpg);
•     background-size:cover;
•     background-attachment: fixed;
• }
• @media only screen and (max-width: 767px){
•     #foot {
•         width:100%;
•     }
• }
• @media only screen and (min-width:768px){
•     #foot {
•         width:25%;
•     }
• }
```

- Jika ditampilkan, halaman website masih terlihat berantakan dan kurang menarik. Sehingga, perlu adanya penambahan kode CSS dengan membuat tag <style></style> di dalam elemen <head>.

```

• <style>
•     *{
•         font-family: 'Instrument', sans-serif;
•         color: #000000;
•     }
•     #lengkap{
•         background-color: #1E90FF;
•         color: #FFFFFF;
•         text-decoration: none;
```

```
●      }
●      #lengkap:hover{
●          background-color:#1265b7;
●      }
●      #bodi{
●          max-width: 50%;
●          margin-left: 50%;
●      }
●
●      #create > a{
●          width: 100%;
●          color:white;
●      }
●      .custom-btn {
●          width: 20%;
●          margin-left:40%;
●          height: 40px;
●          color: #fff;
●          border-radius: 5px;
●          padding: 10px 25px;
●          background: transparent;
●          cursor: pointer;
●          transition: all 0.3s ease;
●          position: relative;
●          display: inline-block;
●          box-shadow: inset 2px 2px 2px 0px rgba(255,255,255,.5),
●          7px 7px 20px 0px rgba(0,0,0,.1),
●          4px 4px 5px 0px rgba(0,0,0,.1);
●          outline: none;
●      }
●      .btn-4 {
●          background: #1E90ff;
●          border: none;
●      }
```

```
● .btn-4:before {  
●     height: 0%;  
●     width: 2px;  
● }  
●  
● .btn-4:hover {  
●     box-shadow: 4px 4px 6px 0 rgba(255,255,255,.5),  
●     -4px -4px 6px 0 rgba(116, 125, 136, .5),  
●     inset -4px -4px 6px 0 rgba(255,255,255,.2),  
●     inset 4px 4px 6px 0 rgba(0, 0, 0, .4);  
● }  
●  
● #feed:hover{  
●     border: 2px #1E90FF solid;  
●     transition-duration: 15ms;  
●     box-shadow: none;  
● }  
●  
● }  
●  
● #summary-btn {  
●     background-color: #1E90FF;  
●     color: white;  
●     border: none;  
●     border-radius: 5px;  
●     cursor: pointer;  
● }  
● div::-webkit-scrollbar {  
●     width: 0px;  
● }  
● .card-body img:hover{  
●     transform: scale(1.1);  
●     transition: transform 0.3s ease;  
● }  
● #summary-btn:hover {  
●     background-color: #0056b3;
```

```

•   }
•   #tombol a{
•     color:#1E90ff;
•     text-decoration: none;
•     text-align: right;
•   }
•   #tombol a:hover{
•     color: #1265b7;
•     text-decoration: underline;
•   }
•   #tombol{
•     max-width:100%;
•   }
•   </style>

```

- Agar lebih dinamis, tambahkan pula kode JavaScript dengan membuat tag `<script></script>` setelah elemen `<footer></footer>`.

```

•   <script src="https://cdn.jsdelivr.net/npm/aos@2.3.4/dist-aos.js">
•   </script>
•
•   <script>
•     AOS.init(); // animasi aos
•     var typed = new Typed("#auto-type", {
•       // ini tulisan berjalan
•         strings: ["Search Your Choice College"],
•         typeSpeed: 100, // timer muncul
•         backSpeed: 100, // timer menghilang
•         loop: true,
•       });
•   </script>

```

- Agar halaman website lebih responsive, gunakan `@media only screen and (max-width:767px)`. Sehingga, CSS akan disesuaikan dengan lebar layar yang digunakan. Selain `max-width`, juga dapat menggunakan `min-width`.

```

•   @media only screen and (min-width: 768px){
•     #head{
•       font-size: 10em
•     }
•   }

```

```
●      }
●      #bodi{
●          font-size: 1em;
●      }
●      #g_icon{
●          position: absolute;top: 50%;left: 50%;transform: translate(-50%, -50%);width: 100%;height: auto;
●      }
●      #icon{
●          width:50%;padding-top:50%;position:relative;justify-content:center;align-items:center;margin:auto;
●      }
●      #icon:hover{
●          transform: scale(1.1);
●          transition: transform 0.3s ease;
●          box-shadow: 0px 0px 5px 2px #1E90FF;
●      }
●      #respon {
●          display: grid;
●          grid-template-columns: repeat(4, 1fr);
●          width: 100%;
●          height: auto;
●          gap:0.3em;
●          margin: auto
●
●
●      }
●      #feed{
●          transition-duration: 0.5s;
●          box-shadow: 0px 0px 20px 0px rgba(19, 142, 176, 0.5);
●          min-width:32vw;
●          width: 32vw;
●          max-height:32vw;
●          justify-content: center;
●          box-sizing: border-box;
●      }
```

```
•      #feedb{  
•          width:8vw;  
•          height:8vw;  
•          margin-left:20%;  
•      }  
•  
•      }  
•      @media only screen and (max-width: 767px){  
•          #feed{  
•              transition-duration: 0.5s;  
•              box-shadow: 0px 0px 20px 0px rgba(19, 142, 176, 0.5);  
•              min-width:90vw;  
•              width: 90vw;  
•              max-height:90vw;  
•              justify-content: center;  
•              box-sizing: border-box;  
•          }  
•          form{  
•              font-size: 0.5em;  
•              text-align: center;  
•          }  
•          #feedb{  
•              width:25vw;  
•              height:25vw;  
•              margin-left:20%;  
•          }  
•          .person>h4,.person >p{  
•              font-size:3vw;  
•          }  
•          q{  
•              font-size: 0.7em;  
•              word-spacing: 0.3 em;  
•          }  
•  
•
```

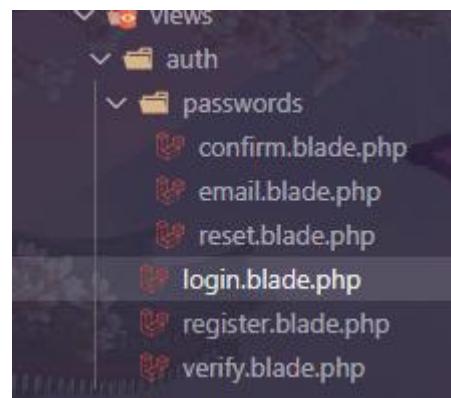
```

•      #head{
•          font-size: 3em;
•      }
•      #body{
•          font-size: 0.5em
•      }
•      #g_icon{
•          position: absolute; top: 50%; left: 50%; transform: translate(-50%, -50%); width: 100%; height: auto;
•      }
•      #icon{
•          width: 50%; padding-top: 50%; position: relative; justify-content: center; align-items: center; margin: auto;
•      }
•      #icon:hover{
•          margin-top: -15%;
•      }
•      #respon {
•          grid-template-columns: repeat(auto-fit, minmax(200px, 1fr));
•          gap: 1em;
•          width: 100%;
•          max-width: 1200px;
•          margin: 0 auto;
•      }
•
•      }

```

- Halaman *home* telah jadi. Namun, halaman tersebut dapat ditampilkan setelah proses pembuatan Controller dan pengaturan route pada pembahasan selanjutnya. Sebelum menuju pembahasan yang dimaksud, berikut ini merupakan halaman lain yang perlu ditampilkan.
- Masih menggunakan cara yang sama, hanya saja file ini dibuat di dalam folder *auth*. Oleh sebab itu, buatlah folder *auth* di dalam folder *resources/views* berisi folder *password* (berisi file *confirm.blade.php*,

email.blade.php, dan reset.blade.php), file login.blade.php, register.blade.php, dan verify.blade.php.



Gambar 3.2.9.1. Pembuatan File Login

- Berikut ini kode yang harus dibuat pada halaman *confirm.blade.php*.

```

• @extends('layouts.app')
•
• @section('content')
• <div class="container">
•   <div class="row justify-content-center">
•     <div class="col-md-8">
•       <div class="card">
•         <div class="card-header">{{ __('Confirm Password') }}</div>
•
•         <div class="card-body">
•           {{ __('Please confirm your password before continuing.') }}
•
•         <form method="POST" action="{{ route('password.confirm') }}">
•           @csrf
•
•           <div class="row mb-3">
•             <label for="password" class="col-md-4 col-form-label text-md-end">{{ __('Password') }}</label>
•
•             <div class="col-md-6">
•               <input id="password" type="password" class="form-control" name="password" required
•                 @error('password') is-invalid @enderror
•                 autocomplete="current-password">

```

```
•         @error('password')
•             <span class="invalid-feedback" role="alert">
•                 <strong>{ { $message } }</strong>
•             </span>
•         @enderror
•     </div>
•
•
•         <div class="row mb-0">
•             <div class="col-md-8 offset-md-4">
•                 <button type="submit" class="btn btn-primary">
•                     {{ __('Confirm Password') }}
•                 </button>
•
•
•             @if (Route::has('password.request'))
•                 <a class="btn btn-link" href="{{ route('password.request') }}"
•             } }">
•                 {{ __('Forgot Your Password?') }}
•             </a>
•         @endif
•
•             </div>
•
•         </div>
•
•         </div>
•
•         </div>
•
•         </div>
•
•     </div>
•
•     </div>
•
•     @endsection
```

- Kode pada halaman *email.blade.php*.

```

●   <h1 class="text-center" style="font-family:Cooper black">{ { __('Reset Password') } }</h1>
●   <div class="container">
●       <div style="display:flex;margin:0">
●           
●           <div style="width:50%">
●               <div>
●                   <div class="card" style="height:580px;background-color:rgba(24, 137, 172, 0.2);border:0px">
●                       <h2 class="text-center m-5">Isi Formulir Berikut</h2>
●
●
●           <div class="card-body">
●               @if (session('status'))
●                   <div class="alert alert-success" role="alert">
●                       { { session('status') } }
●                   </div>
●               @endif
●               <form method="POST" action="{!! route('password.email') !!}">
●                   @csrf
●                   <div class="row mb-3">
●                       <label for="email" class="col-md-4 col-form-label text-md-end">{ { __('Email Address') } }</label>
●
●                       <div class="col-md-6">
●                           <input id="email" type="email" class="form-control" @error('email') is-invalid @enderror name="email" value="{!! old('email') !!}" required autocomplete="email" autofocus>
●
●                           @error('email')
●                               <span class="invalid-feedback" role="alert">
●                                   <strong>{ { $message } }</strong>
●                               </span>
●                           @enderror
●                       </div>
●                   </div>
●                   <div class="row mb-0">

```

- Kode pada halaman *reset.blade.php*.

```

●          <label for="email" class="col-md-4 col-form-label text-md-
end">{ { __('Email Address') } }</label>
●
●          <div class="col-md-6">
●              <input id="email" type="email" class="form-control
@error('email') is-invalid @enderror" name="email" value="{{ old('email') }}"
required autocomplete="email" autofocus>
●
●          @error('email')
●              <span class="invalid-feedback" role="alert">
●                  <strong>{{ $message }}</strong>
●              </span>
●          @enderror
●      </div>
●  </div>
●
●  <div class="row mb-5">
●      <label for="password" class="col-md-4 col-form-label text-md-
end">{ { __('Password') } }</label>
●
●      <div class="col-md-6">
●          <input id="password" type="password" class="form-control
@error('password') is-invalid @enderror" name="password" required
autocomplete="new-password">
●
●          @error('password')
●              <span class="invalid-feedback" role="alert">
●                  <strong>{{ $message }}</strong>
●              </span>
●          @enderror
●      </div>
●  </div>
●
●  <div class="row mb-3">
●      <label for="password-confirm" class="col-md-4 col-form-label
text-md-end">{ { __('Confirm Password') } }</label>

```

```

•
•           <div class="col-md-6">
•               <input id="password-confirm" type="password" class="form-
control" name="password_confirmation" required autocomplete="new-password">
•           </div>
•       </div>
•       <div class="row mb-0">
•           <div class="col-md-6 offset-md-4">
•               <button type="submit" class="btn btn-primary">
•                   {{ __('Reset Password') }}
•               </button>
•           </div>
•       </div>
•   </form>
• </div>
• @endsection

```

- Kode pada halaman *login.blade.php*.

```

•     @extends('layouts.app')
•
•     @section('content')
•         <style>
•             @media only screen and (min-width: 1101px){
•                 .d-grid{
•                     grid-template-columns: repeat(2,1fr);
•                     height:100%;
•
•                 }
•                 #left{
•                     width:100%;
•                 }
•             }

```

```
•
•
•    }
•    #left{
•        background-color: #1E90FF;}
•    @media only screen and (max-width:1100px){
•        .d-grid{
•            grid-template-columns: repeat(1,1fr);
•            width: 100%;
•            height:auto;
•        }
•        #left{
•            background-color: #1E90FF;
•            width:100%;
•        }
•        #right{
•            height:100%
•        }
•        #gmb{
•            width: 100vw;
•        }
•    }
•    #id{
•        text-decoration: none;
•    }
•    #id:hover{
•        text-decoration: underline;
•    }
•
•
•    </style>
•    <div class="d-grid">
•        <div id="right">
•            
•        </div>
•        <div id="left" class="p-5">
```

```

●
●      <h2 class="text-center fw-bold mb-5 text-light fs-1">{{__('Entry Your
  Account')}}</h2>
●
●
●      <div class="card-body">
●          <form method="POST" action="{{ route('login') }}">
●              @csrf
●
●          <div class="row mb-3">
●              <label for="email" class="col-form-label text-md-start text-light fs-
  4">{{__('Email Address')}}</label>
●
●          <div>
●              <input placeholder="example@gmail.com" id="email"
  type="email" class="form-control @error('email') is-invalid @enderror"
  name="email" value="{{ old('email') }}" required autocomplete="email" autofocus>
●
●              @error('email')
●                  <span class="invalid-feedback" role="alert">
●                      <strong>{{ $message }}</strong>
●                  </span>
●                  @enderror
●              </div>
●          </div>
●
●          <div class="row mb-2">
●              <label for="password" class="text-light fs-4 col-form-label text-
  md-start">{{__('Password')}}</label>
●
●          <div>
●              <input placeholder="*****" id="password"
  type="password" class="form-control @error('password') is-invalid @enderror"
  name="password" required autocomplete="current-password">
●
●              @error('password')

```

```

•           <span class="invalid-feedback" role="alert">
•               <strong>{{ $message }}</strong>
•           </span>
•           @enderror
•       </div>
•   </div>
•   <div class="d-flex">
•       <div class="float-left">
•           <a id='link' class="btn btn-link text-light" href="{{ route('register') }}"
•               >{{ __('Dont Have Account? ') }}</a>
•       </div>
•       <div class="float-right">
•           <a id='link' class="btn btn-link text-light" href="{{
•               route('password.request') }}">{{ __('Forgot Your Password?') }}</a>
•       </div>
•   </div>
•   <div class="row mb-5">
•
•       <div class="form-check">
•           <input class="form-check-input m-auto" type="checkbox"
• name="remember" id="remember" {{ old('remember') ? 'checked' : '' }}>
•
•           <label class="form-check-label text-light fs-6"
• for="remember">{{ __('Remember Me') }}</label>
•
•       </div>
•   </div>
•
•   <button type="submit" class="btn btn-primary w-100">
•       {{ __('Login') }}</button>

```

```

•           </button>
•
•           @if (Route::has('password.request'))
•
•               @endif
•
•           </form>
•       </div>
•   @endsection
•

```

- Kode pada halaman *register.blade.php*.

```

•   @extends('layouts.app')
•
•   @section('content')
•       <style>
•           *{
•               color:#ffffff;
•           }
•
•           @media only screen and (min-width: 1001px){
•               .d-grid{
•                   grid-template-columns: repeat(2, 1fr);
•                   height:auto;
•
•               }
•               #right{
•                   width:50%;
•               }
•               #left{
•                   width:50%;
•                   padding:auto;
•               }
•
•           }
•           #left{
•               background-color: #1E90FF;
•
•           }

```

```

●      width:100%;

●      }

● @media only screen and (max-width:1000px){

● .d-grid{
●     grid-template-columns: repeat(1,1fr);
●     width: 100%;
●     height:auto;
● }
● #gmb{
●     width: 100%
● }
● }
● </style>
● <div class="d-grid">
●   <div id="right">
●     
●   </div>
●   <div id="left" class="p-5 m-auto">
●
●     <h2 class="text-center fw-bold mb-3">{ { __('Create Free Account')
} }</h2>
●
●
●   <div class="card-body">
●     <form method="POST" action="{ { route('register') } }">
●       @csrf
●       <div class="row mb-3">
●         <label for="name" class="col-form-label text-md-start">{ {
●           __('Name') } }</label>
●
●         <div>
●           <input placeholder="Otong Surotong" id="name" type="text"
●             class="form-control @error('name') is-invalid @enderror" name="name" value="{
●               old('name') }" required autocomplete="name" autofocus>
●
●           @error('name')

```

```

•           <span class="invalid-feedback" role="alert">
•               <strong>{{ $message }}</strong>
•           </span>
•           @enderror
•       </div>
•   </div>
•   <div class="row mb-3">
•       <label for="email" class="col-form-label text-md-start">{{
•           __('Email Address') }}</label>
•
•       <div>
•           <input placeholder="example@gmail.com" id="email"
• type="email" class="form-control @error('email') is-invalid @enderror"
• name="email" value="{{ old('email') }}" required autocomplete="email">
•
•           @error('email')
•               <span class="invalid-feedback" role="alert">
•                   <strong>{{ $message }}</strong>
•               </span>
•           @enderror
•       </div>
•   </div>
•
•   <div class="row mb-3">
•       <label for="password" class="col-form-label text-md-start">{{
•           __('Password') }}</label>
•
•       <div>
•           <input placeholder="*****" id="password"
• type="password" class="form-control @error('password') is-invalid @enderror"
• name="password" required autocomplete="new-password">
•
•           @error('password')
•               <span class="invalid-feedback" role="alert">
•                   <strong>{{ $message }}</strong>
•               </span>

```

- Kode pada halaman *verify.blade.php*.

```

●    @section('content')
●    <div class="container">
●        <div class="row justify-content-center">
●            <div class="col-md-8">
●                <div class="card">
●                    <div class="card-header">{{ __('Verify Your Email Address') }}</div>
●
●
●                <div class="card-body">
●                    @if (session('resent'))
●                        <div class="alert alert-success" role="alert">
●                            {{ __('A fresh verification link has been sent to your email address.') }}
●                        </div>
●                    @endif
●
●
●                    {{ __('Before proceeding, please check your email for a verification link.') }}
●
●                    {{ __('If you did not receive the email') }},
●                    <form class="d-inline" method="POST" action="{{ route('verification.resend') }}">
●                        @csrf
●                        <button type="submit" class="btn btn-link p-0 m-0 align-baseline">{{ __('click here to request another') }}</button>
●                    </form>
●                </div>
●            </div>
●        </div>
●    </div>
●    @endsection

```

- Buat halaman profil untuk setiap *user* yang telah melakukan login. Buat folder *users* berisi file *index.blade.php*, *view.blade.php*, dan *show.blade.php* di dalam folder *views*. Tampilan informasi singkat pada setiap *user* diletakkan di dalam file *index.blade.php* sebagaimana berikut.
- @extends('layouts.app')

```

●  @section('content')
●  <style>
●      @media (min-width:767px){
●          #btn {
●              height: 38px
●          }
●      }
●      #foto {
●          object-fit: cover;
●          object-position: center;
●          width: 20vw;
●          height: 20vw
●      }
●  </style>
●  <link rel="stylesheet" href="css/footer.css">
●  <div class="container mt-5">
●      <div class="row">
●          <div class="col-12">
●              <div class="card">
●                  <div class="card-body d-flex p-4 h-auto flex-wrap">
●                      <div class="d-inline">
●                          
●                          <form method="POST" action="{{ route('users.deleteProfile',
●                             auth()->user()->id) }}" enctype="multipart/form-data">
●                              @csrf
●                              <button style="
●                                  margin-top:-5%;
●                                  margin-left:15%;
●                              " class="btn btn-danger position-absolute rounded-circle"
● title="Hapus Foto Profil" name="profil" value="hapus"></button>
●                      </form>
●                  </div>
●                  <div class="d-inline m-auto">
●                      <h3 class="fw-bold">{{ $mine->name }}</h3>

```

```

•          <h6 class="fw-bold">Email</h6>
•          <h6>{ ${mine->email} }</h6>
•      </div>

•
•          <a id="btn" href="{ route('users.edit',$mine->id) }"
•              class="btn btn-primary ml-auto rounded-0">
•              Profile Update &rarr;
•          </a>
•          <form method="post" action="{ route('users.destroy',$mine->id) }"
•              id="hapusForm">
•              @csrf
•              @method('DELETE')
•              <button type="submit" class="btn btn-danger ml-auto rounded-
0">Account Delete (-)</button>
•          </form>
•      </div>
•      </div>
•      </div>
•      </div>
•      </div>
•      </div>
•      <div class="row">
•          <div class="col-12">
•              <div class="card m-5">
•                  <div class="card-body">
•                      <table class="table table-striped table-bordered">
•                          <thead>
•
•                              <tr>
•                                  <th>No</th>
•                                  <th>Photo</th>
•                                  <th>Users</th>
•                                  <th>Email</th>
•                                  <th></th>
•                              </tr>
•                          </thead>

```

```

•      <tbody>
•          @php
•              $no =1;
•          @endphp
•          @foreach ($rows as $row)
•              <tr>
•                  <td>{ ${$no++} }</td>
•                  <td></td>
•                  <td style="word-break: break-all;">{ {$row->name} }</td>
•                  <td style="word-break: break-all;">{ {$row->email} }</td>
•              </tr>
•          @endforeach
•      </tbody>
•  </table>
•  </div>
•  </div>
•  </div>
•  </div>
•  <script>
•      document.querySelectorAll('#hapusForm').forEach(form=> {
•          form.addEventListener('submit', function(event) {
•              var konfirmasi = confirm('Yakin?');
•              if (!konfirmasi) {
•                  event.preventDefault();
•              }
•          });
•      });
•  </script>
•  @endsection

```

- Sedangkan pada file *update.blade.php* berisi formulir data *user* yang sedang login. Pada halaman ini *user* dapat mengubah data dirinya. Sehingga, dibutuhkan elemen <form></form> untuk menginput perubahan yang diinginkan. Jika *type input* di dalamnya terdapat *type* berupa file, tambahkan *enctype="multipart/form-data"*. Berikut kode selengkapnya.

```
● @extends('layouts.app')
● @section('content')
● <style>
●   label {
●     margin: 2vw 0;
●     font-weight: bold;
●     font-size: 1.2rem
●   }
●   .custom-btn {
●     width: 20%;
●     height: 40px;
●     color: #fff;
●     border-radius: 5px;
●     padding: 10px 25px;
●     background: transparent;
●     cursor: pointer;
●     transition: all 0.3s ease;
●     position: relative;
●     display: inline-block;
●     box-shadow: inset 2px 2px 2px 0px rgba(255, 255, 255, .5),
●     7px 7px 20px 0px rgba(0, 0, 0, .1),
●     4px 4px 5px 0px rgba(0, 0, 0, .1);
●     outline: none;
●   }
●
●   .btn-4 {
●     background: #1E90ff;
●     border: none;
●   }
● 
```

```

●   .btn-4:before {
●       height: 0%;
●       width: 2px;
●   }
●
●   .btn-4:hover {
●       box-shadow: 4px 4px 6px 0 rgba(255,255,255,.5),
●       -4px -4px 6px 0 rgba(116, 125, 136, .5),
●       inset -4px -4px 6px 0 rgba(255,255,255,.2),
●       inset 4px 4px 6px 0 rgba(0, 0, 0, .4);
●   }
●   </style>
●   <div class="container mt-5">
●       <h1 class="text-center fw-bold">Profile Update</h1>
●       <form method="post" action="{!! route('users.update', $row->id) }" enctype="multipart/form-data">
●           <input type="hidden" name="_token" value="{!! csrf_token() !!}">
●           @method('PUT')
●           <div class="form-group">
●               <label for="name">Nama Pengguna</label>
●               <input type="text" class="form-control" name="name" value="{!! $row->name !!}">
●           </div>
●           <div class="form-group">
●               <label for="profil">Foto Profil</label>
●               <input type="file" class="form-control" name="profil" value="{!! $row->profil !!}">
●           </div>
●           <div class="form-group">
●               <label for="email">Email</label>
●               <input type="email" class="form-control" name="email" value="{!! $row->email !!}">
●           </div>
●           <div class="form-group">
●               <label for="password">Password</label>

```

- <input type="password" class="form-control" name="password" value="{!! \$row->password!!}">
- </div>
-
-

- <button type="submit" class="custom-btn btn-4 my-5 w-100">Update</button>
- </form>
- </div>
- @endsection

- Sedangkan kode pada halaman *view.blade.php* sebagai berikut.

- @extends('layouts.app')
- @section('content')
- <div class="container">
- <form method="post" action="{{ route('users.update', \$row->id) }}">
- <input type="hidden" name="_token" value="{{ csrf_token() }}"/>
- @method('PUT')
- <div class="form-group">
- <label for="name">Nama Pengguna</label>
- <input type="text" class="form-control" name="name" value="{!! \$row->name!!}" readonly>
- </div>
- <div class="form-group">
- <label for="email">Email</label>
- <input type="email" class="form-control" name="email" value="{!! \$row->email!!}" readonly>
- </div>
- <div class="form-group">
- <label for="password">Password</label>
- <input type="password" class="form-control" name="password" value="{!! \$row->password!!}" readonly>
- </div>
-

- <button type="submit" class="btn btn-primary">Perbarui</button>
- </form>
- </div>

- @endsection
- Kemudian, tampilkan soal Tes Psikologi. Buka folder *resources/views*, buat folder baru *tesaji*, kemudian buat 4 file baru dengan nama *soal.blade.php*, *show.blade.php*, *update.blade.php*, dan *detresult.blade.php*.
- Berikut ini kode untuk halaman *soal.blade.php*.

```

● @extends('layouts.app')
● @section('content')
● <head>
●   <script src="https://unpkg.com/sweetalert/dist/sweetalert.min.js"></script>
●   <link rel="stylesheet" href="css/footer.css">
●   <link href="https://unpkg.com/aos@2.3.1/dist-aos.css" rel="stylesheet">
● </head>
● @if ($rows)
● @foreach ( $rows as $row )
●   <div class="w-75 m-auto mt-5 shadow-lg rounded-3 p-5">
●     <div class="d-flex justify-content-center">
●       
●     </div>
●     <div data-aos="fade-up" data-aos-duration="1000">
●       <h1 class="fw-bold m-5 text-center">Welcome in This Test...!</h1>
●       <p class="text-center">Tes psikologi adalah prosedur yang digunakan untuk mengukur dan mengevaluasi berbagai aspek psikologi seseorang, seperti kognisi, emosi, kepribadian, sikap, perilaku, dan fungsi mental lainnya. Tujuan dari tes ini adalah untuk mendapatkan informasi tentang karakteristik dan potensi psikologis seseorang sehingga mereka dapat lebih memahami diri mereka lebih baik . .
●       Tes psikologi digunakan dalam berbagai konteks, seperti konseling, penilaian klinis, pendidikan, seleksi karyawan, dan penelitian. Tes ini dirancang dengan standar ilmiah yang ketat untuk memastikan bahwa hasilnya benar, valid, dan akurat.</p>
●   </div>
●   <div class="d-flex justify-content-end">
●     <button
●       onclick="notifikasi()"
●       class="btn btn-primary text-light fw-bold py-2 px-4 rounded">
●       Start Now &rarr;;

```

```

●      </button>
●      </div>
●      </div>
●      <footer class="mt-5">
●          <div style="height:auto;background-color:rgba(76, 153, 253, 0.8)">
●              <h1 style="text-align:center;font-family: Vivaldi;font-size:8rem;color:blue">Psychic</h1>
●          <div class="d-flex justify-content-around flex-wrap">
●              <div class="ml-auto p-3" id="foot">
●                  <h3>Ingin sukses? jangan salah jurusan. Temukan potensimu di website ini...!</h3>
●                  
●              </div>
●              <div class="ml-auto p-3" id="foot">
●                  <h3 class="fw-bold"><a class="text-decoration-none fw-bold" style="color:blue" href="#myCarousel">Fitur</a></h3>
●                  <ul>
●                      <li>Home</li>
●                      <li>Explore RIASEC</li>
●                      <li>Explore Major</li>
●                      <li>Explore College</li>
●                      <li>Contact</li>
●                      <li>Psych Test</li>
●                  </ul>
●              </div>
●              <div class="ml-auto p-3" id="foot">
●                  <h3 class="fw-bold"><a class="text-decoration-none fw-bold" style="color:blue" href="{{ route('courses.index') }}">RIASEC</a></h3>
●                  <ul>
●                      <li>Dr. John Holland</li>
●                      <li>Realistic</li>
●                      <li>Investigative</li>
●                      <li>Artistic</li>
●                      <li>Social</li>
●                      <li>Enterprising</li>
●                      <li>Conventional</li>
●                  </ul>
●              </div>

```

```

●      </ul>
●      </div>
●      <div class="ml-auto p-3" id="foot">
●          <h3 class="fw-bold"><a class="text-decoration-none fw-bold"
● style="color:blue" href="{{ route('testimoni.create') }}">Contact Us</a></h3>
●          <div class="d-flex flex-wrap mb-3">
●              <a href="mailto:azzahrasidqiana22@gmail.com" class="my-auto m-3 fw-
● bold text-dark d-flex align-items-center" style="flex-shrink: 1; min-width: 0;">
●                  
●                  <span style="word-break: break-all; margin-left:
● 5px;">azzahrasidqiana22@gmail.com</span>
●          </a>
●          </div>
●          <div class="d-flex flex-wrap mb-3">
●              <a href="https://www.instagram.com/sidqianaazzahra" class="my-auto m-
● 3 fw-bold text-dark d-flex align-items-center" style="flex-shrink: 1; min-width: 0;">
●                  
●                  <span style="word-break: break-all; margin-left:
● 5px;">sidqianaazzahra</span>
●          </a>
●          </div>
●
●          <div class="d-flex flex-wrap mb-3">
●              <a href="https://web.telegram.org/a/#1101791976" class="my-auto m-3
● fw-bold text-dark d-flex align-items-center" style="flex-shrink: 1; min-width: 0;">
●                  
●                  <span style="word-break: break-all; margin-left:
● 5px;">ErvaKurnia</span>
●          </a>
●          </div>
●
●          <div class="d-flex flex-wrap mb-3">

```

- ```
• <a
• href="https://wa.me/6282142635182?text=Halo,%20apa%20ada%20yang%20bisa%20
• %saya%20%bantu%20%senang%20berbicara%20%dengan%20Anda" class="my-auto m-
• 3 fw-bold text-dark d-flex align-items-center" style="flex-shrink: 1; min-width: 0;">>
•
• +62 821-4263-
• 5182
•
• </div>
• </div>
• </div>
• <div style="background-color:blue" class="d-flex mt-5 p-3">
• © 2024 Psychic Test. All Rights
• Reserved.<footer>
• </div>
• </div>
• </footer>
• <script src="https://unpkg.com/aos@2.3.1/dist-aos.js"></script>
• <script>
• AOS.init();
• function notifikasi() { // menampilkan notifikasi
• swal({
• content: {
• element: "div",
• attributes: {
• innerHTML: `
•
• <h1>Warning...!</h1>
•
• <li style="text-align:left">Pastikan ada jaringan internet dan Anda
• sudah melakukan login
• <li style="text-align:left">Terdapat 2 bagian soal yang terdiri dari
• 60 soal Pendalaman minat dan 60 soal Perbandingan karakter
```

```

● <li style="text-align:left">Tidak ada jawaban benar/salah, silakan
 pilih jawaban yang sesuai dengan diri Anda masing-masing
● <li style="text-align:left">Tidak ada batas waktu mengerjakan,
 namun tes ini biasanya memakan waktu kurang lebih 30 menit
● <li style="text-align:left">Pastikan semua jawaban telah diisi untuk
 mendapatkan hasil maksimal sesuai kepribadian yang Anda miliki
● <li style="text-align:left">Anda dapat maju ataupun mundur,
 terdapat daftar soal, namun tidak ada pengingat bahwa soal tersebut telah
 dikerjakan
● <li style="text-align:left">Setiap mengerjakan 1 soal, tekan tombol
 "submit" hingga muncul notifikasi "Terkirim". Kemudian, Anda baru boleh menekan
 tombol "Next" untuk lanjut pada soal berikutnya <li
 style="text-align:left">Setelah menjawab semua soal, Anda dapat menekan tombol
 "Akhir Tes" pada halaman soal terakhir, kemudian tombol "Lihat Hasil" untuk
 menemukan hasilnya"<!--li>
● <li style="text-align:left">Anda harus mengakhiri Tes ini dengan
 menekan tombol 'Akhir Tes' di akhir pertanyaan
● <li style="text-align:left">Jika Anda tidak menekannya, Tes
 dianggap belum selesai dan dapat Anda lanjutkan di lain waktu
● <li style="text-align:left">Jika Anda sudah siap, tekan tombol "Yes,
 I'm Agree"<!--li>
●
● <p>Selamat mengerjakan...!</p>
● `

● }
● },
● button: "Yes, I'm Agree"
●).then((value) => {
● window.location.href = "{ { route('tesuji.show', $row->id) } }";
●);
●
● }
● </script>
● @endforeach
● @endif
● @endsection
</pre>

```

- Berikut ini kode untuk halaman *show.blade.php*.

```
● @extends('layouts.app')
● @section('content')
● <head>
● <style>
● * {
● font-family: 'Instrument', sans-serif;
● }
●
● @media screen and (min-width: 768px) {
● #list {
● max-height: 25vw;
● margin-left: -10%;
● grid-template-columns: repeat(3, 1fr);
● position: absolute;
● max-width: 25%;
● }
● .radio-button {
● display: inline-block;
● box-shadow: 10px 10px 10px rgba(161, 180, 225, 0.4);
● border: 1px solid;
● width: 22%;
● border-radius: 10px;
● color: #1E90FF;
● cursor: pointer;
● font-size: 2vw;
● text-align: center;
●
● gap: 2%;
● }
● #format2 {
● display: inline-block;
● box-shadow: 10px 10px 10px rgba(161, 180, 225, 0.4);
● border: 1px solid;
● width: 45%;
```

```
• height: 100%;
• border-radius: 10px;
• color: #1E90FF;
• cursor: pointer;
• font-size: 2vw;
• text-align: center;
•
• gap:2%;
• }
•
• }
```

•

•

•

•

•

•

•

•

•

• @media screen and (max-width: 767px) {

• #list {

• max-height: 25vw;  
• margin-left: -10%;  
• grid-template-columns: repeat(1, 1fr);  
• position: absolute;  
• max-width: 100%;

• }

• .radio-button {

• display: inline-block;  
• box-shadow: 20px 20px 20px black;  
• width: 90%;  
• border-radius: 10px;  
• color: #1E90FF;  
• cursor: pointer;  
• font-size: 1.5rem;  
• text-align: center;  
• box-shadow: 0px 0px 10px 0px #1e90ff;  
• gap:2%;

• }

```
• }
•
•
•
•
• .radio-button input[type="radio"] {
• display: none;
• }
•
• .radio-button:hover {
• background-color: rgba(161, 180, 225, 0.2);
• box-shadow: 0px 0px 10px 0px #1e90ff;
• }
• .radio-button input[type="radio"]:checked + div {
• border: 2px solid #1E90FF;
• background-color: #1E90FF;
• color: white;
• width: 20%;
• height: auto;
• border-radius: 50%;
• margin: auto;
• }
•
• .container {
• display: flex;
• justify-content: space-around;
• }
•
• details {
• width: auto;
• margin-left: auto;
• }
• #pilgan{
• max-width:100%;
• }
• .radio-button input[type="radio"]:checked + #huruf{
```

```

● border-radius:10px;
● }
●
● </style>
● <script src="https://unpkg.com/sweetalert/dist/sweetalert.min.js"></script>
● </head>
●
● <h1 class="text-center m-5 fw-bold text-dark">Psychic Test</h1>
● <div class="bg-light shadow-lg p-5 rounded-3">
● <div class="container">
●
● <h3>No: {!! $row->id !!}/120</h3>
●
● @if (auth()->check())
● @if(auth()->user()->email == 'sidqianaazzahra1@gmail.com')
● <div class="mx-3">
● id) }}">
● </div>
● @endif
● @endif
● <details>
● <summary class="text-light" id="summary-btn">
●
● </summary>
● <div class="shadow-lg p-3 overflow-y-auto rounded-2 d-grid" id="list">
●
● @for ($list = 1; $list <= 120; $list++)
● <a class="text-light" style="text-decoration: none"
● href="/tesuji/{{ $list }}">
● <div class="w-auto bg-primary m-2 p-2 rounded-3">{{ $list }}</div>
●
● @endfor
● </div>
● </details>

```

```

● </div>
● <hr>
● <h1 class="m-5 text-dark">{!! $row->soal !!}</h1>
● <form method="get" id="jawab" action="{{ route('tesuji.show', $id) }}>
● @if ($row->id <= 60)
● <div id="pilgan" class="d-flex flex-wrap">
● @foreach($jawaban as $pilih)
● <label class="radio-button p-4 mx-auto">
● <input type="radio" name='pilgan' value='{!! $pilih->huruf!!}'>
● <div>{!! $pilih->huruf !!}</div>
●
● <p class="text-dark" >{!! $pilih->jawaban !!}</p>
● </label>
● @endforeach
● </div>
● @elseif ($row->id==120)
● <div id="pilgan" class="d-flex flex-wrap">
● @foreach($jawaban as $pilih)
● <label class="radio-button p-4 mx-auto" id="format2">
● <input type="radio" name='pilgan' value='{!! $pilih->huruf!!}'>
● <div style="margin:auto" id="huruf">{!! $pilih->huruf !!}</div>
●
● <p class="text-dark">{!! $pilih->jawaban !!}</p>
● </label>
● @endforeach
● </div>
● @else
● <div id="pilgan" class="d-flex flex-wrap">
● @foreach($jawaban as $pilih)
● <label class="radio-button p-4 mx-auto" id="format2">
● <input type="radio" name='pilgan' value='{!! $pilih->huruf!!}'>
● <div style="margin:auto" id="huruf">{!! $pilih->huruf !!}</div>
●

```

```

• <p class="text-dark">{ !! $pilih->jawaban !!}</p>
• </label>
• @endforeach
• </div>
• @endif
•

•

• @if ($row->id > 1 && $row->id < 120)
• <div class="d-flex justify-content-between">
• id - 1) !!}" class="btn btn-primary">
Previous <<
• <button onclick="terkirim()" type="submit" class="btn btn-primary ml-
auto">Submit</button>
• id + 1) !!}" class="btn btn-success
ml-auto"> Next >>
• </div>
• @elseif ($row->id == 120)
• {{-- <button value="Selesai" name="status">Akhiri Tes</button> --}}
• <div class="d-flex justify-content-between">
• id - 1) !!}" class="btn btn-primary">
Previous <<
• <button onclick="terkirim()" class="btn btn-primary"
type="submit">Submit</button>
• <div
• onclick="Shownotifikasi()"
• class="btn btn-danger text-light fw-bold py-2 px-4 rounded">
Lihat Hasil
• </div>
• </div>
• @else
• <div class="d-flex justify-content-between">
• Preview <<
• <button onclick="terkirim()" type="submit" class="btn btn-
primary">Submit</button>
• id + 1) !!}" class="btn btn-
success"> Next >>

```

```

● </div>
● @endif
● </form>
● </div>
●
● <script>
● const summary = document.getElementById('summary-btn');
●
● document.querySelector("details").addEventListener("toggle", function() {
● summary.textContent = this.open ? "" : "";
● });
●
● function Shownotifikasi() { // notifikasi
● swal({
● title:"Berhasil",
● text:"Semua Jawaban Anda Telah Terkirim.\n Tekan tombol berikut untuk
mengetahui hasilnya",
● icon:"success",
● button: "Lihat Hasil",
● timer:null
● }).then((value) => {
● // Setelah tombol ditekan, redirect ke link
● window.location.href = "{ {route('result')} }";
● });
●
●
● }
●
● function terkirim() {
● swal({
● title: "Terkirim",
● icon: "success",
● button: false,
● timer: 5000,
● });
● }
● </script>
● @endsection

```

- Berikut ini kode untuk halaman *update.blade.php*. Halaman ini hanya dapat diakses oleh admin

```

• @extends('layouts.app')
• @section('content')
• <div class="container">
• <form method="post" action="{{ route('tesiji.update', $row->id) }}"
• enctype="multipart/form-data">
• @csrf
• @method('PUT')
• <div class="form-group mt-5">
• <label for="soal">Soal</label>
• <input type="text" class="form-control mt-5" name="soal"
• value="{{ !! $row->soal !! }}>
• </div>
• <button type="submit" class="btn btn-primary mt-5 w-
• 100">perbaharui</button>
• </form>
• </div>
• @endsection

```

- Berikut ini kode untuk halaman *detresult.blade.php*.

```

• @extends('layouts.app')
• @section('content')
• <style>
• .card-body img:hover{
• transform: scale(1.1);
• transition: transform 0.3s ease;
• }
• #rekom:hover{
• border: 3px #1e90ff solid;
• transition: transform 0.3s ease;
• }
• h1{
• font-size:3rem;
• }
• #tombol a{

```

```
• color:#1E90ff;
• text-decoration: none;
• text-align: right;
• }
• #tombol a:hover{
• color: #1265b7;
• text-decoration: underline;
• }
• #tombol{
• max-width:100%;
• }
• #respon {
• overflow-x: auto;
• gap:1em;
• width: 100%;
• margin: 0 auto;
• }
• hr{
• color:#1e90ff;
• font-weight:bold
• }
• #result {
• border-left:5px #1E90ff solid;
• box-shadow:0px 0px 5px #1e90ff;
• height: 60%;
• }
• #unggul {
• transition-duration: 0.5s;
• border:5px solid #1e90ff;
• justify-content: center;
• box-sizing: border-box;
• }
• h6 {
• font-size:2vw;
• font-weight:bold
```

```
• }
• h2 {
• margin-top:50px;
• margin-bottom:50px;
• font-weight:bold;
• }
• hr{
• height:2px;
• background-color: #1e90ff;
• }
• @media only screen and (max-width:850px){
• #flex > div {
• width:80%;
• }
• #ilustrasi {
• width:70vw;
• }
• #rekom{
• transition-duration: 0.5s;
• min-width: 70vw;
• height:auto;
• justify-content: center;
• box-sizing: border-box;
• }
• #unggul {
• border:5px solid #1e90ff;
• transition-duration: 0.5s;
• width: 50vw;
• height:auto;
• justify-content: center;
• box-sizing: border-box;
• }
• #unggul img {
• width:20vw;
• }
• }
```

```
• #div {
• width: 100%;
• margin-left: auto;
• }
•
• }
• }
• @media only screen and (min-width:851px){
• #flex > div {
• width:40%;
• }
• #ilustrasi {
• width:40vw;
• }
• #unggul {
• transition-duration: 0.5s;
• border:5px solid #1e90ff;
• width: 30vw;
• max-height:32vw;
• justify-content: center;
• box-sizing: border-box;
• }
• #rekom {
• transition-duration: 0.5s;
• min-width: 20vw;
• height:auto;
• justify-content: center;
• box-sizing: border-box;
• }
• .card-body > img {
• max-width:20vw;
• }
• footer{
• background-image: url(http://localhost:8/img/footer.jpg);
• background-size:cover;
• background-attachment: fixed;
```

```

● }
● @media only screen and (max-width: 767px){
● #foot {
● width:100%;
● }
● }
● @media only screen and (min-width:768px){
● #foot {
● width:25%;
● }
● }
●
● }
● </style>
● <script src="https://unpkg.com/sweetalert/dist/sweetalert.min.js"></script>
● <link href="https://unpkg.com/aos@2.3.1/dist-aos.css" rel="stylesheet">
● <header>
● <div class="w-100 d-flex flex-wrap h-auto p-5 justify-content-evenly"
● style="background-image: linear-gradient(to left,rgb(0, 128, 255),
● aqua);height:auto;">
● <h1 data-aos="fade-up" data-aos-duration="1000" class="fw-bold text-light
● my-auto" style="max-width:80%;order:2;flex-shrink:1">Hai, { auth()->user()->name } !</h1>
●
● </div>
● </header>
● <main class="px-3">
● <div id="flex" class="d-flex flex-wrap justify-content-around mt-5">
● <div class="px-3"
● data-aos="fade-up-right"
● data-aos-anchor-placement="center-bottom"
● data-aos-duration="1500">
● <p style="text-align:justify">Selamat,
● Anda telah menjawab semua pertanyaan yang tersedia. Kini, saatnya Anda
● mengetahui hasilnya. Perhitungan ini didasarkan pada tingkat keminatan Anda pada

```

masing - masing bidang. Melalui kebiasaan dan kecenderungan hati terhadap suatu hal, psikologis dapat menentukan potensi yang Anda miliki. Anda bisa mencoba untuk mengembangkan bidang tersebut, sebab segala pekerjaan yang dilandasi dengan rasa senang akan dimaksimalkan sebaik mungkin.</p>

- <p style="text-align:justify">Tingkat kesuksesan Anda dinilai dari seberapa besar usahanya, dan kemudahan dalam berusaha dinilai dari seberapa fokus Anda menjalaninya. Mulailah dari sekarang, kembangkan potensi yang Anda miliki hingga berhasil meraih apa yang dicitakan. Semangat dan semoga sukses di masa depan...!</p>
- </div>
- <div id="result" class="p-4 ml-auto"
- data-aos="fade-up-left"
- data-aos-anchor-placement="center-bottom"
- data-aos-duration="1500">
- @foreach(\$hasil as \$tipe => \$skor)
- @foreach ( \$type as \$typ)
- @if(\$tipe==\$typ->tipe\_kepribadian)
- 
- @endif
- @endforeach
- <span class="my-auto fw-bold">{!!\$tipe}: {{ number\_format((\$skor / \$total) \* 100, 2) }}%</span>
- <hr>
- @endforeach
- </div>
- </div>
- 
- <h2 class="text-center" data-aos="flip-left">Kepribadian yang Lebih Unggul</h2>
- <div id="div" class="overflow-x-auto d-flex">
- @foreach(\$hasil as \$tipe => \$skor)
- @if(\$skor/\$total>1/6)
- @foreach ( \$type as \$typ)
- @if(\$tipe==\$typ->tipe\_kepribadian)
- <div id="unggul" data-aos="zoom-in" data-aos-duration="1000" class=" mx-2 p-4 rounded-3 shadow rounded">

- <div class="d-flex justify-content-center">
- 
- </div>
- <h3 class="text-center mt-3">{ \${tipe} }</h3>
- <h3 class="text-center">{ \${number\_format((\$skor / \$total) \* 100, 2)} %</h3>
- </div>
- @endif
- @endforeach
- @endif
- @endforeach
- </div>
- <br>
- <h4 data-aos="fade-up" data-aos-duration="1000" class="text-center">Berdasarkan hasil tes yang Anda lakukan, Anda cenderung memiliki tipe kepribadian tertinggi pada
- @foreach(\$hasil as \$tipe => \$skor)
- @if(\$skor==max(\$hasil))
- <strong>{ \${tipe} } dengan nilai { \${number\_format((\$skor / \$total) \* 100, 2)} %</strong>
- @endif
- @endforeach
- .</h4>
- <div style="background-image:linear-gradient(to right, #ffffff, rgba(70, 238, 238,0.5));" class="p-3 px-5 d-flex justify-content-end">
- <a onclick="notifikasi()" style="margin-left:auto;text-decoration:none;" href="{{ route('courses.index') }} "><h4>See More &rarr;</h4></a>
- </div>
- <h2 data-aos="flip-left" class="text-center">Rekomendasi Pertama</h2>
- <ul>
- <li>{ \${tipe1} }</li>
- <div id="respon" class="d-flex overflow-x-auto">
- @foreach (\$saran1 as \$ord=>\$id)
- @foreach (\$major as \$row)
- @if (\$row->id==\$id)

```

● <div data-aos="zoom-in" id="rekom" class="card-group mt-5 mb-5 rounded-
2 shadow-lg p-3" style="background-color:white;">
● <div class="card-body">
●
● <h3 class="fw-bold mt-3">{ ${row->title} }</h3>
●
●
● </div>
●
● <div id="tombol" class="card-footer btn-group rounded-5 py-1" >
● <div>
● id) }}" style="font-
size:0.8rem">Learn More →
● </div>
● </div>
●
● </div>
●
● @endif
●
● @endforeach
●
● @endforeach
●
● <div id="respon" style="display:grid;position:relative">
●
● <h2 data-aos="flip-left" class="text-center">Rekomendasi Kedua</h2>
●
●
● { ${tipe2} }
●
● <div id="respon" class="d-flex overflow-x-auto">
● @foreach ($saran2 as $ord=>$id)
●
● @foreach($major as $row)
●
● @if ($row->id==$id)
●
● <div id="rekom" data-aos="zoom-in" class="card-group mt-5 mb-5
rounded-2 shadow-lg p-3" style="background-color:white;">
● <div class="card-body">
●
● <h3 class="fw-bold mt-3">{ ${row->title} }</h3>
●
●
● </div>

```

```

• <div id="tombol" class="card-footer btn-group rounded-5 py-1" >
• <div>
• id) }}" style="font-size:0.8rem">Learn More →
• </div>
• </div>
• @endif
• @endforeach
• @endforeach
<div id="respon" style="display:grid;position:relative">

•
•
• <h2 data-aos="flip-left" class="text-center">Rekomendasi Ketiga</h2>

• {{ $tipe3 }}
• <div id="respon" class="d-flex overflow-x-auto">
• @foreach ($saran3 as $ord=>$id)
• @foreach($major as $row)
• @if ($row->id==$id)
• <div id="rekom" data-aos="zoom-in" class="card-group mt-5 mb-5 rounded-2 shadow-lg p-3" style="background-color:white;">
• <div class="card-body">
•
• <h3 class="fw-bold mt-3">{{ $row->title }}</h3>
• </div>
• <div id="tombol" class="card-footer btn-group rounded-5 py-1" >
• <div>
• id) }}" style="font-size:0.8rem">Learn More →
• </div>
• </div>
• </div>

```

```

• @endif
• @endforeach
• @endforeach
• <div id="respon" style="display:grid;position:relative">
•
• </main>
• <script src="https://unpkg.com/aos@2.3.1/dist-aos.js"></script>
• <script>
• AOS.init();
• </script>
• @endsection
•

```

- Selanjutnya, pengguna website perlu melihat hasil tes yang pernah dikerjakan sebelumnya. Sehingga, perlu dibuat file *result.blade.php* dengan menyalin kode dari halaman *detresult.blade.php*, dan juga file *history.blade.php*. Berikut ini adalah kode dari file *history.blade.php*.

```

• @extends('layouts.app')
• @section('content')
• <style>
• #card {
• background-color: rgba(0,0,0,0.08)
• }
• .hai {
• background-image: linear-gradient(to right, #1e90ff, aqua);
• color:white;
• padding:5%;
• }
• </style>
• <link rel="stylesheet" href="css/footer.css">
• <script src="https://unpkg.com/typed.js@2.1.0/dist/typed.umd.js"></script>
•
• <h3 class="hai"></h3>
• <div class="p-5">
• <h1 class="text-center fw-bold">Test History</h1>
• </div>

```

```

• <div id="card" class="card my-5 p-5 overflow-x-auto">
• <table class="col-12 mx-auto table table-bordered table-striped">
• <thead>
• <th>No</th>
• <th>Tanggal Ujian</th>
• <th>Tipe Paling Unggul</th>
• <th>Persentase</th>
• <th>Status</th>
• <th>Selengkapnya</th>
• </thead>
• @php
• $no=1;
• @endphp
• @foreach ($rows as $row)
•
• <tr>
• <td>{ ${$no++}}</td>
• <td>{ ${$row->created_at}}</td>
• <td>{ ${$row->tipe_id}}</td>
• <td>{ ${$row->hasil}} %</td>
• <td>{ ${$row->status}}</td>
• @if($row->status==='Selesai')
• <td>id) !!}">Learn More →</td>
• @else
• <td>Learn More →</td>
• @endif
• </tr>
• @endforeach
• </table>
•
• </div>
• <script>
• var typed = new Typed(".auto-type", {
• strings: ['Hello, inilah hasil Tes Psikologi yang Anda kerjakan sebelumnya'],

```

```

• typeSpeed: 50,
• backSpeed: 50,
• loop: true,
• });
• </script>
• @endsection

```

- Selain fitur Tes Psikologi, website ini juga menyediakan fitur lain semacam kumpulan informasi mengenai kampus dan jurusan yang ada di Indonesia. Buatlah folder *jurusan,campus*, dan *testimoni* berisi file *index.blade.php*, *show.blade.php*, *create.blade.php*, dan *update.blade.php*. Ini adalah contoh file yang ada di dalam folder *jurusan*. Untuk folder *campus* dan *testimoni*, cukup menyalin file dari folder *jurusan* dan mengubah beberapa kata saja, semisal *jurusan* diganti *campus*.

- Kode pada file *index.blade.php*

```

• @extends('layouts.app')
•
• @section('content')
• <style>
• @media (max-width: 768px) {
• #respon {
• grid-template-columns: repeat(auto-fit, minmax(200px, 1fr));
• gap:1em;
• width: 100%;
• max-width: 1200px;
• margin: 0 auto;
• }
• h3{
• font-size:5vw;
• }
• .div>img{
• height:25vw;
• }
• }
• @media (min-width: 767px) {
• #respon {

```

```
• display: grid;
• grid-template-columns: repeat(4, 1fr);
• width: 70%;
• height: auto;
• gap: 1em;
• margin: auto
•
• }
• h3{
• font-size: 1.5vw;
• font-weight: bold
• }
• .div>img{
• height: 15vw
• }
• }
• #div:hover{
• box-shadow: none;
• border: 3px #d5d6d7 solid;
• }
• #tombol{
• width: 100%;
• }
• #tombol > a{
• color: #1e90ff;
• text-decoration: none;
• }
• #tombol > a:hover{
• color: #1265b7;
• text-decoration: underline;
• }
• .pagination {
• font-size: 14px;
• padding: 8px 12px;
• margin-left: auto;
```

```
● color: white;
● }
● .radio-button {
● display: inline-block;
● border: 1px solid #1E90FF;
● color: #1E90FF;
● cursor: pointer;
● text-align: center;
● width: auto;
● }
● .radio {
● border: none;
● background-color: #ffffff;
● color: #1E90FF;
● }
● .radio > div {
● padding: 0.5vw;
● }
● .radio.active {
● background-color: #1E90FF;
● color: #ffffff;
● }
●
● .container {
● display: flex;
● justify-content: space-around;
● }
●
● details {
● width: auto;
● margin-left: auto;
● }
● #pilgan{
● max-width: 100%;
● }
```

```

● .radio-button button:checked + #huruf{
● border-radius:10px;
● }
● </style>
● <link rel="stylesheet" href="css/footer.css">
● <link href="https://unpkg.com/aos@2.3.1/dist-aos.css" rel="stylesheet">
● <div data-aos="fade-down" data-aos-duration="800">
● <h1 class="text-center fw-bold mt-5">Browse Majors</h1>
● <div class="card-header bg-transparent">
● <div class="d-flex justify-content-between">
● <div class="m-auto">
● <form method="GET" action="">
● <div class="input-group">
● <input type="text" name="nama" class="form-control"
● placeholder="Search..." >
● <div class="input-group-append">
● <button type="submit" class="btn btn-primary" ></button>
● @if (auth()->check())
● @if(auth()->user()->email == 'sidqianaazzahra1@gmail.com')
● Create
● @endif
● @endif
● </div>
● </div>
● </form>
● </div>
● </div>
● </div>
● <form data-aos="fade-down" method='GET' action="" style="margin:5%">
● <div style="border: none" class="d-flex justify-content-center flex-wrap my-4">
● <label class="radio-button">
● <button class="radio {{ request('tipe') == 7 ? 'active' : '' }}" name="tipe"
● value=7>
● <div>All</div>

```

```

● </button>
● </label>
● @foreach($types as $data)
● <label class="radio-button">
● <button class="radio" {{ request('tipe') == $data->id ? 'active' : '' }}>
● name="tipe" value={{ !$data->id !! }}>
● <div>{{ !$data->tipe_keprabadian !! }}</div>
● </button>
● </label>
● @endforeach
● </div>
● </form>
● <div id="respon" class="d-grid position-relative mt-5">
● @if ($rows)
● @foreach ($rows as $row)
● <div data-aos="zoom-out"
● data-aos-duration="500"
● id="div" class="card-group rounded-2 shadow-lg"
● style="padding:5%;background-color:white;height:auto">
● <div class="card-body" style="">
●
● <h3 class="fw-bold mt-3">{{ $row->title }}</h3>
● </div>
● <div id="tombol" class="card-footer btn-group rounded-5 py-1" >
●
● id) }}">Learn More →
● </div>
● @if (auth()->check())
● @if(auth()->user()->email == 'sidqianaazzahra1@gmail.com')
● <div class="d-flex mt-3">
● id) }}">
● <form method="post" action="{{ route('jurusan.destroy', $row->id) }}"
● id="hapusForm">
● @csrf
● @method('DELETE')
● </form>
● </div>
● @endif
● @endif
● </div>
● @endforeach
● </div>

```

- <button type="submit" style="border:none;background:none;"></button>
- </form>
- </div>
- @endif
- @endif
- </div>
- @endforeach
- @endif
- </div>
- <div class="mt-5 d-flex justify-content-center">
- { \${rows->links()} }
- </div>
- <script src="https://unpkg.com/aos@2.3.1/dist-aos.js"></script>
- <script>
- AOS.init();
- 
- </script>
- @endsection
- 

- Kode pada file *show.blade.php*

```
@extends('layouts.app')
@section('content')
<head>
<style>
footer{
background-image: url(http://localhost:8/img/footer.jpg);
background-size:cover;
background-attachment: fixed;
}
@media only screen and (max-width:767px){
img,p {
width:90%;
height:90%;
}
}
```

```

● #foot {
● width:100%;
● }
● }
● @media only screen and (min-width:768px){
● img,p{
● width:40%;
● height:40%;
● }
● #foot {
● width:25%;
● }
● }
● </style>
● <link href="https://unpkg.com/aos@2.3.1/dist-aos.css" rel="stylesheet">
● </head>
● <main>
● <div class="w-100 d-flex flex-wrap" style="background-image: linear-gradient(to right,rgb(0, 128, 255), aqua);height:auto;padding:10vw;">
● <h1 data-aos="fade-up" data-aos-duration="1000" class="fw-bold text-light" style="max-width:80%;font-size:10vw">{ !! $row->title !! }</h1>
● </div>
●

● <div class="pt-5 d-flex justify-content-around flex-wrap">
●
● <p data-aos="fade-up-left" data-aos-duration="1000" style="text-align: justify">{ !!$row->content!! }</p>
● </div>
● kembali
● </main>
● </div>
● </div>
● </footer>
● <script src="https://unpkg.com/aos@2.3.1/dist-aos.js"></script>
● <script>

```

- AOS.init();
  - </script>
  - @endsection
  - Kode pada file *create.blade.php*. Pada file *create* dan *update*, sesuaikan dengan kolom pada masing – masing tabel.
- ```

• @extends('layouts.app')
• @section('content')
<style>
• @media only screen and (min-width: 768px){
•     #grid{
•
•     }
•         grid-template-columns: repeat(2,1fr)
•     }
• @media only screen and (max-width: 767px){
•     #grid{
•
•         grid-template-columns: repeat(1,1fr)
•     }
•     }
• .custom-btn {
•     width: 20%;
•     height: 40px;
•     color: #fff;
•     border-radius: 5px;
•     padding: 10px 25px;
•     background: transparent;
•     cursor: pointer;
•     transition: all 0.3s ease;
•     position: relative;
•     display: inline-block;
•     box-shadow: inset 2px 2px 2px 0px rgba(255,255,255,.5),
•                 7px 7px 20px 0px rgba(0,0,0,.1),
•                 4px 4px 5px 0px rgba(0,0,0,.1);
•     outline: none;

```



```

●          <label for="files">Gambar</label>
●          <input type="file" class="form-control" name='files' required>
●          </div>
●          <div class="form-group mt-4" >
●              <label for="content">isi</label>
●              <textarea type="text" class="form-control" style="height:30px"
● name='content' required></textarea>
●          </div>
●          <div class="form-group mt-4" >
●              <label for="tipeSatu">TipeSatu</label>
●              <textarea type="text" class="form-control" style="height:30px"
● name='tipeSatu' required></textarea>
●          </div>
●          <div class="form-group mt-4" >
●              <label for="tipeDua">TipeDua</label>
●              <textarea type="text" class="form-control" style="height:30px"
● name='tipeDua' required></textarea>
●          </div>
●          <div class="form-group mt-4" >
●              <label for="tipeTiga">TipeTiga</label>
●              <textarea type="text" class="form-control" style="height:30px"
● name='tipeTiga' required></textarea>
●          <button type="submit" class="custom-btn btn-4 mt-4 w-
100">simpan</button>
●          </div>
●          </form>
●          </div>
●          </div>
●          @endsection

```

- Kode pada file *update.blade.php*

```

●      @extends('layouts.app')
●      @section('content')
●      <style>
●          .custom-btn {
●              width: 20%;

```

```
•     height: 40px;
•     color: #fff;
•     border-radius: 5px;
•     padding: 10px 25px;
•     background: transparent;
•     cursor: pointer;
•     transition: all 0.3s ease;
•     position: relative;
•     display: inline-block;
•     box-shadow: inset 2px 2px 2px 0px rgba(255,255,255,.5),
•                 7px 7px 20px 0px rgba(0,0,0,.1),
•                 4px 4px 5px 0px rgba(0,0,0,.1);
•     outline: none;
• }
•
• .btn-4 {
•     background: #1E90ff;
•     border: none;
• }
•
• .btn-4:before {
•     height: 0%;
•     width: 2px;
• }
•
• .btn-4:hover {
•     box-shadow: 4px 4px 6px 0 rgba(255,255,255,.5),
•                 -4px -4px 6px 0 rgba(116, 125, 136, .5),
•                 inset -4px -4px 6px 0 rgba(255,255,255,.2),
•                 inset 4px 4px 6px 0 rgba(0, 0, 0, .4);
• }
• </style>
• <div class="container mt-5">
•     <h1 class="text-center fw-bold">Major Update</h1>
```

```

●      <form method="post" action="{{ route('jurusan.update', $row->id) }}">
●          @csrf
●          @method('PUT')
●      <div class="form-group mt-4">
●          <label for="title">Jurusan</label>
●          <input type="text" class="form-control" name="title"
●              value="{{ !! $row->title!! }}>
●      </div>
●      <div class="form-group mt-4">
●          <label for="files">Gambar</label>
●          <input type="file" class="form-control" name="files"
●              value="{{ !! $row->files!! }}>
●      </div>
●      <div class="form-group mt-4">
●          <label for="content">Isi</label>
●          <textarea type="text" class="form-control" name="content"
●              value="{{ !! $row->content!! }}></textarea>
●      </div>
●      <div class="form-group mt-4" >
●          <label for="tipeSatu">TipeSatu</label>
●          <textarea type="text" class="form-control" name='tipeSatu'
●              value="{{ !!$row->TipeSatu!! }}></textarea>
●      </div>
●      <div class="form-group mt-4" >
●          <label for="tipeDua">TipeDua</label>
●          <textarea type="text" class="form-control" name='tipeDua'
●              value="{{ !!$row->TipeDua!! }}></textarea>
●      </div>
●      <div class="form-group mt-4" >
●          <label for="tipeTiga">TipeTiga</label>
●          <textarea type="text" class="form-control" name='tipeTiga'
●              value="{{ !!$row->TipeTiga!! }}></textarea>
●      </div>
●

```

```

•      <div class="form-group">
•          <label for="activated">Publikasi/label>
•          <select name="activated" class="form-select" required>
•              <option value="">silahkan pilih</option>
•              <option value="1" @if($row->activated==1)selected
•                  @endif>ya</option>
•              <option value="0" @if($row->activated==0) selected
•                  @endif>tidak</option>
•          </select>
•      </div>
•      <button type="submit" class="custom-btn btn-4 my-5 w-
100">Update</button>
•  </form>
• </div>
• @endsection

```

- Namun, terdapat kode tambahan pada halaman *show.blade.php* di folder *campus*, karena halaman ini perlu memunculkan lebih banyak informasi mengenai kampus di Indonesia, termasuk memunculkan diagram mengenai daya tampung masing – masing kampus. Berikut kodennya.

```

•  @extends('layouts.app')
•  @section('content')
•  <style>
•  .gedung {
•      width: 100vw;
•      height: auto;
•  }
•  hr{
•      border: 0;
•      height: 3px;
•      background: linear-gradient(to right, #1e90ff, aqua);
•      box-shadow: 0px 0px 3px rgba(0,0,0,0.5);
•  }
•  video {
•      width: 100%;
•      height: 100%;

```

```
●      object-fit: cover;
●      z-index: 1;
●  }
●  @media only screen and (max-width:767px){
●      iframe,p {
●          width:90%;
●          height:90%;
●      }
●      #dayaTampungChart {
●          position:relative;
●          width:50%;
●      }
●      #foot {
●          width:100%;
●      }
●      #gedung {
●          height:40vh;
●      }
●
●  }
●  @media only screen and (min-width:768px){
●      iframe,p{
●          width:40%;
●          height:40%;
●      }
●      #dayaTampungChart {
●          position:relative;
●          width:25%;
●      }
●      #foot {
●          width:25%;
●      }
●      #gedung {
●          height:80vh;
●      }
● }
```

```

●      }
●      footer{
●          background-image: url(http://localhost:8/img/footer.jpg);
●          background-size:cover;
●          background-attachment: fixed;
●      }
●      #gedung::-webkit-scrollbar {
●          width: 0px;
●          height: 0px;
●      }
●      </style>
●      <link rel="stylesheet" href="css/footer.css">
●      <link href="https://unpkg.com/aos@2.3.1/dist/aos.css" rel="stylesheet">
●      <script src="https://cdn.jsdelivr.net/npm/chart.js"></script>
●      <div id="gedung" class="overflow-y-auto overflow-x-hidden shadow">
●          
●      </div>
●      <div data-aos="fade-right" data-aos-anchor-placement="bottom-bottom" data-aos-duration="1000" class="w-100 d-flex flex-wrap gap-3 p-3 h-auto">
●          
●          <div class="my-auto">
●              <h1 class="fw-bold">{$row->title!!}</h1>
●              <h4 class="fw-bold">{$row->status!!}</h4>
●          </div>
●      </div>
●
●      <div data-aos="fade-up" data-aos-duration="1000" class="pt-5 d-flex justify-content-around flex-wrap">
●          <p style="text-align: justify">{$row->content!!}</p>
●          {$row->lokasi!!}
●      </div>
●      <iframe style="height:40vw; width:90%; margin-left:5%; margin-top:10%" src="https://www.youtube.com/embed/{$row->profil!!}" frameborder="0" allow="accelerometer; autoplay; clipboard-write; encrypted-media; gyroscope; picture-in-picture" allowfullscreen></iframe>
●

```

- <div data-aos="fade-down" data-aos-duration="1000" style="margin-left:10%>
- <div class="d-flex flex-wrap mt-5 gap-5 gap-5">
- <div>
-
- </div>
-
- <h4 class="fw-bold"></h4>
- <h4 class="my-auto">Terakreditasi { !!\$row->akreditasi!! }</h4>
- </div>
- <hr class="w-75">
- <div class="d-flex flex-wrap mt-5 gap-5">
- <div>
- </div>
- <h4 class="fw-bold"></h4>
- <h4 class="my-auto">Rp. { !!\$row->biaya!! }</h4>
- </div>
- <hr class="w-75">
- <div class="d-flex flex-wrap mt-5 gap-5">
- <div>
- </div>
- <h4 class="fw-bold"></h4>
- <h4 class="my-auto">{ !!\$row->alamat!! }</h4>
- </div>
- <hr class="w-75">
- <div class="d-flex flex-wrap mt-5 gap-5">
- <div>
-
-
- </div>
-
- <h4 class="fw-bold"></h4>

```

●      <h4 class="my-auto"><a href="https://{$row->website}">https://{$row->website}</a></h4>
●      </div>
●      <hr class="w-75">
●      </div>
●      </div>
●      </div>
●      <div class="d-flex flex-wrap mt-5 gap-5 justify-content-center">
●      <div>
●       </div>
●
●      <h4 class="my-auto">Daya Tampung</h4>
●      <h4 class="fw-bold">:</h4>
●      <h4 class="fw-bold">Sekitar {$row->dayaTampung} %</h4>
●      </div>
●      <canvas id="dayaTampungChart" class="m-auto">
●      </canvas>
●      <a class="btn btn-primary m-5" href="/campus">kembali</a>
●      <script src="https://unpkg.com/aos@2.3.1/dist-aos.js"></script>
●      <script>
●      AOS.init();
●      var ctx = document.getElementById('dayaTampungChart').getContext('2d');
●      var dayaTampung = {$row->dayaTampung}; // Mengambil data
persentase dari Blade
●
●      var myPieChart = new Chart(ctx, {
●          type: 'pie', // Menggunakan tipe 'pie' untuk diagram lingkaran
●          data: {
●              labels: ['Daya Tampung', 'Sisa'],
●              datasets: [{{
●                  data: [dayaTampung, 100 - dayaTampung], // Data persentase
●                  backgroundColor: ['#1e90ff', 'blue'],
●                  borderColor: ['#1e90ff', '#1e90ff'],
●                  borderWidth: 1
●              }}]

```

```

•      },
•      options: {
•          responsive: true,
•          plugins: {
•              legend: {
•                  position: 'top',
•              },
•              tooltip: {
•                  callbacks: {
•                      label: function(context) {
•                          var label = context.label || "";
•                          var value = context.raw || 0;
•                          return label + ':' + value + '%';
•                      }
•                  }
•              }
•          }
•      });
•
•      </script>
•  @endsection

```

- Sedangkan dalam halaman *create.blade.php* folder *testimoni*, perlu memunculkan alamat admin menggunakan elemen *<iframe></iframe>* sebagai berikut.

```

•  <div class="flex mt-5">
•      <div class="box"><iframe
src="https://www.google.com/maps/embed?pb=!1m18!1m12!1m3!1d3952.99622116
6599!2d113.37477170000001!3d-
7.7902235!2m3!1f0!2f0!3f0!3m2!1i1024!2i768!4f13.1!3m3!1m2!1s0x2dd7aaaee11e
d2d9%3A0xe84d43c739cb23e2!2sPesantren%20Zainul%20Hasan%20Genggong!5e
0!3m2!1sid!2sid!4v1726978105203!5m2!1sid!2sid"
•          height="600" style="border:0;" allowfullscreen="" loading="lazy"
referrerpolicy="no-referrer-when-downgrade">
•      </iframe>

```

- Lokasi

695H+3J6, PIP Lt. 1 Pajarakan, Jl. Raya Condong, Gerojokan, Karangbong, Kec. Pajarakan, Kabupaten Probolinggo, Jawa Timur 67281

Email

azzahrasidqiana22@gmail.com

Country

Indonesia
- </div>
- </div>
- </div>

- Berikut CSS tambahan pada file *create.blade.php* di folder *testimoni*.

```
<style>
@media screen only and (max-width:767px){
    .box {
        flex-basis:90%;
    }
}
.box {
    width:500px;
    margin-bottom:5%;
}
```

```
●      }
●      .box div, iframe{
●          width:100%
●      }
●      @media screen only and (min-width:768px){
●
●
●      .box {
●          flex-basis:90%;
●          margin-bottom:5px;
●      }
●      }
●      .flex {
●          display: flex;
●          margin:auto;
●          justify-content:space-around;
●          flex-wrap: wrap;
●      }
●      .custom-btn {
●          width: 20%;
●          height: 40px;
●          color: #fff;
●          border-radius: 5px;
●          padding: 10px 25px;
●          background: transparent;
●          cursor: pointer;
●          transition: all 0.3s ease;
●          position: relative;
●          display: inline-block;
●          box-shadow: inset 2px 2px 2px 0px rgba(255,255,255,.5),
●          7px 7px 20px 0px rgba(0,0,0,.1),
●          4px 4px 5px 0px rgba(0,0,0,.1);
●          outline: none;
●      }
●
●      .btn-4 {
```

```

●      background: #1E90ff;
●      border: none;
●      }
●
●      .btn-4:before {
●          height: 0%;
●          width: 2px;
●      }
●
●      .btn-4:hover {
●          box-shadow: 4px 4px 6px 0 rgba(255,255,255,.5),
●                      -4px -4px 6px 0 rgba(116, 125, 136, .5),
●                      inset -4px -4px 6px 0 rgba(255,255,255,.2),
●                      inset 4px 4px 6px 0 rgba(0, 0, 0, .4);
●      }
●      </style>
```

- Terakhir, buat folder *courses* berisi pengetahuan singkat mengenai metode RIASEC. Buat file *index.blade.php* dan *show.blade.php*. Begini kode pada file *index.blade.php*.

```

●      @extends('layouts.app')
●      @section('content')
●      <head>
●      <style>
●      *{
●          font-family: 'Instrument',sans-serif;
●      }
●      header{
●          height: 25vw;
●          background-image: linear-gradient(to right,rgb(0, 128, 255), aqua);
●          padding-block-start: 10%;
●      }
●      h1 {
●          font-size: 8vw;
●          text-shadow: 10px 10px 20px black;
●      }
```

```
• p{  
•     text-align: justify;  
• }  
• h2{  
•     font-weight: bold;  
• }  
• /* #sejarah{  
•     min-width:200px;  
• } */  
• @media only screen and (min-width:768px){  
•     #sejarah{  
•         width:55%;  
•     }  
•     #holland{  
•         width:40%;  
•         height:60%;  
•         margin-left: 5%;  
•     }  
• }  
• @media only screen and (max-width:767px){  
•     #sejarah{  
•         width:100%;  
•         order: 2;  
•     }  
•     #holland{  
•         width:100%;  
•         height:auto;  
•         order: 1;  
•     }  
• }  
• .item{  
•     width:30%;  
•     margin-bottom:30px;  
• }
```

```

● #icon {
●     border-radius:10px;
●     width: 40%;
●     margin-left:30%;
● }
● #icon:hover{
●     box-shadow:0px 0px 15px 3px rgba(0, 128, 255,0.8);
● }
● </style>
● <script src="https://unpkg.com/typed.js@2.1.0/dist/typed.umd.js"></script>
● <link rel="stylesheet" href="css/footer.css">
●
● <link href="https://unpkg.com/aos@2.3.1/dist-aos.css" rel="stylesheet">
● </head>
● <main>
●     @if(session('alert'))
●     <div id="alert" style="position: absolute; top: 20px; right: 20px; background-color: #f44336; color: white; padding: 15px; border-radius: 5px;">
●         {{ session('alert') }}
●     </div>
●     @endif
●     <header>
●         <h1 class="text-center fw-bold text-light">Apa itu <span class="auto-type"></span></h1>
●     </header>
●     <article class="p-5">
●         <div data-aos="fade-up" data-aos-anchor-placement="bottom-bottom" data-aos-duration="1000">
●             <h2>Pengertian RIASEC</h2>
●             <p>RIASEC merupakan tes psikologi minat bakat yang dikemukaan oleh Dr. John Holland, seorang psikolog Amerika dan Profesor Emeritus Sosiologi di Universitas Johns Hopkins yang menciptakan model pengembangan karier, Holland Occupational Themes, yang dikenal sebagai Holland Codes. Melalui metode ini, seseorang akan dikelompokkan menuju 6 tipe kepribadian untuk memahami arah kecenderungan potensi yang dimiliki. Dengan demikian, nilai produktivitas akan lebih efektif jika orang tersebut lebih menitikberatkan pengembangan dan

```

- pendalaman terhadap minat dan bakat yang sesuai. RIASEC yang merupakan singkatan dari Realistic, Investigative, Artistic, Social, Enterprising, dan Conventional.</p>
- </div>
 - <h2>Penemu RIASEC</h2>
 - <div class="container d-flex flex-wrap">

Dr. John Holland merupakan seorang psikolog terkenal kelahiran Obama pada tahun 1919. Penemuan teorinya mengenai Tes Psikologi RIASEC amat berpengaruh di berbagai belahan dunia, khususnya pada abad ke 20. Sejak kecil, ia dibesarkan di dalam hati- tanah Amerika, bersama dengan dua saudara laki-laki dan saudara perempuan, dipandu oleh sekolah ibu guru dan ayah eksekutif periklanan, yang keduanya “memiliki ditandai minat intelektual “American Psychological Association, 1995. Dia tinggal di rumah untuk kuliah, menghadiri Universitas Kota Omaha / University of Omaha, jurusan psikologi, matematika, dan Perancis dan lulus pada tahun 1942. Seperti banyak pria muda di tahun 1940-an, dia masuk Angkatan Darat setelah kuliah. Mungkin karena ia mengambil jurusan psikologi, Swasta Karier militer Belanda termasuk posisi sebagai pewawancara klasifikasi, pengujian tes, asisten psikologis, dan Wechsler Intelligence Test administrator. Pada tahun 1966, Holland berpendapat bahwa masing - masing orang yang berada di lingkungan realistik, intelektual, artistik, sosial, atau konvensional mayoritas memiliki tipe kepribadian yang sama. Ia juga mengakui bahwa orang yang memiliki minat berbeda dan bekerja pada lingkungan yang berbeda adalah orang yang memiliki sejarah hidup yang berbeda pula. Sehingga, pilihan karir ialah suatu ekspresi atau suatu perluasan kepribadian dalam dunia kerja yang diikuti oleh identifikasi berikutnya dengan stereotipe pekerjaan yang spesifik. Perbandingan antara diri (self) dengan persepsi terhadap suatu pekerjaan dan penerimaan atau penolakan adalah penentu utama dalam pilihan karir. Keseuaian antara tinjauan diri (self) seseorang dengan penetapan pemilihan pekerjaan ialah berhubungan dengan model gaya pribadi. Hal ini memicu Holland untuk mengembangkan pikirannya dalam mengklasifikasi tipe kepribadian seseorang menjadi enam bagian sebagai berikut. </p>
 -
 - </div>
 -


```

●      <h2 class="text-center">Jenis Tipe Kepribadian</h2>
●      <h3 class="text-center">Berdasarkan Metode RIASEC</h3>
●      <br>
●      <div class="container d-flex justify-content-evenly flex-wrap">
●          @if($rows)
●              @foreach($rows as $row)
●                  <div data-aos="zoom-in" data-aos-anchor-placement="bottom-bottom"
●                      data-aos-duration="1000" class="item">
●                      <a href="{{ route('courses.show', $row->id) }}"></a>
●                      <br>
●                      <h5 class="text-center mt-2 fs-6">{!!$row->tipe_kepribadian!!}</h5>
●                  </div>
●                  <br>
●
●                  @endforeach
●                  @endif
●              </div>
●
●          </article>
●      </main>
●      <script src="https://unpkg.com/aos@2.3.1/dist/aos.js"></script>
●      <script>
●          AOS.init();
●          var typed = new Typed(".auto-type", {
●              strings:
●                  ["RIASEC?", "Realistic?", "Investigative?", "Artistic?", "Social?", "Enterprising?",
●                  "Conventional?"],
●                  typeSpeed: 100,
●                  backSpeed: 100,
●                  loop: true,
●              });
●          // Menghilangkan notifikasi setelah beberapa detik (misalnya 3 detik)
●          setTimeout(function() {
●              var alertElement = document.getElementById('alert');
●              if (alertElement) {

```

- alertElement.style.transition = "opacity 0.5s ease";
- alertElement.style.opacity = '0';
- setTimeout(function() {
 alertElement.remove();
 }, 500); // Menghapus elemen setelah efek transition selesai
- }
- }, 3000);
- </script>
- @endsection
- Kode file *show.blade.php*.

```

• @extends('layouts.app')
• @section('content')
• <head>
• <style>
•   .title{
•     background-color: rgba(17, 140, 207, 0.4);
•     position: absolute;
•     top: 0;
•     left: 0;
•     width: 100%;
•     padding: 15vh;
•     height:60vh;
•     z-index: 2;
•   }
• 
• 
•   h1{
•     font-size: 10vw;
•   }
• 
• 
•   header {
•     position: relative;
•     width: 100%;
•     height: 45vh;
•     overflow: hidden;
•   }

```

```
●
●   video {
●     position: absolute;
●     top: 0;
●     left: 0;
●     width: 100%;
●     height: 100%;
●     object-fit: cover;
●     z-index: 1;
●   }
●   p{
●     text-align: justify;
●   }
●   h2{
●     font-weight: bold;
●   }
●   #jurusan {
●     box-shadow: inset 0px 0px 40px 2px #7da6ce;
●   }
●   #jurusan:hover{
●     box-shadow: inset 0px 0px 40px 2px #1e90ff;
●   }
●
●   /*
●   #sejarah{
●     min-width:200px;
●   }
●   */
●   @media only screen and (min-width:768px){
●     #sejarah{
●       width:55%;
●     }
●     #holland{
●       width:40%;
●       height:60%;
●       margin-left: 5%;
●     }
●   }
```

```
●      }
●      #jurusan {
●          width:40%;
●      }
●      #foot {
●          width:25%;
●      }
●      }
●      @media only screen and (max-width:767px){
●          #sejarah{
●              width:100%;
●              order: 2;
●          }
●          #holland{
●              width:100%;
●              height:auto;
●              order: 1;
●          }
●          #jurusan {
●              width:80%;
●          }
●          #foot {
●              width:100%;
●          }
●      }
●      footer{
●          background-image: url(http://localhost:8/img/footer.jpg);
●          background-size:cover;
●          background-attachment: fixed;
●      }
●      </style>
●      <link href="https://unpkg.com/aos@2.3.1/dist/aos.css" rel="stylesheet">
●      </head>
●      <header>
●          <div class="title">
```

```

●      <h1 data-aos="fade-up" data-aos-duration="1000" class="text-center text-light fw-bold">{ ${row->tipe_kepribadian} }</h1>
●      </div>
●      <video autoplay loop muted plays-inline style="height:45vh;width:100%">
●          <source src="/img/RIASEC/{ ${row->video} }.mp4">
●      </video>
●      </header>
●      <article class="p-5">
●          <div class="container d-flex flex-wrap">
●              <p data-aos="fade-up-right" data-aos-duration="1000"
id="sejarah">{ ${row->content} }</p>
●              
●          </div>
●          <br>
●          <h2 class="text-center" data-aos="flip-left" data-aos-anchor-placement="top-
center">Jurusan Yang Sesuai</h2>
●          <div data-aos="zoom-in-up" data-aos-anchor-placement="top-center" data-
aos-duration="1000" class="mt-4 d-flex flex-wrap justify-content-around">
●              @for ($i=1;$i<=6;$i++)
●                  <div class="p-3 text-center mb-3 rounded-3" id="jurusan">Jurusan { ${i} } :
{ ${row->$i} }</div>
●                  <br>
●              @endfor
●          </div>
●          </article>
●          <script src="https://unpkg.com/aos@2.3.1/dist/aos.js"></script>
●          <script>
●              AOS.init();
●          </script>
●      @endsection

```

3.1.1 Pembuatan Controller

- Pemakaian tombol dalam website pasti mengarahkan sebuah fungsi, di mana fungsi tersebut dideklarasikan dalam halaman *Controller*. Buat file

baru pada folder *Http\Controllers* sesuai nama halaman yang dijalankan. Misalnya, *CampusController.php*.

- Di dalam file Controller, ketik kode seperti ini.

```
• <?php
• namespace App\Http\Controllers;
• use Illuminate\Http\Request;
• class CampusController extends Controller
• {
• }
```

- Kemudian, deklarasi setiap fungsi yang ingin dibuat dengan kode sebagai berikut.

```
• public function NamaFungsi($parameter)
• {
• \\\ di sini logika fungsinya
• Contoh : $var1= '1234567890';
• \\\ pengambilan data melalui App\Models
• $data=\App\Models>NamaTabel::all();
•         return view('halaman_tujuan', [
•             'var1'=>$var1,
•             'data'=>$data
•         ]);
• }
```

- Jika data yang diambil berupa input, sertakan *Request \$request* pada parameter fungsinya. Misal pada halaman update dan sebagainya.

```
• public function update(request $request,$id )
• {
•
•
•     $data = [
•         'title' =>$request['title'],
•         'content' =>$request['content'],
•         'files' =>$namaFile,
•         'activated' =>$request['activated'],
•     ];
•
•
•     \App\Models\Jurusan::where('id', $id)->update($data);
```

- - return redirect()->to('jurusan');
 - }
 - Untuk pengambilan data berupa file yang diinput dari *device*, tambahkan kode berikut.
- ```

• if($request->hasFile('files')) {
• $file = $request->file('files');
• $ekstensi = $file->getClientOriginalExtension();
• $tanggal = str_replace(['-','.',':',''],",\Carbon\Carbon::now());
• $namaFile = 'gambar'.'$tanggal'.'$ekstensi';
• $file->move(public_path('img'), $namaFile);
• $data['files'] = $namaFile;
• }

```
- Berikut ini contoh kode selengkapnya untuk *CampusController.php*
- ```

•      <?php
•      namespace App\Http\Controllers;
•      use Illuminate\Http\Request;
•      class CampusController extends Controller
•      {
•
•      public function index(Request $request)
•      {
•          $model = \App\Models\Campus::orderBy('title','asc');
•          if (isset($request['nama'])) {
•              $model->where('title', 'Like', '%'.$request['nama'].'%');
•          }
•          return view('campus.index', [
•              'rows' => $model->paginate(40)
•          ]);
•      }
•      public function create()
•      {
•          return view('campus.create');
•      }
•      public function show($id)

```

```

•    {
•        return view('campus.show', [
•            'row' => \App\Models\Campus::where('id', $id)->first()
•        ]);
•    }
•    public function edit($id)
•    {
•        return view('campus.update', [
•            'row' => \App\Models\Campus::where('id', $id)->first()
•        ]);
•    }
•    public function store(request $request )
•    {
•        $file = $request->file('files');
•        $ekstensi = $file->getClientOriginalExtension();
•        $tanggal = str_replace(['-', ':', ','], '', Carbon\Carbon::now());
•        $namaFile = 'gedung-' . $tanggal . '.' . $ekstensi;
•        $file->move(public_path('img'), $namaFile);
•
•        $logo = $request->file('logo');
•        $ekstensi = $logo->getClientOriginalExtension();
•        $namalogo = 'logo-' . $tanggal . '.' . $ekstensi;
•        $logo->move(public_path('img'), $namalogo);
•
•        \App\Models\Campus::create([
•            'title' => $request['title'],
•            'content' => $request['content'],
•            'files' => $namaFile,
•            'logo' => $namalogo,
•            'status' => $request['status'],
•            'biaya' => $request['biaya'],
•            'lokasi' => $request['lokasi'],
•            'profil' => $request['profil'],
•            'alamat' => $request['alamat'],
•            'akreditasi' => $request['akreditasi'],

```

```

•     'website'=>$request['website'],
•     'dayaTampung'=>$request['dayaTampung']
• );
• return redirect()->to('/campus')->with('success','berhasil');
}
public function update(request $request,$id )
{
•
•
•     if($request->hasFile('files')) {
•         $file = $request->file('files');
•         $ekstensi = $file->getClientOriginalExtension();
•         $tanggal = str_replace(['-','.',',',' '],",\Carbon\Carbon::now());
•         $namaFile = 'gedung-'.$tanggal.'.'.$ekstensi;
•         $file->move(public_path('img'), $namaFile);
•         $data['files'] = $namaFile;
•
•         $logo = $request->file('logo');
•         $ekstensi = $logo->getClientOriginalExtension();
•         $namalogo = 'logo-'.$tanggal.'.'.$ekstensi;
•         $logo->move(public_path('img'), $namalogo);
•         $data['logo'] = $namalogo;
•     }
•     $data = [
•         'title' =>$request['title'],
•         'content' =>$request['content'],
•         'files' =>$namaFile,
•         'logo'=>$namalogo,
•         'status' =>$request['status'],
•         'biaya'=>$request['biaya'],
•         'lokasi'=>$request['lokasi'],
•         'profil'=>$request['profil'],
•         'alamat'=>$request['alamat'],
•         'akreditasi'=>$request['akreditasi'],
•         'website'=>$request['website'],
•         'dayaTampung'=>$request['dayaTampung']
}

```

```
•     ];
•
•
•         \App\Models\Campus::where('id', $id)->update($data);
•
•
•             return redirect()->to('campus');
•
•         }
•
•     public function destroy($id)
•
•     {
•
•         $model = \App\Models\Campus::where('id', $id);
•
•         $row = $model->first();
•
•         \Illuminate\Support\Facades\File::delete(public_path('img/' . $row->files));
•
•         $model->delete();
•
•         return redirect()->to('campus');
•
•     }
•
• }
```

- Halaman Controller lain masih menggunakan cara yang sama seberti kode di atas, hanya berbeda pada penyesuaian nama tabel dan algoritma pemrogramannya.

3.2.10 Rancangan Kode Perhitungan Skor

- Tujuan utama website ini ialah mengetahui minat seseorang setelah mengerjakan tes psikologi yang disediakan. Sehingga, perlu ada rancangan perhitungan skor pada masing - masing soal yang dijawab. Dalam website ini, setiap jawaban memiliki nilai berbeda sesuai tabel jawaban.

| | <input type="checkbox"/> | <input type="checkbox"/> Ubah | <input type="checkbox"/> Salin | <input type="checkbox"/> Hapus | id | created_at | update_at | soal_id | huruf | jawaban | nilai | tipe_id |
|--|--------------------------|-------------------------------|--------------------------------|--------------------------------|-----------|---------------------|---------------------|----------------|--------------|----------------|--------------|----------------|
| | | | | | 1 | 2024-05-22 06:27:20 | 2024-06-10 11:05:12 | 1 | A | Sangat setuju | 3 | 1 |
| | | | | | 2 | 2024-05-22 06:27:20 | 2024-06-10 11:05:12 | 1 | B | Setuju | 2 | 1 |
| | | | | | 3 | 2024-05-20 06:04:07 | 2024-06-10 11:05:12 | 1 | C | Lumayan | 1 | 1 |
| | | | | | 4 | 2024-05-20 06:04:07 | 2024-06-10 11:05:12 | 1 | D | Tidak setuju | 0 | 1 |
| | | | | | 5 | 2024-06-10 11:08:33 | 2024-06-10 11:08:33 | 2 | A | Sangat setuju | 3 | 2 |
| | | | | | 6 | 2024-06-10 11:08:33 | 2024-06-10 11:08:33 | 2 | B | Setuju | 2 | 2 |
| | | | | | 7 | 2024-06-10 11:08:33 | 2024-06-10 11:08:33 | 2 | C | Lumayan | 1 | 2 |
| | | | | | 8 | 2024-06-10 11:08:33 | 2024-06-10 11:08:33 | 2 | D | Tidak setuju | 0 | 2 |
| | | | | | 9 | 2024-06-10 11:08:33 | 2024-06-10 11:08:33 | 3 | A | Sangat setuju | 3 | 3 |
| | | | | | 10 | 2024-06-10 11:08:33 | 2024-06-10 11:08:33 | 3 | B | Setuju | 2 | 3 |
| | | | | | 11 | 2024-06-10 11:08:33 | 2024-06-10 11:08:33 | 3 | C | Lumayan | 1 | 3 |
| | | | | | 12 | 2024-06-10 11:08:33 | 2024-06-10 11:08:33 | 3 | D | Tidak setuju | 0 | 3 |
| | | | | | 13 | 2024-06-10 11:08:33 | 2024-06-10 11:08:33 | 4 | A | Sangat setuju | 3 | 4 |
| | | | | | 14 | 2024-06-10 11:08:33 | 2024-06-10 11:08:33 | 4 | B | Setuju | 2 | 4 |
| | | | | | 15 | 2024-06-10 11:08:33 | 2024-06-10 11:08:33 | 4 | C | Lumayan | 1 | 4 |
| | | | | | 16 | 2024-06-10 11:08:33 | 2024-06-10 11:08:33 | 4 | D | Tidak setuju | 0 | 4 |

Gambar 3.2.11.1. Tabel Jawaban

- Algoritma perhitungan skor dilakukan dengan menambah setiap nilai pada masing – masing tipe kepribadian. Sedangkan hasil akhir berupa presentase atau perbandingan total skor pada setiap tipe. Rumusnya total nilai setiap tipe dibagi total nilai semua tipe dikali 100 %. Berikut kode selengkapnya.

```

•    public function result(Request $request)
•    {
•        \App\Models\HasilTes::where('user_id',auth()->user()->id)
•        ->update([
•            'status'=>'Selesai'
•        ]);
•        $tes_user =
•            \App\Models\HasilTes::orderBy('created_at','desc')-
•            >where('user_id',auth()->user()->id)->first();
•
•        if (!$tes_user) {
•            return redirect()->back()->with('error', 'Tidak
•            ada jawaban yang disimpan.');
•        }
•        $huruf = \App\Models\Jawaban::pluck('huruf')-
•        >toArray();

```

```

•      $nilai = \App\Models\Jawaban::pluck('nilai')-
•          >toArray();
•      $tipe_id = \App\Models\Jawaban::pluck('tipe_id')-
•          >toArray();
•      $tipeSatu = \App\Models\Jurusan::pluck('tipeSatu')-
•          >toArray();
•      $tipeDua = \App\Models\Jurusan::pluck('tipeDua')-
•          >toArray();
•      $tipeTiga = \App\Models\Jurusan::pluck('tipeTiga')-
•          >toArray();
•      $idJurusan = \App\Models\Jurusan::pluck('id')-
•          >toArray();
•      $icons=\App\Models\Tipe::pluck('icon')->toArray();
•      $major = \App\Models\Jurusan::all();
•      $type=\App\Models\Tipe::all();
•      $hasil = [
•          'Realistic' => 0,
•          'Investigative' => 0,
•          'Artistic' => 0,
•          'Social' => 0,
•          'Enterprising' => 0,
•          'Conventional' => 0
•      ];
•      $RIASEC = [
•          'Realistic' => 1,
•          'Investigative' => 2,
•          'Artistic' => 3,
•          'Social' => 4,
•          'Enterprising' => 5,
•          'Conventional' => 6
•      ];
•      for ($i = 1; $i <= 60; $i++) {
•          $index = ($i - 1) * 4;
•          $tipe = $tipe_id[$index];
•          // $huruf_user='A';

```

```

•           $huruf_user =
•             \App\Models\HasilTesDetail::where('hasil_tes_id',$tes_user-
•               >id)->where('soal_id',$i)->pluck('jawaban_id')->first();
•   // perhitungan skor soal nomor 61 sampai 120
•
•
•       for ($j = 0; $j < 4; $j++) {
•         if ($huruf_user == $huruf[$index + $j]) {
•           switch ($tipe) {
•             case 1:
•               $hasil['Realistic'] +=
•                 $nilai[$index + $j];
•               break;
•             case 2:
•               $hasil['Investigative'] +=
•                 $nilai[$index + $j];
•               break;
•             case 3:
•               $hasil['Artistic'] +=
•                 $nilai[$index + $j];
•               break;
•             case 4:
•               $hasil['Social'] += $nilai[$index
•                 + $j];
•               break;
•             case 5:
•               $hasil['Enterprising'] +=
•                 $nilai[$index + $j];
•               break;
•             default:
•               $hasil['Conventional'] +=
•                 $nilai[$index + $j];
•               break;
•           }
•         }

```

```
•          }
•      }
• // perhitungan skor soal nomor 61 sampai 120
•     for ($i = 61; $i <= 120; $i++) {
•         $index = 240 + ($i - 61) * 2;
•         $huruf_user =
•             \App\Models\HasilTesDetail::where('hasil_tes_id',$tes_user->id)->where('soal_id',$i)->pluck('jawaban_id')->first();
•             // $huruf_user='A';
•         for ($j = 0; $j <= 1; $j++) {
•             if ($huruf_user == $huruf[$index + $j]) {
•                 $tipe = $tipe_id[$index + $j];
•                 switch ($tipe) {
•                     case 1:
•                         $hasil['Realistic'] += 1;
•                         break;
•                     case 2:
•                         $hasil['Investigative'] += 1;
•                         break;
•                     case 3:
•                         $hasil['Artistic'] += 1;
•                         break;
•                     case 4:
•                         $hasil['Social'] += 1;
•                         break;
•                     case 5:
•                         $hasil['Enterprising'] += 1;
•                         break;
•                     default:
•                         $hasil['Conventional'] += 1;
•                         break;
•                 }
•             }
•         }
•     }
• }
```

```

•
•           $total = array_sum($hasil);
•   // mengurutan tipe tertinggi hingga terendah berdasarkan skor
•           arsort($hasil);
•   $rankTipe = array_keys($hasil);
•   $jurusan = [
•           'pertama' => [],
•           'kedua' => [],
•           'ketiga' => []
•       ];
•   // memunculkan jurusan yang sesuai berdasarkan kolom
•   'TipeSatu', 'TipeDua', maupun 'TipeTiga' pada tabel 'jurusan'
•   foreach ($tipeSatu as $a => $tipe) {
•       if ($tipe == $RIASEC[$rankTipe[0]]) {
•           $jurusan['pertama'][$a] = $idJurusan[$a];
•       }
•   }
•   foreach ($tipeDua as $a => $tipe) {
•       if ($tipe == $RIASEC[$rankTipe[1]]) {
•           $jurusan['kedua'][$a] = $idJurusan[$a];
•       }
•   }
•   foreach ($tipeTiga as $a => $tipe) {
•       if ($tipe == $RIASEC[$rankTipe[2]]) {
•           $jurusan['ketiga'][$a] = $idJurusan[$a];
•       }
•   }
•   \App\Models\HasilTes::orderBy('created_at', 'desc')
• ->where('user_id', auth()->user()->id)->first()
• ->update([
•           'tipe_id'=>$rankTipe[0],
•           'hasil'=>(max($hasil) / $total) * 100
•       ]);
•   return view('result', [

```

```

•          'nilai'=>$nilai,
•          'hasil' => $hasil,
•          'total' => $total,
•          'saran1' => $jurusan['pertama'],
•          'saran2' => $jurusan['kedua'],
•          'saran3' => $jurusan['ketiga'],
•          'tipe1' => $rankTipe[0],
•          'tipe2' => $rankTipe[1],
•          'tipe3' => $rankTipe[2],
•          'icon'=>$icons,
•          'major' => $major,
•          'type'=>$type
•      ]);
•  }

```

- Kode di atas sudah disertai cara menampilkan rekomendasi jurusan yang sesuai dengan tipe yang relatif unggul. Langkah selanjutnya, letakkan kode tersebut pada halaman Controller.

3.2.11 Eksport Hasil ke PDF

- Selanjutnya, penambahan fitur eksport hasil tes dalam bentuk file PDF. Fitur ini berguna untuk mendukung perencanaan akademik siswa. Sehingga, tenaga pendidik khususnya guru BK dapat mengatur metode pembelajaran setelah mengetahui hasil tes masing – masing siswa melalui file PDF.
- Langkah pertama, ketik perintah di bawah ini pada terminal, kemudian jalankan satu per satu.

```

composer require barryvdh/laravel-dompdf
php artisan vendor:publish --
provider="Barryvdh\DomPDF\ServiceProvider"
php artisan route:clear
php artisan config:clear
php artisan route:cache

```

- Tambahkan kode di bawah ini pada file config/app.php

```

•  'providers' => [Barryvdh\DomPDF\ServiceProvider::class],
•  'aliases' => ['PDF' => Barryvdh\DomPDF\Facade::class]

```

- Buat file Controller dengan nama *EksporController.php*. Buat *public function* dengan nama *ekspor*.

```

• <?php
•
• namespace App\Http\Controllers;
•
• use Barryvdh\DomPDF\Facade\Pdf;
• use App\Models\HasilTes;
• use Illuminate\Http\Request;
•
• class EksporController extends Controller
• {
•     public function ekspor()
•     {
•         if (!auth()->check()) {
•             return redirect()->route('login')->with('error',
• 'Silakan login terlebih dahulu.');
•         }
•
•         $user = auth()->user();
•         $hasilTes = HasilTes::where('user_id', auth()->user()->id)->orderBy('created_at', 'desc')
•             ->where('status', 'Selesai')->get();
•         if (!$hasilTes) {
•             return redirect()->back()->with('error', 'Hasil
• tes tidak ditemukan!');
•         }
•         $types=\App\Models\Tipe::all();
•         $majors=\App\Models\Jurusan::all();
•         $pdf = Pdf::loadView('hasilTesPDF', compact('user',
• 'hasilTes', 'majors', 'types'));
•         return $pdf->download('hasil-tes-' . $user->name .
• '.pdf');
•     }
• }

```

- Langkah terakhir, buat isi file yang akan diekspor ke dalam file php. Letakkan file ini pada folder *view* dengan nama *hasilTesPDF.blade.php*, dan isi file sebagai berikut.

```

• <!DOCTYPE html>
• <html lang="en">
• <head>
•     <meta charset="UTF-8">

```

```

•      <meta name="viewport" content="width=device-width,
initial-scale=1.0">
•      <title>Hasil Tes PDF</title>
•      <style>
•          p{
•              font-size: 14px;
•              font-weight: bolder;
•          }
•          h1 {
•              text-align:center;
•          }
•      </style>
•  </head>
•  <body>
•      <div>
•          <h1>Hasil Tes {{ $user->name }}</h1>
•      </div>
•      @php
•          $no=1;
•      @endphp
•      @foreach ($hasilTes as $hasil)
•          <hr>
•              <p>No : {{$no++}}</p>
•              <p>Tanggal : {{$hasil->created_at->format('d-m-Y')}}</p>
•              <p>Tipe : {{$hasil->tipe_id}}</p>
•              <p>Skor : {{$hasil->hasil}} %</p>
•              <p>Rekomendasi Jurusan :
•                  @foreach($types as $type)
•                      @if($type->tipe_kepribadian==$hasil->tipe_id)
•                          @php
•                              $id_tipe=$type->id;
•                              $total=0;
•                          @endphp
•                      @endif
•                  @endforeach
•                  @foreach ($majors as $major)
•                      @if($major->tipeSatu==$id_tipe)
•                          <p>{{$major->title}}</p>
•                          @php
•                              $total+=1;
•                          @endphp
•                      @endif
•                  @if ($total==5)
•                      @break
•                  @endif
•              @endforeach

```

- @endforeach
- </body>
- </html>
- File akan terunduh ketika user mengunjungi alamat
http://localhost:8000/ekspor-pdf

3.2.12 Pengaturan Route

- Setelah semua kode dibuat, sambungkan kode tersebut menggunakan halaman route. Pada umumnya setiap halaman memiliki satu route, namun juga bisa tidak. Halaman ini berfungsi menentukan kode mana yang harus dijalankan dalam pemanggilan terkait. Misal, jika *user* menekan label *Psychic* di bagian navbar, maka website akan dialihkan ke halaman utama, yaitu *home*. Berikut strukturnya.

- <?php
-
- use Illuminate\Support\Facades\Route;
- Auth::routes();
- Route::method('/halaman_view',
[App\Http\Controllers\HalamanController::class,
'namaFungsi'])->name('halaman_view');
- Jika halaman hanya bisa diakses bagi pengguna yang sudah login, tambahkan ->middleware('auth');. Sehingga, pengguna yang belum login akan diarahkan ke halaman login.
- Untuk logout, menggunakan route Route::get('/logout',
'App\Http\Controllers\Auth\LoginController@logout')->name('logout');
- Sesuaikan pada masing-masing halaman yang akan ditampilkan, sehingga kode selengkapnya akan seperti ini.

- <?php
-
- use Illuminate\Support\Facades\Route;
- Auth::routes();
- Route::get('/logout',
'App\Http\Controllers\Auth\LoginController@logout')->name('logout');

- Route::get('/', [App\Http\Controllers\HomeController::class, 'index'])->name('home');
- Route::get('/home', [App\Http\Controllers\HomeController::class, 'index'])->name('home');
- Route::get('/beranda', [App\Http\Controllers\BerandaController::class, 'index'])->name('beranda');
- Route::get('/uji', [App\Http\Controllers\halamancontroller::class, 'uji'])->name('uji');
- Route::get('/uji2', [App\Http\Controllers\halamancontroller::class, 'uji2'])->name('uji2');
- Route::get('/hallo', [App\Http\Controllers\HalloController::class, 'index'])->name('hallo');
- Route::get('/history', [App\Http\Controllers\HistoryController::class, 'history'])->name('history');
- Route::get('/ujian', [App\Http\Controllers\halamancontroller::class, 'ujian'])->name('ujian');
- Route::get('/result', [App\Http\Controllers\Hasilcontroller::class, 'result'])->name('result');
- Route::resource('mapel', App\Http\Controllers\MataPelajaranController::class);
- Route::resource('coba', App\Http\Controllers\CobaController::class);
- Route::resource('users', App\Http\Controllers\UserController::class)->middleware('auth');
- Route::resource('jurusan', App\Http\Controllers\JurusanController::class);

- Route::resource('testimoni',
App\Http\Controllers\TestimoniController::class);
- Route::resource('courses',
App\Http\Controllers\CoursesController::class);
- Route::resource('campus',
App\Http\Controllers\CampusController::class);
- Route::resource('tesuji',
App\Http\Controllers\TesUjiController::class)->middleware('auth');
- Route::get('/tesuji/detresult/{id}',
[App\Http\Controllers\TesUjiController::class, 'detresult'])->name('tesuji.detresult');
- use App\Http\Controllers\EksporController;
- Route::get('/ekspor-pdf', [EksporController::class, 'ekspor'])->name('ekspor-pdf');
- use App\Http\Controllers\UserController;
-
- Route::get('/users', [UserController::class, 'index'])->name('users.index');
- Route::post('/users/{id}/deleteProfile',
[UserController::class, 'deleteProfile'])->name('users.deleteProfile');
-
- use App\Http\Controllers\TesUjiController;
- Route::get('/tesuji/reset', [TesUjiController::class, 'reset'])->name('tesuji.reset');
- Route::get('hasil', [TesUjiController::class, 'hasil'])->name('hasil');

BAB IV

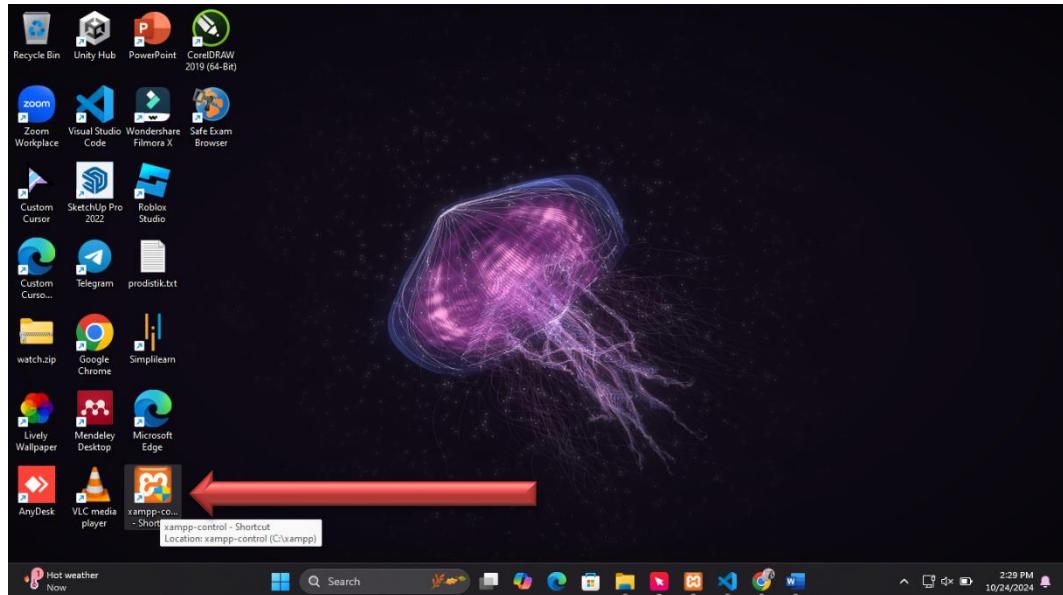
UJI COBA DAN HASIL

Setelah proses perencanaan dan perancangan selesai, dilanjutkan dengan uji coba guna mengetahui hasil dari produk yang telah dibuat.

Berikut langkah - langkahnya:

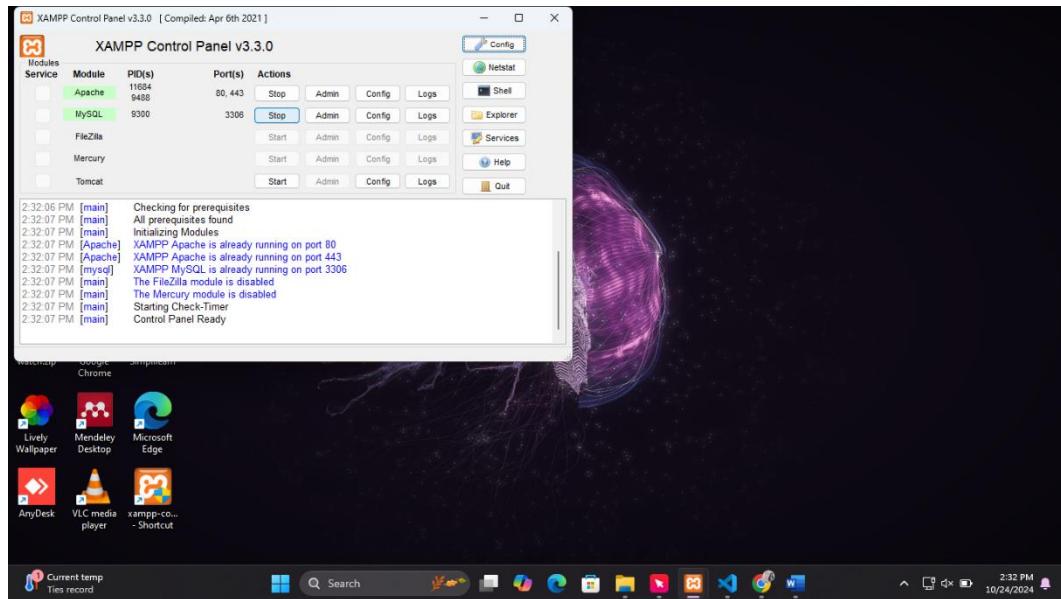
4.1 Uji Coba

1. Pada tampilan awal layar, buka aplikasi *XAMPP Browser*.



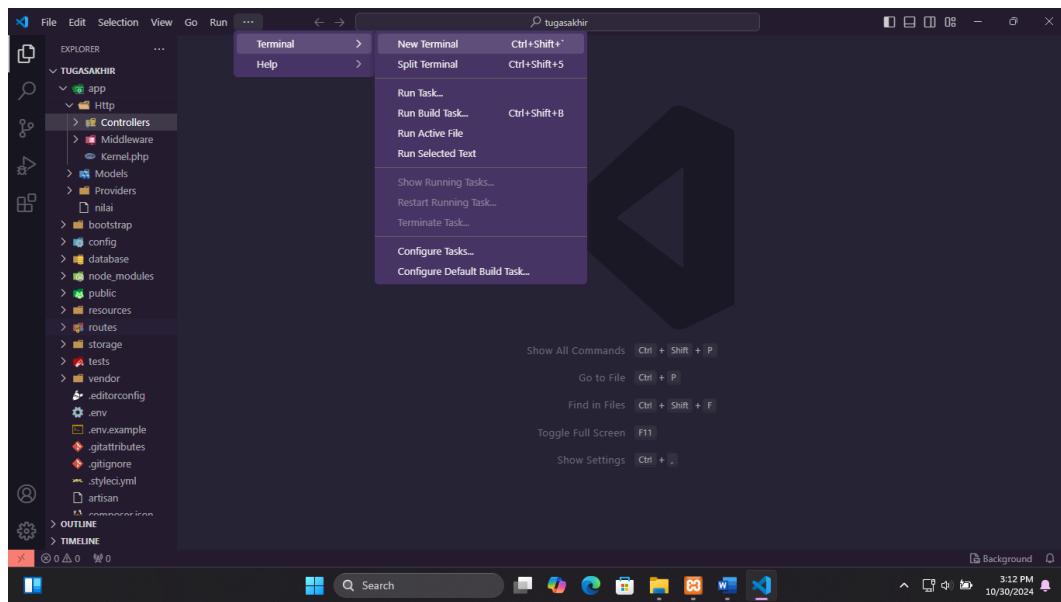
Gambar 4.1.1. Tampilan Awal Layar

2. Pastikan *Apache* dan *MySQL* dalam keadaan aktif. Jika tidak, maka aktifkan dengan menekan tombol *Start*.



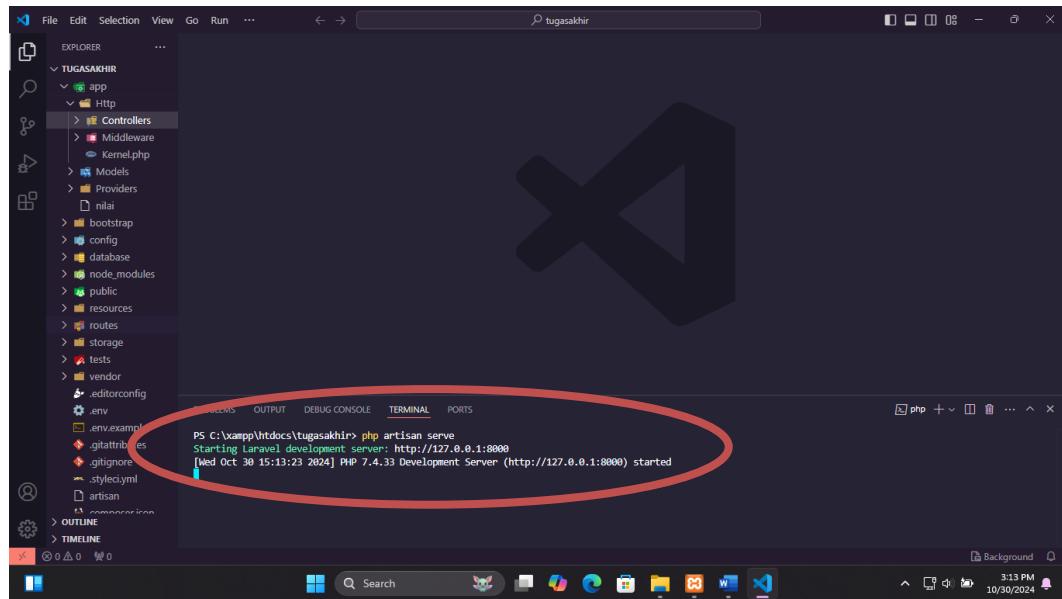
Gambar 4.1.2. Aktifkan XAMPP

3. Aktifkan terminal pada aplikasi *Visual Studio Code* dengan menekan tombol *terminal* di bagian atas, klik *new terminal*, atau dengan cara lain klik *Ctrl + Shift + ‘*.



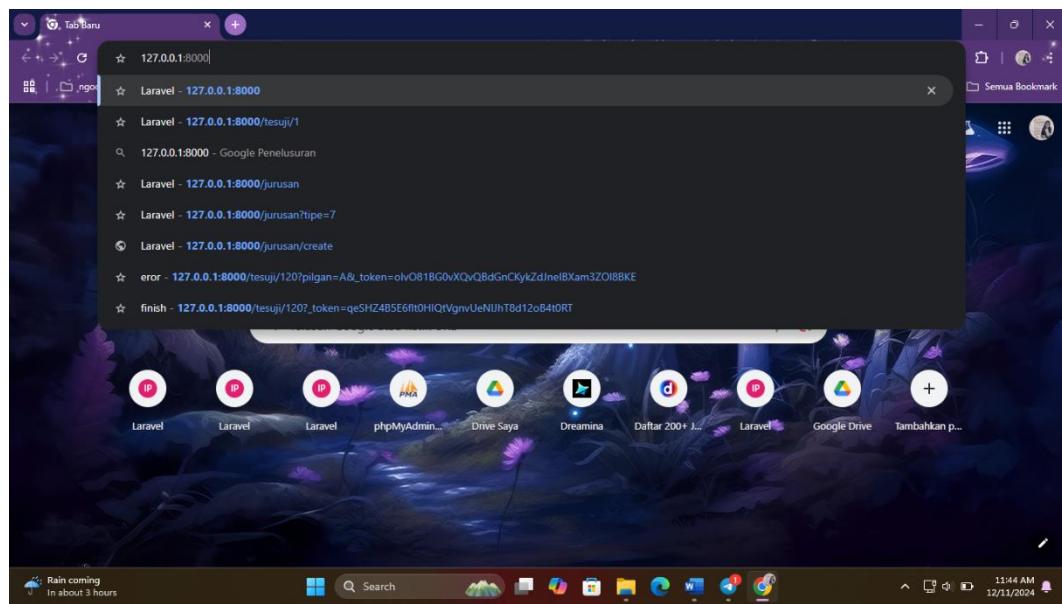
Gambar 4.1.3. Buka Terminal

4. Ketik ‘php artisan serve’ di bagian terminal lalu tekan enter hingga muncul tulisan berwarna hijau seperti ini. Hal ini menunjukkan bahwa terminal berhasil dijalankan.



Gambar 4.1.4. Menjalankan Terminal

5. Buka *Google Chrome* untuk melihat hasilnya. Ketik alamat <http://127.0.0.1:8000>. Browser akan memunculkan tampilan *home* sebagai halaman utama.

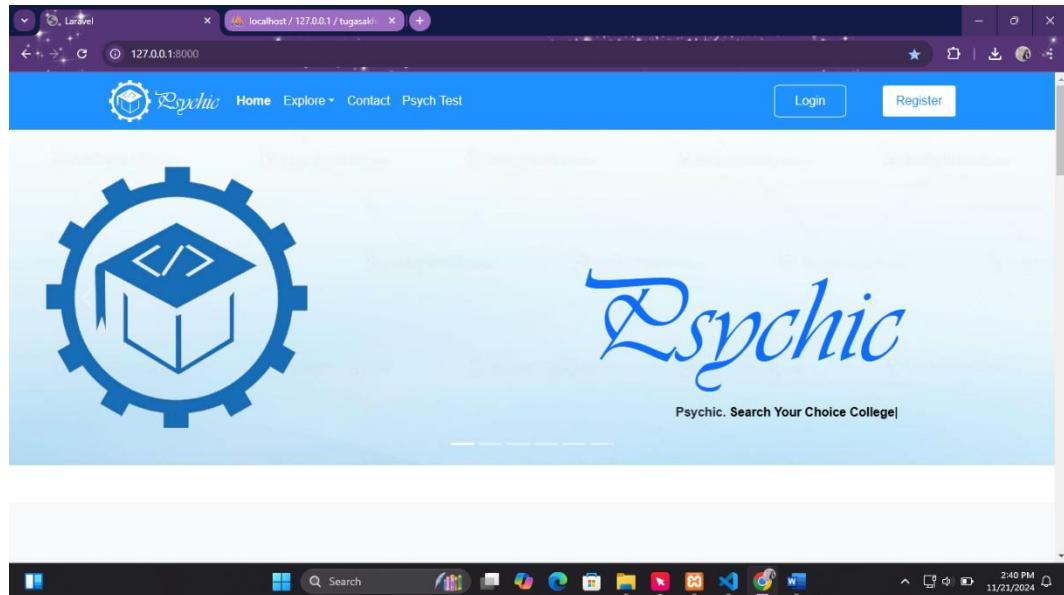


Gambar 4.1.5. Ketik ID Address

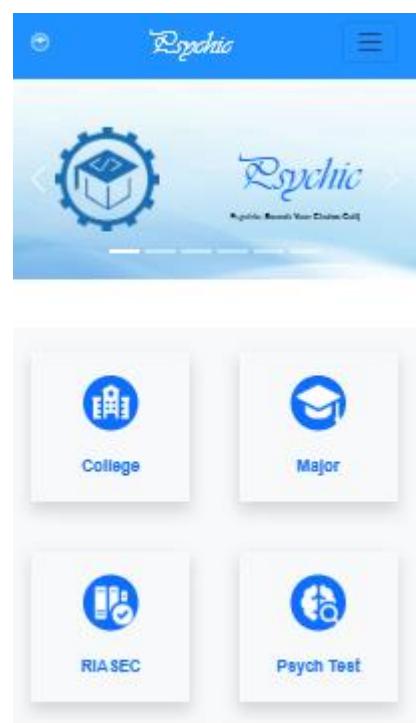
4.2. Hasil

Berikut gambar beserta penjelasan singkat mengenai website *Psychic*.

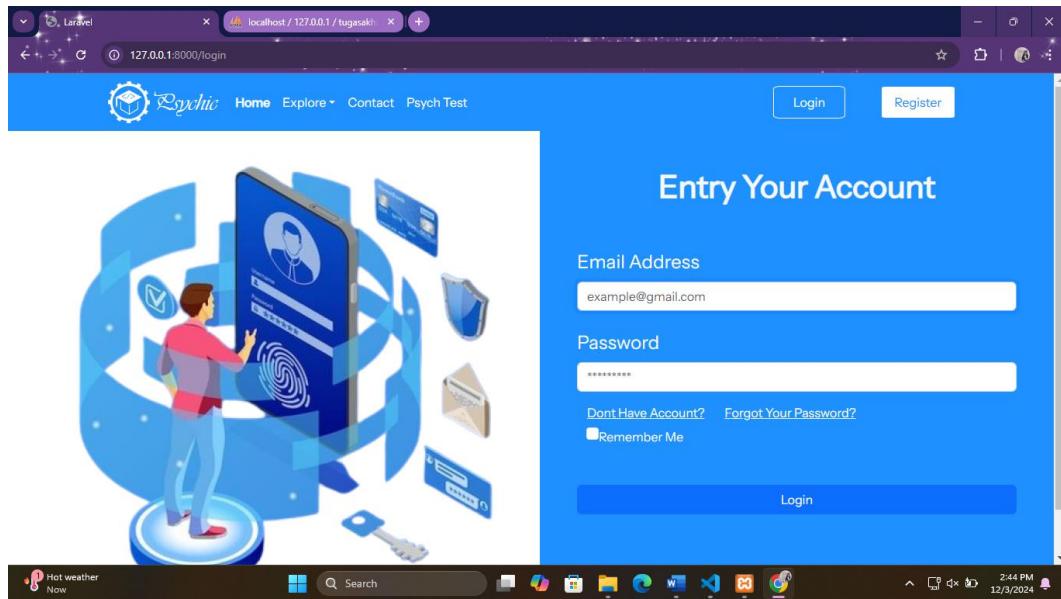
1. Tampilan halaman website.



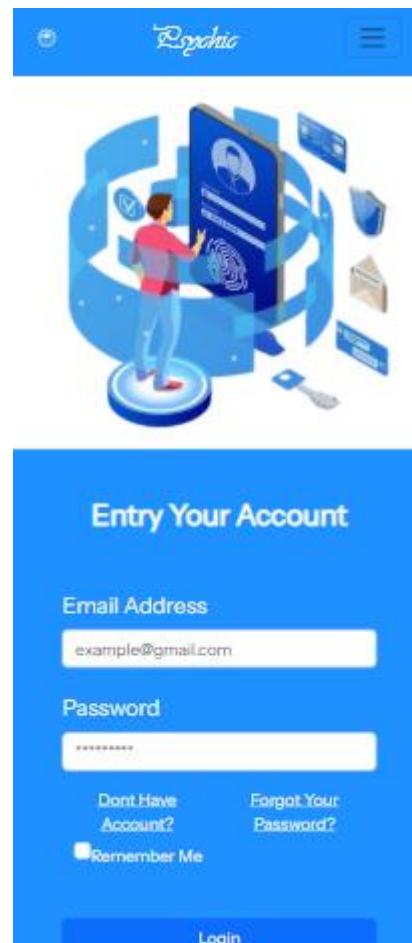
Gambar 4.2.1. Tampilan Halaman Home Dekstop



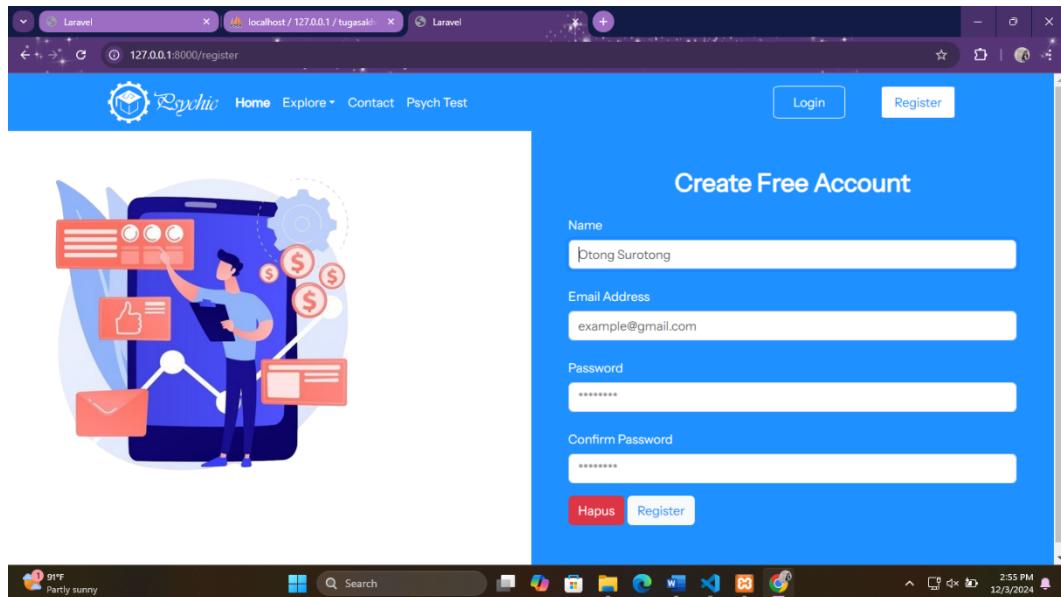
Gambar 4.2.2. Tampilan Halaman Home Android



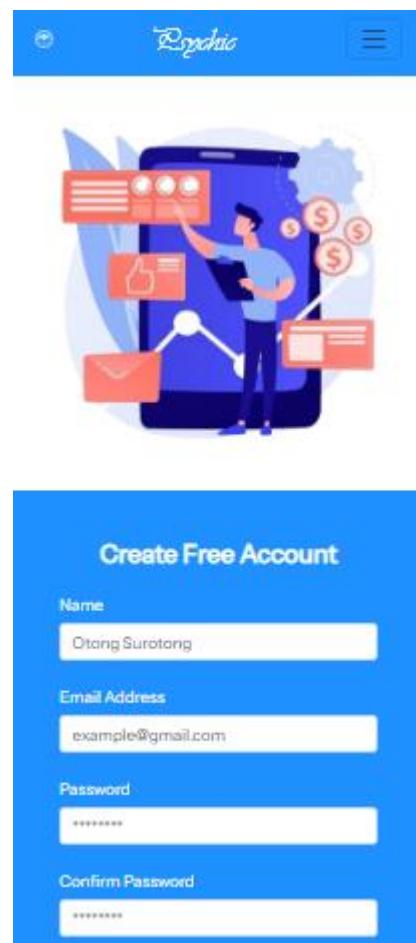
Gambar 4.2.3. Tampilan Halaman Login Dekstop



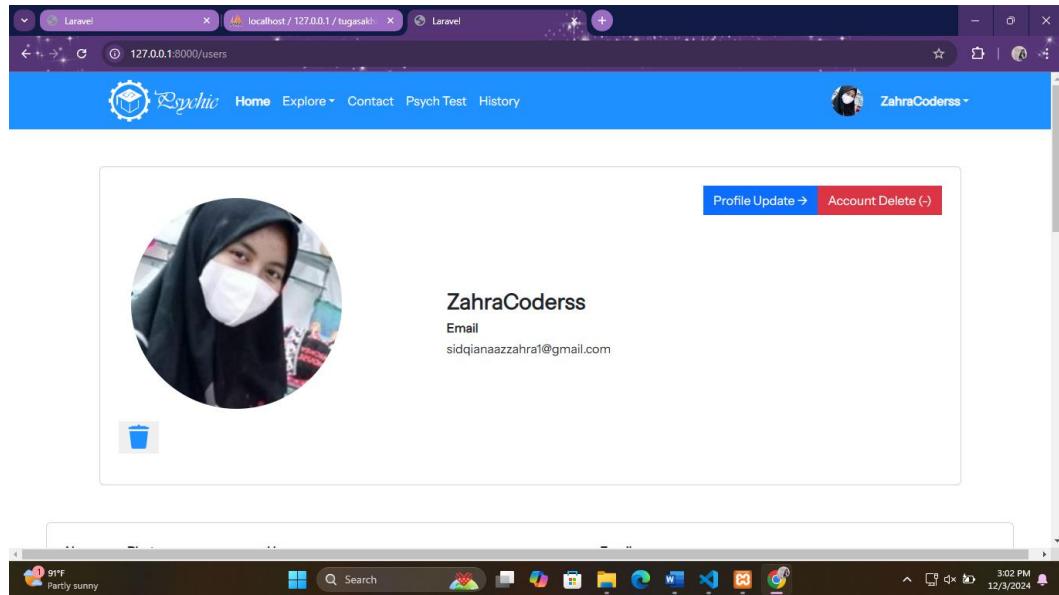
Gambar 4.2.4. Tampilan Halaman Login Android



Gambar 4.2.5. Tampilan Halaman Register Dekstop



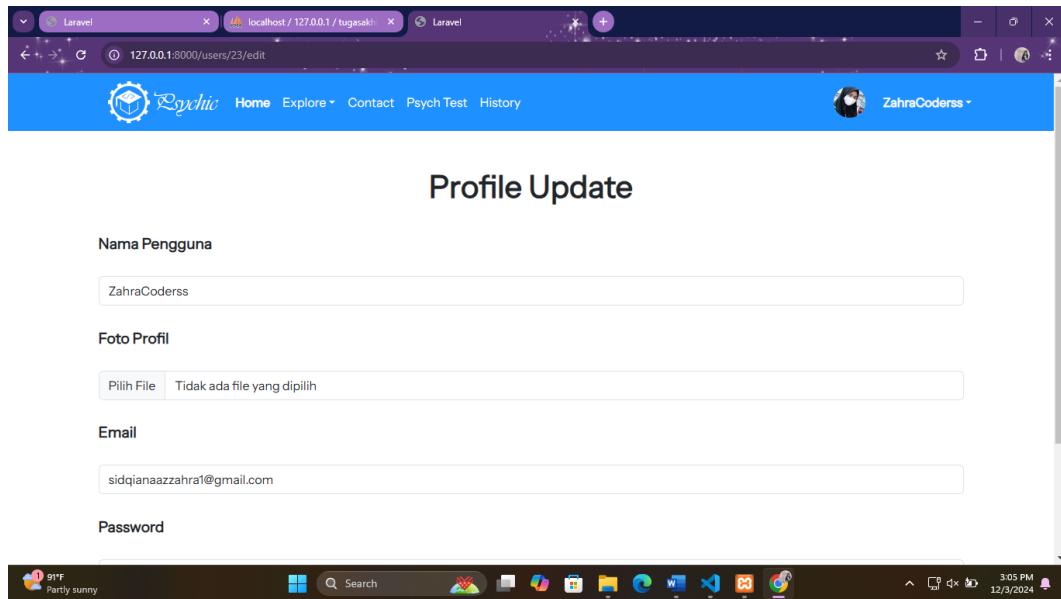
Gambar 4.2.6. Tampilan Halaman Register Android



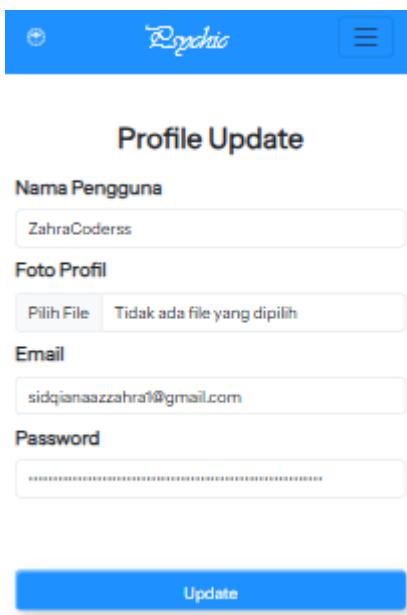
Gambar 4.2.7. Tampilan Halaman Profile Dekstop

| No | Photo | Users | Email |
|----|-------|--------------|----------------------------|
| 1 | | testing | testing@gmail.com |
| 2 | | siapakamu | siapakamu@gmail.com |
| 3 | | akunbaru | baru@gmail.com |
| 4 | | Otong87 | otong@gmaill.com |
| 5 | | ZahraCoderss | sidqianaazzahra1@gmail.com |
| 6 | | Nur Daril | alhamdulillah |

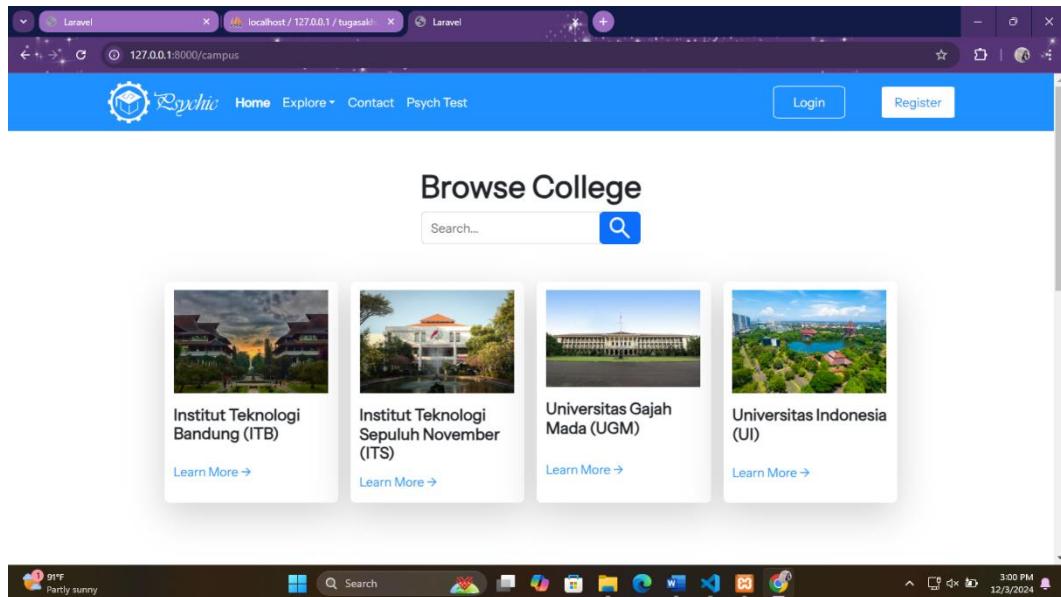
Gambar 4.2.8. Tampilan Halaman Profile Android



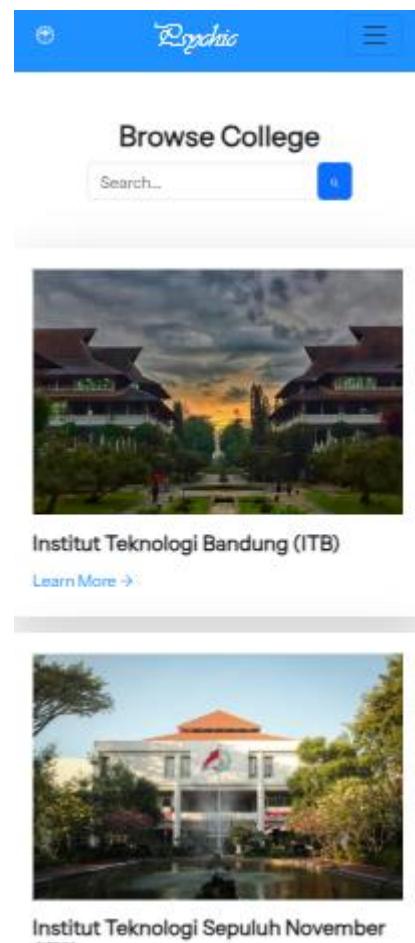
Gambar 4.1.9. Tampilan Profile Update Dekstop



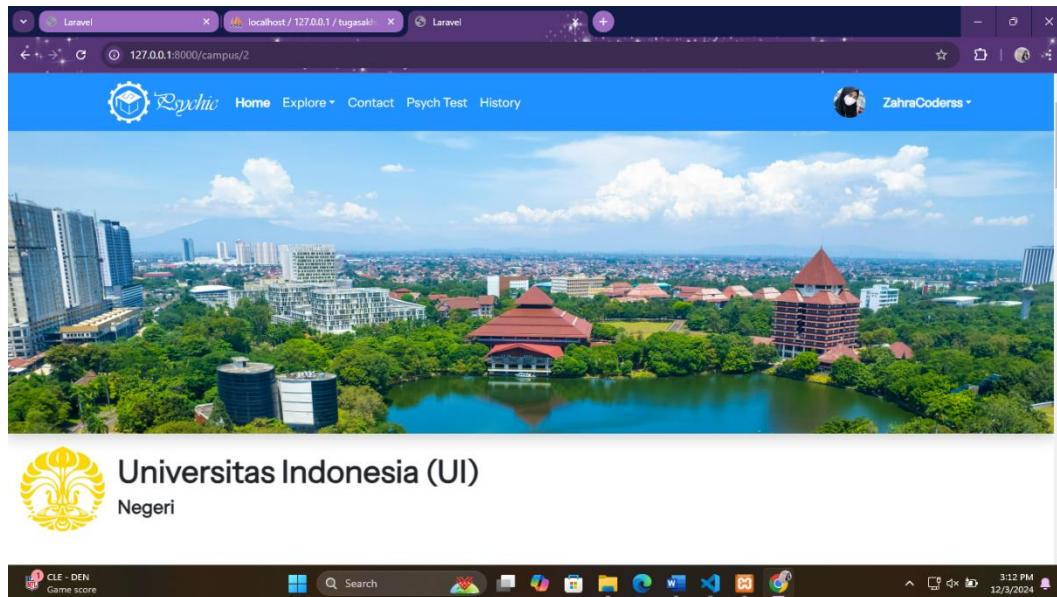
Gambar 4.2.10. Tampilan Profile Update Android



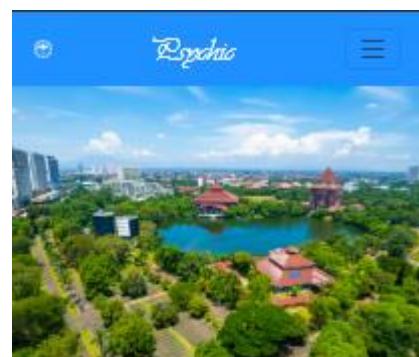
Gambar 4.2.11. Tampilan Halaman Campus Index Dekstop



Gambar 4.2.12. Tampilan Halaman Campus Index Android

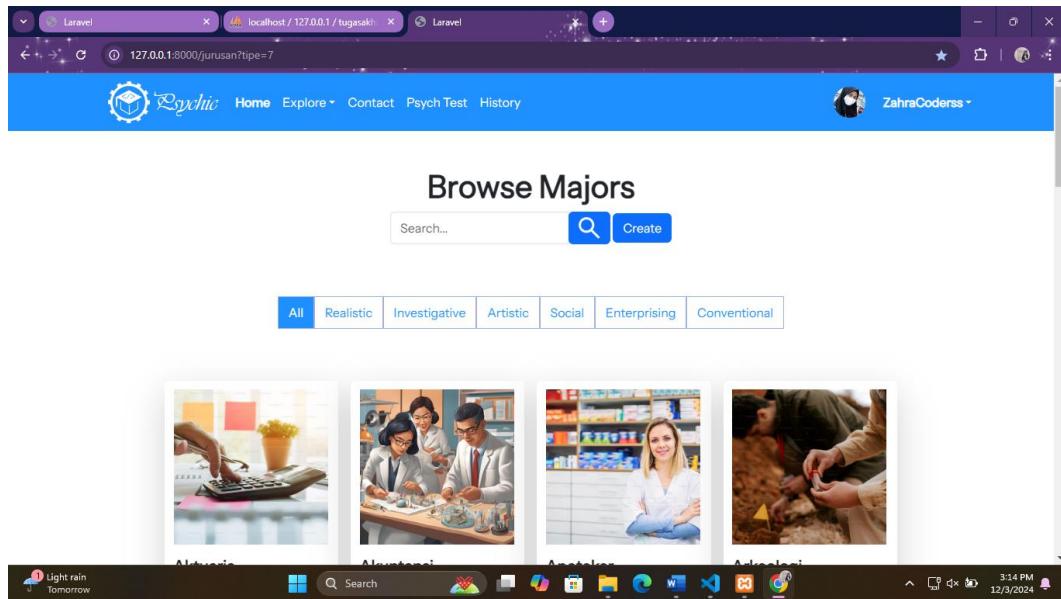


Gambar 4.2.13. Tampilan Halaman Campus Show Dekstop

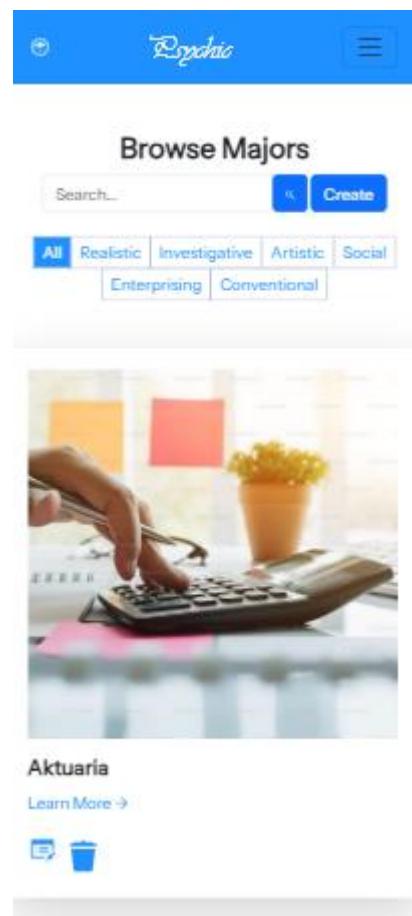


Universitas Indonesia (UI) adalah salah satu universitas riset atau institusi akademik terkemuka di dunia yang terus mengejar pencapaian tertinggi dalam hal penemuan, pengembangan dan difusi pengetahuan secara regional dan global. Dengan prestasi yang terus diraihnya UI berada di peringkat kampus terbaik di Indonesia berdasarkan penilaian Lembaga Pemeringkatan Dunia. Dengan lebih dari 400.000 alumni dalam jaringan yang kuat, UI terus memainkan peran penting di tingkat nasional dan global. Berdasarkan penilaian Quacquarelli Symonds, UI merupakan kampus penghasil lulusan terbaik di Indonesia yang kompeten, inovatif dan efektif di lingkungan kerja. UI terus bekerjasama

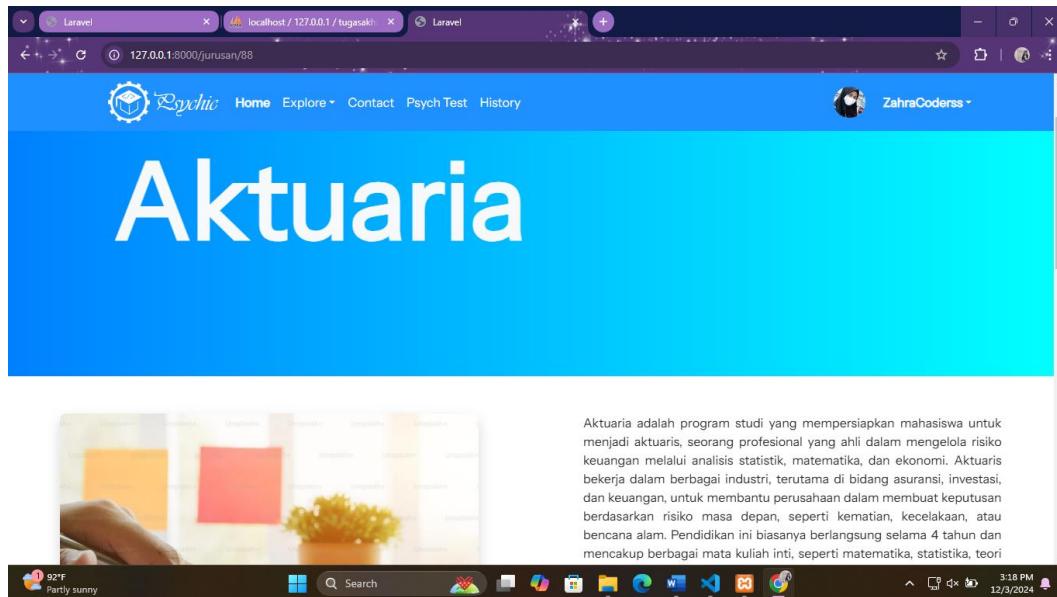
Gambar 4.2.14. Tampilan Halaman Campus Show Android



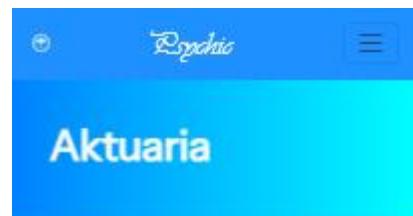
Gambar 4.2.15. Tampilan Halaman Jurusan Index Dekstop



Gambar 4.2.16. Tampilan Halaman Jurusan Index Android

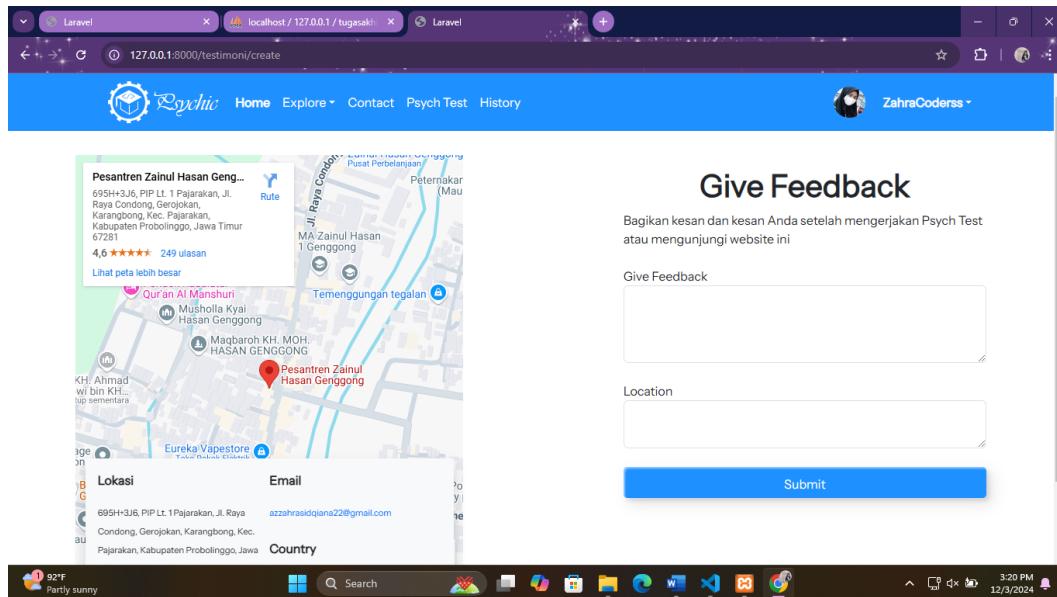


Gambar 4.2.17. Tampilan Halaman Jurusan Show Dekstop

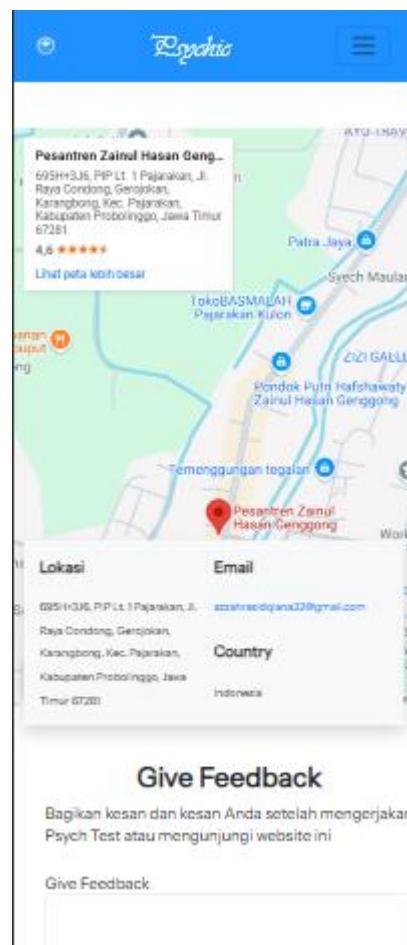


Aktuaria adalah program studi yang mempersiapkan mahasiswa untuk menjadi aktuaris, seorang profesional yang ahli dalam mengelola risiko keuangan melalui analisis statistik, matematika, dan ekonomi. Aktuaris bekerja dalam berbagai industri, terutama di bidang asuransi, investasi, dan keuangan, untuk membantu perusahaan dalam membuat keputusan berdasarkan risiko masa depan, seperti kematian, kecelakaan, atau bencana alam. Pendidikan ini biasanya berlangsung selama 4 tahun dan mencakup berbagai mata kuliah inti, seperti matematika, statistika, teori probabilitas.

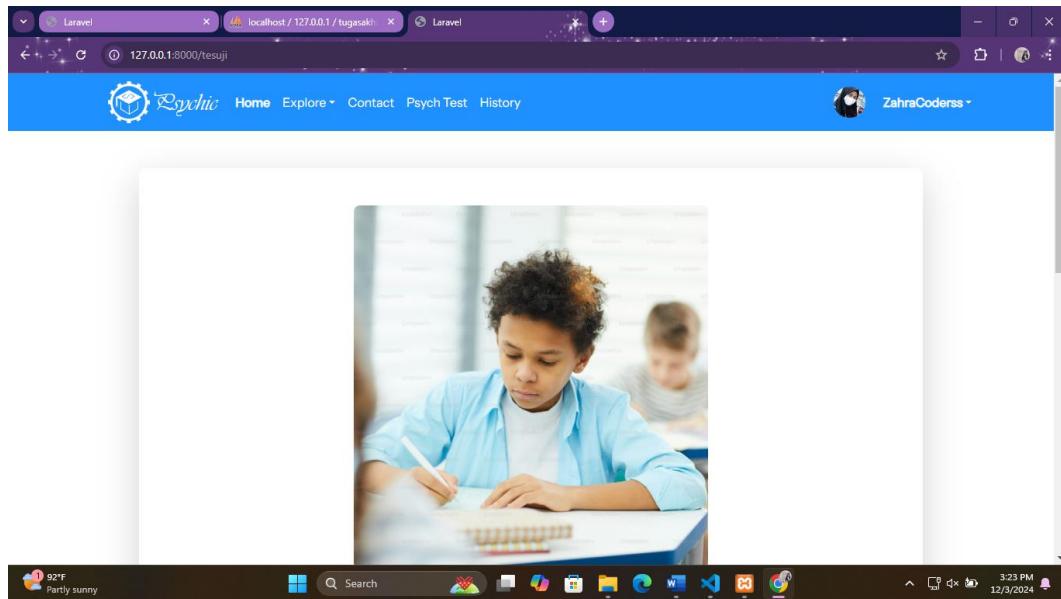
Gambar 4.2.18. Tampilan Halaman Jurusan Show Android



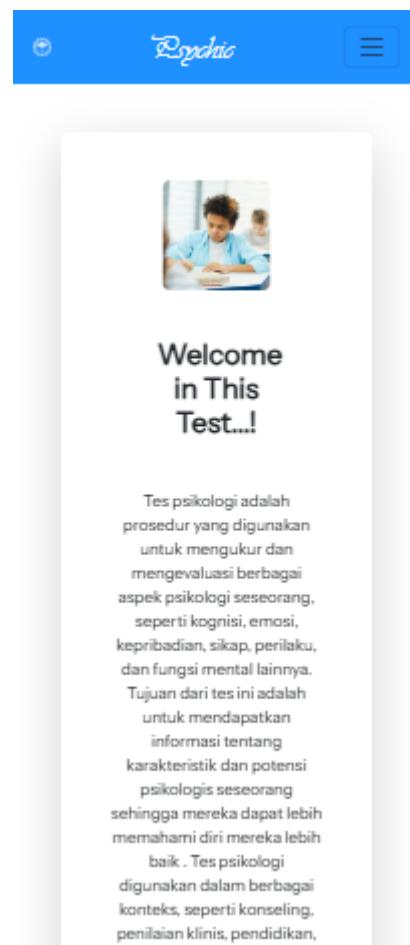
Gambar 4.2.19. Tampilan Halaman Contact Dekstop



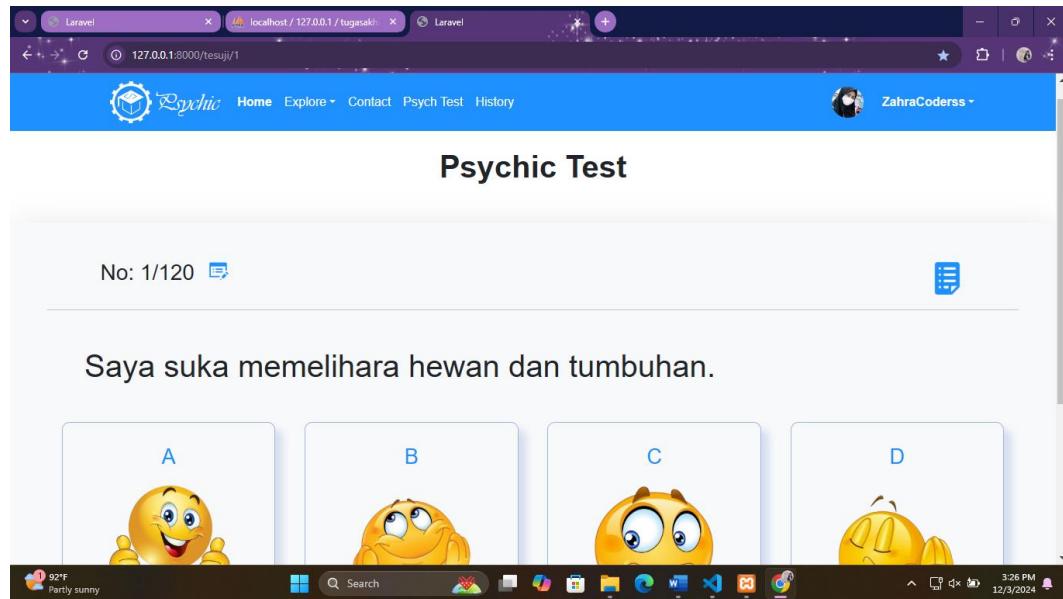
Gambar 4.2.20. Tampilan Halaman Contact Android



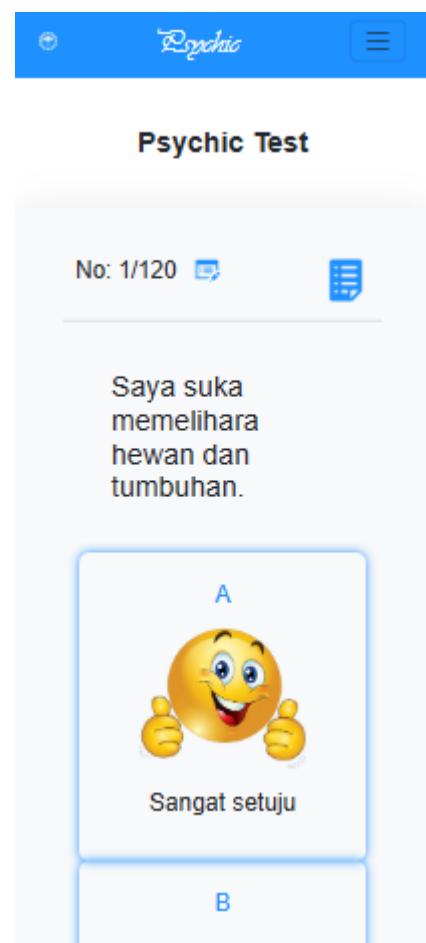
Gambar 4.2.21. Tampilan Halaman Psych Test Dekstop



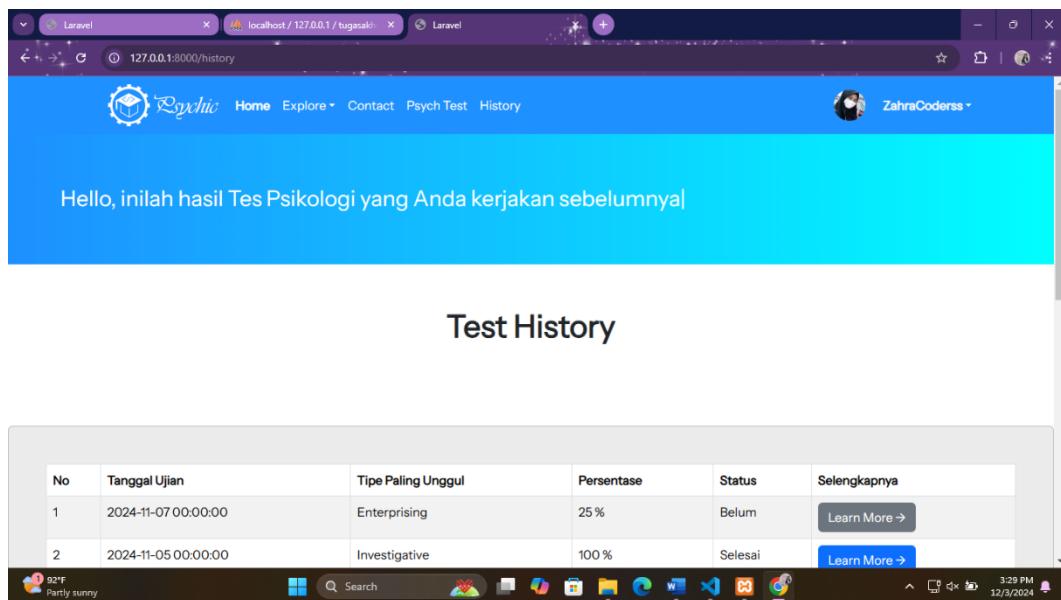
Gambar 4.2.22. Tampilan Halaman Psych Test Android



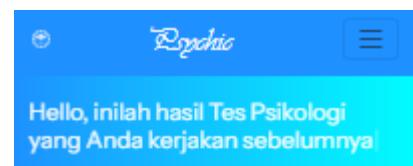
Gambar 4.2.23. Tampilan Halaman Tes Uji Dekstop



Gambar 4.2.24. Tampilan Halaman Tes Uji Android

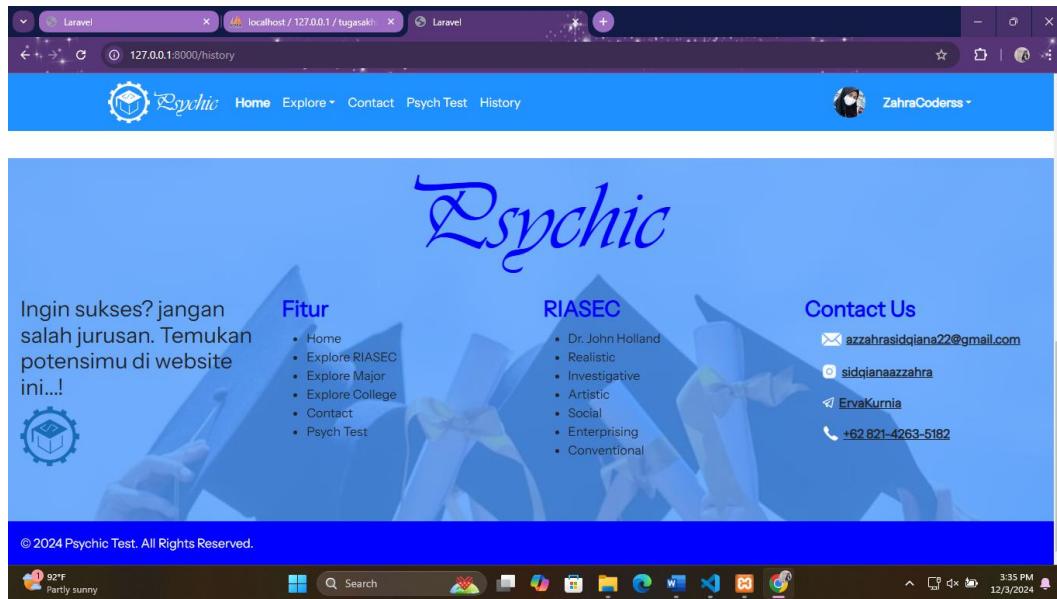


Gambar 4.2.25. Tampilan Halaman History Dekstop



Gambar 4.2.26. Tampilan Halaman History Android





Gambar 4.2.27. Tampilan Footer Dekstop



Gambar 4.2.28. Tampilan Footer Android

2. Analisis Perhitungan Skor Tes

Perhitungan skor masing – masing tipe kepribadian dilakukan dengan menghitung perbandingan skor yang diperoleh, sehingga memunculkan tipe dengan skor yang lebih unggul pada pengguna. Dalam tes ini, skor yang diberikan pada masing – masing jawaban tentu berbeda. Pada tipe soal pertama, jawaban *Sangat Setuju* memiliki skor yang lebih tinggi dibanding jawaban lainnya, dan skor tersebut akan diberikan pada tipe yang sesuai dengan soal yang disajikan. Sedangkan tipe soal kedua, semua memiliki skor yang sama pada setiap jawaban, yaitu 1. Skor ini akan diberikan pada tipe yang sesuai dengan jawaban yang dipilih. Jumlah soal dan skor masing – masing tipe kepribadian telah diseimbangkan satu sama lain, sehingga soal tersebut tidak dapat ditambah maupun dikurangi. Berikut perbandingan perhitungan secara manual dengan koding. Misalnya, pada proses perhitungan data berikut.

| | | Ubah | | Salin | | Hapus | id | created_at | updated_at | hasil_tes_id | soal_id | jawaban_id |
|--------------------------|--|------|--|-------|--|-------|----|------------|------------|--------------|---------|------------|
| <input type="checkbox"/> | | Ubah | | Salin | | Hapus | 31 | 2024-11-05 | 2024-11-06 | | 6 | 1 C |
| <input type="checkbox"/> | | Ubah | | Salin | | Hapus | 32 | 2024-11-05 | 2024-11-06 | | 6 | 2 B |
| <input type="checkbox"/> | | Ubah | | Salin | | Hapus | 33 | 2024-11-05 | 2024-11-06 | | 6 | 3 A |
| <input type="checkbox"/> | | Ubah | | Salin | | Hapus | 34 | 2024-11-05 | 2024-11-06 | | 6 | 120 B |
| <input type="checkbox"/> | | Ubah | | Salin | | Hapus | 35 | 2024-11-06 | 2024-11-06 | | 6 | 4 kosong |
| <input type="checkbox"/> | | Ubah | | Salin | | Hapus | 36 | 2024-11-06 | 2024-11-06 | | 6 | 5 kosong |
| <input type="checkbox"/> | | Ubah | | Salin | | Hapus | 37 | 2024-11-06 | 2024-11-06 | | 6 | 14 B |
| <input type="checkbox"/> | | Ubah | | Salin | | Hapus | 38 | 2024-11-06 | 2024-11-06 | | 6 | 15 A |
| <input type="checkbox"/> | | Ubah | | Salin | | Hapus | 39 | 2024-11-06 | 2024-11-06 | | 6 | 16 C |
| <input type="checkbox"/> | | Ubah | | Salin | | Hapus | 40 | 2024-11-06 | 2024-11-06 | | 6 | 17 B |

Gambar 4.2.29. Contoh Data Jawaban User

| <input type="checkbox"/> | Ubah | Salin | Hapus | 3 | 2024-09-20 06:04:07 | 2024-06-10 11:05:12 | 1 | C | Lumayan | 1 1 |
|--------------------------|------|-------|-------|-----|---------------------|---------------------|-----|---|---|-----|
| <input type="checkbox"/> | Ubah | Salin | Hapus | 6 | 2024-06-10 11:08:33 | 2024-06-10 11:08:33 | 2 | B | Setuju | 2 2 |
| <input type="checkbox"/> | Ubah | Salin | Hapus | 9 | 2024-06-10 11:08:33 | 2024-06-10 11:08:33 | 3 | A | Sangat setuju | 3 3 |
| <input type="checkbox"/> | Ubah | Salin | Hapus | 54 | 2024-06-10 12:00:23 | 2024-06-03 12:00:23 | 14 | B | Setuju | 2 2 |
| <input type="checkbox"/> | Ubah | Salin | Hapus | 57 | 2024-06-10 12:00:23 | 2024-06-10 12:00:23 | 15 | A | Sangat setuju | 3 3 |
| <input type="checkbox"/> | Ubah | Salin | Hapus | 63 | 2024-06-10 12:09:24 | 2024-06-10 12:09:24 | 16 | C | Lumayan | 1 4 |
| <input type="checkbox"/> | Ubah | Salin | Hapus | 66 | 2024-06-10 12:09:24 | 2024-06-10 12:09:24 | 17 | B | Setuju | 2 5 |
| <input type="checkbox"/> | Ubah | Salin | Hapus | 360 | 2024-08-15 15:46:14 | 2024-06-26 14:10:08 | 120 | B | Pendataan masyarakat yang yang berhak mendapat ban... | 1 6 |

Gambar 4.2.30. Contoh Data Skor

- Perhitungan manual

Berdasarkan tabel di atas, pengguna memiliki total skor pada tipe:

1. Realistic : 1
2. Investigative : $2+2 = 4$
3. Artistic : $3+3 = 6$
4. Social : 1
5. Enterprising : 2
6. Conventional : 1

Sehingga, total skor keseluruhan sebanyak 15. Skor akhir masing – masing tipe kepribadian diperoleh dari rumus

$$\frac{\text{skor masing – masing tipe}}{\text{total skor semua tipe}} \times 100\% = \text{hasil akhir masing – masing tipe}$$

Misalnya $\frac{4}{15} \times 100\% = 26.6666 \dots \%$. Sehingga, skor untuk tipe Investigative adalah 26.6666...% atau dibulatkan menjadi 26.67%. Begitupun tipe lainnya.

Sedangkan Tipe yang dikatakan unggul apabila hasil skornya melebihi $\frac{100}{6}$, karena tes ini terdiri atas enam tipe berbeda.

- Perhitungan koding

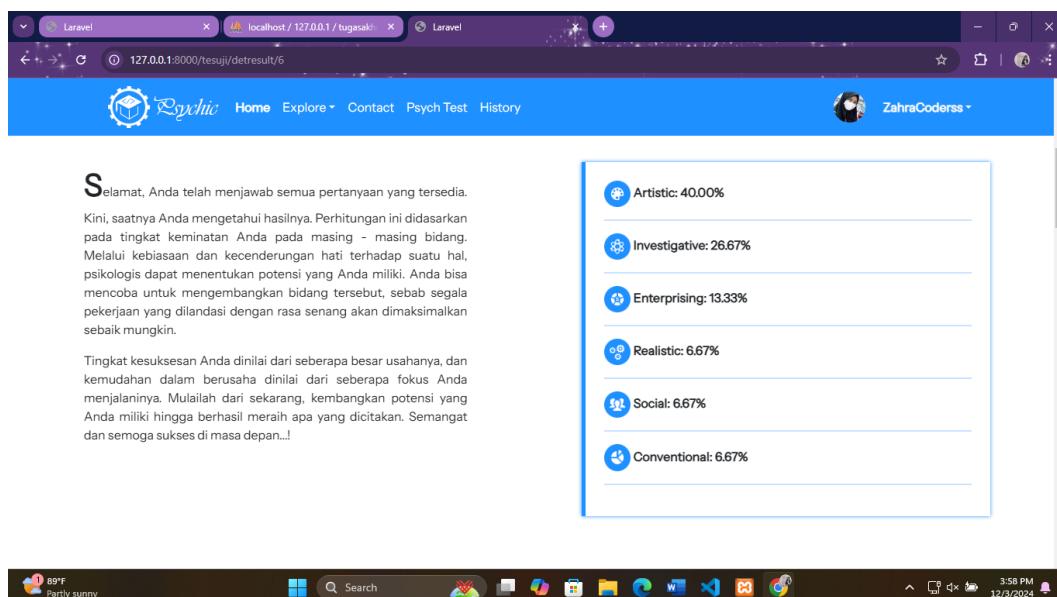
Sebelum melakukan perhitungan, diperlukan deklarasi variabel yang memuat data skor masing – masing soal dengan syarat id tes sesuai dengan tes yang dimaksud. Kemudian, diperlukan fungsi *if*

else dan *for loop* untuk menjumlahkan masing – masing skor sesuai tipe pada tabel jawaban. Begitupun variabel *total* yang memuat jumlah semua skor yang didapat. Skor akhir masing – masing tipe kepribadian diperoleh dari rumus

$$\frac{\text{skor masing – masing tipe}}{\text{total skor semua tipe}} \times 100\% = \text{hasil akhir masing – masing tipe}$$

Kode perhitungan skor tercantum pada BAB III halaman *HasilController*.

Hasilnya akan terlihat sama dengan perhitungan secara manual, yakni skor pada tipe Investigative bernilai 26.67%.



Gambar 4.2.31. Hasil Perhitungan Skor Secara Koding

BAB V

PENUTUP

5.1 Kesimpulan

Rancangan dan Pembuatan Website ‘Psychic’ Sebagai Sarana Tes Psikologi Berdasarkan Metode RIASEC Oleh Dr. John Holland Menggunakan Laravel 8 merupakan panduan yang dibutuhkan selama proses pembuatannya. Mulai dari bahan hingga kode yang harus dijalankan. Website Psychic ditujukan kepada siswa SMA atau sederajat yang ingin melanjutan studi ke perguruan tinggi. Melalui tes psikologi yang telah disediakan, diharapkan mampu memberikan rekomendasi terbaik mengenai jurusan perkuliahan sesuai tipe kepribadian masing – masing pengguna. Di samping itu, website ini juga menyediakan fitur penunjang mengenai informasi kampus dan jurusan yang ada di Indonesia sebagai gambaran tentang studi yang diminati.

Proses Perancangan dan Pembuatan Website ‘Psychic’ Sebagai Sarana Tes Psikologi Berdasarkan Metode RIASEC Oleh Dr. John Holland Menggunakan Laravel 8 diawali dengan tiga proses instalasi, yaitu instalasi XAMPP, Composer, dan Laravel. Website ini dibuat menggunakan bahasa pemrograman PHP. Sehingga, dibutuhkan aplikasi *Code Editor* seperti *Visual Studio Code* atau semacamnya. Alangkah baiknya untuk mendalami pemrograman dasar seperti HTML, CSS, JavaScript, PHP, dan Database terlebih dahulu. Kemudian, dilanjutkan dengan mengikuti langkah - langkah penerapan kode, mulai dari pembuatan tabel database, perancangan tampilan, hingga proses perhitungan skor. Terakhir, perlu adanya proses uji coba manual dan otomatis agar mengetahui apakah kode yang dijalankan sudah sesuai rencana. Oleh sebab itu, laporan ini dibuat sebagai pemandu dalam memahami proses pembuatan website ‘Psychic’.

5.6 Kritik dan Saran

Kritik dan saran sangat diharapkan dari pihak manapun demi perbaikan di masa selanjutnya, mengingat penulis masih menduduki tahap pemula dalam bidang programming sehingga banyak kekurangan dalam pembuatan website ‘Psychic’ ini.

DAFTAR PUSTAKA

Agusriandi.2018.*Dasar-Dasar Penguasaan Pemrograman Web*.Yogyakarta:CV Budi Utama.

Enterprise, J.2021.*PHP Edisi Lengkap*.Jakarta:PT Gramedia.

<https://akupintar.id/info-pintar/-/blogs/tes-mbti-16-tipe-kepribadian-klasifikasi-dan-penjelasannya>, Diakses 5 Maret 2024.

<https://akupintar.id/universitas>, Diakses 5 Maret 2024.

<https://bimba-aueo.com/manakah-lebih-penting-minat-bakat/>, Diakses 5 Maret 2024.

<https://blog.maukuliah.id/mengenal-tes-bakat-holland-riasec-arti-jenis-dan-template-tesnya/>, Diakses 8 Maret 2024.

<https://campus.quipper.com/>, Diakses 8 Maret 2024

<https://cnbcindonesia.com/tech/20220921131159-37-373856/mengenal-apa-itu-internet-sejarah-perkembangan-manfaatnya#:~:text=Internet%20adalah%20singkatan%20dari%20Interconnected,saja%20dan%20dengan%20siapa%20saja>, Diakses .20 Maret 2024

<https://dicoding.com/academies/123/tutorials>, Diakses 21 Maret 2024

https://en.wikipedia.org/wiki/John_L._Holland, Diakses 21 Maret 2024

<https://hellosehat.com/mental/sanguinis/>, Diakses 21 Maret 2024

<https://hostinger.co.id/tutorial/cara-instal-xampp>, Diakses 17 April 2024

<https://hostinger.co.id/tutorial/cara-install-wordpress-di-xampp/>, Diakses 17 April 2024

https://id.wikipedia.org/wiki/Bahasa_pemrograman, Diakses 18 April 2024

https://id.wikipedia.org/wiki/Daftar_perguruan_tinggi_negeri_di_Indonesia, Diakses 1 Mei 2024

https://id.wikipedia.org/wiki/Daftar_perguruan_tinggi_swasta_di_Indonesia,
Diakses 1 Mei 2024.

<https://jakarta.telkomuniversity.ac.id/laravel-definisi-cara-kerja-keunggulan-dan-kekurangan>, Diakses 9 Mei 2024.

<https://lpka.umy.ac.id/kenali-kepribadian-plegmatis-si-penakut/>, Diakses 9 Mei 2024.

<https://pyfahealth.com/blog/kepribadian-melankolis-ciri-ciri-kelebihan-dan-kekurangannya/>, Diakses 9 Mei 2024.

<https://prestasikita.com/2023/01/30/kenali-jenis-tipe-kepribadian-manusia/>,
Diakses 13 Mei 2024.

<https://simumtaz.mazainulhasan1.sch.id/login>, Diakses 28 Mei 2024.

<https://unsplash.com/>, Diakses 18 Agustus 2024.

<https://www.ayongoding.com/install-composer-di-windows/>, Diakses 20 Agustus 2024.

<https://www.bola.com/ragam/read/5432557/arti-kepribadian-koleris-beserta-ciri-kelebihan-dan-kekurangannya?page=2>, Diakses 20 Agustus 2024.

https://www.canva.com/id_id/login/?redirect=%2Fdesign%2FDAGLSMR8SCQ%2FG8gs3gRVx-0RkAtJit4PEw%2Fedit%3Fui%3DeyJljp7IkE_IjoiViIsIkIiOjJCIn0sIkciOnsiQil6dHJ1ZX19, Diakses 22 Agustus 2024.

<https://www.chatgpt.com/>, Diakses 22 Agustus 2024.

<https://www.detik.com/edu/edutainment/d-7318249/angka-pengangguran-di-indonesia-tinggi-kemenko-pmk-ungkap-penyebabnya>, Diakses 22 Agustus 2024.

<https://www.detik.com/edu/seleksi-masuk-pt/d-6716044/daftar-200-jurusan-kuliah-di-indonesia-yang-perlu-kamu-ketahui>, Diakses 15 September 2024.

[, Diakses 15 September 2024](https://www.google.com/search?q=tampilan+halaman+login+blue&sca_esv=9ce04a1690bce6d9&sca_upv=1&rlz=1C1CHBD_idID1020ID1020&udm=2&biw=1366&bih=633&ei=xM-YZoDTBt-OseMPrs232As&ved=0ahUKEwiAxYjCk7CHAxVfR2wGHa7mDbsQ4dUDCB&uact=5&oq=tampilan+halaman+login+blue&gs_lp=Egxnd3Mtd2l6LXNlcnAiG3RhbXBpbGFuIGHhbGFtYW4gbG9naW4gYmx1ZUiwF1DBBFiQE3ABeACQAQCYATgAbQCqgEBNbgbA8gBAPgBAZgCAqACRMICBRAAGIAEwgIEEAAYHpgDAIgGAZIHATKgB4IC&sclient=gws-wiz-serp#vhid=BAenaqITogCifM&vssid=mosaic)

[, Diakses 16 September 2024](https://www.hostinger.co.id/tutorial/cara-install-wordpress-di-xampp/)

[, Diakses 17 September 2024.](https://www.hops.id/hobi/pr-2942601482/langkah-langkah-cara-install-laravel-8-gampang-kok)

[, Diakses 8 November 2024.](https://www.kemdikbud.go.id/main/index.php/tentang-kemdikbud/upt/perguruantingginegeriakademik)

[, Diakses 10 November 2024](https://www.prestasikita.com/2023/01/30/kenali-jenis-tipe-kepribadian-manusia/)

[, Diakses 10 November 2024.](https://www.youtube.com/)

BIOGRAFI PENULIS



Sidqiana Azzahra, atau kerap disapa dengan nama Zahra. Seorang anak sulung dari dua bersaudara yang dilahirkan di Kabupaten Probolinggo, Ahad, 22 Juli 2007. Menempuh pendidikan di MA Zainul Hasan 1 Genggong, Madrasah Mumtaz Berkualitas sejak tahun 2022 silam. Minatnya di bidang programming tumbuh sejak mengikuti program unggulan PRODISTIK di madrasah ini. Harapannya, minat ini tidak berhenti di sini saja, melainkan terus berlanjut ke jenjang berikutnya, yakni jurusan Teknik Informatika, bahkan hingga tercapai cita – citanya sebagai developer di masa mendatang.