

**IMPLEMENTASI APLIKASI ARSITEK.CO BERBASIS
MOBILE DAN *WEBSITE* UNTUK MENGHUBUNGKAN
MASYARAKAT DENGAN TENAGA PROFESIONAL DI
BIDANG ARSITEKTUR**

LAPORAN PROYEK 3



Oleh:

Kelompok 3

- | | |
|---------------------------|----------------|
| 1. Eko Permana | 2003068 |
| 2. Lely Maria Kova | 2003075 |
| 3. Rifki Zahri | 2003085 |

**PROGRAM STUDI D3 TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
POLITEKNIK NEGERI INDRAMAYU
APRIL 2022**

HALAMAN PENGESAHAN
IMPLEMENTASI APLIKASI ARSITEK.CO BERBASIS *MOBILE* DAN
***WEBSITE* UNTUK MENGHUBUNGKAN MASYARAKAT DENGAN**
TENAGA PROFESIONAL DI BIDANG ARSITEKTUR

Disusun oleh :

- | | |
|---------------------------|----------------|
| 1. Eko Permana | 2003068 |
| 2. Lely Maria Kova | 2003075 |
| 3. Rifki Zahri | 2003085 |

Laporan Proyek 3 disetujui oleh:

Dosen	:	<u>Fachrul P. B. M., S.ST., M.Kom.</u>	
Fasilitator		NIP 199204232018031001

Indramayu, 15 Juni 2022
Koordinator Program Studi
D3 Teknik Informatika

Fachrul P. B. M., S.ST., M.Kom.
NIP 199204232018031001

PERNYATAAN

Dengan ini kami menyatakan dengan sesungguhnya bahwa Proyek 3 ini adalah asli hasil karya kami sendiri serta Tugas Proyek 3 ini tidak terdapat karya yang pernah diajukan untuk memenuhi nilai mata kuliah Proyek 3 di suatu Perguruan Tinggi, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis dirujuk dalam naskah ini dan disebutkan dalam daftar pustaka.

Indramayu, 15 Juni 2022

Yang Menyatakan,

Eko Permana

NIM. 2003068

ABSTRAK

Rumah yang indah dan nyaman adalah pencapaian yang ingin didapatkan oleh setiap manusia. Tetapi pengetahuan masyarakat terhadap desain arsitektur masih minim. Sedangkan tenaga konsultan kesulitan menyajikan jasanya. Akhirnya observasi dengan metode scrum pun dilakukan di PT Lima Benua Consultant untuk mengamati proses perencanaan kerja sampai desainnya. Observasi ini menghasilkan sebuah Implementasi Aplikasi Arsitek.co Berbasis Mobile dan Website untuk menghubungkan masyarakat dengan tenaga profesional dibidang arsitektur.

Kata Kunci : *rumah, desain, arsitek, kontruksi*

ABSTRACT

A beautiful and comfortable home is an achievement that every human being wants to get. However, public knowledge of architectural design is still minimal. Meanwhile, the consultants have difficulty in presenting their services. Finally, observations using the scrum method were carried out at PT Lima Benua Consultant to observe the work planning process to the design. This observation resulted in an Implementation of the Mobile and Website-Based Arsitek.co Application to connect the public with professionals in the field of architecture.

Keywords: house, design, architect, construction

MOTTO

KATA PENGANTAR

Puji syukur kehadirat Allah SWT yang telah melimpahkan rahmatnya sehingga dokumen ini dapat diselesaikan tepat pada waktunya dengan judul “Implementasi Aplikasi Arsitek.co Berbasis Mobile dan Website Untuk Menghubungkan Masyarakat Dengan Tenaga Profesional Di Bidang Arsitektur” Dokumen ini disusun demi memenuhi tugas Mata Kuliah Proyek 3 Program Studi Teknik Informatika Diploma III Politeknik Negeri Indramayu.

Dokumen ini diharapkan dapat memberikan kontribusi dalam Pembelajaran dan memenuhi target pencapaian sebagai tugas dari Mata Kuliah Proyek 3. Harapan kami, dokumen ini menjadi salah satu media yang menarik untuk dibaca dan mudah dipahami oleh seluruh pembacanya. Sehingga, kedepannya banyak aplikasi yang tepat sasaran, inovatif dan mampu memicu berkembangnya dunia teknologi di Indonesia.

Penulis mengucapkan banyak Terima Kasih, khususnya kepada:

1. Bapak Iryanto, S.Si., M.Si selaku Ketua Jurusan Teknik Informatika Politeknik Negeri Indramayu.
2. Bapak Fachrul Pralienka Bani Muhamad, M.Kom., selaku Dosen Pembimbing Proyek 3 kelas D3TI 2C.
3. Bapak Mohammad Yani, ST., MT., M.Sc., selaku Dosen Proyek 3 kelas D3TI 2C.
4. Ibu Esti Mulyani, S.Kom., M.Kom., selaku Dosen Proyek 3 kelas D3TI 2C.
5. Orang tua dan keluarga yang selalu memberikan dukungan kepada kami baik itu moril maupun material.
6. Rekan-rekan D3TI 2C, yang selalu memberikan kebahagiaan setiap harinya sehingga kami tidak merasa terbebani dengan adanya tugas ini.

Indramayu, 12 Mei 2022

Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
HALAMAN PENGESAHAN	ii
PERNYATAAN	iii
ABSTRAK.....	iv
ABSTRACT.....	v
MOTTO	vi
KATA PENGANTAR	vii
DAFTAR ISI.....	viii
DAFTAR TABEL.....	x
DAFTAR GAMBAR.....	xi
DAFTAR LAMPIRAN.....	xii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah.....	3
1.4 Tujuan.....	3
1.5 Manfaat.....	4
1.6 Sistematika Penulisan.....	4
BAB II LANDASAN TEORI.....	6
2.1 Flowchart.....	6
2.2 <i>Scrum</i>	7
2.3 PT. Lima Benua Consultant	7
2.4 <i>REST API</i>	7
2.5 <i>Unified Modeling Language (UML)</i>	7
2.6 <i>Entity Relationship Diagram (ERD)</i>	11
2.7 Metode <i>Prototype</i>	12
2.8 Penelitian Terkait	13
BAB III METODE PELAKSANAAN	14
3.1 Teknik Pengumpulan Data	14
3.2 Metode Pengembangan Perangkat Lunak	14
3.3 Flowchart.....	16
3.4 Use Case.....	17

3.5	Activity Diagram.....	18
3.6	Class Diagram	19
3.7	ERD.....	20
BAB IV HASIL DAN PEMBAHASAN		21
4.1	Hasil	21
4.2	Pembahasan.....	21
4.2.1	Hasil <i>Sprint Planning</i>	21
4.2.2	Hasil Daily Scrum	23
4.2.3	Hasil Sprint Review.....	61
4.2.4	Sprint Retrospective	61
BAB V PENUTUP		63
5.1	Kesimpulan.....	63
5.2	Saran.....	63
DAFTAR PUSTAKA		64
LAMPIRAN.....		66

DAFTAR TABEL

Tabel 2. 1 Flowchart	6
Tabel 2. 2 Diagram Use Case	8
Tabel 2. 3 Diagram Aktivitas.....	9
Tabel 2. 4 Class Diagram.....	10
Tabel 2. 5 Seuquence Diagram	11
Tabel 2. 6 ERD	11
Tabel 4.2. 1 Sprint Planning	21
Tabel 4.2. 2 Sprint Review	61
Tabel 4.2. 3 Sprint Restropective.....	61
Tabel 4.2.2 1 Pembagian Pekerjaan	45

DAFTAR GAMBAR

Gambar 4. 1 Halaman Home	23
Gambar 4. 2 Halaman Login.....	24
Gambar 4. 3 Halaman Register Owner	24
Gambar 4. 4 Halaman Register Konsultan	25
Gambar 4. 5 Halaman Register Kontraktor	25
Gambar 4. 6 Halaman Owner	26
Gambar 4. 7Halaman Inspirasi Desain	26
Gambar 4. 8 Halaman Profile	27
Gambar 4. 9 Halanan List Konsultan.....	27
Gambar 4. 10 Halaman Detail Konsultan	28
Gambar 4. 11 Halaman Project Saya (Owner).....	28
Gambar 4. 12 Halaman Transaksi Payment Gateway	29
Gambar 4. 13 Halaman Lelang (Owner).....	29
Gambar 4. 14 Halanan Lelang Saya (Owner)	30
Gambar 4. 15 Halaman Project Konsultan	30
Gambar 4. 16 Halaman Verifikasi Konsultan.....	31
Gambar 4. 17 Halaman Verifikasi Kontraktor.....	31
Gambar 4. 18 Verifikasi Pembayaran Project Desain Konsultan	31
Gambar 4. 19 Verifikasi Pembayaran Project Kontruksi Kontraktor	32
Gambar 4. 20 Login Mobile.....	33
Gambar 4. 21 Choose Register Mobile.....	34
Gambar 4. 22 Register Owner Mobile	35
Gambar 4. 23 Register Profesional Mobile.....	36
Gambar 4. 24 Inspirasi Desain Mobile	37
Gambar 4. 25 Detail Project Desain Mobile	38
Gambar 4. 26 Profile Owner Mobile	39
Gambar 4. 27 Project Saya Mobile	40
Gambar 4. 28 Detail Project Saya Mobile	41
Gambar 4. 29 Lelang Saya Mobile	42
Gambar 4. 30 Desain Saya Mobile	43
Gambar 4. 31 Halaman Verifikasi Profesional Mobile	44

DAFTAR LAMPIRAN

Lampiran 1 Logo Aplikasi	66
Lampiran 2 Poster	66

BAB I PENDAHULUAN

1.1 Latar Belakang

Rumah yang indah dan nyaman untuk ditinggali adalah suatu pencapaian yang ingin didapatkan oleh setiap manusia. Manusia pada umumnya sering menempatkan pembangunan rumah sebagai tujuan dalam hidupnya. Membangun rumah impian terkadang memiliki banyak kendala salah satunya yaitu kesulitan dalam menentukan desain rumah sampai tahap konstruksi. Kesalahan dalam pembuatan rumah dapat berdampak pada penyesalan tersendiri. Oleh karena itu, pembuatan rumah harus dilakukan secara teliti dan matang agar penyesalan tersebut tidak ada. Tentunya hal ini dibutuhkan pengetahuan mengenai desain rumah yang dirancang oleh tenaga arsitek konsultan profesional dan tahap konstruksi yang dikerjakan oleh tenaga kontraktor dengan tim yang berpengalaman.

Tahapan dalam pembuatan rumah memiliki langkah yang terbilang rumit, seperti perencanaan, studi kelayakan, pemaparan, perancangan desain, pelelangan, pelaksanaan, pemeliharaan, dan persiapan pemakaian. Memahami tahapan tersebut akan membantu semua pihak yang terlibat sebelum memulai proyek pembangunan. Tanpa rancangan lengkap dan matang dapat menghambat atau menyebabkan kecelakaan. Tentu hal ini perlu dipahami untuk setiap individu yang ingin membangun sebuah rumah agar mendapatkan kenyamanan jangka panjang.

Pengetahuan masyarakat terhadap portofolio karya atau desain arsitektur terkadang masih minim. Sedangkan tenaga arsitek konsultan kesulitan dalam mencari pengguna jasa atau masyarakat yang ingin menggunakan jasanya. Sehingga dibutuhkan wadah yang terpusat untuk menghubungkan tenaga arsitek konsultan profesional dengan masyarakat atau calon pengguna jasa begitupun sebaliknya. Adanya wadah tersebut dapat menyelesaikan masalah yang dialami oleh keduanya.

Dalam dunia proyek konstruksi bangunan dibutuhkan desain dan rencana anggaran biaya yang sangat detail. Detail dalam konstruksi perlu ditampilkan sebagai persyaratan yang wajib dilakukan. Kesalahan dalam mempersiapkan detail tersebut sering sekali ditemukan sehingga pihak kontraktor bangunan mengalami kebingungan dalam mengerjakan sebuah proyek. Diperlukan informasi yang lengkap untuk memberikan gambaran terhadap calon pengguna jasa agar

mempersiapkan segala hal yang dibutuhkan. Oleh karena itu, perlu adanya wadah yang menyajikan informasi tahapan sebelum memulai konstruksi proyek bangunan.

Adapun masyarakat yang telah mempersiapkan semua persyaratan masih diharuskan mencari tenaga kontraktor profesional agar proyek bangunan miliknya dapat dikerjakan dengan baik. Hal itu dibutuhkan tempat atau media terpusat yang menyajikan berbagai profil kontraktor dan tim yang mendukungnya. Profil kontraktor dapat menarik minat calon pengguna jasa sehingga proses pencarian dapat berlangsung dengan mudah. Setelah tahapan pencarian dilakukan pengguna jasa dapat menyepakati ketentuan awal yang ditentukan oleh kontraktor dan ketentuan akhir sesuai dengan isi dari desain dan rencana anggaran biaya milik pengguna jasa. Dengan demikian, kesepakatan yang diperoleh menjadi awal dari sebuah konstruksi proyek bangunan.

Tempat atau wadah yang terpusat menjadi patokan utama dalam mempertemukan semua pihak yang saling berkaitan. Wadah tersebut memberikan banyak peluang terutama peluang kerja setiap sektor dalam bidang arsitek konsultan dan proyek konstruksi bangunan. Peluang kerja yang menjadi luas sehingga dapat menjadi solusi untuk mengurangi angka pengangguran saat ini. Peluang masyarakat dapat menemukan jasa tenaga arsitek konsultan profesional dan menyelesaikan tahapan konstruksi proyek bangunan dengan sukses.

Berdasarkan yang telah dipaparkan, wadah atau aplikasi untuk membantu masyarakat dalam mencari tenaga arsitek profesional masih sangat minim. Adapun aplikasi sejenis yang mengangkat tema yang sama sejauh ini masih belum diketahui masyarakat luas. Aplikasi sejenis juga akan lebih membantu dalam tahap pengembangan dan langsung diketahui masyarakat secara luas. Dengan demikian, penulis ingin membuat aplikasi agar bisa mempermudah masyarakat dalam mencari kebutuhan seputar desain arsitektur dan tahapan konstruksinya.

Dapat disimpulkan bahwa rancangan aplikasi untuk menghubungkan masyarakat dengan tenaga arsitek konsultan profesional dan tenaga kontraktor bangunan memiliki peluang yang sangat besar. Tentu hal ini mendorong penulis untuk membuat “Implementasi Aplikasi Arsitek.co Berbasis *Mobile* Dan *Website* Untuk Menghubungkan Masyarakat Dengan Tenaga Profesional Di Bidang Arsitektur”. Sistem aplikasi yang dibuat akan melihat dari sudut pandang

pengguna aplikasi. Melalui aplikasi penulis berharap dapat membantu masyarakat dalam mencari informasi terkait desain arsitektur dan tahap pembangunannya serta memudahkan arsitek konsultan profesional dan kontraktor bangunan dalam memasarkan jasanya.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang dijabarkan, maka diperoleh suatu rumusan permasalahan:

1. Bagaimana menyajikan karya desain arsitektur para tenaga arsitek konsultan secara terpusat?
2. Bagaimana cara membantu masyarakat dalam menemukan desain arsitektur karya yang disajikan para tenaga arsitek konsultan?
3. Bagaimana mempertemukan para tenaga arsitek konsultan dengan masyarakat?
4. Bagaimana cara membantu masyarakat dalam menyampaikan ide kustomisasi desain arsitektur kepada para tenaga arsitek konsultan?
5. Bagaimana cara penyajian informasi tahapan konstruksi proyek bangunan kepada masyarakat?
6. Bagaimana menyajikan profil tim para kontraktor proyek bangunan secara terpusat?
7. Bagaimana mempertemukan para tenaga kontraktor proyek bangunan dengan masyarakat?

1.3 Batasan Masalah

Berdasarkan latar belakang yang dijelaskan sebelumnya agar dapat terarah dan permasalahan yang dihadapi tidak terlalu luas serta dengan judul penelitian Proyek 3 yang diajukan maka ditetapkan batasan-batasan terhadap masalah yang diteliti. Adapun beberapa hal yang membatasi dalam penelitian Proyek 3 ini, antara lain:

1. studi kasus PT. Lima Benua Consultant,
2. penggunaan framework flutter pada pengembangan aplikasi mobile,
3. penggunaan framework laravel pada pengembangan aplikasi web.

1.4 Tujuan

Tujuan dari pembuatan aplikasi yang dibuat meliputi:

1. mewadahi karya desain arsitektur para tenaga arsitek konsultan secara terpusat,
2. mempertemukan para tenaga arsitek konsultan dengan masyarakat,
3. mewadahi masyarakat dalam mencari karya desain arsitektur,
4. mewadahi profil tim para kontraktor proyek bangunan secara terpusat,
5. mempertemukan para tenaga kontraktor proyek bangunan dengan masyarakat,
6. memberikan informasi mengenai tahapan konstruksi proyek bangunan.

1.5 Manfaat

Manfaat yang dapat diperoleh dengan dibuatnya aplikasi ini adalah:

1. meningkatkan kenyamanan masyarakat dalam mencari desain arsitektur dan tenaga arsitek konsultan,
2. meningkatkan kenyamanan masyarakat dalam memulai tahapan konstruksi proyek bangunan hingga selesai,
3. membangun pemahaman masyarakat mengenai tahapan konstruksi proyek bangunan,
4. meningkatkan peluang kesejahteraan para tenaga arsitek konsultan dan kontraktor proyek bangunan,
5. memberikan kesempatan peluang kerja bagi tenaga arsitek konsultan dan kontraktor proyek bangunan beserta tim konstruksinya,
6. mengurangi angka pengangguran di bidang arsitektur dan konstruksi proyek bangunan.

1.6 Sistematika Penulisan

Dalam sistematika penulisan laporan proyek 3 ini dibagi menjadi lima bab, dimana setiap bagian memiliki pembahasan yang berbeda-beda tetapi saling terkait antara satu dengan lainnya. Untuk memudahkan penulisan laporan proyek 3 ini, maka akan diuraikan dan dijabarkan setiap bagian secara sistematis agar penulisan laporan tersusun dengan sempurna. Adapun sistematika penulisan laporan ini sebagai berikut.

BAB I PENDAHULUAN

Pada bab pendahuluan dipaparkan mengenai latar belakang masalah, rumusan masalah, batasan masalah, tujuan masalah, serta manfaat penelitian “Implementasi

Aplikasi Arsitek.co Berbasis *Mobile* dan *Website* untuk Menghubungkan Masyarakat dengan Tenaga Profesional di bidang Arsitektur”.

BAB II LANDASAN TEORI

Landasan teori merupakan seperangkat definisi maupun konsep yang telah disusun berdasarkan sumber referensi yang valid tentang teori-teori terkait penyusunan laporan proyek 3 serta beberapa literatur *review* yang berhubungan dengan penelitian. Sumber yang dijadikan referensi berasal dari buku maupun internet.

BAB III METODOLOGI PENELITIAN

Metodologi penelitian menjelaskan tentang tahapan serta metode penelitian yang digunakan dalam membangun Implementasi Aplikasi Arsitek.co Berbasis *Mobile* dan *Website* untuk Menghubungkan Masyarakat dengan Tenaga Profesional di bidang Arsitektur. Pada bagian ini terdapat perancangan aplikasi berupa *Flowchart*, *Unified Model Language* (UML), *Entity Relational Diagram* (ERD), dan rancangan user interface yang dibuat untuk menjelaskan tahapan alur proses serta gambaran nyata dari sistem yang akan dibuat.

BAB IV HASIL DAN PEMBAHASAN

Pada bab ini terdapat hasil penelitian serta implementasi membangun Implementasi Aplikasi Arsitek.co Berbasis *Mobile* dan *Website* untuk Menghubungkan Masyarakat dengan Tenaga Profesional di bidang Arsitektur berdasarkan rancangan yang telah dipaparkan. Pada bagian ini dibahas mengenai implementasi *database*, dan implementasi desain *interface*.

BAB V PENUTUP

Bab ini berisi kesimpulan dan saran yang berkaitan dengan analisa dan optimalisasi aplikasi berdasarkan yang telah diuraikan pada bab-bab sebelumnya. Kesimpulan dihasilkan dari rangkuman berdasarkan poin-poin penting pada penulisan laporan proyek 3 Implementasi Aplikasi Arsitek.co Berbasis *Mobile* dan *Website* untuk Menghubungkan Masyarakat dengan Tenaga Profesional di bidang Arsitektur. Sedangkan saran berisi tentang pendapat atau masukan dari penulis kepada pembaca.


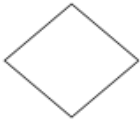








BAB II LANDASAN TEORI

2.1 Flowchart

Flowchart merupakan suatu bagan dengan simbol-simbol tertentu yang menjelaskan urutan proses secara mendetail dan hubungan antara suatu proses dengan proses lainnya dalam suatu program. Diagram ini memberi solusi selangkah demi selangkah untuk penyelesaian masalah yang ada didalam algoritma tersebut.

Flowchart terdiri dari simbol-simbol tertentu, beberapa simbol-simbol yang digunakan dalam *flowchart* dapat dilihat pada Tabel 2.1.

Tabel 2. 1 *Flowchart*

Simbol	Nama	Keterangan
	Garis Alir (<i>Flow Line</i>)	Menunjukkan arah aliran program.
	<i>Decision</i>	Menunjukkan perbandingan pernyataan, penyeleksian data yang memberikan pilihan untuk langkah.
	<i>Terminator</i>	Menujukkan permulaan / akhir program
	<i>Manual Input</i>	Menujukkan pemasukan data secara manual <i>on-line</i> keyboard.
	<i>Proses</i>	Merupakan proses perhitungan/proses pengolahan data.
	<i>Input/Output Data</i>	Menunjukkan proses penginputan/output data, parameter, dan informasi.
	<i>Display</i>	Menunjukkan peralatan output yang digunakan yaitu layar, plotter, printer, dan sebagainya.
	<i>Manual Operation</i>	Menunjukkan pengolahan yang tidak dilakukan oleh komputer/sistem.
	<i>Database</i>	Menunjukkan data-data yang di akses atau disimpan ke <i>database</i> .
	<i>Disk Storage</i>	Menyatakan <i>input</i> berasal dari <i>disk</i> atau <i>output</i> disimpan ke <i>disk</i> .

Sumber: (Malabay, 2016)

2.2 Scrum

Scrum adalah sebuah kerangka kerja yang dapat mengatasi suatu masalah kompleks yang selalu berubah, dan juga dinilai dapat memberikan kualitas produk yang baik sesuai dengan keinginan pengguna secara kreatif dan produktif (Schwaber & Sutherland, 2020).

2.3 PT. Lima Benua Consultant

PT. Lima Benua Consultant adalah perusahaan perancangan konstruksi berbentuk PT. PT. Lima Benua Consultant beralamat di Jl. Raya Plumbon RT. 006 RW. 002, Desa Plumbon, Kecamatan Indramayu, kabupaten Indramayu. PT. Lima Benua Consultant adalah badan usaha berpengalaman yang mengerjakan proyek nasional.

2.4 REST API

REST API RESTful atau *REST API* merupakan implementasi dari API (Application Programming Interface). REST (Representational State Transfer) sendiri merupakan suatu arsitektur metode komunikasi yang melakukan proses transaksi data melalui protokol HTTP yang bertujuan untuk menjadikan performa sistem menjadi lebih baik, cepat, dan mudah untuk dikembangkan terutama dalam transaksi dan komunikasi data (Ainuzzaqy & Nudin, 2020).

2.5 Unified Modeling Language (UML)




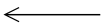





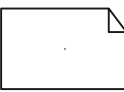
Unified Modeling Language (UML) adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem. UML menjadi bahasa yang bisa digunakan untuk berkomunikasi dalam perspektif objek antara *user* dengan developer, antara developer dengan developer, antara developer analis dengan developer desain, dan antara developer desain dengan developer pemrograman (Suendri, 2018).

UML terdiri atas pengelompokan diagram-diagram sistem menurut aspek atau sudut pandang tertentu. Metodologi UML dapat berjalan pada orientasi objek sehingga sangat disarankan dalam penggunaannya. Beberapa contoh diagram UML diantaranya yaitu *Use Case Diagram*, *Activity diagram* dan *Class diagram* (Subekti et al., 2014).

a. Diagram *Use Case* (*Use Case Diagram*)

Use case diagram menggambarkan fungsionalitas dari sebuah sistem. Sebuah *use case* merepresentasikan sebuah interaksi antara *actor* dengan sistem. Simbol-simbol *use case diagram* dapat dilihat pada Tabel 2.2.

Tabel 2. 2 Diagram *Use Case*

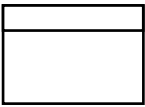

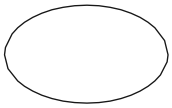




Gambar	Nama	Keterangan
	<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i>
	<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit
	<i>Depedency</i>	Hubungan dimana perubahan yang terjadi pada satu elemen mandiri () akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri
	<i>Generalization</i>	Hubungan dimana objek anak (descendent) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (ancestor)
	<i>Extend</i>	Menspesifikasikan bahwa use case target memperluas perilaku dari use case sumber pada suatu titik yang diberikan
	<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya
	<i>System</i>	Menspesifikasikan paket yang berisi tampilan sistem secara terbatas
	<i>Use Case</i>	Deskripsi dari urutan aksi- aksi yang ditampilkan sistem yang memberi suatu hasil terstruktur bagi aktor
	<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi)
	<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

Sumber: (Hendini, 2016)

b. Diagram Aktivitas (Activity Diagram)

Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *Activity diagram* mesti digunakan sejajar (horizontal) dengan teknik pemodelan lainnya, seperti diagram *Use Case*. *Activity diagram* berfungsi juga untuk menganalisis diagram use case dengan cara mendeskripsikan aktor, *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. *Activity diagram* dapat dibagi menjadi beberapa *object swimlane* untuk menggambarkan objek mana yang bertanggung jawab untuk aktivitas tertentu. *Activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis (Hendini, 2016). Simbol *Activity Diagram* dapat dilihat pada Tabel 2.3.

Tabel 2. 3 Diagram Aktivitas

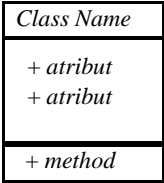





Gambar	Nama	Keterangan
	<i>Swimlane</i>	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.
	<i>Activity</i>	Memperlihatkan bagaimana masing masing kelas antarmuka saling berinteraksi satu sama lain
	<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi
	<i>Desicion</i>	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
	<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
	<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
	<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

Sumber: (Hendini, 2016)

c. Class Diagram

Class diagram adalah gambar grafis mengenai struktur objek statis dari suatu sistem, menunjukkan *class-class* objek yang menyusun sebuah sistem dan juga hubungan antara *class* objek tersebut. *Class Diagram* juga menunjukkan atribut-atribut dan operasi-operasi dari sebuah kelas dan constraint yang berhubungan dengan objek yang dikoneksikan (Hendini, 2016). Simbol-simbol *class diagram* dapat dilihat pada Tabel 2.4.

Tabel 2. 4 Class Diagram



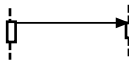

Simbol	Nama	Keterangan
	<i>Class</i>	Merupakan blok - blok pembangun pada pemrograman berorientasi objek. Terdiri dari nama <i>class</i> , atribut dan <i>method</i> pada <i>class</i> .
	<i>Association</i>	Merupakan sebuah relationship paling umum antara dua class. Association melambangkan tipe-tipe relationship dan juga dapat menampilkan hukum-hukum multiplisitas pada sebuah relationship.
	<i>Composition</i>	Jika sebuah class tidak bisa berdiri sendiri dan harus merupakan bagian dari class yang lain, maka class tersebut memiliki relasi Composition terhadap class tempat dia bergantung tersebut.
	<i>Generalization</i>	Relasi antar kelas dengan makna turunan spesialisasi (umum khusus).
	<i>Aggregation</i>	Mengindikasikan keseluruhan bagian relationship dan biasanya disebut sebagai relasi.
	<i>Depedency</i>	Umumnya dependency digunakan untuk menunjukkan operasi pada suatu class yang menggunakan class yang lain.

Sumber: (Hendini, 2016)

d. Sequence Diagram

Sequence diagram merupakan diagram yang menggambarkan kolaborasi dinamis antara sejumlah *object*. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara *object* juga interaksi antara *object*. Simbol-simbol pada *sequence diagram* dapat dilihat pada Tabel 2.5 *sequence diagram*.

Tabel 2. 5 *Seuquence Diagram*

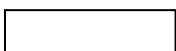

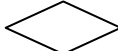

Simbol	Nama	Keterangan
	<i>Actor</i>	Menggambarkan seseorang yang berinteraksi dengan aplikasi.
	<i>Life Line</i>	Objek entity, antarmuka yang saling berinteraksi.
	<i>Object Message</i>	Menggambarkan pesan antar objek yang menunjukkan urutan kejadian yang terjadi.
	<i>Message to Self</i>	Menggambarkan pesan objek itu sendiri yang menunjukkan urutan kejadian.

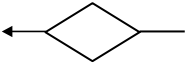
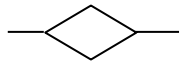
Sumber: (Hendini, 2016)

2.6 Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) merupakan suatu model yang bertujuan menjelaskan hubungan antar data dalam basis data. ERD terdiri dari hubungan antar objek-objek dasar yang dilukiskan dengan menggunakan komponen-komponen data entitas, atribut dan relasi antar entitas (Rohayani et al., 2020). Simbol-simbol *Entity Relationship Diagram* (ERD) dapat dilihat pada Tabel 2.6 ERD.

Tabel 2. 6 ERD

Simbol	Nama	Keterangan
	Entitas	Individu yang mewakili suatu objek dan dapat dibedakan dengan objek yang lain.
	Atribut	Properti yang dimiliki oleh suatu entitas, dimana dapat mendeskripsikan karakteristik dari entitas tersebut.
	Relasi	Menunjukkan hubungan sejumlah entitas yang berbeda.
	Relasi 1 : 1	Relasi yang menunjukkan bahwa setiap entitas pada himpunan entitas pertama berhubungan dengan paling banyak satu entitas pada himpunan entitas kedua

	Relasi 1 : N	Relasi yang menunjukkan bahwa hubungan antara entitas pertama dengan entitas kedua adalah satu banding banyak.
	Relasi N : N	Hubungan ini menunjukkan bahwa setiap entitas pada himpunan entitas yang pertama dapat berhubungan dengan banyak entitas pada himpunan entitas yang kedua.

Sumber: (Edi & Betshani, 2009)

2.7 Metode *Prototype*

Prototype perangkat lunak adalah salah satu metode siklus hidup sistem yang didasarkan pada konsep model bekerja (*working model*). Tujuannya adalah mengembangkan model menjadi sistem final. Artinya sistem akan dikembangkan lebih cepat dari pada metode tradisional dan biayanya menjadi lebih rendah. Ada banyak cara untuk melakukan prototyping, begitu pula dengan penggunaannya.

Ciri dari metode ini adalah pengembang dan pelanggan dapat melihat dan melakukan pengerjaan dengan bagian dari sistem komputer dari sejak awal proses pengembangan (Rizky et al., 2022).

1. Analisa kebutuhan

Di tahap ini pengembang melakukan identifikasi *software* dan semua kebutuhan sistem yang akan dibuat.

2. Membangun prototyping

Membangun prototyping dengan membuat perancangan sementara yang berfokus pada penyajian kepada pelanggan (misalnya dengan membuat *input* dan format *output*).

3. Evaluasi prototyping

Evaluasi ini dilakukan untuk mengetahui apakah prototyping sudah sesuai dengan harapan pelanggan.

4. Mengkodekan sistem

Pada tahap ini prototyping yang sudah disetujui akan diubah ke dalam bahasa pemrograman.

5. Menguji sistem

Di tahap ini dilakukan untuk menguji sistem perangkat lunak yang sudah dibuat.

6. Evaluasi Sistem

Perangkat lunak yang sudah siap jadi akan dievaluasi oleh pelanggan untuk mengetahui apakah sistem sesuai dengan yang diharapkan.

7. Menggunakan sistem

Perangkat lunak yang sudah diuji dan disetujui oleh pelanggan siap digunakan.

2.8 Penelitian Terkait

Penelitian dibuat oleh Dwi Madya Sidik Fadhil (Madya & Fadhil, 2021). Telah mengumpulkan hasil data, analisa, perancangan, implementasi serta pengujian “Aplikasi Konsultasi Desain Interior dan Eksterior untuk Menghubungkan Stakeholder dengan Tenaga Arsitek Profesional Menggunakan Framework Laravel” maka dapat ditarik kesimpulan yaitu Aplikasi ini mempermudah dalam proses memperluas area promosi arsitek profesional ke seluruh Indonesia, khususnya daerah indramayu dan Dengan adanya Aplikasi Konsultasi Desain Interior dan Eksterior untuk Menghubungkan Stakeholder dengan Tenaga Arsitek Profesional Menggunakan Framework Laravel mempermudah masyarakat mencari jasa arsitek profesional.

Penelitian dibuat oleh Alvin Fernanda (Fernanda, 2021). Memiliki analisis kebutuhan, perancangan, pengimplementasian serta pengujian Aplikasi Konsultasi Desain Interior dan Eksterior untuk Menghubungkan Stakeholder dengan Tenaga Arsitek Profesional Menggunakan Framework Flutter Android maka dapat ditarik kesimpulan tujuan pembuatan sistem ini yaitu untuk menampilkan gambaran inspirasi desain interior dan eksterior dan menyediakan fitur untuk membantu masyarakat dalam menemukan tenaga arsitek profesional yang tepat.

BAB III METODE PELAKSANAAN

Pada bab metode pelaksanaan ini berisikan langkah-langkah yang digunakan dalam Pengimplementasian Aplikasi Arsitek.co Berbasis Mobile dan Website Untuk Menghubungkan Masyarakat Dengan Tenaga Profesional Di Bidang Arsitektur pada proyek 3. Dengan adanya metodologi ini proses pembuatan aplikasi dapat dipahami oleh pembaca.

3.1 Teknik Pengumpulan Data

Pengumpulan data dilakukan dengan menggunakan metode pengumpulan data penelitian kualitatif. Adapun tahapan-tahapan pengumpulan data sebagai berikut.

a. Observasi

Pada penelitian ini, observasi dilakukan di PT. Lima Benua Consultant untuk mengamati proses perencanaan desain proyek bangunan, mulai dari proses pengajuan desain, perancangan rencana anggaran biaya, sampai dengan proses pencarian tenaga kontraktor dan melakukan konstruksi proyek bangunan.

b. Studi Pustaka

Studi pustaka dilakukan untuk mengumpulkan data dan informasi dalam bentuk digital maupun sumber buku yang dibaca.

3.2 Metode Pengembangan Perangkat Lunak

Model pengembangan perangkat lunak yang digunakan dalam penelitian ini adalah *agile software development methods*, dengan menggunakan Model *Scrum* (Mahendra & Eby Yanto, 2018). Adapun tahapan-tahapan *scrum* yang dilakukan, sebagai berikut.

1. *Product Backlog*

Bagian pertama ini adalah kumpulan dari hal-hal yang diperlukan dan yang harus tersedia dalam produk. *Produk backlog* berada dalam tanggung jawab *product owner*. *Product Backlog* adalah daftar utama dari semua fungsi yang diinginkan dalam produk. Metode Scrum tidak mewajibkan dokumentasi semua persyaratan pada awal proyek.

2. *Sprint Backlog*

Perencanaan *Sprint* dilakukan dalam pertemuan atau *meeting* antara *product owner* dan tim developer yang akan bekerja sama untuk memilih *product backlog* untuk

dimasukan ke dalam proses sprint. Hasil dari pertemuan tersebut yaitu *Sprint Backlog*.

3. *Sprint*

Sprint merupakan suatu kerangka waktu yang berdurasi paling lama 1 bulan untuk mengembangkan produk yang akan dirilis. Di dalam *Sprint* terdapat 2 bagian pekerjaan:

a. Pertemuan Harian (*Daily Standup Meeting*)

Merupakan pertemuan dimana setiap 24 jam (1 hari), tim pengembangan bertemu untuk membahas proses pengembangan produk.

b. Refleksi Sprint

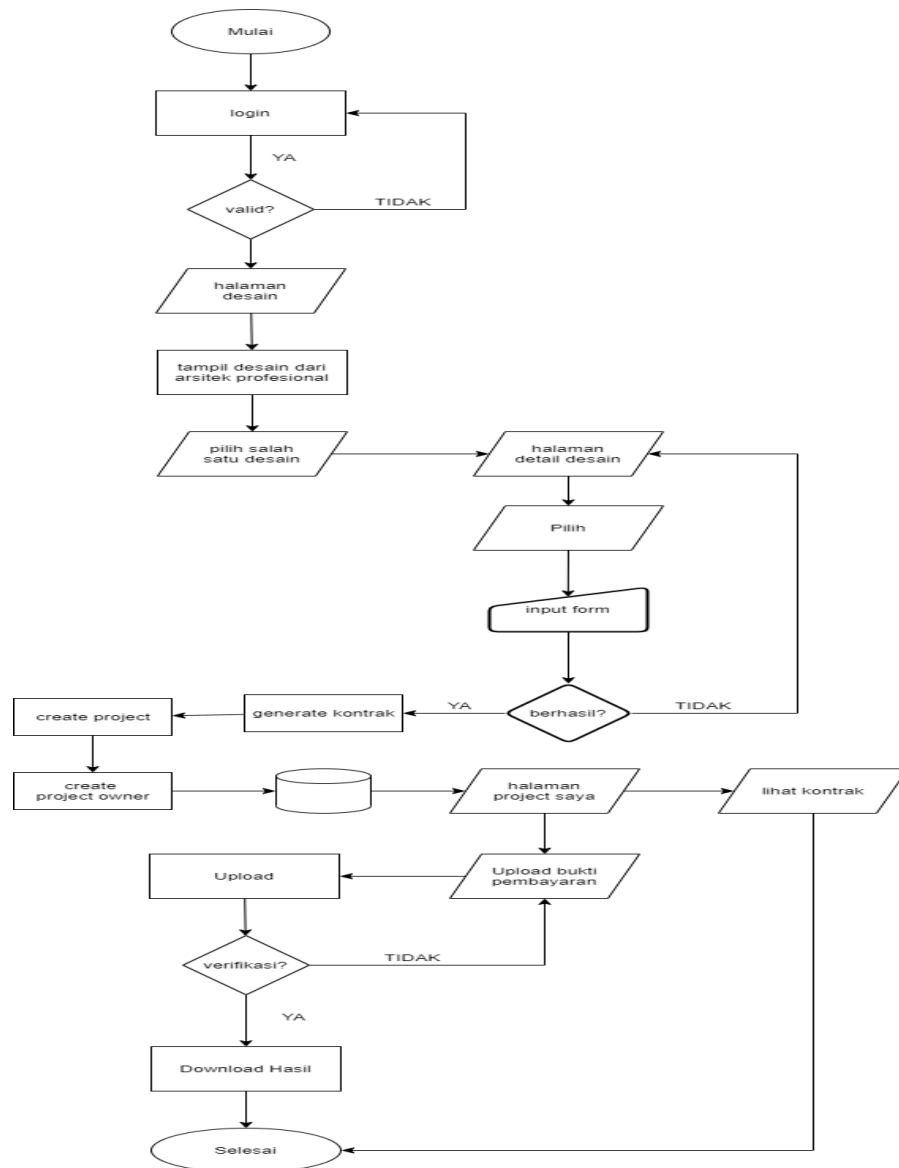
Merupakan pertemuan yang dilakukan setiap bulannya, yang bertujuan untuk membahas hal dari *Sprint Backlog* yang telah berjalan dan telah berhasil dikerjakan, serta dapat memperbaiki dan meningkatkan kualitas produk pada *Sprint* yang berikutnya.

4. *Working increment of the software*

Increment adalah hasil dari *Product Backlog* yang sudah selesai dikerjakan pada *Sprint*. Di akhir *Sprint*, *Increment* harus sudah benar-benar selesai, yang berarti harus dalam keadaan yang *useable*.

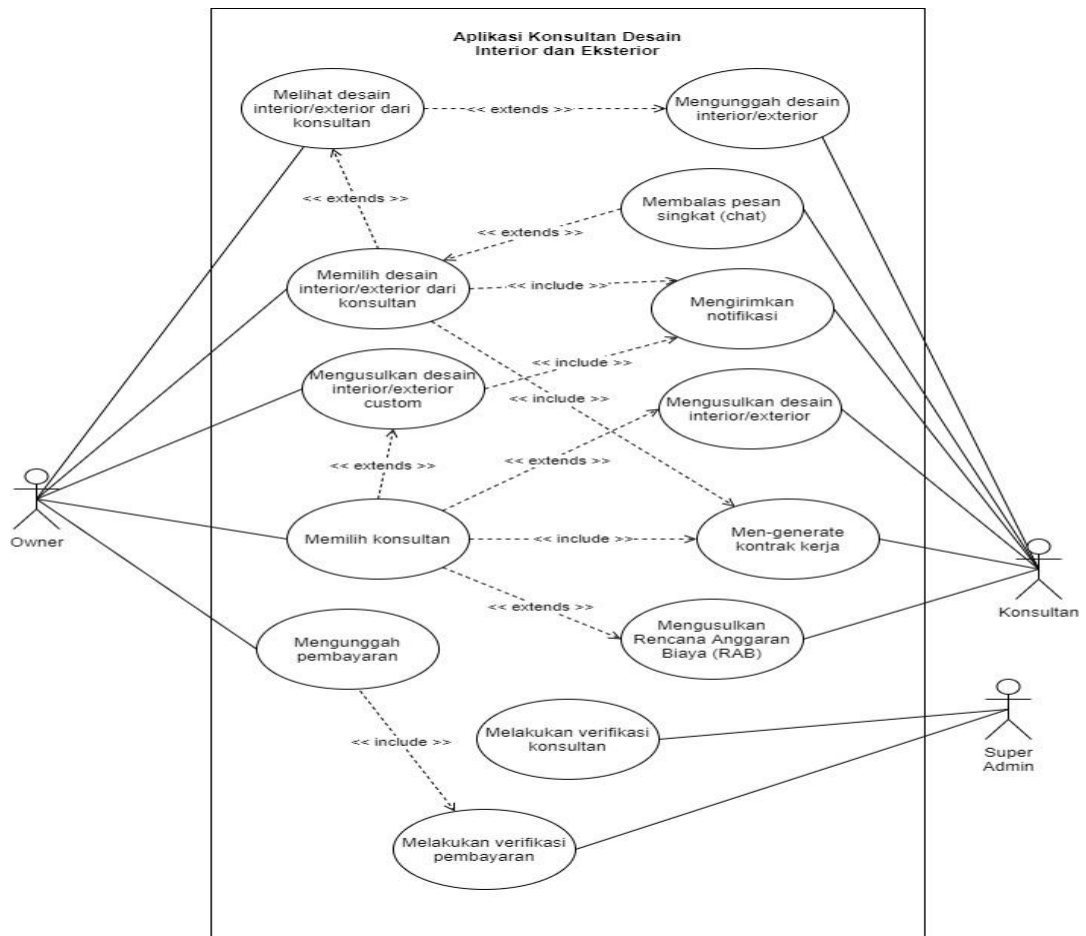
3.3 Flowchart

Flowchart adalah sebuah jenis diagram yang mewakili algoritme, alir kerja atau proses, yang menampilkan langkah-langkah dalam bentuk simbol-simbol grafis, dan urutannya dihubungkan dengan panah.



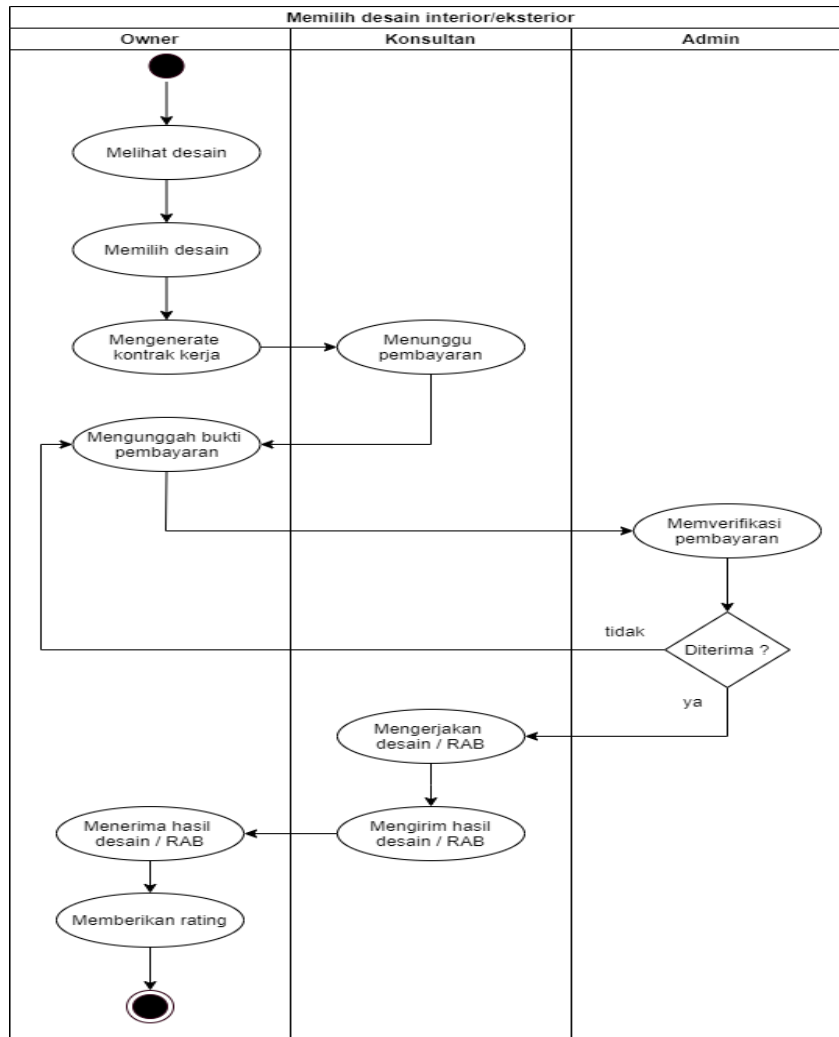
3.4 Use Case

Use Case adalah untuk hanya menggambarkan fungsionalitas yaitu apa yang sistem lakukan dan bukan bagaimana hal itu dilakukan



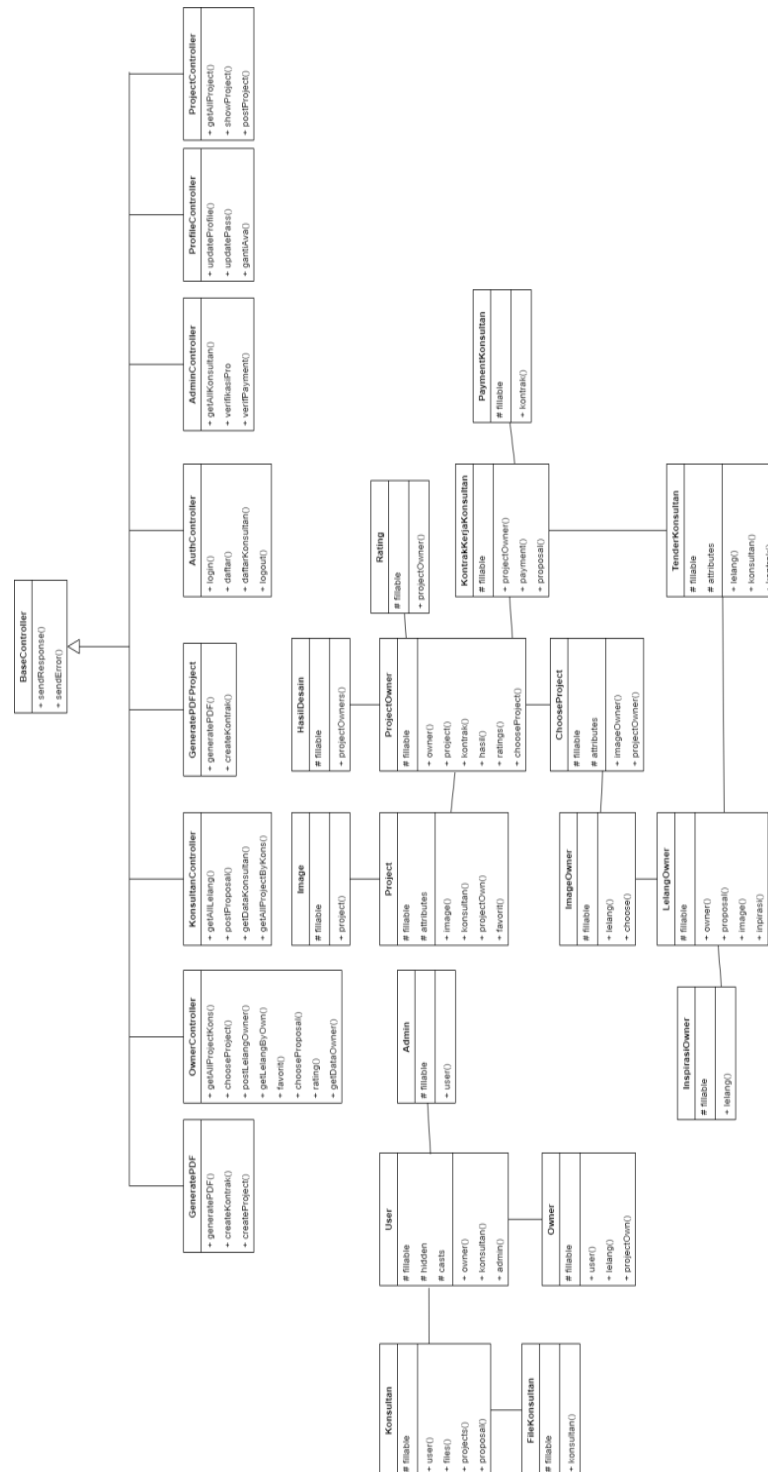
3.5 Activity Diagram

Activity diagram digunakan untuk memodelkan workflow yang menggambarkan kondisi, kendala, aktivitas berurutan dan bersamaan.



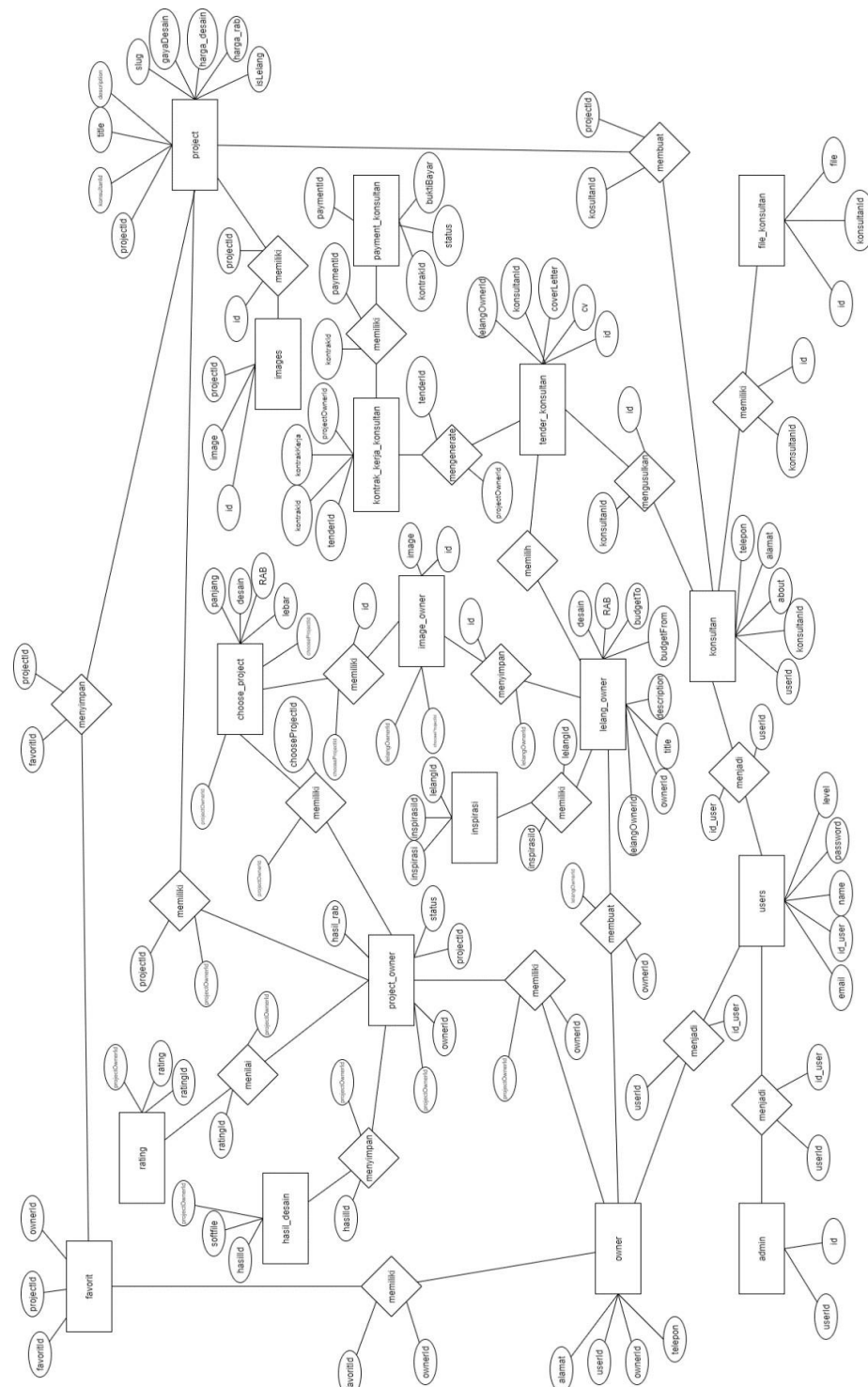
3.6 Class Diagram

Diagram kelas adalah diagram UML yang menggambarkan kelas-kelas dalam sebuah sistem dan hubungannya antara satu dengan yang lain, serta dimasukkan pula atribut dan operasi.



3.7 ERD

Entity Relationship Diagram (ERD) merupakan model data berupa notasi grafis dalam pemodelan data konseptual yang menggambarkan hubungan antara penyimpan.



BAB IV HASIL DAN PEMBAHASAN

4.1 Hasil

Penelitian ini menghasilkan sebuah Implementasi Aplikasi Arsitekco Berbasis *Mobile* Dan *Website* Untuk Menghubungkan Masyarakat Dengan Tenaga Profesional Di Bidang Arsitektur. Aplikasi ini digunakan untuk membantu masyarakat dalam memperoleh gambaran inspirasi desain interior dan eksterior secara terpusat. Aplikasi ini juga membantu masyarakat dalam mencari tenaga arsitek konsultan yang tepat. Selain itu, aplikasi ini membantu menghubungkan masyarakat dengan arsitek konsultan dalam melakukan diskusi atau konsultasi desain.

Selain itu, penelitian ini dapat mengarahkan masyarakat agar bijak dan paham dalam tahapan sebuah proyek konstruksi. Pada aplikasi yang dirancang terdapat pemilihan tim konstruksi proyek bangunan sesuai keinginan dari pengguna aplikasi sehingga kenyamanan saat melihat profil tim konstruksi dan kontraktor menjadi acuan utama dalam perancangan fitur aplikasi. Masyarakat dapat membuat kontrak sesuai dengan kontraktor yang dipilihnya dengan melalui tahapan konstruksi proyek yang sempurna.

4.2 Pembahasan

4.2.1 Hasil *Sprint Planning*

Pada bagian ini dijelaskan mengenai *sprint planning* yang dilakukan selama pengerjaan produk aplikasi. *Sprint planning* dilakukan setiap minggu. Adapun rincian hasil pembahasan dirangkum pada tabel 4.2.1 hasil sprint planning.

Tabel 4.2. 1 *Sprint Planning*

NO	KEGIATAN	TIMELINE
1.	Pembentukan Kelompok	Minggu ke-2 Bulan Februari
2.	Diskusi Topik	Minggu ke-3 Bulan Februari
3.	Penentuan Topik	Minggu ke-3 Bulan Februari
4.	Perancangan Flowchart, Mockup, Dokumen SRS, SPMP	Minggu ke-4 Bulan Februari
5.	Perancangan DFD, UML Diagram Class, <i>Usecase</i> , <i>Database</i> ERD dan membahas alur kerja <i>Role</i> Kontraktor serta memasuki tahap koding dari <i>Source Code</i> dan memahami alur koding	Minggu ke-1 Bulan Maret

6.	Perancangan Diagram Konteks, UML <i>Activity</i> , Sequence, Diagram Statechart, perancangan <i>Migration Database</i> dan membuat Proposal	Minggu ke-2 Bulan Maret
7.	Tahap koding Web Halaman <i>Login</i> , Register, Home setiap role dan menampilkan data desain dan pembuatan API-nya	Minggu ke-3 Bulan Maret
8.	Tahap koding Mobile Halaman Login, Register, Home setiap role dan menampilkan data desain	Minggu ke-4 Bulan Maret
9.	Menyesuaikan fitur chat dengan <i>Firebase</i> , menyelesaikan Proposal dan Presentasi UTS	Minggu ke-1 Bulan April
10.	Membuat <i>Model</i> , <i>View</i> , <i>Controller</i> (MVC) dan <i>Routes</i> dari Role Owner, Konsultan dan Admin secara orientasi objek	Minggu ke-2 Bulan April
11.	Melakukan percobaan <i>Database</i> dengan sistem <i>relational Migration</i> dan data <i>Seeder</i> Laravel serta percobaan API lanjut	Minggu ke-3 Bulan April
12.	Tahap Koding bagian Lelang desain dan cara Konsultan mengirimkan data Proposalnya sesuai permintaan serta pembuatan Dokumen SDD	Minggu ke-4 Bulan April
13.	Perancangan Backend menggunakan <i>library</i> NodeJs untuk menyimpan data <i>table</i> Kontraktor, Konsultan, dan Admin serta percobaan API untuk Mobile Register Kontraktor dan membuat Laporan Proyek	Minggu ke-4 Bulan April
14.	Penerapan <i>Payment Gateway</i> Midtrans sebagai <i>tools</i> transaksi dan penerapan konsep Kontraktor Cabang sebagai Tim dari seorang Kontraktor	Minggu ke-1 Bulan Mei
15.	Menambahkan MVC, <i>Routes</i> , dan <i>Migration</i> baru serta pemanggilan objek gambar sesuai <i>database</i>	Minggu ke-2 Bulan Mei
16.	Membuat verifikasi <i>payment gateway</i> oleh Admin dan penambahan UI <i>Guest Mobile</i>	Minggu ke-3 Bulan Mei
17.	Merancang Seeder baru secara relational, membuat UI data Kontraktor ke Owner dan pembuatan dokumen STD	Minggu ke-4 Bulan Mei
18.	Membuat transaksi Kontraktor dan alur verifikasi sampai membuat kontrak	Minggu ke-1 Bulan Juni
19.	Merancang MVC <i>Profile</i> Owner, Unggah Desain Konsultan <i>Mobile</i> dan API data Kontraktor	Minggu ke-2 Bulan Juni
20.	Menyelesaikan Laporan Proyek, <i>Slide</i> ppt, membuat Poster dan <i>Banner</i> serta Presentasi UAS	Minggu ke-3 Bulan Juni
21.	Kegiatan EXPO 2022	Minggu ke-4 Bulan Juni

4.2.2 Hasil Daily Scrum

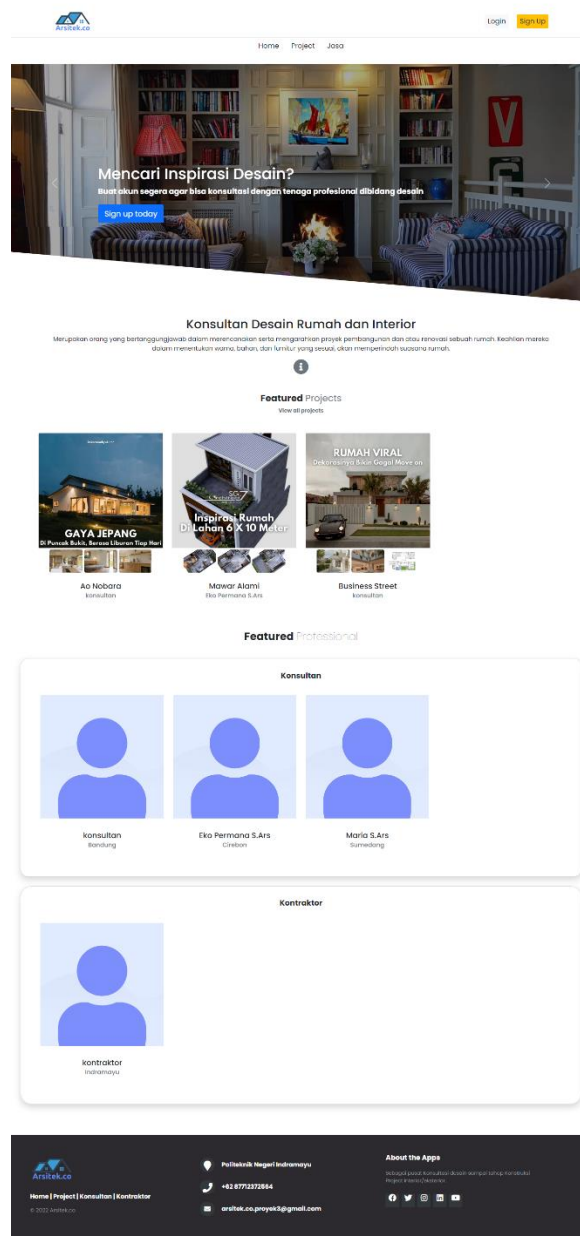
Pada bagian ini dijelaskan mengenai hasil daily scrum yang dilakukan selama pengerjaan produk aplikasi. Hasil *daily scrum* dilakukan setiap seminggu 3 kali. Adapun berikut rincian hasil pembahasannya.

4.2.2.1 Implementasi Desain Antarmuka (*Interfaces*) Website

Antarmuka merupakan cara untuk mempermudah pengguna dalam melakukan aktivitasnya. Antarmuka yang kami buat ini dibuat menggunakan text editor Visual Studio Code menggunakan Framework Laravel.

A. Halaman Home

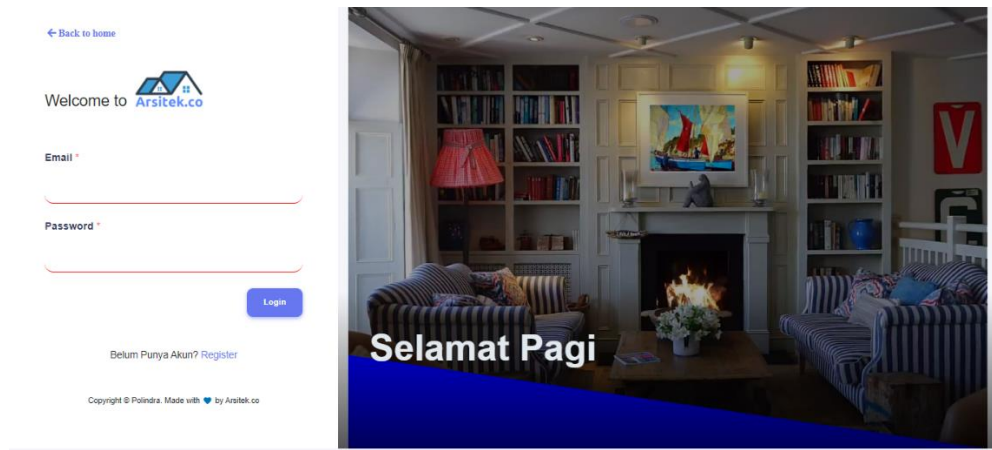
Halaman ini menyajikan konten awal yang interaktif sebelum login.



Gambar 4. 1 Halaman Home

B. Halaman *Login*

Login halaman *login* merupakan halaman awal ketika program dijalankan. Dalam halaman ini user harus menginputkan username dan password untuk masuk kedalam sistem.



Gambar 4. 2 Halaman *Login*

C. Halaman *Register Owner*

Halaman *register Owner* adalah halaman untuk melakukan registrasi sebagai pengguna baru dengan role Owner. Diperlukan pengisian form untuk dijadikan data diri akun owner

Gambar 4. 3 Halaman *Register Owner*

D. Halaman Register Konsultan

Halaman register Konsultan adalah halaman untuk melakukan registrasi sebagai pengguna baru dengan role Konsultan.

The screenshot shows the 'Daftar sebagai Konsultan' (Register as Consultant) form on the Arsitek.co website. The form includes fields for 'Nama Lengkap*' (Full Name), 'Username*', 'Email*', 'Password*', and a 'File' upload section with 'Choose Files' and 'No file chosen' options. Below the fields is a checkbox for 'Saya setuju dengan syarat dan ketentuan' (I agree with the terms and conditions) and a yellow 'Daftar' (Register) button. A link for 'Sudah punya akun? Login' (Already have an account? Login) is provided. The footer contains the Arsitek.co logo, navigation links (Home | Project | Konsultan | Kontraktor), copyright information (© 2022 Arsitek.co), contact details for Politeknik Negeri Indramayu (+62 87712372564, arsitek.co.proyek3@gmail.com), and an 'About the Apps' section.

Gambar 4. 4 Halaman Register Konsultan

E. Halaman Register Kontraktor

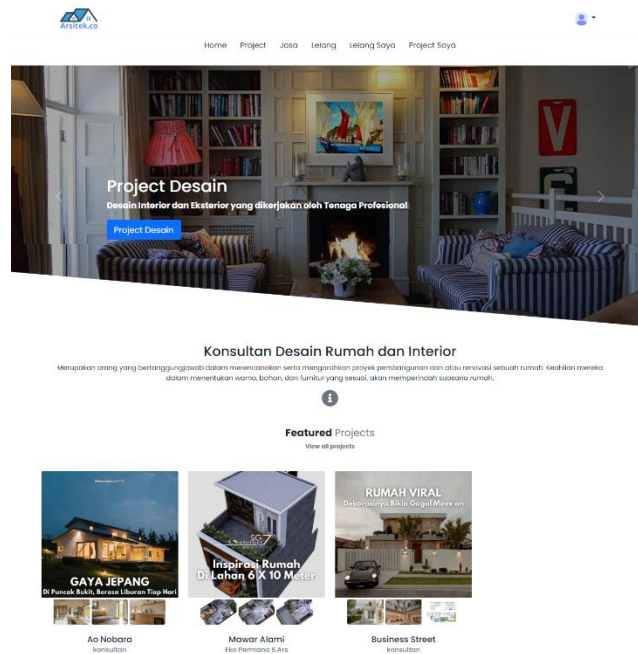
Halaman register Kontraktor adalah halaman untuk melakukan registrasi sebagai pengguna baru dengan role Kontraktor.

The screenshot shows the 'Daftar sebagai Kontraktor' (Register as Contractor) form on the Arsitek.co website. The form includes fields for 'Nama Lengkap*' (Full Name), 'Username*', 'Email*', 'Password*', and a 'File' upload section with 'Choose File' and 'No file chosen' options. Below the fields is a checkbox for 'Saya setuju dengan syarat dan ketentuan' (I agree with the terms and conditions) and a yellow 'Daftar' (Register) button. A link for 'Sudah punya akun? Login' (Already have an account? Login) is provided. The footer contains the Arsitek.co logo, navigation links (Home | Project | Konsultan | Kontraktor), copyright information (© 2022 Arsitek.co), contact details for Politeknik Negeri Indramayu (+62 87712372564, arsitek.co.proyek3@gmail.com), and an 'About the Apps' section.

Gambar 4. 5 Halaman Register Kontraktor

F. Halaman Home Owner

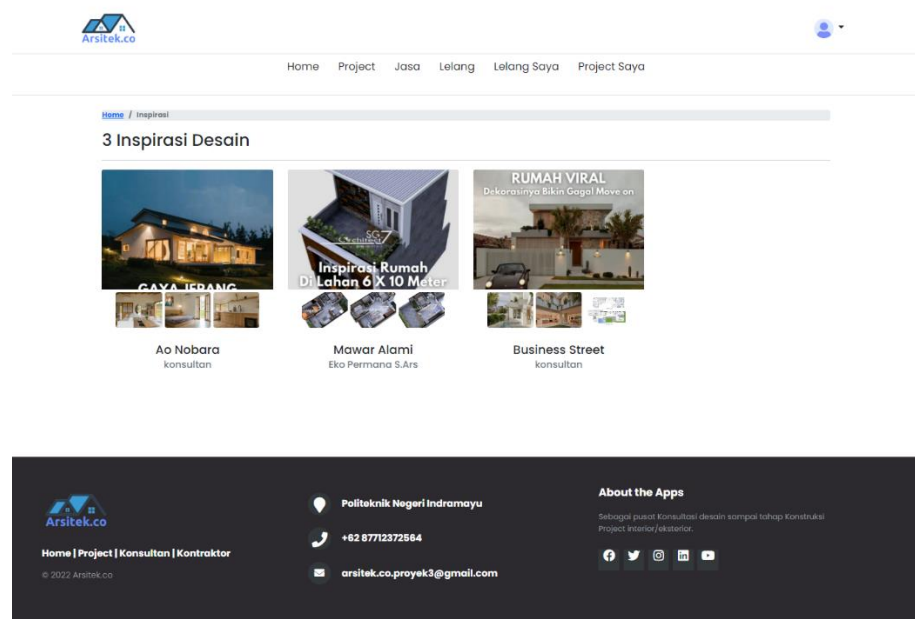
Halaman ini menyajikan konten awal untuk memberikan kesan pertama yang menyenangkan bagi pengguna aplikasi. Halaman ini sama seperti halaman home biasa, namun terdapat konten dan navigasi bar yang berbeda.



Gambar 4. 6 Halaman Owner

G. Halaman Inspirasi Desain

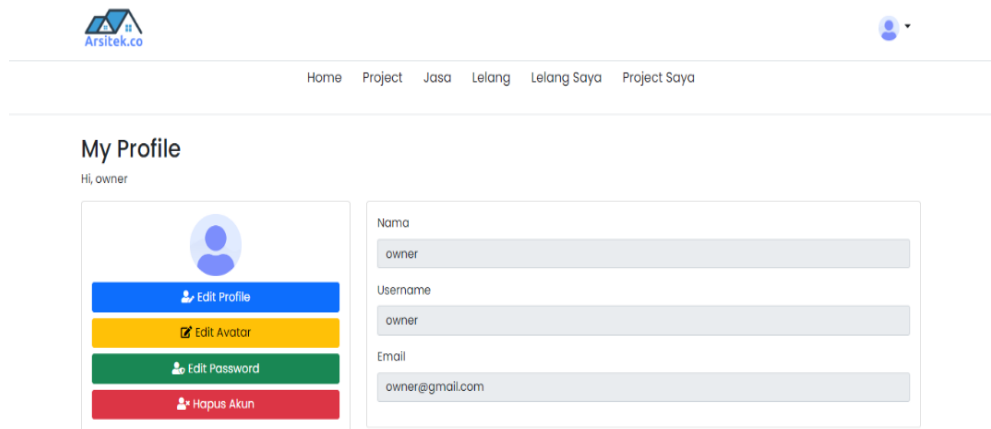
Halaman inspirasi desain adalah halaman untuk melihat desain yang dimiliki/diunggah oleh tenaga profesional bidang arsitektur.



Gambar 4. 7Halaman Inspirasi Desain

H. Halaman Profile

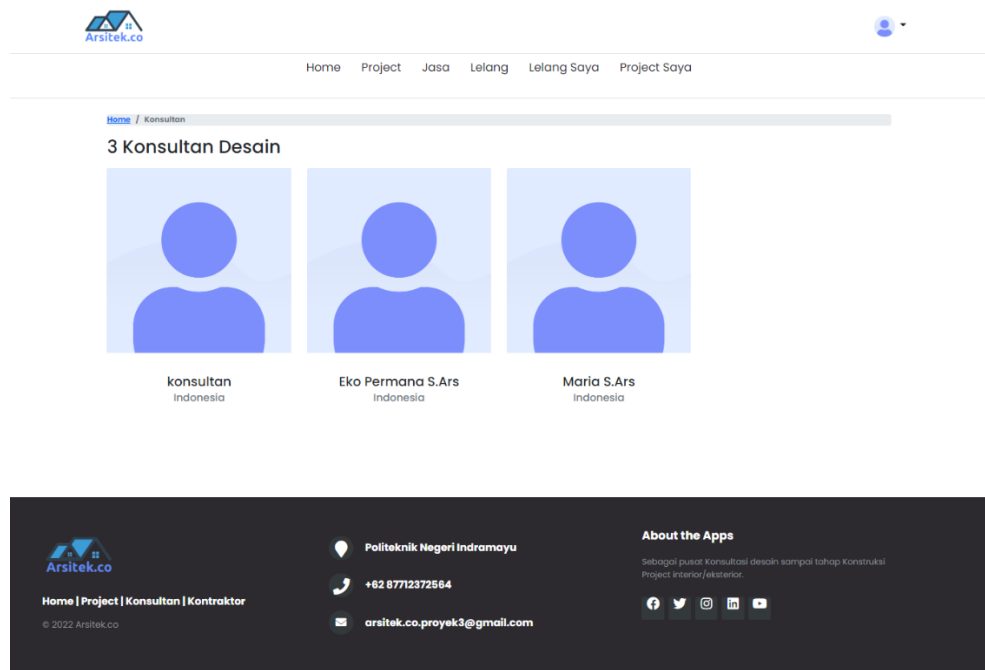
Halaman profile adalah halaman untuk melakukan melihat informasi data tentang diri anda.



Gambar 4. 8 Halaman Profile

I. Halaman List Konsultan

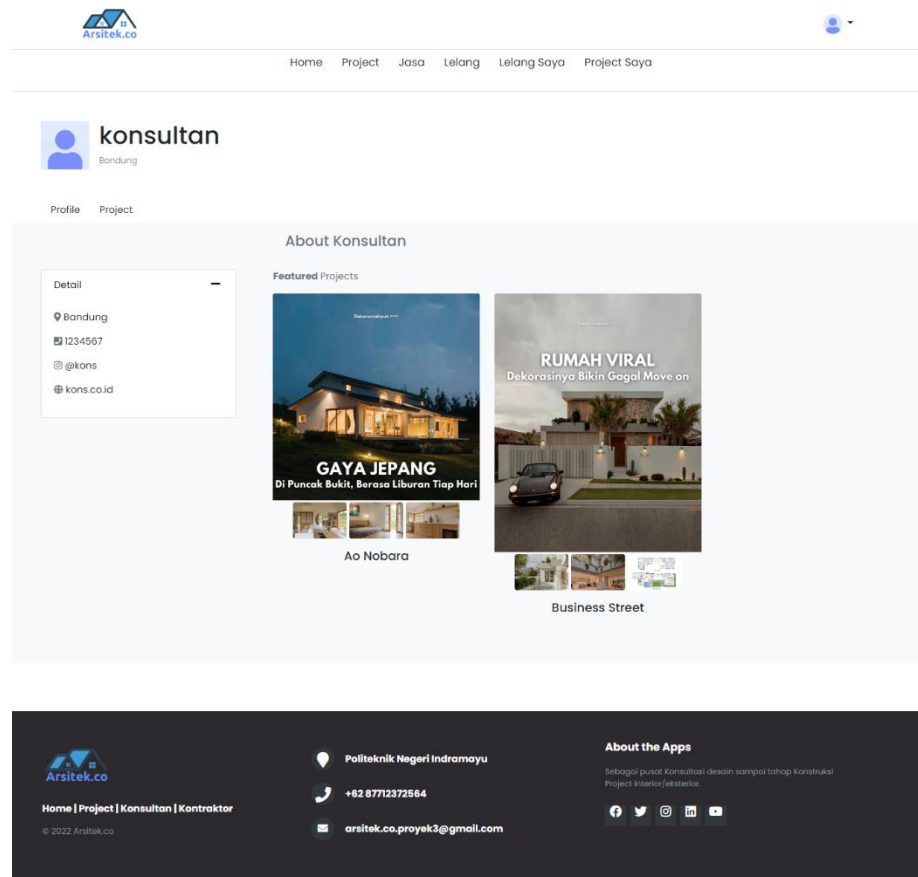
Halaman List Konsultan menyajikan list dari semua role Konsultan yang aktif.



Gambar 4. 9 Halaman List Konsultan

J. Halaman Detail Konsultan

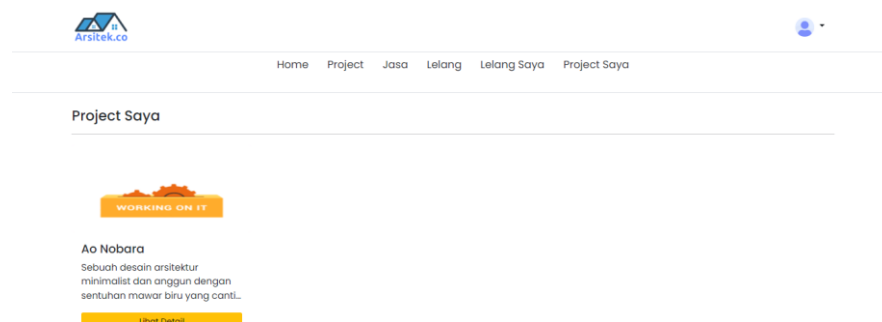
Halaman ini berisi informasi dari Konsultan secara detail



Gambar 4. 10 Halaman Detail Konsultan

K. Halaman Project Saya Bagian Desain (Owner)

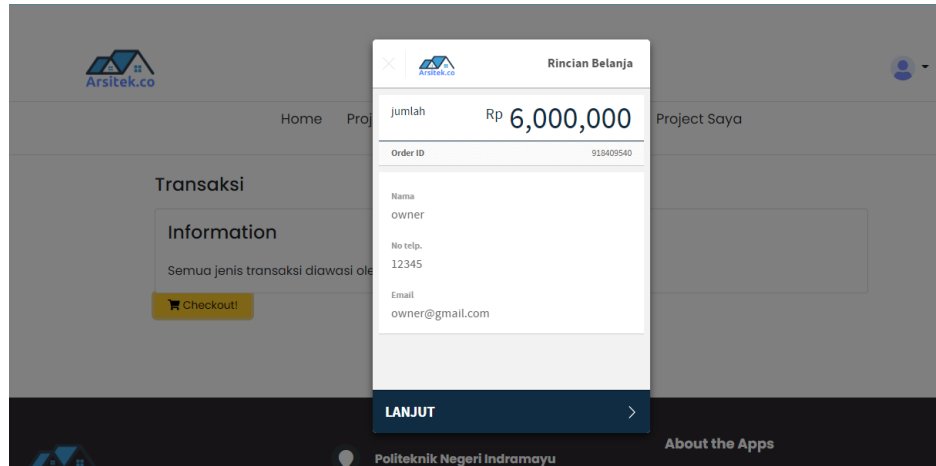
Halaman project saya Owner adalah halaman untuk melihat project/kontrak kerja sama yang sedang berjalan.



Gambar 4. 11 Halaman Project Saya (Owner)

L. Halaman Transaksi Payment Gateway

Halaman transaksi payment gateway adalah halaman payment atau pembayaran project desain maupun konstruksi dengan memakai tools Midtrans.



Gambar 4. 12 Halaman Transaksi Payment Gateway

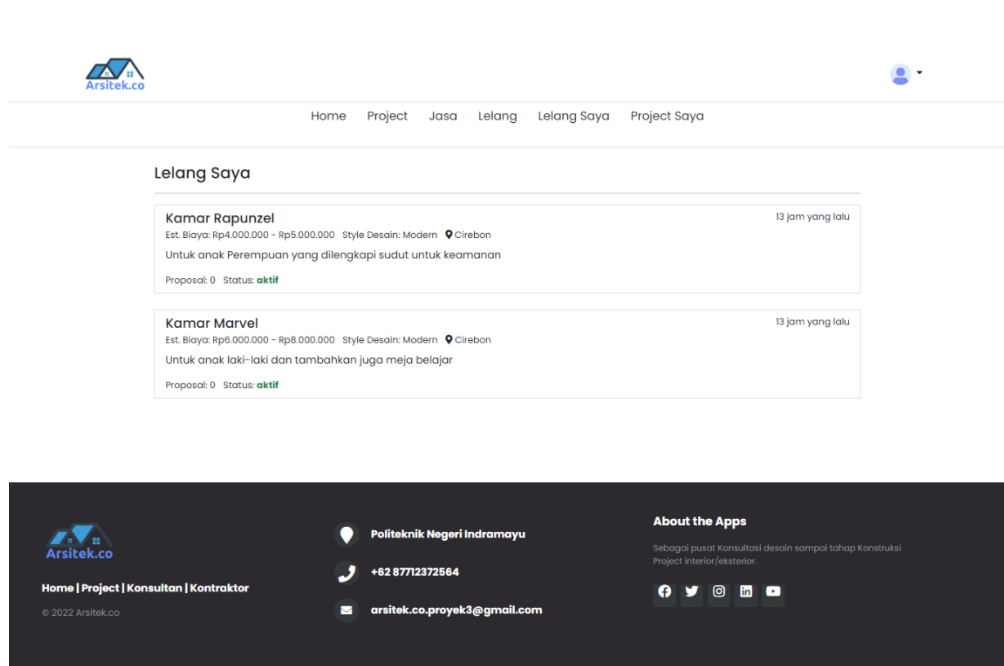
M. Halaman Lelang (Owner)

Halaman ini berisi form yang harus di isi untuk memulai pelelangan desain

Gambar 4. 13 Halaman Lelang (Owner)

N. Halaman Lelang Saya (Owner)

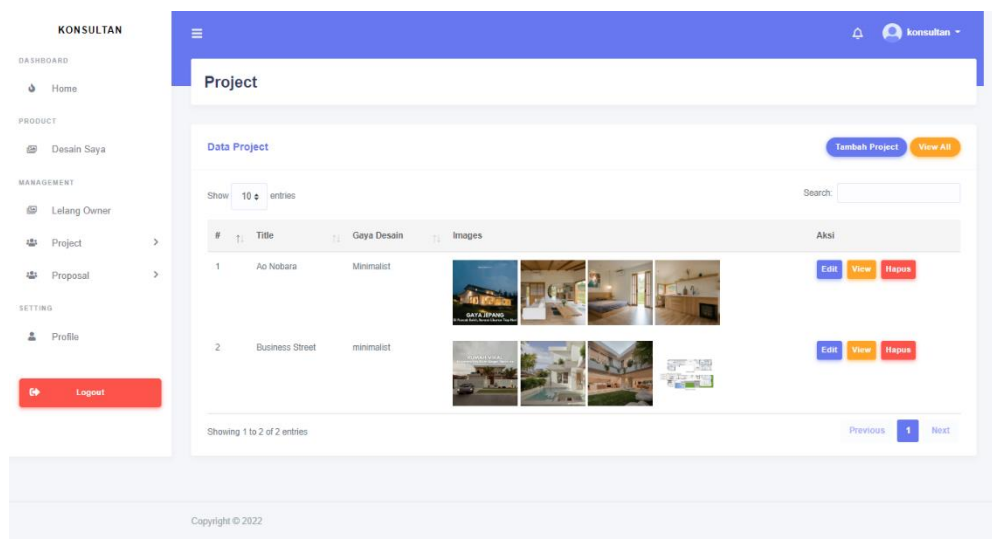
Halaman lelang saya Owner adalah halaman untuk Owner melakukan kustomisasi desain yang diinginkan.



Gambar 4. 14 Halaman Lelang Saya (Owner)

O. Halaman Project Konsultan

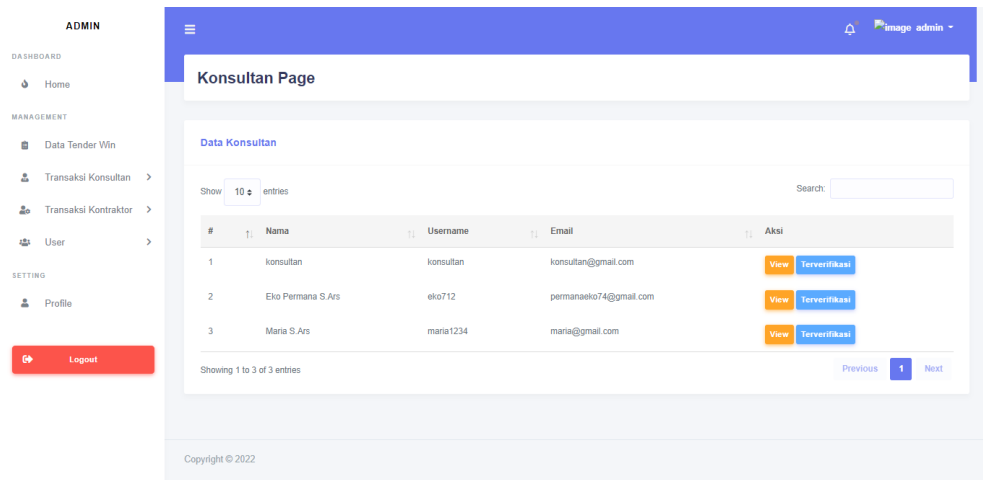
Halaman project Konsultan adalah halaman untuk Konsultan mengunggah desain yang ia miliki.



Gambar 4. 15 Halaman Project Konsultan

P. Halaman Verifikasi Konsultan

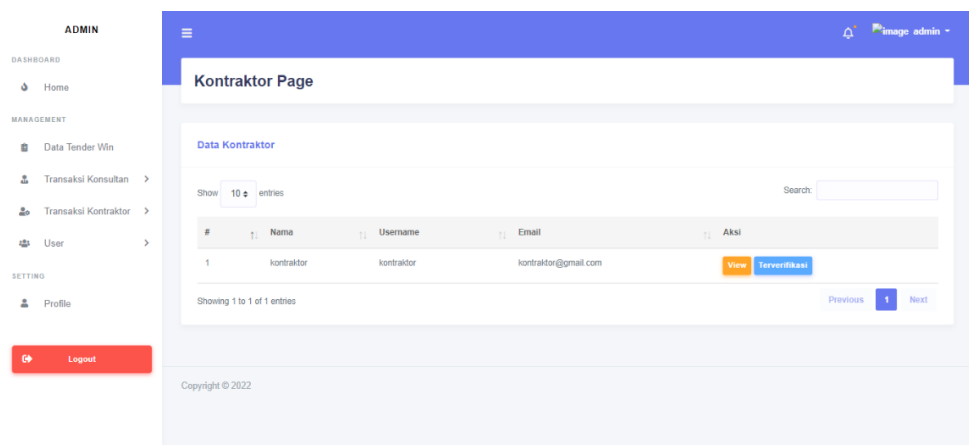
Halaman verifikasi Konsultan adalah halaman untuk Admin melakukan verifikasi registrasi akun konsultan.



Gambar 4. 16 Halaman Verifikasi Konsultan

Q. Halaman Verifikasi Kontraktor

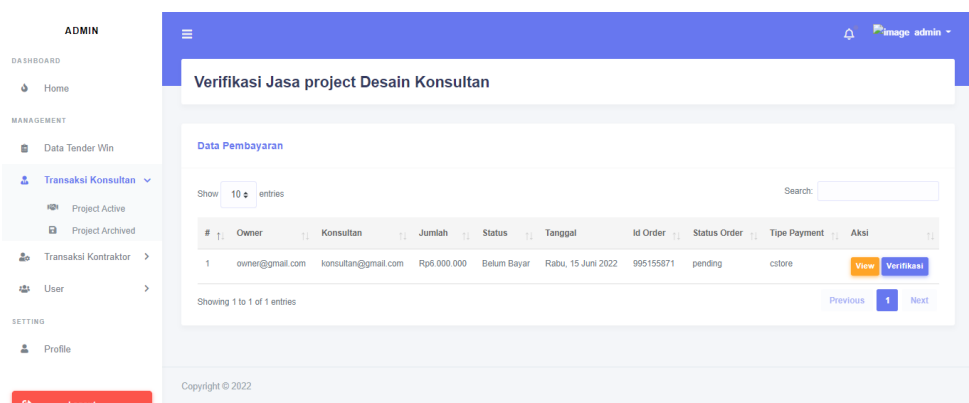
Halaman verifikasi Konsultan adalah halaman untuk Admin melakukan verifikasi registrasi akun konsultan.



Gambar 4. 17 Halaman Verifikasi Kontraktor

R. Halaman Verifikasi Pembayaran Project Desain Konsultan

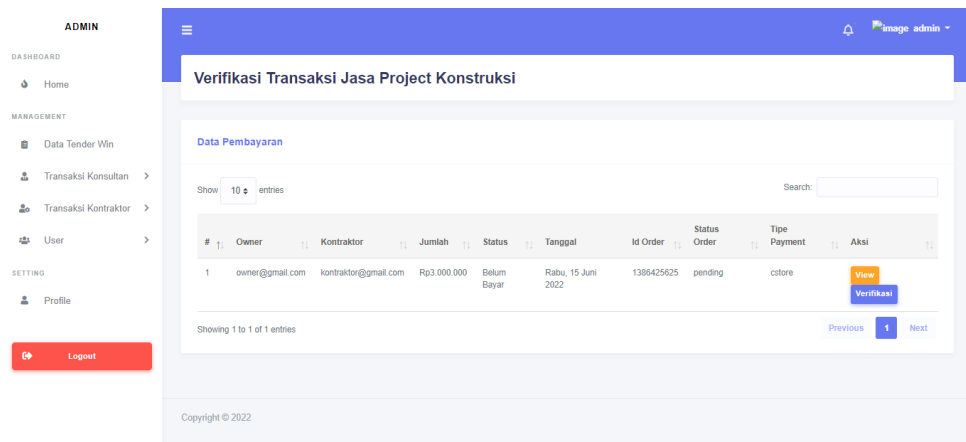
Halaman verifikasi pembayaran project desain konsultan adalah halaman untuk Admin melakukan verifikasi pembayaran project desain Owner.



Gambar 4. 18 Verifikasi Pembayaran Project Desain Konsultan

S. Halaman Verifikasi Pembayaran Project Konstruksi Kontraktor

Halaman verifikasi pembayaran project konstruksi adalah halaman untuk Admin melakukan verifikasi pembayaran project konstruksi Owner.



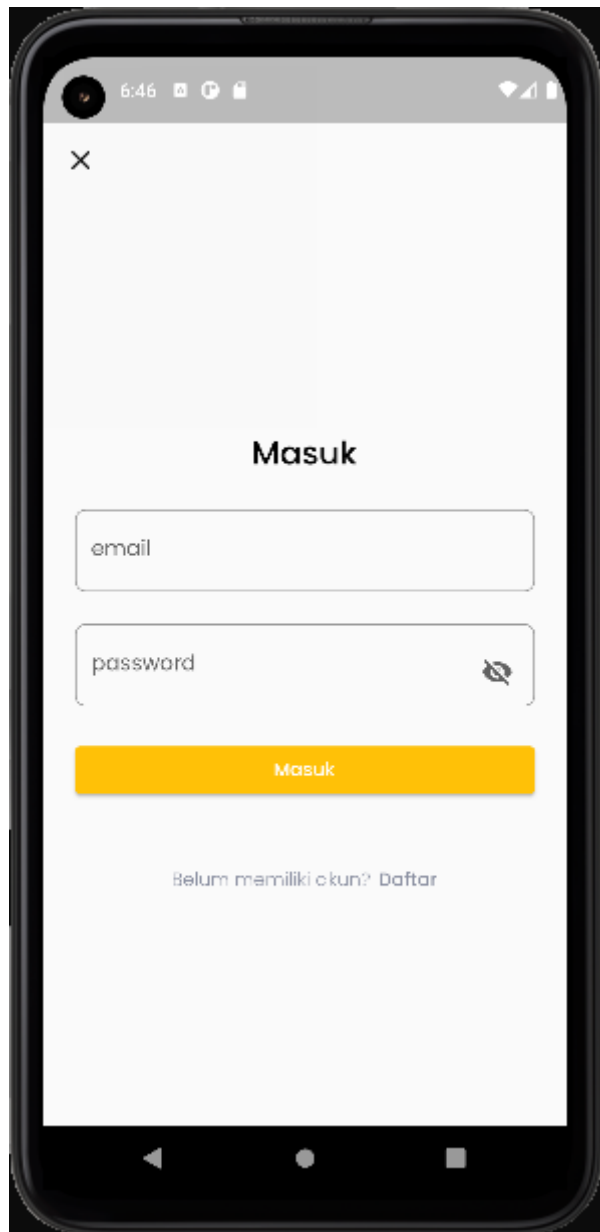
Gambar 4. 19 Verikasi Pembayaran Project Kontruksi Kontraktor

4.2.2.2 Implementasi Desain Antarmuka (Interfaces) Mobile

Antarmuka merupakan cara untuk mempermudah pengguna dalam melakukan aktivitasnya. Antarmuka yang kami buat ini dibuat menggunakan text editor Visual Studio Code menggunakan Framework Flutter.

A. Halaman Login

Login halaman login merupakan halaman awal ketika program dijalankan. Dalam halaman ini user harus menginputkan username dan password untuk masuk kedalam sistem.



Gambar 4. 20 Login Mobile

B. Halaman Choose Register

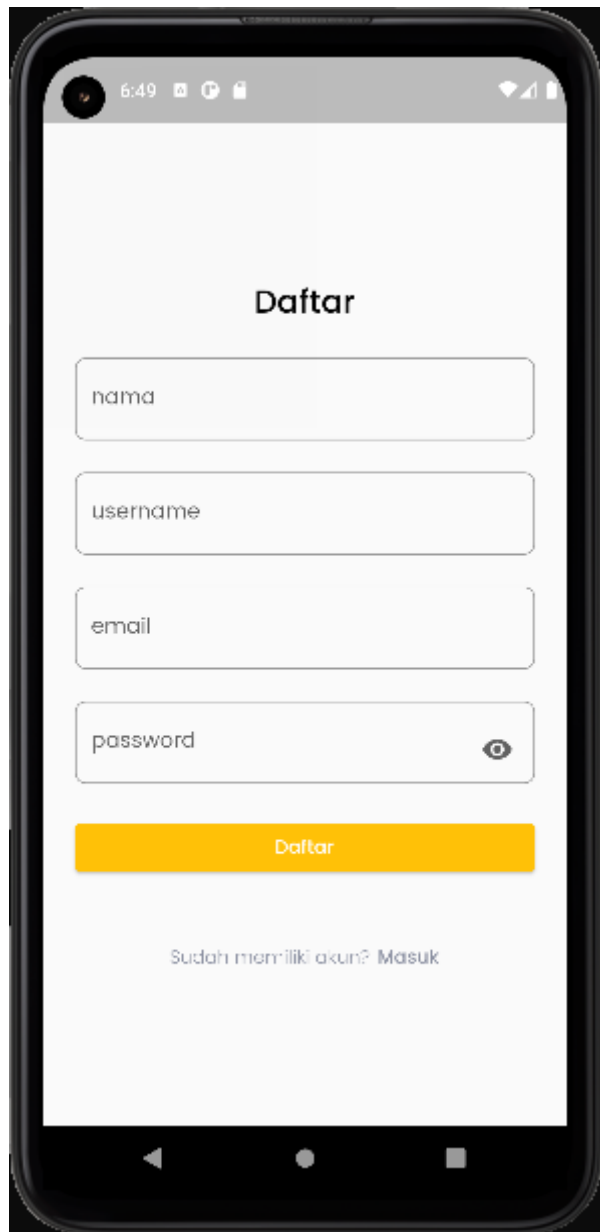
Halaman register Owner adalah halaman untuk melakukan registrasi sebagai pengguna baru dengan role Owner.



Gambar 4. 21 Choose Register Mobile

C. Halaman Register Owner

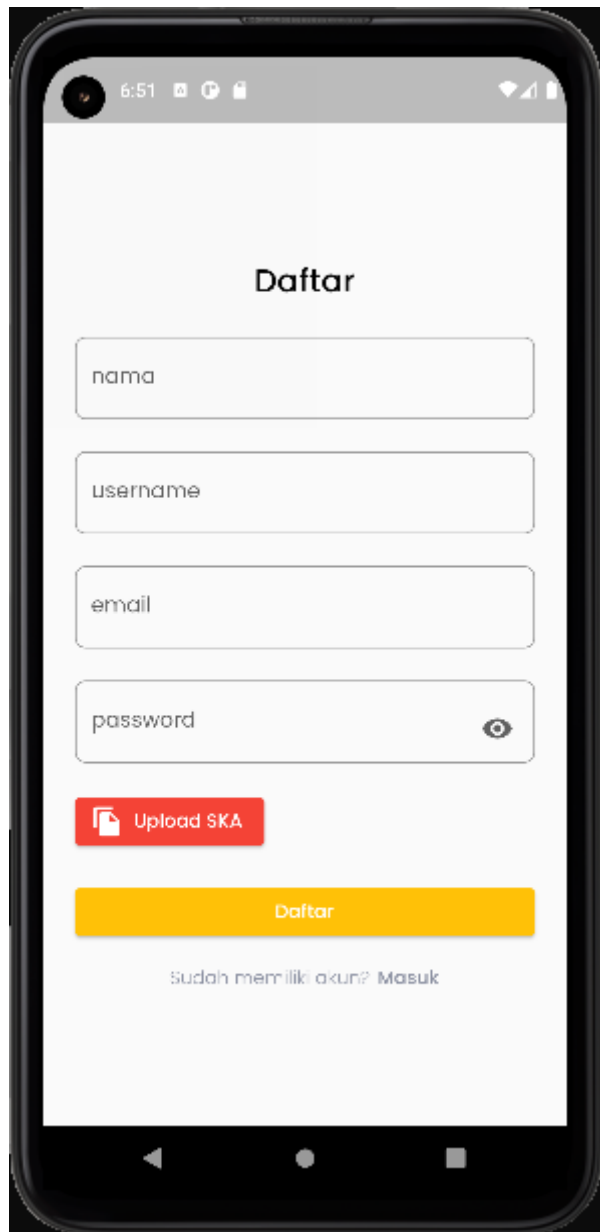
Halaman register Owner adalah halaman untuk melakukan registrasi sebagai pengguna baru dengan role Owner.



Gambar 4. 22 Register Owner Mobile

D. Halaman Register Professional

Halaman register Professional adalah halaman untuk melakukan registrasi sebagai pengguna baru dengan role Kontraktor atau Konsultan ketika memilih role di halaman choose register.



Gambar 4. 23 Register Profesional Mobile

E. Halaman Inspirasi Desain

Halaman inspirasi desain adalah halaman untuk melihat desain yang dimiliki/diunggah oleh tenaga profesional arsitek.



Gambar 4. 24 Inspirasi Desain Mobile

F. Halaman Detail Project Desain

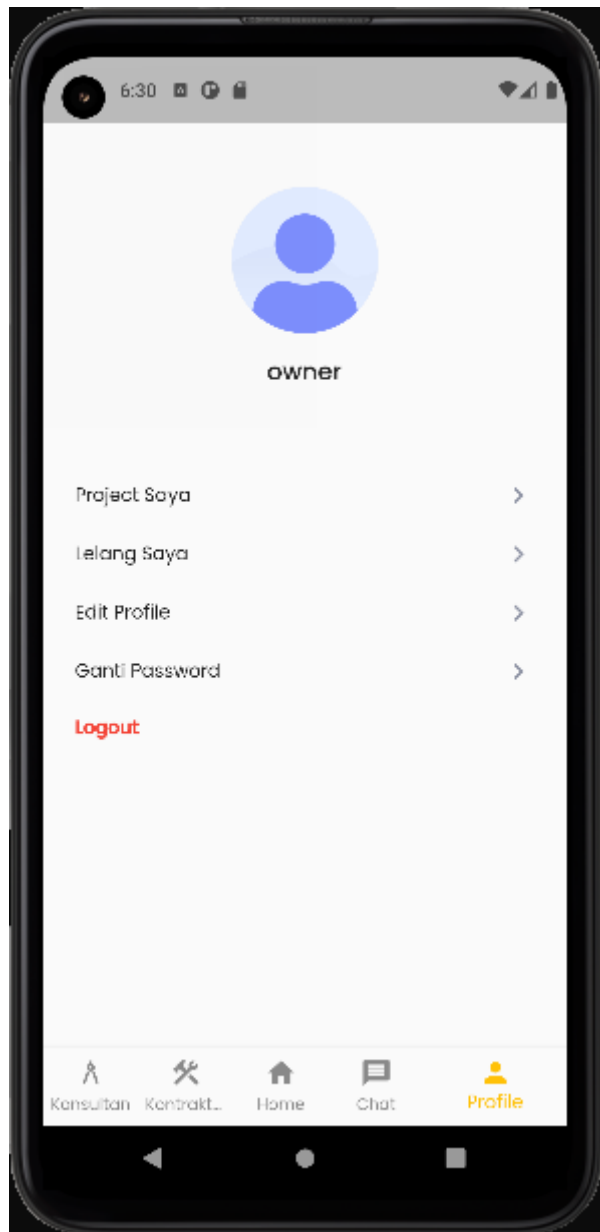
Halaman detail project desain adalah berisi detail informasi mengenai desain.



Gambar 4. 25 Detail Project Desain Mobile

G. Halaman Profile Owner

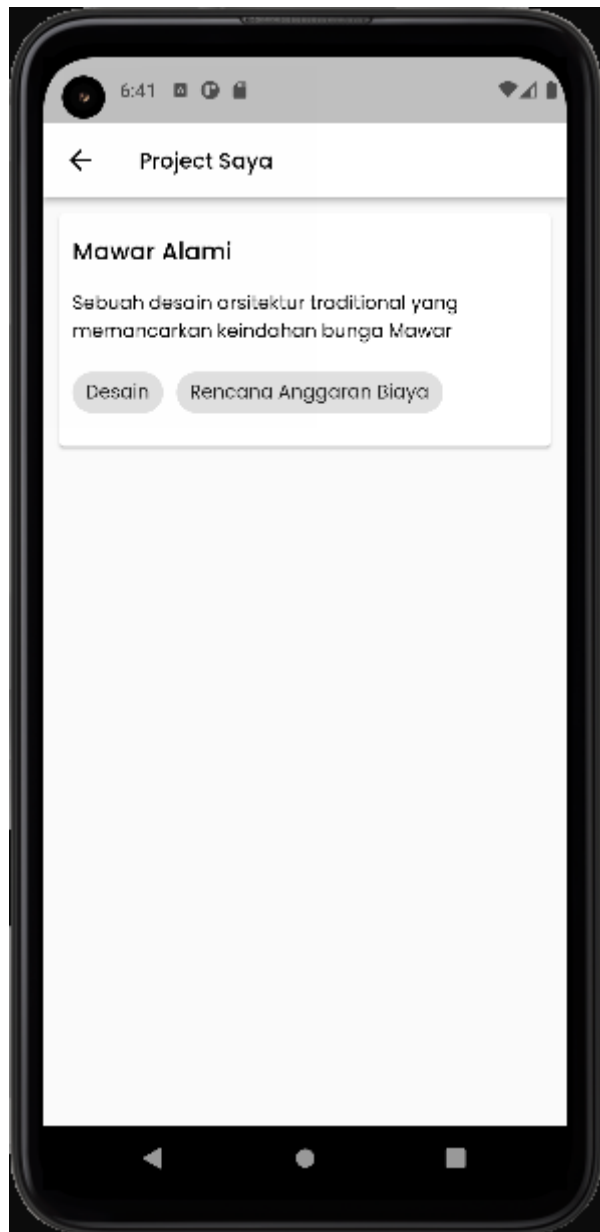
Halaman profile adalah halaman untuk melihat informasi data owner.



Gambar 4. 26 Profile Owner Mobile

H. Halaman Project Saya

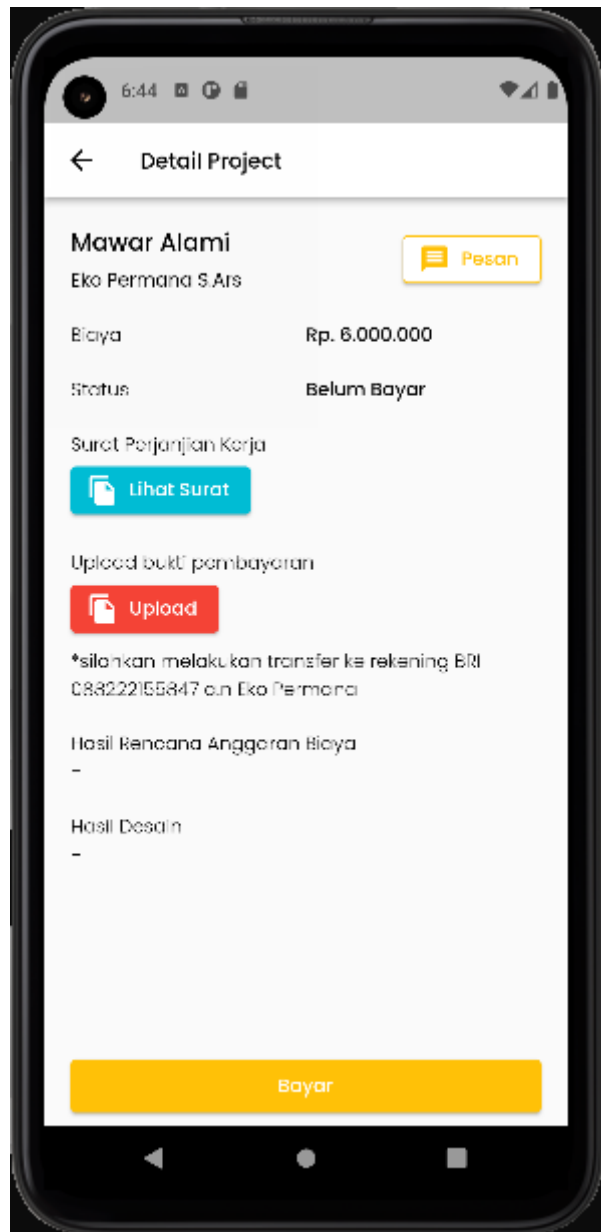
Halaman project saya adalah halaman owner untuk melihat project/kontrak kerja sama yang sedang berjalan.



Gambar 4. 27 Project Saya Mobile

I. Halaman Detail Project Saya

Halaman detail project saya adalah melihat informasi project dan cara pembayarannya.



Gambar 4. 28 Detail Project Saya Mobile

J. Halaman Lelang Saya Owner

Halaman lelang saya Owner adalah halaman untuk Owner melakukan kustomisasi desain yang diinginkan.



Gambar 4. 29 Lelang Saya Mobile

K. Halaman Desain Saya

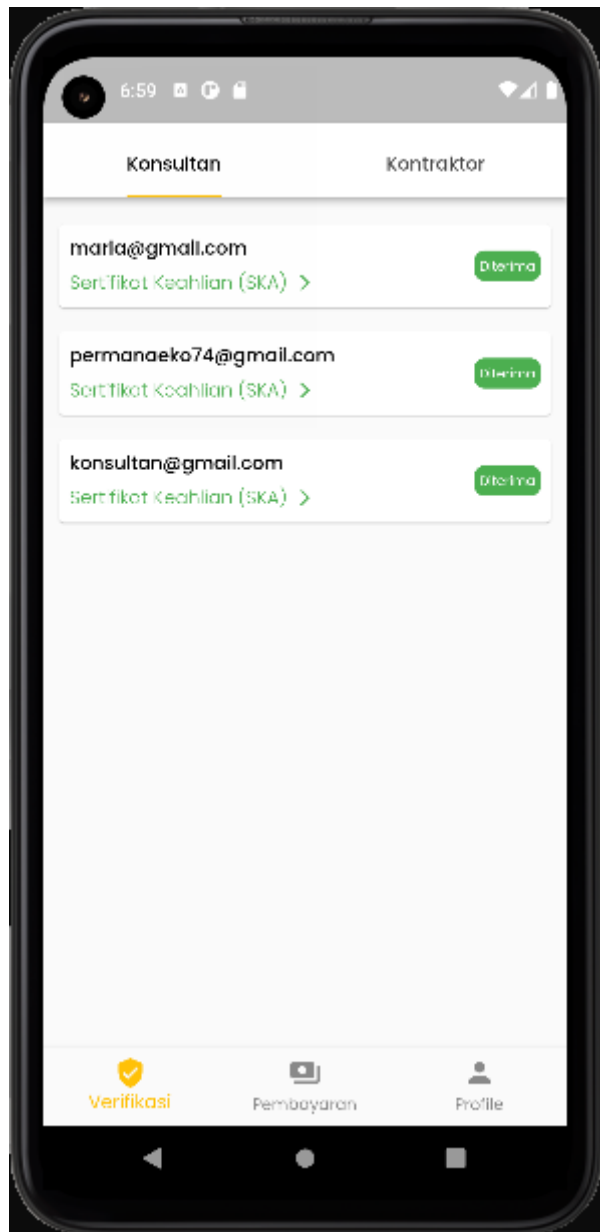
Halaman desain saya adalah halaman untuk Konsultan mengunggah desain yang ia miliki.



Gambar 4. 30 Desain Saya Mobile

L. Halaman Verifikasi Profesional

Halaman verifikasi profesional adalah halaman untuk Admin melakukan verifikasi registrasi akun dan sertifikat konsultan dan kontraktor.



Gambar 4. 31 Halaman Verifikasi Profesional Mobile

4.2.2.3 Kode program

Kode program adalah suatu rangkaian pernyataan atau deklarasi yang ditulis dalam bahasa pemrograman komputer yang terbaca manusia.

A. Potongan Kode Program Website

B. Potongan Kode Program Mobile

4.2.2.4 Pembagian Pekerjaan

Pembagian kerja adalah analisis jabatan yang merupakan suatu aktivitas dalam menentukan apa pekerjaan yang dilakukan dan siapa yang harus melakukan tugas tersebut. Pada tabel 4.2.2.1 dibawah ini akan ada pembagian pekerjaan.

Tabel 4.2.2 1 Pembagian Pekerjaan

Item Pekerjaan yang Diselesaikan		
NO	Target	Dikerjakan Oleh
1.	Riset/diskusi topik	Eko, Lely, Rifki
2.	Analisis Kebutuhan	Eko, Lely, Rifki
3.	Membuat Proposal	Eko, Lely
4.	<i>Design</i> DFD	Lely
5.	<i>Design</i> Flowchart Sistem	Eko
6.	<i>Design</i> ERD	Lely
7.	<i>Design</i> Diagram Use Case	Eko
8.	<i>Design</i> Class Diagram	Eko
9.	<i>Design</i> Diagram Konteks	Lely
10.	<i>Design</i> Diagram Sequence	Lely
11.	<i>Design Mockup Website</i>	Rifki
12.	<i>Design Mockup Mobile</i>	Rifki
13.	UI dan Backend Register Owner Web	Lely
14.	UI dan Backend Register Kontraktor Web	Lely
15.	UI dan Backend Register Kontraktor Mobile	Lely
16.	UI dan Backend Login Web	Eko
17.	Mencari Asset NodeJs	Eko
18.	Mencari Asset Template Admin	Lely
19.	UI Home, Navbar, Konten dan Footer	Eko
20.	UI dan Backend Detail Konsultan	Lely
21.	UI dan Backend Detail Desain Konsultan	Lely
22.	UI dan Backend Detail Kontraktor	Eko
23.	UI dan Backend Detail Tim/Cabang Kontraktor	Eko
24.	UI Informasi Tahapan Konstruksi	Eko
25.	CRUD, UI dan Backend Desain Konsultan	Lely
26.	CRUD, UI dan Backend Cabang Kontraktor	Eko
27.	CRUD, UI dan Backend Lelang Owner	Lely
28.	CRUD, UI dan Backend Profile Owner	Lely
29.	CRUD, UI dan Backend Profile Konsultan	Lely
30.	CRUD, UI dan Backend Profile Kontraktor	Eko
31.	CRUD, UI dan Backend Profile Admin	Eko
32.	UI dan Backend Memilih Desain Konsultan	Eko
33.	Backend Download Kontrak Kerja Owner dengan Konsultan	Eko
34.	UI dan Backend Detail Pemesanan Desain	Eko
35.	UI dan Backend Payment Gateway Desain Konsultan	Eko
36.	UI dan Backend Memilih Tim/Cabang dari Kontraktor	Eko
37.	UI dan Backend Detail Pemesanan Jasa Konstruksi	Eko
38.	UI dan Backend Payment Gateway Jasa Konstruksi	Eko

39.	UI dan Backend Sidebar Konsultan, Kontraktor, dan Admin	Eko
40.	UI Dashboard Konsultan	Lely
41.	UI dan Backend Project Saya (Konsultan)	Eko
42.	UI dan Backend Project Saya Archive (Konsultan)	Eko
43.	UI dan Backend Lelang Owner (Konsultan)	Eko
44.	UI dan Backend mengirim Proposal (Konsultan)	Lely
45.	UI Dashboard Kontraktor	Eko
46.	UI dan Backend Find work (Kontraktor)	Eko
47.	UI dan Backend Jobs Active	Eko
48.	UI dan Backend Jobs Archive	Eko
49.	UI dan Backend Dashboar Admin	Eko
50.	Backend Tabel Last Login	Eko
51.	UI dan Backend Data Tender Win	Lely
52.	UI dan Backend Transaksi Desain	Eko
53.	UI dan Backend Transaksi jasa Konstuksi	Eko
54.	UI dan Backend Verifikasi Akun	Lely
55.	API Authentication	Lely
56.	API Admin	Lely
57.	API GeneratePDF	Lely
58.	API Generate PDFProject	Eko
59.	API Konsultan	Lely
60.	API Kontraktor	Lely
61.	API Owner	Lely
62.	API Profile	Lely
63.	API Project	Lely
64.	API Midtrans	Eko
65.	Mobile UI Guest Mobile	Lely
66.	Mobile UI dan Backend Login	Lely
67.	Mobile UI dan Backend Register	Lely
68.	Mobile UI dan Backend Home Owner	Eko
69.	Mobile UI dan Backend UI Cari Jasa Desain	Lely
70.	Mobile UI dan Backend Detail Desain	Eko
71.	Mobile UI dan Backend Detail Konsultan	Lely
72.	Mobile UI dan Backend Detail Kontraktor	Lely
73.	Mobile UI dan Backend Memilih Desain	Lely
74.	Mobile UI dan Backend Pembayaran	Eko
75.	Mobile UI dan Backend Chat by Owner	Eko
76.	Mobile UI dan Backend Lelang Saya	Lely
77.	Mobile UI dan Backend Project Saya	Lely
78.	Mobile UI dan Backend Profile Owner	Lely
79.	Mobilr UI dan Backend Home Konsultan	Lely
80.	Mobile UI dan Backend Data Proposal Konsultan	Lely
81.	Mobile UI dan Backend Mengirim File Hasil	Lely

	Konsultan	
82.	Mobile CRUD, UI dan Backend Desain Saya by Konsultan	Lely
83.	Mobile CRUD, UI dan Backend Profile Konsultan	Lely
84.	Mobile UI dan Backend Chat by Konsultan	Lely
85.	Mobile UI Home Kontraktor	Lely
86.	Mobile UI Find Work Kontraktor	Eko
87.	Mobile CRUD, UI dan Backend Profile Kontraktor	Eko
88.	Mobile UI dan Backend Verifikasi Pembayaran by Admin	Eko
89.	Mobile UI dan Backend Verifikasi Akun by Admin	Eko
90.	Mobile CRUD, UI dan Backend Profile Admin	Eko

4.2.2.5 API

API adalah singkatan dari Application Programming Interface. API sendiri merupakan interface yang dapat menghubungkan satu aplikasi dengan aplikasi lainnya.

A. API Owner

```

class OwnerController extends BaseController
{

    public function getAllProjectKons()
    {
        $project = Project::with('images', 'konsultan.user',
        'projectOwn.ratings', 'projectOwn.owner.user')->where('isLelang',
        "0")->get();

        return $this->sendResponse($project, 'Data loaded successfully');
    }

    public function detailProjectKons(Project $project)
    {
        $project = Project::with('images', 'konsultan.user')->where('id',
        $project->id)->first();

        return $this->sendResponse($project, 'Data loaded successfully');
    }

    public function getProjectByOwn()
    {

```

```

        $project = ProjectOwner::with('project.konsultan.user',
        'kontrak.payment', 'hasil', 'ratings', 'chooseProject')->where('ownerId',
        $this->getOwnerId()->owner->id)->orderBy('id', 'DESC')->get();

        return $this->sendResponse($project, 'Data loaded successfully');
    }

    public function chooseProject(Request $request)
    {
        $image = array();
        $validator = Validator::make($request->all(), [
            'rab' => Rule::requiredIf($request->has('rab')),
            'panjang' => 'required',
            'lebar' => 'required',
            'desain' => Rule::requiredIf($request->has('desain')),
        ]);
        if ($validator->fails()) {
            return $this->sendError('Validation Error.', $validator->errors());
        }

        $projectOwner = ProjectOwner::create([
            'ownerId' => $this->getOwnerId()->owner->id,
            'projectId' => $request->projectId,
        ]);
        $input = [
            'projectOwnerId' => $projectOwner->id,
            'panjang' => $request->panjang,
            'lebar' => $request->lebar,
        ];
        if ($request->has('rab')) {
            $input += ['RAB' => $request->rab];
        }
        if ($request->has('desain')) {
            $input += ['desain' => $request->desain];
        }
    }

```

```

$choose = ChooseProject::create($input);

if ($request->hasFile('image')) {
    $img = $request->file('image');
    $path = 'img/owner/';
    foreach ($img as $key) {
        $filename = $key->hashName();
        $key->storeAs($path, $filename, 'files');
        $image[] = ImageOwner::create([
            'chooseProjectId' => $choose->id,
            'image' => $filename,
        ]);
    }
}

if ($request->has('telepon')) {
    Owner::firstWhere('id', $this->getOwnerId()->owner->id)->update(['telepon' => $request->telepon]);
}

if ($request->has('alamat')) {
    Owner::firstWhere('id', $this->getOwnerId()->owner->id)->update(['alamat' => $request->alamat]);
}

$data = ProjectOwner::with('owner.user.owner')->where('id', $projectOwner->id)->get();
$generate = new GeneratePDFProject();
$generate->generatePDF($projectOwner->id);

// Mengirim notifikasi ke konsultan
$user = User::where('id', $request->userId)->first();
$owner = User::where('id', Auth::user()->id)->first();

```

```

$SERVER_API_KEY =
'AAAAODXY9xI:APA91bEJBxQ3kKubZRAQTloCk_2aYGXE-
xNUI571Oka9fIKCBwi-
J0p4r__syz4_cpJuVTEDzbCSUJ0YdI_hN66KVjk8MmyqgpwBlIRT
OnhAGe60DgL08q4D0cdyGCGsumJOacD_crt0';

// $SERVER_API_KEY = 'AAAAe3lvlp:APA91bEg_-
VVYnHn12FPjiuyLzvJAaqCiZZHilXP3XImA99x8oEYJU5MEmd
Xwi3wcoooBlJml3uwXnTucZ0a0w2jvwI2NCLinqjmF7CxyAd8p6cx
XOG4Ebjw_lQdA8hO1PNJQU5fiY';

$firebase = [
    "to" => $user->fireBaseToken,
    // "registration_ids" => $user->fireBaseToken,
    "notification" => [
        "title" => "Architect App",
        "body" => "$owner->name membeli desain anda",
    ]
];

$dataString = json_encode($firebase);

$headers = [
    'Authorization: key=' . $SERVER_API_KEY,
    'Content-Type: application/json',
];

$ch = curl_init();

curl_setopt($ch, CURLOPT_URL,
'https://fcm.googleapis.com/fcm/send');
curl_setopt($ch, CURLOPT_POST, true);
curl_setopt($ch, CURLOPT_HTTPHEADER, $headers);
curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
curl_setopt($ch, CURLOPT_POSTFIELDS, $dataString);

// curl_exec($ch);

```

```

$response = curl_exec($ch);

// dd($response);

// return $this->sendResponse(compact('data', 'response'), 'Berhasil
memilih project');
return $this->sendResponse($data, 'Berhasil memilih project');
// return $this->sendResponse(compact('choose', 'image'), 'Berhasil
memilih project');
}

public function postLelangOwner(Request $request)
{
    $foto = array();
    $inspirasi = array();
    $validator = Validator::make($request->all(), [
        'title' => 'required|min:3|string',
        'description' => 'required|min:3|string',
        'desain' => 'required',
        'budgetFrom' => 'required',
        'budgetTo' => 'required',
        'rab' => 'required',
        // 'kontraktor' => 'required',
        'style' => 'required',
        'panjang' => 'required',
        'lebar' => 'required',
    ]);
    if ($validator->fails()) {
        return $this->sendError('Validation Error.', $validator->errors());
    }
    $input = [
        'title' => $request->title,
        'description' => $request->description,
        'budgetFrom' => $request->budgetFrom,

```

```

        'budgetTo' => $request->budgetTo,
        'desain' => $request->desain,
        'RAB' => $request->rab,
        'status' => 0,
        'ownerId' => $this->getOwnerId()->owner->id,
        'gayaDesain' => $request->style,
        'panjang' => $request->panjang,
        'lebar' => $request->lebar
    ];

    if ($request->has('telepon')) {
        Owner::firstWhere('id',          $this->getOwnerId()->owner->id)-
        >update(['telepon' => $request->telepon]);
    }

    if ($request->has('alamat')) {
        Owner::firstWhere('id',          $this->getOwnerId()->owner->id)-
        >update(['alamat' => $request->alamat]);
    }

    $lelang = LelangOwner::create($input);
    if ($request->hasFile('image')) {
        $img = $request->file('image');
        $path = 'img/lelang/tkp/';
        $no = 1;
        foreach ($img as $key) {
            $filename = $key->hashName();
            $key->storeAs($path, $filename, 'files');
            $image = new ImageOwner(['lelangOwnerId' => $lelang->id,
            'image' => $filename]);
            $foto = tap($lelang->image()->save($image)->get());
        }
    }
}

```



```

if ($request->hasFile('inspirasi')) {
    $ins = $request->file('inspirasi');
    $path = 'img/lelang/tkp/';
    $no = 1;
    foreach ($ins as $key) {
        $filename = $key->hashName();
        $key->storeAs($path, $filename, 'files');
        $inspirasi = InspirasiOwner::create(['inspirasi' => $filename,
        'lelangOwnerId' => $lelang->id]);
    }
}

$client = Auth::user()->name;

$SERVER_API_KEY =
'AAAAODXY9xI:APA91bEJBxQ3kKubZRAQTloCk_2aYGXE-
xNUI571Oka9fIKCBwi-
J0p4r__syZ4_cpJuVTEDzbCSUJ0YdI_hN66KVjk8MmyqgpwBlIRT
OnhAGe60DgLO8q4D0cdyGCGsumJOacD_crt0';

// $SERVER_API_KEY = 'AAAAe3lvlp:APA91bEg_-
VVYnHn12FPjiuyLzvJAaqCiZZHilXP3XImA99x8oEYJU5MEmd
Xwi3wcooBlJml3uwXnTucZ0a0w2jvwI2NCLinqjmF7CxyAd8p6cx
XOG4Ebjjw_lQdA8hO1PNJQU5fiY';

$firebase = [
    "to" => "/topics/konsultan",
    // "registration_ids" => $user->fireBaseToken,
    "notification" => [
        "title" => "Architect App",
        "body" => "$client mengusulkan desain custom. Lihat apakah
        kamu cocok dengan pekerjaannya",
    ]
];

$dataString = json_encode($firebase);

$headers = [

```

```

        'Authorization: key=' . $SERVER_API_KEY,
        'Content-Type: application/json',
    ];

    $ch = curl_init();

    curl_setopt($ch, CURLOPT_URL,
'https://fcm.googleapis.com/fcm/send');
    curl_setopt($ch, CURLOPT_POST, true);
    curl_setopt($ch, CURLOPT_HTTPHEADER, $headers);
    curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
    curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
    curl_setopt($ch, CURLOPT_POSTFIELDS, $dataString);

    // curl_exec($ch);
    $response = curl_exec($ch);

    return $this->sendResponse(compact('lelang'), 'Lelang berhasil
ditambahkan');

    // return $this->sendResponse(compact('lelang','inspirasi','foto'),
'Lelang berhasil ditambahkan');
}

public function putLelangOwner(Request $request, LelangOwner
$lelang)
{
    $validator = Validator::make($request->all(), [
        'title' => 'required|min:3|string',
        'description' => 'required|min:3|string',
        'desain' => 'required',
        'budget' => 'required',
        'rab' => 'required',
        'kontraktor' => 'required',
    ]);

    if ($validator->fails()) {

```

```

        return $this->sendError('Validation Error.', $validator->errors());
    }

    $input = [
        'title' => $request->title,
        'description' => $request->description,
        'desain' => $request->desain,
        'budgetFrom' => $request->budgetFrom,
        'budgetTo' => $request->budgetTo,
        'RAB' => $request->rab,
        'kontraktor' => $request->kontraktor,
        'status' => 0,
        'ownerId' => $this->getOwnerId()->owner->id
    ];

    $data = tap(LelangOwner::where('id', $lelang->id))->update($input)->first();
    return $this->sendResponse($data, 'Lelang berhasil diedit');
}

public function deleteLelangOwn(LelangOwner $lelang)
{
    $delete = LelangOwner::destroy($lelang->id);
    return $this->sendResponse($delete, 'lelang berhasil dihapus');
}

public function getLelangByOwn()
{
    $data = LelangOwner::with('image')->get();
    // return view('public.project', compact('data'));
    // return view('public.mylelang', compact('data'));
    return $this->sendResponse($data, 'Data loaded successfully');
}

public function getAllKons()

```

```

{
    $data = Konsultan::with('user', 'files', 'projects.images',
    'projects.konsultan.user', 'projects.projectOwn.ratings',
    'projects.projectOwn.owner.user')->whereHas('user', function ($q) {
        $q->where('is_active', 1);
    })->get();
    return $this->sendResponse($data, 'Data Loaded successfully');
}

public function getProposalByOwn(Request $request)
{
    $data = TenderKonsultan::with('konsultan.user')-
    >where('lelangOwnerId', $request->lelangId)->get();
    return $this->sendResponse($data, 'Data Loaded successfully');
}

public function getProposal()
{
    $data = TenderKonsultan::with('lelang.owner.user',
    'konsultan.user')->get();
    return $this->sendResponse($data, "");
}

public function chooseProposal(Request $request)
{
    $data = tap(TenderKonsultan::where('id', $request->proposalId)-
    >update(['status' => 1])->first();

    TenderKonsultan::where('lelangOwnerId', $request->lelangId)-
    >where('status', 0)->update(['status' => 2]);

    LelangOwner::firstWhere('id', $request->lelangId)->update(['status'
    => 1]);

    $generate = new GeneratePDF();
    $generate->generatePDF($request->proposalId);

    $proposal = TenderKonsultan::where('id', $request->proposalId)-
    >with('lelang.owner.user', 'konsultan.user')->first();

```

```

$konsToken = $proposal->konsultan->user->fireBaseToken;
$ownName = $proposal->lelang->owner->user->name;

$SERVER_API_KEY =
'AAAAODXY9xI:APA91bEJBxQ3kKubZRAQTIoCk_2aYGXE-
xNUI571Oka9fIKCBwi-
J0p4r__syz4_cpJuVTEDzbCSUJ0YdI_hN66KVjk8MmyqgpwBlIRT
OnhAGe60DgL08q4D0cdyGCGsumJOacD_crt0';

// $SERVER_API_KEY = 'AAAAe3lvlp:APA91bEg_-
VVYnHn12FPjiuyLzvJAaqCiZZHilXP3XImA99x8oEYJU5MEmd
Xwi3wcoooBlJml3uwXnTucZ0a0w2jvwI2NCLinqjmF7CxyAd8p6cx
XOG4Ebjw_lQdA8hO1PNJQU5fiY';

$firebase = [
    "to" => $konsToken,
    // "registration_ids" => $user->fireBaseToken,
    "notification" => [
        "title" => "Architect App",
        "body" => "Proposal anda diterima oleh $ownName",
    ]
];

$dataString = json_encode($firebase);

$headers = [
    'Authorization: key=' . $SERVER_API_KEY,
    'Content-Type: application/json',
];

$ch = curl_init();

curl_setopt($ch, CURLOPT_URL,
'https://fcm.googleapis.com/fcm/send');
curl_setopt($ch, CURLOPT_POST, true);
curl_setopt($ch, CURLOPT_HTTPHEADER, $headers);
curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
curl_setopt($ch, CURLOPT_POSTFIELDS, $dataString);

```

```

        // curl_exec($ch);
        $response = curl_exec($ch);

        return $this->sendResponse($data, 'Kontrak Kerja telah dibuat');
    }

    public function uploadPayment(Request $request)
    {
        $bukti = $request->file('bukti');
        $path = 'img/bukti/';
        if ($request->hasFile('bukti')) {

            $filename = $bukti->hashName();
            $bukti->storeAs($path, $filename, 'files');
            $field = [
                'kontrakKonsultanId' => $request->kontrakId,
                'buktiBayar' => $filename,
                'status' => 0
            ];
            $payment = PaymentKonsultan::create($field);
            return $this->sendResponse($payment, 'Mohon menunggu verifikasi');
        }
        return $this->sendError('error', 'Terjadi kesalahan');
    }

    public function rating(Request $request)
    {
        if ($request->has('rating')) {
            $rating = Rating::create([
                'projectOwnerId' => $request->projectOwnerId,
                'rating' => $request->rating
            ]);
        }
    }

```

```

        ProjectOwner::firstWhere('id', $request->projectOwnerId)-
        >update(['status' => "1"]);
    } else {
        $rating = ProjectOwner::firstWhere('id', $request-
        >projectOwnerId)->update(['status' => "1"]);
    }
    return $this->sendResponse(compact('rating'), 'Terimakasih');
}

public function getDataOwner()
{
    $data = User::with('owner')->where('id', Auth::user()->id)->first();
    return $this->sendResponse($data, 'Data berhasil diambil');
}

public function favorit(Request $request)
{
    $favorits = Favorit::all();
    foreach ($favorits as $favs) {
        if ($favs->ownerId == $this->getOwnerId()->owner->id &&
        $favs->projectId == $request->projectId) {
            Favorit::where(['ownerId' => $this->getOwnerId()->owner->id,
            'projectId' => $request->projectId])->delete();
        } else {
            Favorit::create(['ownerId' => $this->getOwnerId()->owner->id,
            'projectId' => $request->projectId]);
        }
    }
    // $favs = Favorit::create(['ownerId'=>$this->getOwnerId()->owner-
    >id,'projectId'=>$request->projectId]);
    return $this->sendResponse($favorits, 'Berhasil menyimpan desain
    favorit');
}

public function hapusFavorit(Request $request)
{

```

```

    $fav = Favorit::destroy($request->idFavorit);
    return $this->sendResponse($fav, 'Berhasil menghapus favorit');
}

// Midtrans Payment
public function payment_handler(Request $request){
    $json = json_decode($request->getContent());

    $signature_key = hash('sha512',$json->order_id . $json->status_code . $json->gross_amount . env('MIDTRANS_SERVER_KEY'));

    if($signature_key != $json->signature_key){
        return abort(404);
    }

    //status berhasil
    $order = Order::where('order_id', $json->order_id)->first();
    return $order->update(['status'=>$json->transaction_status]);
}

public function getAllKontraktor()
{
    $data = Kontraktor::with('user', 'files', 'cabang.images', 'cabang.kontraktor.user')->whereHas('user', function ($q) {
        $q->where('is_active', 1);
    })->get();
    return $this->sendResponse($data, 'Data Loaded successfully');
}

public function getAllCabangKontraktor()
{
    $cabang = KontraktorCabang::with('images', 'kontraktor.user')->where('isLelang', "0")->get();
    return $this->sendResponse($cabang, 'Data loaded successfully');
}

```



```

    }
    public function detailCabangKontraktor(KontraktorCabang $cabang)
    {
        $cabang = KontraktorCabang::with('images', 'kontraktor.user')-
        >where('id', $cabang->id)->first();
        return $this->sendResponse($cabang, 'Data loaded successfully');
    }
}

```

4.2.3 Hasil Sprint Review

Pada bagian ini dijelaskan mengenai hasil sprint review yang dilakukan selama pengerjaan produk aplikasi. Sprint review merupakan hasil diskusi dengan dosen fasilitator yang dilakukan setiap jumat. Adapun rincian hasil pembahasan dirangkum pada tabel 4.2.2.

Tabel 4.2. 2 Sprint Review

NO	KEGIATAN	TIMELINE
1.	Diskusi Topik	Bulan Februari
2.	Pembagian Tugas	Bulan Maret
3.	Pembuatan Mockup	Bulan Maret
4.	Relasi Tabel Database	Bulan Maret
5.	Alur Sistem Proyek	Bulan April
6.	Penambahan Mobile Guest	Bulan April
7.	Pembayaran Sistem Menggunakan Midtrans	Bulan April
8.	Alur Kontraktor	Bulan April

4.2.4 Sprint Retrospective

Pada bagian ini dijelaskan mengenai sprint retrospective yang dilakukan selama pengerjaan produk aplikasi. Sprint retrospective dilakukan untuk kinerja tim. Adapun rincian hasil pembahasan dirangkum pada tabel 4.2.4

Tabel 4.2. 3 Sprint Restropective

NO	KEGIATAN	TIMELINE
1.	Pembuatan Backlog	Bulan Maret
2.	Pembuatan Trello	Bulan Maret
3.	Usulan pindah lokasi (ruangan) pengerjaan proyek (produk aplikasi)	Bulan Maret

4.	Rutinitas daily meeting	Bulan Maret
5.	Pembuatan Seeder	Bulan April

BAB V PENUTUP

5.1 Kesimpulan

Berdasarkan hasil analisis kebutuhan, perancangan, pengimplementasian serta pengujian Implementasi Aplikasi Arsitek.co Berbasis Mobile dan Website Untuk Menghubungkan Masyarakat Dengan Tenaga Profesional Di Bidang Arsitektur maka dapat ditarik kesimpulan :

1. Aplikasi ini membantu stakeholder dalam memperoleh gambaran inspirasi atau referensi desain yang diunggah para arsitek profesional.
2. Aplikasi ini mempermudah para tenaga profesional dalam mempromosikan inspirasi desain yang mereka punya.
3. Masyarakat dapat melihat inspirasi desain bahkan mengajukan inspirasi desain sesuai kebutuhan mereka pada aplikasi ini.

5.2 Saran

Aplikasi ini masih terdapat beberapa kekurangan, oleh karena itu penulis menyarankan beberapa hal perlu diperhatikan seperti:

1. dimungkinkan pengembangan lebih mendalam pada tahap konstruksi bangunan,
2. penggunaan payment gateway bisa digunakan dengan cara lain.

DAFTAR PUSTAKA

- Ainuzzaqy, F., & Nudin, S. R. (2020). Rancang Bangun Aplikasi Siska (Sistem Informasi Karier) Menggunakan Laramen. *Jurnal Manajemen Informatika*, 10(02), 11–20.
- Edi, D., & Betshani, S. (2009). Analisis Data dengan Menggunakan ERD dan Model Konseptual Data Warehouse. *Jurnal Informatika*, 5(1), 71–85. <http://www.itmaranatha.org/jurnal.informatika>
- Fernanda, A. (2021). *Program studi teknik informatika jurusan teknik informatika politeknik negeri indramayu agustus 2021*.
- Hendini, A. (2016). Pemodelan Uml Sistem Informasi Monitoring Penjualan Dan Stok Barang. *Jurnal Khatulistiwa Informatika*, 2(9), 107–116. <https://doi.org/10.1017/CBO9781107415324.004>
- Madya, D. W. I., & Fadhil, S. (2021). *Program studi teknik informatika jurusan teknik informatika politeknik negeri indramayu agustus 2021*.
- Mahendra, I., & Eby Yanto, D. T. (2018). Sistem Informasi Pengajuan Kredit Berbasis Web Menggunakan Agile Development Methods Pada Bank Bri Unit Kolonel Sugiono. *Jurnal Teknologi Dan Open Source*, 1(2), 13–24. <https://doi.org/10.36378/jtos.v1i2.20>
- Malabay. (2016). Pemanfaatan Flowchart Untuk Kebutuhan Deskripsi Proses Bisnis. *Jurnal Ilmu Komputer*, 12(1), 21–26. <https://digilib.esaunggul.ac.id/pemanfaatan-flowchart-untuk-kebutuhan-deskripsi-proses-bisnis-9347.html>
- Rizky, A., Walihadi, A. S. J., Anwar, T., Haryanto, B. A., Idfitri, M., & Safina, H. A. (2022). Perancangan Sistem Informasi Teknologi Website Dalam Pembelajaran Online Berbasis Framework Laravel Di Era Revolusi 4.0. *BEST Journal (Biology Education, Sains and Technology)*, 5(1), 190–196. <https://www.jurnal.uisu.ac.id/index.php/best/article/view/4665>
- Rohayani, Y., Penulis, I., & Bina Sarana Informatika, U. (2020). Penerapan Metode Waterfall Dalam Perancangan Sistem Informasi Akuntansi Pengiriman Barang. *Jurnal Sistem Informasi Akuntansi*, 01(01), 1–09. <https://ejournal.bsi.ac.id/ejurnal/index.php/justian>
- Schwaber, K., & Sutherland, J. (2020). Scrum Guide V7. *Agile Metrics : Agile Health Metrics for Predictability*, November, 133–152.
- Subekti, M., Lukman, L., Indrawan, D., & Putra, G. (2014). Perancangan Case Tools untuk Diagram Use Case, Activity, dan Class untuk Permodelan Uml Berbasis Web Menggunakan HTML5 dan PHP. *ComTech: Computer, Mathematics and Engineering*



Applications, 5(2), 625. <https://doi.org/10.21512/comtech.v5i2.2199>

Suendri. (2018). Implementasi Diagram UML (Unified Modelling Language) Pada Perancangan Sistem Informasi Remunerasi Dosen Dengan Database Oracle (Studi Kasus: UIN Sumatera Utara Medan). *Jurnal Ilmu Komputer Dan Informatika*, 3(1), 1–9. <http://jurnal.uinsu.ac.id/index.php/algorithm/article/download/3148/1871>

LAMPIRAN





Lampiran 1 Logo Aplikasi

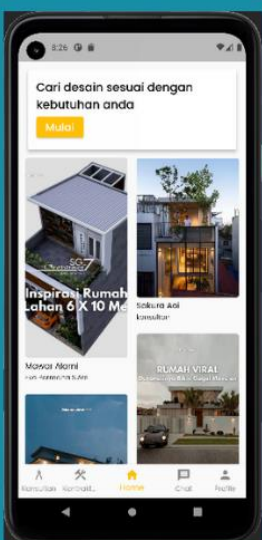


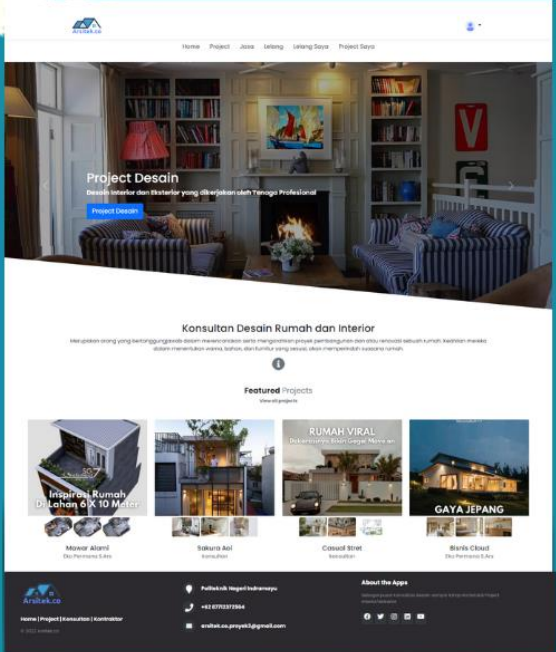
ARSITEK.CO

Eko Permana | Lely Maria Kova | Rifki Zahri
kelompok-3 | D3TI.2C



"sebagai pusat desain arsitektur dan tim kontraktor dalam menyajikan jasanya kepada masyarakat"





TEKNIK INFORMATIKA
POLITEKNIK NEGERI INDRAMAYU

Lampiran 2 Poster