

**PERANCANGAN APLIKASI *TO-DO LIST* PADA MAHASISWA
PENDIDIKAN PROFESI DOKTER BERBASIS GAMIFIKASI**

(Skripsi)

Oleh:

**SABRI RAMADHANI
NPM 1715061012**



**FAKULTAS TEKNIK
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2024**

**PERANCANGAN APLIKASI *TO-DO LIST* PADA MAHASISWA
PENDIDIKAN PROFESI DOKTER BERBASIS GAMIFIKASI**

Oleh

SABRI RAMADHANI

Skripsi

**Sebagai Salah Satu Syarat untuk Mencapai Gelar
SARJANA TEKNIK**

Pada

**Program Studi S-1 Teknik Informatika
Jurusan Teknik Elektro
Fakultas Teknik Universitas Lampung**



**FAKULTAS TEKNIK
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2024**

ABSTRAK

PERANCANGAN APLIKASI *TO-DO LIST* PADA MAHASISWA PENDIDIKAN PROFESI DOKTER BERBASIS GAMIFIKASI

Oleh

SABRI RAMADHANI

Pada tahap pendidikan profesi dokter, mahasiswa menangani kasus nyata dan menerapkan ilmu preklinik melalui analisis, evaluasi, dan implementasi sambil mengoptimalkan waktu untuk mencapai kompetensi yang harus dicapai. Manajemen waktu adalah pendekatan untuk mengoptimalkan waktu agar lebih efektif, tetapi banyak individu, termasuk mahasiswa profesi dokter, masih menggunakan metode manual seperti pencatatan di kertas yang kurang efisien dalam menjalankan aktivitas. Penelitian ini bertujuan untuk mengembangkan sebuah aplikasi *To-Do List* berbasis gamifikasi. Gamifikasi mengacu pada penerapan elemen game untuk meningkatkan pengalaman pengguna dalam konteks non-game, yang berguna memotivasi mahasiswa profesi dokter agar selalu produktif dalam menjalani aktivitas selama tahap pendidikan profesi dokter. Aplikasi ini memungkinkan mahasiswa profesi dokter mengelola tugas dengan lebih efisien dan memungkinkan dokter pembimbing memberikan dan memantau tugas kepada mahasiswa berdasarkan stase. Selain fitur tugas dan stase, aspek gamifikasi seperti level, koin, peringkat, dan pencapaian, yang dapat mendukung aplikasi. Aplikasi *To-Do List* berbasis gamifikasi ini dikembangkan menggunakan *framework* Laravel dan *framework* Flutter dengan metode pengembangan *Scrumban*. Pengujian aplikasi dilakukan menggunakan metode *Unit Testing*, *Black Box Testing* dan *Usability Testing*. Pengujian menggunakan metode *Unit Testing* pada *VS Code* dengan 9 skenario dan *Black Box Testing* dengan 54 skenario menunjukkan bahwa aplikasi berjalan sesuai dengan kebutuhan fungsional dan non-fungsional yang telah ditetapkan. Pengujian *Usability Testing* menggunakan UEQ pada lima dari enam skala memiliki nilai rata-rata berada di atas 1.25 yang mengindikasikan bahwa hasilnya masuk dalam kategori baik.

Kata Kunci: Pendidikan Profesi Dokter, *To-Do List*, Gamifikasi, Flutter, Laravel, *Scrumban*, *Black Box Testing*

ABSTRACT

DESIGN OF A GAMIFICATION-BASED TO-DO LIST APPLICATION FOR MEDICAL PROFESSIONAL EDUCATION STUDENTS

By

SABRI RAMADHANI

During the stage of medical professional education, students handle real cases and apply preclinical knowledge through analysis, evaluation, and implementation while optimizing time to achieve the required competencies. Time management is an approach to optimizing time more effectively, but many individuals, including medical professional education students, still use manual methods such as paper note-taking, which are less efficient for conducting activities. This research aims to develop a gamification-based To-Do List application. Gamification refers to the application of game elements to enhance user experience in a non-game context, which helps motivate medical professional education students to remain productive during their professional education stage. This application allows medical professional education students to manage tasks more efficiently and enables supervising doctors to assign and monitor tasks for students based on their rotations. In addition to task and rotation features, gamification aspects such as levels, coins, rankings, and achievements can support the application. The gamification-based To-Do List application is developed using the Laravel framework and Flutter framework with the Scrumban development method. Application testing is conducted using Unit Testing, Black Box Testing, and Usability Testing methods. Testing using Unit Testing on VS Code with 9 scenarios and Black Box Testing with 54 scenarios shows that the application operates according to the established functional and non-functional requirements. Usability Testing using UEQ on five of six scales shows an average score above 1.25, indicating that the results fall into the good category.

Keywords: Medical Professional Education, To-Do List, Gamification, Flutter, Laravel, Scrumban, Black Box Testing

Judul Skripsi : **PERANCANGAN APLIKASI *TO-DO LIST* PADA MAHASISWA PENDIDIKAN PROFESI DOKTER BERBASIS GAMIFIKASI**

Nama Mahasiswa : **Sabri Ramadhani**

Nomor Pokok Mahasiswa : 1715061012

Program Studi : Teknik Informatika

Jurusan : Teknik Elektro

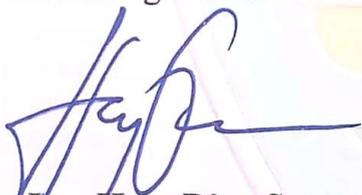
Fakultas : Teknik

MENYETUJUI

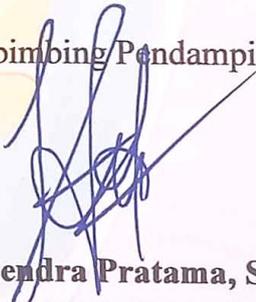
1. Komisi Pembimbing

Pembimbing Utama

Pembimbing Pendamping



Ir/Eng. Hery Dian Septama, S.T.
NIP. 198509152008121001



Mahendra Pratama, S.T., M.Eng.
NIP. 199112152019031013

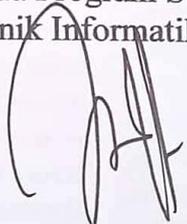
2. Mengetahui

Ketua Jurusan Teknik Elektro

Ketua Program Studi
Teknik Informatika



Herlinawati, S.T., M.T.
NIP. 197103141999032001



Yessi Mulyani, S.T., M.T.
NIP. 197312262000122001

MENGESAHKAN

1. Tim Penguji

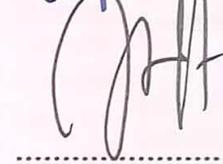
Ketua : Ir. Ing. Hery Dian Septama, S.T.



Sekretaris : Mahendra Pratama, S.T., M.Eng.



Penguji : Yessi Mulyani, S.T., M.T.



2. Dekan Fakultas Teknik


Dr. Eng. Helmy Fitriawan, S.T., M.Sc. }

NIP. 197509282001121002

Tanggal Lulus Ujian Skripsi : 13 Juni 2024

SURAT PERNYATAAN

Saya yang bertanda tangan di bawah ini, dengan ini menyatakan bahwa skripsi berjudul “Perancangan Aplikasi *To-Do List* pada Mahasiswa Pendidikan Profesi Dokter. Berbasis Gamifikasi” adalah karya asli yang saya buat sendiri. Seluruh isi dari skripsi ini telah disusun sesuai dengan prinsip-prinsip penulisan karya ilmiah yang berlaku di Universitas Lampung. Saya menyadari dan bertanggung jawab penuh atas keaslian karya ini. Apabila di masa mendatang terbukti bahwa skripsi ini merupakan plagiat atau dibuat oleh pihak lain, saya bersedia menerima segala bentuk sanksi yang ditetapkan sesuai dengan regulasi hukum dan akademik yang berlaku.

Bandar Lampung, 13 Juni 2024

Pembuat Pernyataan,



Sabri Ramadhani

NPM. 1715061012

RIWAYAT HIDUP



Penulis dilahirkan di Bandar Lampung pada tanggal 28 Desember 1999. Penulis merupakan anak pertama dari 3 bersaudara dari pasangan Bapak Abdurrahman Zainuddin dan Ibu Susanti. Penulis menyelesaikan pendidikannya di TK Aisyiyah Teluk Betung pada 2005, SD Negeri 4 Sawah Brebes pada tahun 2011, SMP Negeri 5 Bandar Lampung pada tahun 2014 dan SMK Negeri 4 Bandar Lampung pada tahun 2017. Pada tahun 2017 penulis terdaftar sebagai mahasiswa Program Studi Teknik Informatika Universitas Lampung melalui jalur SBMPTN.

Selama menjalani proses perkuliahan secara aktif, penulis mengikuti Himpunan Mahasiswa Teknik Elektro (HIMATRO) sebagai anggota Departemen Sosial dan Kewirausahaan Periode 2018. Pada organisasi luar kampus, penulis juga mengikuti komunitas Gerakan Digital Ekosistem Nusantara (GRADIEN). Penulis juga pernah melaksanakan kerja praktik di PT. Kazee Digital Indonesia pada tahun 2021 yang bertempat di Bandung dan masuk dalam divisi Android Development. Topik yang dibuat oleh penulis saat melakukan kerja praktik adalah “Pengembangan Sistem *Kazee Monitoring* Berbasis Mobile Menggunakan *Framework Flutter*”. Kemudian pada tahun 2023 penulis mengaplikasikan ilmu di bidang akademis dengan melaksanakan Kuliah Kerja Nyata (KKN) di Kelurahan Kalirejo, Kecamatan Kalirejo, Lampung Tengah.

Prestasi yang pernah dicapai penulis antara lain adalah sebagai penerima program beasiswa Lintasarta DigiSchool Tahun 2020 yang diselenggarakan Lintasarta dan Dicoding, penerima program beasiswa Indosat Ooredoo Hutchison (IDCamp) Tahun 2023 pada bidang Multi-Platform App Developer, dan sebagai penerima Program Google Arcade Fasilitator 2024 pada bidang Komputasi Awan dan Kecerdasan Buatan. Keahlian penulis adalah pengembangan *mobile apps* dengan *framework Flutter* dan pengembangan *back-end* dengan *framework Laravel*.

MOTTO

“Jika kamu tidak sanggup menahan lelahnya belajar, maka kamu harus sanggup menahan perihnya kebodohan”

(Imam Syafi’i)

“Ikatlah (catatlah) ilmu dengan menulis”

(Ali bin Abi Thalib)

“I fear not the man who has practiced 10,000 kicks once, but I fear the man who has practiced one kick 10,000 times”

(Bruce Lee)



Dengan menyebut nama Allah Yang Maha Pengasih lagi Maha Penyayang. *Alhamdulillah* dengan segala rahmat dan karunia-Nya, skripsi ini dapat diselesaikan dengan baik dan lancar. Semoga dengan keberhasilan yang telah dicapai ini saya dapat menuju masa depan yang lebih baik dan dapat menggapai cita-cita serta selalu berada di jalan-Mu.

KUPERSEMBAHKAN KARYA ILMIAH INI
TERUNTUK:

“Ibunda Susanti dan Ayahanda Abdurrahman Zainuddin atas dukungan dan kasih sayang tiada batas yang telah diberikan mulai dari saya lahir ke dunia ini sampai saya sudah besar seperti sekarang ini. Terima kasih kepada Ibu dan Ayah atas doa yang tak henti-hentinya dipanjatkan serta pengorbanan yang tak terhitung nilainya. Semoga dengan ilmu dan cita-cita yang saya dapatkan kelak akan menjadi amal jariyah bagi Ibu dan Ayah”

“Terima kasih untuk adik-adikku Risnanda Febriani dan Ratri Adzani Rahman yang selalu menemani, bercanda gurau dan bermain denganku saat di rumah. Semoga kalian kelak menjadi pribadi yang lebih sukses dari kakakmu”

“Diriku sendiri. Maaf untuk malam-malam panjang dengan mata yang sulit tertidur, kepala yang sakit, dan lelah pikiran. Terima kasih telah berjuang dan bekerja sama selama ini. Kamu hebat.”

“Terima kasih kepada teman-teman Teknik Informatika 2017 yang telah menemani dan membantu saya selama perkuliahan di kampus tercinta Universitas Lampung. Terima kasih atas kenangan yang kalian berikan selama perkuliahan mulai dari proses pengenalan kampus sampai dengan akhir semester. Semoga kelak kita semua akan menjadi orang-orang yang sukses.”

“Terima kasih kepada teman-teman lainnya yang selalu memberikan semangat dan motivasi saat saya jatuh dan yang selalu mengingatkan saya untuk bersyukur saat saya bahagia. Semoga Allah selalu memberikan kita kebahagiaan pada apa pun jalan yang kita pilih di masa depan.”

SANWACANA

Puji syukur penulis panjatkan kepada Allah SWT, yang telah memberikan yang karunia serta ridho-Nya sehingga penulis dapat melaksanakan dan menyelesaikan penelitian ini yang berjudul “Perancangan Aplikasi *To-Do List* pada Mahasiswa Pendidikan Profesi Dokter Berbasis Gamifikasi”. Penelitian ini merupakan salah satu syarat untuk menyelesaikan kurikulum mata kuliah penelitian skripsi pada Program Studi Teknik Informatika, Jurusan Teknik Elektro, Fakultas Teknik, Universitas Lampung.

Pelaksanaan penelitian ini penulis mendapatkan bantuan, bimbingan serta pengarahan dari berbagai pihak. Maka dari itu, penulis mengucapkan terima kasih sebanyak-banyaknya kepada:

1. Allah SWT yang senantiasa memberikan kemudahan dan kelancaran kepada penulis serta Rasulullah Muhammad SAW yang telah menjadi suri tauladan selama penelitian berlangsung;
2. Ibu dan Ayah serta keluarga penulis yang selalu memberikan motivasi dan dukungan kepada penulis;
3. Bapak Dr. Eng. Helmy Fitriawan, S.T., M.Sc., selaku Dekan Fakultas Teknik Universitas Lampung;
4. Ibu Herlinawati, S.T., M.T., selaku Ketua Jurusan Teknik Elektro Universitas Lampung;
5. Ibu Yessi Mulyani, S.T., M.T., selaku Ketua Program Studi Teknik Informatika Universitas Lampung sekaligus Penguji yang telah membimbing dan membantu proses kelancaran pengerjaan penelitian;
6. Bapak Ir. Ing. Hery Dian Septama, S.T., selaku Pembimbing Utama penelitian yang selalu meluangkan waktunya untuk memberikan bimbingan dan dukungan;

7. Bapak Mahendra Pratama, S.T., M.Eng., selaku Pembimbing Pendamping penelitian yang selalu memberikan motivasi dan memberikan bimbingan kepada penulis untuk menjadi lebih baik;
8. Bapak Ir. Gigih Forda Nama, S.T., M.T.I., IPM., selaku Pembimbing Akademik yang telah memberikan bimbingan selama perkuliahan di setiap semester;
9. Mbak Rika Asliana, S.IP., selaku Admin Program Studi Teknik Informatika yang telah memberikan bantuan dalam proses administrasi penelitian;
10. GRADIEN selaku komunitas sekaligus wadah penulis dalam melakukan pembelajaran di luar perkuliahan;
11. Teman-teman Teknik Informatika 2017 yang selalu mendukung penulis;
12. Semua pihak yang turut serta dalam membantu menyelesaikan penelitian dan tidak mungkin penulis sebutkan satu persatu.

Penulis menyadari bahwa dalam penulisan laporan penelitian ini masih bisa disempurnakan kembali. Oleh karena itu, penulis mengharapkan saran dan kritik yang bersifat membangun dari para pembaca. Penulis berharap laporan skripsi ini dapat bermanfaat bagi banyak pihak.

Bandar Lampung, 13 Juni 2024
Penulis,

Sabri Ramadhani
NPM. 1715061012

DAFTAR ISI

	Halaman
DAFTAR ISI	i
DAFTAR TABEL	iv
DAFTAR GAMBAR	v
I. PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	3
1.3. Tujuan Penelitian	3
1.4. Manfaat Penelitian	3
1.5. Batasan Masalah	3
1.6. Sistematika Penulisan	4
II. TINJAUAN PUSTAKA	5
2.1. <i>Co-Assistant</i>	5
2.2. <i>To-Do List</i>	5
2.3. Gamifikasi	6
2.4. <i>Android</i>	8
2.5. Flutter	8
2.6. Laravel	9
2.7. MVVM	10
2.8. MVC	11
2.9. <i>Scrumban</i>	11
2.10. UEQ	12
2.11. Penelitian Terkait	13
2.11.1. Analisis dan Perancangan <i>User Interface</i> dan <i>User Experience</i> Aplikasi <i>Mobile E-Learning</i> Berbasis Gamifikasi dengan Metode <i>Design Thinking</i>	13

2.11.2. <i>Effectiveness of Innovative Gamified Learning Among Undergraduate Medical Students</i>	14
2.11.3. <i>Impact of Gamification on Student Engagement in Graduate Medical Studies</i>	15
2.11.4. Pembangunan Tampilan Antarmuka Sistem Informasi <i>Logbook</i> Koas Program Studi Pendidikan Dokter Fakultas Kedokteran Universitas XYZ	15
2.11.5. Pengembangan Sistem Informasi <i>To Do List</i> Berbasis Website dalam Meninjau Kegiatan Mahasiswa Undiksha	16
2.11.6. Pengembangan Aplikasi <i>To Do List</i> dalam Meningkatkan Produktivitas Mahasiswa Universitas Brawijaya Pada Masa Pandemi	16
2.11.7. Perancangan Aplikasi <i>ToDo List</i> Berbasis Android Menggunakan <i>Flutter SDK</i> dan <i>Database SQLite</i>	17
III. METODOLOGI PENELITIAN	21
3.1. Waktu dan Tempat Penelitian	21
3.2. Jadwal Penelitian	21
3.3. Alat dan Bahan Penelitian	22
3.3.1. Alat Penelitian	22
3.3.2. Bahan Penelitian	23
3.4. Tahapan Penelitian	23
3.4.1. <i>Goals</i>	24
3.4.2. <i>Story Queue</i>	24
3.4.3. <i>Analysis</i>	26
3.4.4. <i>Development</i>	29
3.4.5. <i>Test</i>	30
3.4.6. <i>Deployment</i>	30
3.4.7. <i>Done</i>	30
IV. HASIL DAN PEMBAHASANN	Error! Bookmark not defined.
4.1. Hasil	Error! Bookmark not defined.
4.1.1. Perancangan	Error! Bookmark not defined.
4.1.1.1. <i>Information Architecture</i>	Error! Bookmark not defined.

4.1.1.2. <i>Lo-Fi Wireframe</i>	Error! Bookmark not defined.
4.1.1.3. <i>Design System</i>	Error! Bookmark not defined.
4.1.1.4. <i>Mockup</i>	Error! Bookmark not defined.
4.1.1.5. <i>Entity Relationship Diagram (ERD)</i>	Error! Bookmark not defined.
4.1.1.6. Algoritma Gamifikasi	Error! Bookmark not defined.
4.1.2. Pengembangan	Error! Bookmark not defined.
4.1.2.1. Pengembangan <i>Backend</i>	Error! Bookmark not defined.
4.1.2.2. Pengembangan Aplikasi	Error! Bookmark not defined.
4.1.3. Pengujian	Error! Bookmark not defined.
4.1.3.1. <i>Unit Testing</i>	Error! Bookmark not defined.
4.1.3.2. <i>Usability Testing</i>	Error! Bookmark not defined.
4.2. Pembahasan	Error! Bookmark not defined.
4.2.1. Metode Pengembangan	Error! Bookmark not defined.
4.2.1.1. <i>Goals</i>	Error! Bookmark not defined.
4.2.1.2. <i>Story Queue</i>	Error! Bookmark not defined.
4.2.1.3. <i>Analysis</i>	Error! Bookmark not defined.
4.2.1.4. <i>Development</i>	Error! Bookmark not defined.
4.2.1.5. <i>Test</i>	Error! Bookmark not defined.
4.2.1.6. <i>Deployment</i>	Error! Bookmark not defined.
4.2.1.7. <i>Done</i>	Error! Bookmark not defined.
4.2.2. Capaian Penelitian	Error! Bookmark not defined.
V. KESIMPULAN DAN SARAN	185
5.1. Kesimpulan	185
5.2. Saran	186
DAFTAR PUSTAKA	187

DAFTAR TABEL

	Halaman
Tabel 2.1. Elemen Gamifikasi	7
Tabel 2.2. Contoh Item pada UEQ	12
Tabel 2.3. Perbandingan Penelitian Terkait	18
Tabel 3.1. Jadwal Penelitian	21
Tabel 3.2. Alat Penelitian	22
Tabel 3.3. <i>Story Queue</i>	25
Tabel 3.4. Kebutuhan Fungsional Aplikasi	26
Tabel 3.5. Kebutuhan Non-Fungsional Aplikasi	27
Tabel 4.1. Gamifikasi pada <i>Character</i>	53
Tabel 4.2. Gamifikasi pada <i>Event</i>	54
Tabel 4.3. Daftar <i>Route</i>	57
Tabel 4.4. Daftar <i>Model</i>	62
Tabel 4.5. Daftar <i>Factory</i>	64
Tabel 4.6. Daftar <i>Seeder</i>	65
Tabel 4.7. Daftar <i>Controller</i>	68
Tabel 4.8. Hasil Pengujian Menggunakan Metode <i>Black Box</i>	154
Tabel 4.9. Hasil Uji Reliabilitas	163
Tabel 4.10. Data Respons UEQ	164
Tabel 4.11. Data Hasil Transformasi UEQ	165
Tabel 4.12. Hasil Rata-Rata, Varian, dan Simpangan Baku UEQ	166
Tabel 4.13. Kualitas Pragmatis dan Hedonis	168
Tabel 4.14. Analisis <i>Benchmark</i> UEQ	169
Tabel 4.15. <i>Product Backlog</i> dan <i>Sprint Backlog</i>	173

DAFTAR GAMBAR

	Halaman
Gambar 2.1. Konsep Gamifikasi	6
Gambar 2.2. <i>Framework Flutter</i>	9
Gambar 2.3. Tampilan Aplikasi <i>E-Learning</i>	14
Gambar 3.1. Tahapan pada metode <i>Scrumban</i>	23
Gambar 3.2. <i>Use Case Diagram</i>	28
Gambar 4.1. <i>Information Architecture</i>	32
Gambar 4.2. <i>Wireframe</i> Halaman <i>Splashscreen</i> dan <i>Login</i>	33
Gambar 4.3. <i>Wireframe</i> Halaman Daftar Tugas	33
Gambar 4.4. <i>Wireframe</i> Halaman Tambah Tugas	34
Gambar 4.5. <i>Wireframe</i> Halaman Daftar Stase	34
Gambar 4.6. <i>Wireframe</i> Halaman Tugas dan Anggota Stase	35
Gambar 4.7. <i>Wireframe</i> Halaman Profil dan Notifikasi	35
Gambar 4.8. <i>Wireframe</i> Halaman Peringkat, Pencapaian dan Toko	36
Gambar 4.9. <i>Wireframe</i> Halaman Pengaturan, Perbarui Profil dan Ubah Password	36
Gambar 4.10. Logo Aplikasi Gagas	37
Gambar 4.11 <i>Typography</i>	38
Gambar 4.12. <i>Color Palette</i>	39
Gambar 4.13. <i>Icon</i>	40
Gambar 4.14. <i>Avatar Illustration</i>	41
Gambar 4.15. <i>Stase Illustration</i>	41
Gambar 4.16. <i>Screen Illustration</i>	42
Gambar 4.17. <i>Dialog Illustration</i>	42
Gambar 4.18. <i>Snackbar Illustration</i>	42

Gambar 4.19. <i>Achievement Illustration</i>	43
Gambar 4.20. <i>Store Illustration</i>	43
Gambar 4.21. <i>Button</i>	44
Gambar 4.22. <i>App Bar</i>	44
Gambar 4.23. <i>Text Field</i>	45
Gambar 4.24. <i>Chip</i>	45
Gambar 4.25. <i>Snackbar</i>	45
Gambar 4.26. Rancangan Halaman <i>Splashscreen</i> dan <i>Login</i>	46
Gambar 4.27. Rancangan Halaman Tugas	46
Gambar 4.28. Rancangan Halaman Tambah Tugas	47
Gambar 4.29. Rancangan Halaman Daftar Stase	47
Gambar 4.30. Rancangan Halaman Tugas dan Anggota Stase	48
Gambar 4.31. Rancangan Halaman Peringkat dan Profil	48
Gambar 4.32. Rancangan Halaman Pencapaian	49
Gambar 4.33. Rancangan Halaman Toko	49
Gambar 4.34. Rancangan Halaman Barang	50
Gambar 4.35. Rancangan Halaman Peraturan, Perbarui Profil dan Ubah Sandi	50
Gambar 4.36. <i>Entity Relationship Diagram</i>	52
Gambar 4.37. Tampilan Kode <i>Route</i>	61
Gambar 4.38. Tampilan Kode <i>Model</i>	63
Gambar 4.39. Tampilan Kode <i>Migration</i>	64
Gambar 4.40. Tampilan Kode <i>Factory</i>	65
Gambar 4.41. Tampilan Kode <i>Seeder</i>	67
Gambar 4.42. Tampilan Kode <i>Controller</i>	72
Gambar 4.43. Tampilan Halaman <i>Login</i>	74
Gambar 4.44. Tampilan Kode <i>User Model</i>	74
Gambar 4.45. Tampilan Kode <i>Login Screen</i>	75
Gambar 4.46. Tampilan Kode <i>User Provider</i>	76
Gambar 4.47. Tampilan <i>Splashscreen</i>	77
Gambar 4.48. Tampilan Kode <i>Splashscreen</i>	77
Gambar 4.49. Tampilan Kode <i>Check Login</i> pada <i>User Provider</i>	78
Gambar 4.50. Tampilan Tombol <i>Logout</i> Aplikasi	79

Gambar 4.51. Tampilan Kode Tombol <i>Logout</i>	79
Gambar 4.52. Tampilan Kode <i>Logout</i> pada <i>User Provider</i>	80
Gambar 4.53. Tampilan Halaman Daftar Tugas dengan Subtugas	81
Gambar 4.54. Tampilan Halaman Daftar Tugas dengan Pelaksana	81
Gambar 4.55. Tampilan Kode <i>Tugas Wrapper Model</i>	82
Gambar 4.56. Tampilan Kode <i>Character Model</i>	82
Gambar 4.57. Tampilan Kode <i>Tugas Model</i>	83
Gambar 4.58. Tampilan Kode <i>Subtugas Model</i>	83
Gambar 4.59. Tampilan Kode <i>Pelaksana Model</i>	84
Gambar 4.60. Tampilan Kode <i>Tugas Screen</i>	84
Gambar 4.61. Tampilan Kode <i>Index</i> pada <i>Tugas Provider</i>	85
Gambar 4.62. Tampilan Halaman Tambah Tugas	86
Gambar 4.63. Tampilan <i>Dialog</i> Berhasil Menambah Tugas	87
Gambar 4.64. Tampilan Kode <i>Tambah Tugas Screen</i>	87
Gambar 4.65. Tampilan Kode <i>Store</i> pada <i>Tugas Provider</i>	88
Gambar 4.66. Tampilan Halaman Detail Tugas Distribusi	89
Gambar 4.67. Tampilan Kode <i>Detail Tugas Screen</i>	90
Gambar 4.68. Tampilan Kode <i>Show</i> pada <i>Tugas Provider</i>	90
Gambar 4.69. Tampilan Dialog Berhasil Memperbarui Tugas	91
Gambar 4.70 Tampilan Kode <i>Update</i> pada <i>Tugas Provider</i>	92
Gambar 4.71. Tampilan <i>Button</i> Hapus Tugas	93
Gambar 4.72. Tampilan Dialog Konfirmasi Hapus Tugas	93
Gambar 4.73. Tampilan Kode <i>Destroy</i> pada <i>Tugas Provider</i>	94
Gambar 4.74. Tampilan <i>Button</i> Izin Tugas	95
Gambar 4.75. Tampilan Dialog Konfirmasi Izin Tugas	95
Gambar 4.76. Tampilan Kode <i>Gamifikasi Model</i>	96
Gambar 4.77. Tampilan Kode <i>Permission</i> pada <i>Tugas Provider</i>	97
Gambar 4.78. Tampilan <i>Button</i> Batalkan Izin Tugas	98
Gambar 4.79. Tampilan Dialog Konfirmasi Batalkan Izin Tugas	98
Gambar 4.80. Tampilan Kode <i>Cancel Permission</i> pada <i>Tugas Provider</i>	99
Gambar 4.81. Tampilan <i>Snackbar</i> saat Tugas Selesai	100
Gambar 4.82. Tampilan Kode <i>Complete</i> pada <i>Tugas Provider</i>	100

Gambar 4.83. Tampilan Subtugas saat Selesai	101
Gambar 4.84. Tampilan Kode <i>Complete Subtugas</i> pada <i>Tugas Provider</i>	102
Gambar 4.85. Tampilan Halaman Daftar Stase	102
Gambar 4.86. Tampilan Kode <i>Sirkulasi Model</i>	103
Gambar 4.87. Tampilan Kode <i>Stase Screen</i>	104
Gambar 4.88. Tampilan Kode <i>Index</i> pada <i>Sirkulasi Provider</i>	105
Gambar 4.89. Tampilan Halaman Daftar Tugas Stase	106
Gambar 4.90. Tampilan Kode <i>Sirkulasi Wrapper Model</i>	106
Gambar 4.91. Tampilan Kode <i>Detail Stase Screen</i>	107
Gambar 4.92. Tampilan Kode <i>Task</i> pada <i>Sirkulasi Provider</i>	108
Gambar 4.93. Tampilan Halaman Anggota Stase	108
Gambar 4.94. Tampilan Kode <i>Anggota Model</i>	109
Gambar 4.95. Tampilan Kode <i>Dosen Model</i>	109
Gambar 4.96. Tampilan Kode <i>Mahasiswa Model</i>	110
Gambar 4.97. Tampilan Kode <i>Anggota Screen</i>	110
Gambar 4.98. Tampilan Kode <i>Member</i> pada <i>Sirkulasi Provider</i>	111
Gambar 4.99. Tampilan Halaman Peringkat	112
Gambar 4.100. Tampilan Kode <i>Peringkat Model</i>	112
Gambar 4.101. Tampilan Kode <i>Peringkat Screen</i>	113
Gambar 4.102. Tampilan Kode <i>Index</i> pada <i>Peringkat Provider</i>	114
Gambar 4.103. Tampilan Halaman Profil Pengguna	115
Gambar 4.104. Tampilan Kode <i>Profil Model</i>	116
Gambar 4.105. Tampilan Kode <i>Profil Screen</i>	116
Gambar 4.106. Tampilan Kode <i>Show</i> pada <i>Profil Provider</i>	117
Gambar 4.107. Tampilan Halaman Perbarui Profil	118
Gambar 4.108. Tampilan Kode <i>Perbarui Profil Screen</i>	118
Gambar 4.109. Tampilan Kode <i>Update</i> pada <i>Profil Provider</i>	119
Gambar 4.110. Tampilan Halaman Ubah Kata Sandi	120
Gambar 4.111. Tampilan Kode <i>Ubah Password Screen</i>	120
Gambar 4.112. Tampilan Kode <i>Change Password</i> pada <i>Profil Provider</i>	121
Gambar 4.113. Tampilan Halaman Daftar Pencapaian	122
Gambar 4.114. Tampilan Kode <i>Pencapaian Model</i>	122

Gambar 4.115. Tampilan Kode <i>Pencapaian Screen</i>	123
Gambar 4.116. Tampilan Kode <i>Index</i> pada <i>Pencapaian Provider</i>	124
Gambar 4.117. Tampilan <i>Dialog</i> Detail <i>Pencapaian</i>	124
Gambar 4.118. Tampilan Kode <i>Achievement Dialog</i>	125
Gambar 4.119. Tampilan Kode <i>Show</i> pada <i>Pencapaian Provider</i>	125
Gambar 4.120. Tampilan <i>Dialog</i> Sukses Klaim <i>Pencapaian</i>	126
Gambar 4.121. Tampilan Kode <i>Helper Dialog</i> untuk Sukses Klaim <i>Pencapaian</i>	127
Gambar 4.122. Tampilan Kode <i>Claim</i> pada <i>Pencapaian Provider</i>	127
Gambar 4.123. Tampilan Halaman Daftar Barang Toko	128
Gambar 4.124. Tampilan kode <i>Toko Wrapper Model</i>	129
Gambar 4.125. Tampilan kode <i>Toko Model</i>	129
Gambar 4.126. Tampilan Kode <i>Toko Screen</i>	129
Gambar 4.127. Tampilan Kode <i>Index</i> pada <i>Toko Provider</i>	130
Gambar 4.128. Tampilan <i>Full-Screen Dialog</i> Detail Toko	131
Gambar 4.129. Tampilan Kode <i>Store Dialog</i>	131
Gambar 4.130. Tampilan Kode <i>Show</i> pada <i>Toko Provider</i>	132
Gambar 4.131. Tampilan <i>Dialog</i> Beli Barang Toko	132
Gambar 4.132. Tampilan Kode <i>Confirm Buy Dialog</i>	133
Gambar 4.133. Tampilan Kode <i>Buy</i> pada <i>Toko Provider</i>	133
Gambar 4.134. Tampilan Halaman Daftar Barang	134
Gambar 4.135. Tampilan Kode <i>Barang Model</i>	134
Gambar 4.136. Tampilan Kode <i>Barang Screen</i>	135
Gambar 4.137. Tampilan Kode <i>Index</i> pada <i>Barang Provider</i>	136
Gambar 4.138. Tampilan <i>Dialog</i> Detail Barang	137
Gambar 4.139. Tampilan Kode <i>Item Dialog</i>	137
Gambar 4.140. Tampilan Kode <i>Show</i> pada <i>Barang Provider</i>	138
Gambar 4.141. Tampilan <i>Dialog</i> Sukses Menggunakan Barang	138
Gambar 4.142. Tampilan Kode <i>Helper Dialog</i> Gunakan Barang	139
Gambar 4.143. Tampilan Kode <i>Use</i> pada <i>Barang Provider</i>	139
Gambar 4.144. Tampilan <i>Dialog QR Code</i> Barang	140
Gambar 4.145. Tampilan <i>Floating Action Button</i> Tukar Barang	141

Gambar 4.146. Tampilan Proses Menukarkan Barang	141
Gambar 4.147. Tampilan <i>Dialog</i> saat Berhasil Tukar Barang	142
Gambar 4.148. Tampilan Kode <i>QR Code Dialog</i>	142
Gambar 4.149. Tampilan Kode <i>Floating Action Button</i> Tukar Barang	143
Gambar 4.150. Tampilan Kode <i>Redeem</i> pada <i>Barang Provider</i>	143
Gambar 4.151. Tampilan Kode <i>Check Redeem</i> pada <i>Barang Provider</i>	144
Gambar 4.152. Tampilan Halaman Daftar Notifikasi	145
Gambar 4.153. Tampilan Kode <i>Notifikasi Model</i>	145
Gambar 4.154. Tampilan Kode <i>Notifikasi Screen</i>	146
Gambar 4.155. Tampilan Kode <i>Index</i> pada <i>Notifikasi Provider</i>	147
Gambar 4.156. Tampilan Halaman Detail Notifikasi	147
Gambar 4.157. Tampilan Kode <i>Detail Notifikasi Screen</i>	148
Gambar 4.158. Tampilan Kode <i>Show</i> pada <i>Notifikasi Provider</i>	148
Gambar 4.159. Tampilan <i>Dialog</i> Konfirmasi Hapus Notifikasi	149
Gambar 4.160. Tampilan Kode <i>Helper Dialog</i> Hapus Notifikasi	149
Gambar 4.161. Tampilan Kode <i>Destroy</i> pada <i>Notifikasi Provider</i>	150
Gambar 4.162. <i>Unit Test</i> pada Fitur-fitur Tugas	151
Gambar 4.163. Tampilan Kode <i>Unit Testing Read</i> Data Tugas	152
Gambar 4.164. Tampilan Kode <i>Unit Testing Insert</i> Data Tugas	152
Gambar 4.165. Tampilan Kode <i>Unit Testing Update</i> Data Tugas	153
Gambar 4.166. Tampilan Kode <i>Unit Testing Delete</i> Data Tugas	154
Gambar 4.167. Diagram Nilai Rata-rata Enam Skala UEQ	167
Gambar 4.168. Diagram Kualitas Pragmatis dan Hedonis	168
Gambar 4.169. Grafik Analisis <i>Benchmark</i>	169
Gambar 4.170. Label Warna pada <i>Taskbar</i>	170
Gambar 4.171. <i>Role Marker</i> pada <i>Taskbar</i>	171
Gambar 4.172. <i>Product Backlog</i> pada <i>Sprint</i> ke-1	172
Gambar 4.173. <i>Definition of Done</i>	172
Gambar 4.174. <i>Product Backlog</i> pada <i>Sprint</i> ke-2	176
Gambar 4.175. <i>Product Backlog</i> pada <i>Sprint</i> ke-3	177
Gambar 4.176. <i>Product Backlog</i> pada <i>Sprint</i> ke-4	177
Gambar 4.177. <i>Product Backlog</i> pada <i>Sprint</i> ke-5	178

Gambar 4.178. <i>Product Backlog</i> pada <i>Sprint</i> ke-6	178
Gambar 4.179. <i>Product Backlog</i> pada <i>Sprint</i> ke-7	178
Gambar 4.180. <i>Product Backlog</i> pada <i>Sprint</i> ke-8	179
Gambar 4.181. <i>Product Backlog</i> pada <i>Sprint</i> ke-9	179
Gambar 4.182. <i>Product Backlog</i> pada <i>Sprint</i> ke-10	179
Gambar 4.183. <i>Product Backlog</i> pada <i>Sprint</i> ke-11	180
Gambar 4.184. <i>Product Backlog</i> pada <i>Sprint</i> ke-12	180
Gambar 4.185. <i>Product Backlog</i> pada <i>Sprint</i> ke-13	181
Gambar 4.186. <i>Product Backlog</i> pada <i>Sprint</i> ke-14	181
Gambar 4.187. <i>Product Backlog</i> pada <i>Sprint</i> ke-15	181
Gambar 4.188. <i>Product Backlog</i> pada <i>Sprint</i> ke-16	182
Gambar 4.189. Ngrok dengan Status Aktif	183

I. PENDAHULUAN

1.1. Latar Belakang

Mahasiswa kedokteran merupakan individu yang sedang menjalani program pendidikan ilmiah, praktik profesional, pelatihan klinis, serta magang, dengan tujuan untuk memperoleh kualifikasi sebagai dokter, dokter spesialis, dan dokter subspecialis sesuai dengan persyaratan yang ditetapkan [1]. Tuntutan pendidikan di fakultas kedokteran mewajibkan mahasiswa untuk mengikuti masa studi preklinik di universitas sebelum melanjutkan karier sebagai dokter di rumah sakit. Seseorang dapat mengikuti Program Pendidikan Profesi Dokter jika mereka telah memenuhi syarat lulus dalam tahap pendidikan akademik dan telah memperoleh gelar Sarjana Kedokteran [2].

Dalam tahap pendidikan profesi, mahasiswa akan menghadapi situasi nyata yang melibatkan kasus dan penanganan pasien. Proses ini memerlukan penerapan ilmu yang telah dipelajari pada tahap preklinik, melalui analisis, evaluasi, modifikasi, dan implementasi. Untuk mengakomodasi proses ini, tahap pendidikan profesi dibagi menjadi 13 stase yang dilaksanakan dalam rentang waktu yang berbeda [3]. Mahasiswa harus memiliki kemampuan dalam mengoptimalkan penggunaan waktu dengan efektif mengingat adanya banyak kompetensi yang harus dicapai dan penyesuaian jadwal pelayanan dokter pendidik klinis [4].

Manajemen waktu merupakan suatu pendekatan yang dinamis dan digunakan untuk mengidentifikasi strategi yang digunakan dalam memanfaatkan waktu dalam melaksanakan berbagai tugas dan aktivitas untuk mencapai sebuah tujuan tertentu. Manajemen waktu yang efektif, yang melibatkan pengaturan tujuan dan prioritas dalam kegiatan, dapat mengoptimalkan efisiensi kerja,

mengurangi tekanan dan tuntutan pekerjaan yang berlebihan, serta meningkatkan hasil akademik yang lebih baik [5].

Pada umumnya, individu biasanya masih mengandalkan metode manual dalam menjalankan aktivitas. Seperti halnya mahasiswa pendidikan profesi dokter yang seringkali menghadapi situasi di mana mahasiswa harus menyelesaikan berbagai aktivitas secara bersamaan. Dalam mengatasi hal tersebut, mahasiswa akan mencatat daftar aktivitas yang harus dikerjakan terlebih dahulu. Umumnya, mahasiswa masih menggunakan metode manual dengan cara mencatat daftar kegiatan tersebut di atas kertas. Menggunakan metode manual akan menyebabkan kurang efisien dan efektifnya waktu yang digunakan dalam menyelesaikan setiap pekerjaan yang ada [6].

Gamifikasi merupakan istilah yang dipinjam dari bahasa Inggris yaitu *gamification*. Gamifikasi mengacu pada penerapan elemen dan desain yang biasanya ditemukan dalam *game* untuk meningkatkan pengalaman pengguna dalam situasi atau konteks yang bukan permainan. Pengalaman pengguna ini mencakup rangkaian efek yang dihasilkan dari interaksi dengan sistem, termasuk pengalaman estetika (yaitu tingkat kepuasan semua indera pengguna), pengalaman makna (yaitu makna yang melekat pada produk) dan pengalaman emosional. Dalam beberapa tahun terakhir, gamifikasi telah diterapkan di berbagai bidang dan telah memotivasi orang untuk mengubah perilaku untuk mengembangkan keterampilan dan untuk mendorong inovasi [7].

Aplikasi *To Do List* berbasis gamifikasi ini diharapkan dapat menjadi solusi dari masalah yang telah dijabarkan sebelumnya, sehingga mahasiswa pendidikan profesi dokter dapat mengelola daftar tugasnya dengan lebih efisien selama menjalankan tahap profesi (*co-assistant*). Selain itu, penerapan gamifikasi dapat meningkatkan motivasi dan semangat mahasiswa dalam menjalankan semua kegiatan di rumah sakit. Pengembangan aplikasi ini menggunakan *framework* Laravel untuk pengembangan *backend* dan *framework* Flutter untuk pengembangan mobile dengan *Scrum* sebagai metode pengembangan aplikasinya.

1.2. Rumusan Masalah

Rumusan masalah dalam penelitian ini yaitu bagaimana membangun aplikasi *To-Do List* berbasis gamifikasi menggunakan *framework* Laravel dan Flutter sehingga memungkinkan mahasiswa pendidikan profesi dokter mengelola daftar tugasnya selama menjalankan tahap profesi (*co-assistant*) dengan metode pengembangan aplikasi *Scrumban*.

1.3. Tujuan Penelitian

Tujuan dari penelitian ini adalah membangun aplikasi *To-Do List* pada mahasiswa pendidikan profesi dokter berbasis gamifikasi yang dapat mengelola daftar tugas selama menjalankan tahap profesi (*co-assistant*).

1.4. Manfaat Penelitian

Adapun manfaat yang diharapkan dalam penelitian ini yaitu:

1. Menerapkan ilmu yang telah diperoleh selama kuliah di prodi Teknik Informatika Universitas Lampung dalam menyelesaikan permasalahan.
2. Menambah wawasan pengetahuan bagi penulis terkait *framework* Laravel, *framework* Flutter dan metode *Scrumban* serta penerapan gamifikasi pada mahasiswa pendidikan profesi dokter.
3. Memperoleh rancangan aplikasi *To-Do List* dengan gamifikasi yang dapat meningkatkan motivasi dalam mengelola daftar tugas mahasiswa pendidikan profesi dokter selama menjalankan tahap profesi (*co-assistant*).

1.5. Batasan Masalah

Batasan masalah dalam penelitian ini adalah sebagai berikut:

1. Pengembangan aplikasi *To-Do List* ini menggunakan *framework* Laravel dan Flutter dengan metode *Scrumban*.
2. Aplikasi ini dapat digunakan bagi mahasiswa pendidikan profesi dokter di Universitas Lampung.

1.6. Sistematika Penulisan

Sistematika yang digunakan pada laporan ini adalah pembagian menjadi beberapa bab pembahasan antara lain sebagai berikut:

BAB I : PENDAHULUAN

Bab ini menguraikan secara umum mengenai latar belakang penelitian, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan penelitian, dan sistematika penulisan.

BAB II : TINJAUAN PUSTAKA

Bab ini membahas mengenai prinsip, dasar teori, dan penelitian terkait yang berfungsi sebagai sumber dalam memahami permasalahan mengenai Perancangan Aplikasi *To-Do List* pada Mahasiswa Pendidikan Profesi Dokter Berbasis Gamifikasi.

BAB III : METODE PENELITIAN

Bab ini membahas mengenai metode penelitian yang digunakan dalam Perancangan Aplikasi *To-Do List* pada Mahasiswa Pendidikan Profesi Dokter Berbasis Gamifikasi.

BAB IV : HASIL DAN PEMBAHASAN

Bab ini berisi tentang pembahasan serta hasil yang diperoleh dalam penelitian berdasarkan metode penelitian yang digunakan, yaitu *Scrumban*.

BAB V : KESIMPULAN DAN SARAN

Bab ini berisi kesimpulan dari hasil penelitian dan saran-saran sebagai masukan untuk penelitian lanjutan di masa mendatang.

DAFTAR PUSTAKA

II. TINJAUAN PUSTAKA

2.1. *Co-Assistant*

Co-Assistant atau yang dapat disingkat menjadi *Co-Ass* atau Koas, merupakan tugas magang bagi mahasiswa kedokteran di rumah sakit yang bertujuan untuk memperkenalkan dunia kedokteran. Tujuannya adalah agar mahasiswa koas tidak merasa terkejut atau tidak sanggup saat memulai karier kedokteran yang sebenarnya di masa mendatang. Dalam program ini, mahasiswa koas akan mendapatkan bimbingan dari dokter pembimbing yang memiliki keahlian di bidang spesifiknya masing-masing. Masa koas diberlakukan setelah seorang mahasiswa lulusan S1 fakultas kedokteran menyelesaikan studinya untuk mendapatkan gelar dokter. Mahasiswa koas akan menjalani rotasi pada berbagai bagian di rumah sakit seperti bedah, saraf, forensik, dan lainnya selama kurang lebih dua tahun [8].

2.2. *To-Do List*

To-Do List merupakan sebuah metode yang bertujuan untuk memberikan dukungan kepada pengguna dalam mencatat, menata, dan mengelola pekerjaan atau kegiatan rutin sehari-hari. Aplikasi yang berfokus pada *To-Do List* dapat menyediakan aksesibilitas serta fleksibilitas bagi pengguna, karena dapat diakses dengan mudah dan cepat melalui perangkat *mobile* seperti *smartphone* atau tablet. Aplikasi *To-Do List* menawarkan berbagai fitur seperti pencatatan tugas, pengaturan prioritas, pengingat waktu, dan penentuan status tugas (selesai atau belum selesai). Aplikasi ini dapat sangat berguna bagi dokter muda di rumah sakit dalam mengatur kegiatan mereka. Dengan aplikasi ini, mahasiswa pendidikan

profesi dokter dapat lebih terorganisir dan lebih mudah dalam menjalankan aktivitasnya tanpa harus sibuk menjadwalkan kegiatannya secara manual [9].

2.3. Gamifikasi

Gamifikasi merupakan istilah yang dipinjam dari bahasa Inggris yaitu *gamification*. Gamifikasi adalah penggunaan elemen dan desain pada *game* untuk meningkatkan pengalaman pengguna dalam konteks *non-game*. Pengalaman ini didefinisikan sebagai seluruh rangkaian efek yang ditimbulkan pengguna dengan interaksi dengan sistem, termasuk pengalaman estetika (tingkat kepuasan semua indera pengguna), pengalaman makna (makna yang melekat pada produk) dan pengalaman emosional [10].



Gambar 2.1. Konsep Gamifikasi

Menurut Jusuf, model pembelajaran gamifikasi memiliki beberapa kelebihan dibandingkan model pembelajaran lainnya, antara lain:

1. Proses memperoleh pengetahuan yang lebih menyenangkan.
2. Dapat memotivasi siswa agar memenuhi aktivitas pembelajarannya.
3. Dapat membantu siswa dalam meningkatkan konsentrasi dan memahami materi pelajaran.
4. Dapat memberi kesempatan siswa untuk terlibat dalam kompetisi, eksplorasi, dan prestasi pada lingkungan pendidikan [11].

Gamifikasi menggunakan berbagai elemen *game* untuk mendapatkan tanggapan berarti dari pengguna. Terdapat beberapa elemen dasar *game* yang dapat diimplementasikan pada gamifikasi seperti pada Tabel 2.1 [12].

Tabel 2.1. Elemen Gamifikasi [12]

Elemen Gamifikasi	Keterangan
<i>Point</i>	Penggambaran numerik kemajuan pengguna sehubungan dengan tugas yang diusulkan (misalnya poin pengalaman).
<i>Badge</i>	Ilustrasi pencapaian pengguna. Setiap <i>badge</i> diperoleh setelah menyelesaikan serangkaian tugas tertentu.
<i>Customization</i>	Kapasitas pengguna untuk menyesuaikan profil avatar, detail pribadi, dan aspek-aspek tertentu dari sistem sesuai dengan preferensi pengguna.
<i>Leaderboard</i>	Papan peringkat terkait dengan peringkat pengguna dibandingkan dengan yang lain untuk memastikan kemajuan pengguna, memberikan signifikasi pada elemen lain (misalnya poin dan level).
<i>Level</i>	Bentuk evaluasi perkembangan keterampilan pengguna dari waktu ke waktu, biasanya dengan mempertimbangkan berbagai aspek, seperti akumulasi poin atau tugas yang diselesaikan.
<i>Challenges / Quest</i>	Tugas yang membutuhkan upaya signifikan, berfungsi sebagai <i>checkpoint</i> bagi pengguna untuk mengevaluasi pemahaman pengguna.
<i>Feedback</i>	Memberikan data timbal balik yang relevan (misalnya, umpan balik visual tentang tugas yang tertunda, tugas yang diselesaikan, kemajuan pembelajaran secara keseluruhan, poin yang diperoleh, dan level yang dicapai) untuk meningkatkan motivasi pengguna untuk terlibat dalam kegiatan lebih lanjut.
<i>Progress Bar / Progress Tracking</i>	Membantu setiap pengguna dalam memantau dan mengawasi kinerja pengguna.

2.4. *Android*

Android adalah sebuah sistem operasi yang dikembangkan oleh Google untuk smartphone. *Android* merupakan platform *open source* yang terbuka untuk semua orang: *developer*, *designer* dan pengembang aplikasi. Artinya, semakin banyak orang yang dapat bereksperimen, membayangkan, dan menciptakan hal-hal yang belum pernah dilihat dunia sebelumnya [13]. Sistem operasi ini berbasis kernel Linux dan dirancang agar dapat digunakan dengan sentuhan. Pengguna dapat berinteraksi dengan perangkat menggunakan sentuhan, gesekan, atau ketukan pada layar. Selain itu, *Android* adalah sebuah sistem *open source* yang memungkinkan pengembang untuk melakukan modifikasi, pembaruan, dan distribusi secara bebas. Selain itu sifat *open source* ini membuat para pengembang aplikasi bebas membuat aplikasi dengan kode-kode sumber yang dikeluarkan Google. Pengembangan android menggunakan konsep *Object Oriented Programming* (OOP) dan salah satu *framework* yang dapat digunakan adalah Flutter.

2.5. *Flutter*

Flutter adalah *framework open source* yang diciptakan oleh Google untuk membangun aplikasi *multi-platform* yang menarik dan dapat di kompilasi secara *native* hanya dengan satu sumber kode saja. Hal ini juga berlaku untuk pemisahan UI dan kode yang biasa terdapat pada bahasa-bahasa pengembangan yang lain. Flutter membuat satu *codebase* yang cukup untuk UI dan *logic*. Flutter mengimplementasikan kodenya dengan *widget*. *Widget* di dalam Flutter dapat berupa komponen visual maupun sekadar penampung bagi *widget-widget* yang lain [14].



Gambar 2.2. *Flutter Framework*

Flutter ditulis dalam bahasa C, C++, dan Dart, serta menggunakan *Skia Graphics Engine*. Flutter menawarkan berbagai *widget* yang sepenuhnya dapat disesuaikan untuk membangun antarmuka asli, termasuk perpustakaan *Material Design* yang menarik dan *widget Cupertino* (bergaya iOS), API yang fleksibel, *scrolling* alami yang halus, kesadaran *platform*, dan *hot reload* yang membantu membangun UI dengan cepat tanpa kehilangan state di emulator, simulator, dan perangkat keras untuk iOS dan Android [15].

2.6. Laravel

Laravel adalah *framework* aplikasi web dengan sintaksis yang ekspresif dan elegan, menyediakan struktur dan pembuatan aplikasi dari awal. Laravel memberikan pengalaman pengembang yang luar biasa dengan fitur-fitur seperti *dependency injection*, lapisan abstraksi *database* yang ekspresif, pengujian unit dan integrasi, serta banyak lagi. Bagi pengguna baru dalam *framework* web PHP atau memiliki pengalaman bertahun-tahun, Laravel adalah *framework* yang cepat dan mutakhir sehingga membantu pengguna memulai sebagai pengembang web atau meningkatkan keahlian ke tingkat berikutnya [16].

Framework yang dikembangkan oleh Taylor Otwell pada April 2011 ini merupakan salah satu *framework* PHP terbaik berdasarkan data dari Google Trend. Hal tersebut dikarenakan Laravel bersifat *open source* dan memiliki sintaksis yang ekspresif dan elegan untuk mempermudah dan mempercepat proses pengembangan website. Salah satu fitur unggulan yang diberikan Laravel adalah

Artisan, yang dapat menyederhanakan perintah-perintah PHP lewat *command line* untuk mengelola website seperti membuat database, membuat *controller*, melihat *route* dan banyak lagi. Laravel dibangun dengan konsep MVC (*Model-View-Controller*) dan berada di bawah lisensi MIT. [17].

2.7. MVVM

Model-View-ViewModel (MVVM) pertama kali diperkenalkan oleh John Gossman, seorang Arsitek Perangkat Lunak di Microsoft yang berfokus pada *Windows Presentation Foundation* dan teknologi Silverlight, pada tahun 2005 melalui blog pribadinya. MVVM mewakili perkembangan dari *Presentation Model (PM) pattern*, yang awalnya diperkenalkan oleh Martin Fowler pada tahun 2004. Tujuan utama dari *PM pattern* adalah untuk memisahkan dan merangkum tampilan visual antarmuka pengguna dari *presentation logic*. Selain itu, *PM pattern* bertujuan untuk membuat *presentation logic* dapat disesuaikan di berbagai antarmuka pengguna sambil meminimalkan dependensi antara antarmuka pengguna dan elemen lainnya [18].

Model mewakili data dan logika bisnis, *View* menampilkan elemen UI kepada pengguna, dan *ViewModel* bertindak sebagai perantara antara *Model* dan *View*, menangani interaksi pengguna, pengikatan data, dan logika khusus UI lainnya. Salah satu manfaat utama MVVM adalah penggunaan kembali komponen, yang memungkinkan pengembangan secara independen pada UI dan logika bisnis. Pemisahan ini meningkatkan fleksibilitas, karena modifikasi pada UI dapat dilakukan tanpa memengaruhi bisnis yang mendasarinya atau logika *backend*. *Pattern* ini juga meningkatkan pengalaman pengguna dengan memastikan antarmuka yang responsif dan interaktif sambil merampingkan proses pengembangan melalui pemisahan masalah yang jelas [19].

2.8. MVC

Model-View-Controller (MVC) adalah konsep arsitektur yang lebih tradisional dibandingkan dengan MVVM dalam pengembangan aplikasi berbasis web. *Model* bertanggung jawab untuk mengoordinasikan, mempersiapkan, memanipulasi, dan mengelola data (yang diambil dari *database*) berdasarkan arahan yang diberikan oleh *Controller*. *View* bertanggung jawab untuk menyajikan informasi visual kepada pengguna sesuai instruksi yang diberikan oleh *Controller*. *Controller* bertanggung jawab untuk mengawasi tugas yang diberikan pada *Model* dan menentukan *View* mana yang harus ditampilkan atas permintaan pengguna [17].

2.9. Scrumban

Scrumban adalah gabungan dari metode *development Agile Scrum* dan *Kanban* untuk mencoba menggunakan fitur dari kedua model pengembangan perangkat lunak. Keuntungan utama dalam *Scrumban* adalah membuat tim anggota dapat bekerja lebih kreatif lagi dalam mengembangkan metode baru. Karena di dalam *Scrumban* juga mengandung sifat dasar dari *Scrum* dan juga fleksibilitas *Kanban*. *Scrumban* juga adalah metode yang paling ringan, di mana tidak ada alat atau metode yang sepenuhnya sempurna atau yang bisa membantu tim sepanjang waktu atau di dalam setiap kegiatan aktivitas mereka. Dengan *Scrumban* ini dapat melengkapi prinsip *Scrum* dengan praktik dari *Kanban* dan menghasilkan metodologi yang luar biasa [20].

Secara singkat *Scrumban* adalah sebuah metodologi yang membuat *Scrum* lebih ramping dan ini membantu berkolaborasi dengan menggunakan metode *Kanban* yang bisa dipakai seperti *workflow board*, *WIP*, *team rules*, dan lain-lain. Itu membuat *Scrum* lebih fleksibel terhadap perubahan yang begitu cepat, dan itu semua ter-*synchronized* dengan *Scrum* seperti *sprint planning*, *sprint review*. Di dunia industry, *Scrumban* menjadi sangat populer, karena *Scrumban* memiliki pengembangan dan pemeliharaan (*development and maintenance*) [21].

2.10. UEQ

User Experience Questionnaire (UEQ) adalah metode pengujian dengan kuesioner untuk mengukur pengalaman pengguna. Dikembangkan oleh Laugwitz, Schrepp, dan Held pada tahun 2005, yang pada awalnya UEQ hanya tersedia dalam bahasa Jerman. Sekarang, UEQ sudah diterjemahkan ke lebih dari 30 bahasa. Metode ini mempermudah analisis dan pengolahan data, serta unggul dibandingkan metode pengujian lainnya karena memberikan gambaran pengalaman pengguna yang komprehensif, mencakup aspek kegunaan (*usability*) hingga pengalaman pengguna (*user experience*) [22].

Item dalam *User Experience Questionnaire* (UEQ) berbentuk diferensial semantik, yaitu setiap item diwakili oleh dua istilah dengan makna yang berlawanan. Urutan istilah diacak untuk setiap item, sehingga setengah dari item dimulai dengan istilah positif dan setengahnya lagi dimulai dengan istilah negatif. *User Experience Questionnaire* menggunakan skala tujuh tahap untuk mengurangi bias kecenderungan sentral yang umum pada jenis item ini. Skala untuk setiap pertanyaan diberi nilai dari -3 hingga +3, di mana -3 mewakili jawaban paling negatif, 0 jawaban netral, dan +3 jawaban paling positif. Contoh item pada UEQ bisa dilihat pada Tabel 2.2 [23].

Tabel 2.2. Contoh Item pada UEQ [23]

	1	2	3	4	5	6	7	
Tidak menarik	o	o	o	o	o	o	o	Menarik

UEQ terdiri dari 6 skala dengan total 26 item:

1. *Attractiveness*: Kesan keseluruhan tentang produk. Apakah pengguna menyukai atau tidak menyukai produk tersebut?
2. *Perspicuity*: Apakah mudah untuk mengenal produk? Apakah mudah untuk mempelajari cara menggunakan produk tersebut?
3. *Efficiency*: Apakah pengguna dapat menyelesaikan tugas mereka tanpa usaha yang tidak perlu?
4. *Dependability*: Apakah pengguna merasa memiliki kendali atas interaksi tersebut?

5. *Stimulation*: Apakah menggunakan produk tersebut menyenangkan dan memotivasi?
6. *Novelty*: Apakah produk tersebut inovatif dan kreatif? Apakah produk menarik minat pengguna? [23]

2.11. Penelitian Terkait

Terdapat beberapa penelitian terkait yang dijadikan sebagai perbandingan serta rujukan mengenai metode serta hasil yang dicapai pada penelitian ini. Berikut merupakan ulasan dari beberapa penelitian terkait:

2.11.1. Analisis dan Perancangan *User Interface* dan *User Experience* Aplikasi *Mobile E-Learning* Berbasis Gamifikasi dengan Metode *Design Thinking*

Penelitian yang dilakukan oleh Azzahra Agitha Karina pada tahun 2023 ini bertujuan untuk menganalisis kebutuhan serta merancang pengalaman pengguna pada aplikasi *mobile e-learning* yang berbasis gamifikasi dengan menerapkan metode *Design Thinking*. Sehingga dari penelitian ini diharapkan dapat memberikan manfaat untuk menunjang pembelajaran bagi Mahasiswa Program Profesi Dokter dalam menjalani masa praktik di rumah sakit. Penelitian ini menghasilkan desain prototype dengan 7 desain fitur utama yaitu *Sign In*, Pembelian Karakter *Astrodoc*, *Clinical Scenarios to Solve* pada *Journey*, Permainan *Word Detective* dan *Medicine Wheel* pada *Galaboard*, *To Do List*, dan *Reminder*. Pengukuran yang dilakukan menggunakan metode SUS dengan dua kali iterasi memperoleh skor akhir 82 yang masuk ke dalam nilai B dengan kategori *excellent* [3].



Gambar 2.3. Tampilan Aplikasi *E-Learning* [3]

2.11.2. *Effectiveness of Innovative Gamified Learning Among Undergraduate Medical Students*

Penelitian ini dilakukan oleh M H Ohn, K M Ohn, U D Souza, S Yusof, dan Z Ariffin dari Universitas Malaysia Sabah pada tahun 2019 ini bertujuan untuk menyelidiki keefektifan pembelajaran gamifikasi di antara mahasiswa kedokteran. Penelitian ini membandingkan *gamified online digital* dengan pembelajaran elektronik konvensional (*e-learning*) di tingkat sarjana kedokteran. Terdapat masing-masing dua puluh satu pertanyaan penilaian pengetahuan dan keterampilan dengan dua puluh pertanyaan pilihan ganda dan satu pertanyaan pemeriksaan klinis terstruktur objektif. Hasil dari penelitian mengungkapkan bahwa pembelajaran berbasis gamifikasi lebih efektif daripada *e-learning* konvensional dalam meningkatkan akurasi diagnostic *Electrocardiogram* (ECG) sedangkan pembelajaran berbasis gamifikasi sama efektifnya dengan *e-learning* konvensional untuk meningkatkan keterampilan interpretasi ECG [24].

2.11.3. *Impact of Gamification on Student Engagement in Graduate Medical Studies*

Penelitian ini dilakukan oleh Ralai Andriamiarisoa dari Walden University pada tahun 2018. Tujuan dari penelitian ini adalah mengidentifikasi apakah terjadi peningkatan yang signifikan dari waktu ke waktu dalam pembelajaran kooperatif, tingkat kognitif, dan keterampilan pribadi ketika menggunakan metode pembelajaran berbasis gamifikasi. *Student Engagement Scoring* (SES) yang telah divalidasi digunakan untuk mengumpulkan data dari mahasiswa kedokteran tahun kedua di sekolah kedokteran California Selatan. Penelitian dilakukan selama 3 bulan yang terbagi dalam 3 periode waktu (TP). Hasil yang diperoleh adalah peningkatan signifikan terlihat antara TP 1 dan TP 2, diikuti oleh penurunan signifikan antara TP 2 dan TP 3. Hasil ini menunjukkan fakta bahwa diperlukan lebih banyak penelitian untuk lebih memahami apakah jenis implementasi gamifikasi tertentu merugikan keterlibatan siswa dalam pendidikan kedokteran, atau apakah prinsip desain yang lebih baik harus dieksplorasi untuk menghasilkan komponen pembelajaran yang efektif yang dapat berdampak positif terhadap keterlibatan siswa dalam pendidikan kedokteran [25].

2.11.4. *Pembangunan Tampilan Antarmuka Sistem Informasi Logbook Koas Program Studi Pendidikan Dokter Fakultas Kedokteran Universitas XYZ*

Penelitian yang dilakukan oleh Ananda Arief Pradipto Mathias pada tahun 2023 ini bertujuan untuk membangun antarmuka Sistem Informasi *Logbook* Koas Program Studi Pendidikan Dokter (PSPD) Fakultas Kedokteran (FK) Universitas XYZ untuk mempermudah penilaian dan pencatatan *logbook* oleh mahasiswa dan dosen. Aplikasi sistem informasi yang dibangun merupakan aplikasi yang berbasis *mobile* dengan menggunakan *framework* Flutter. Penelitian ini menyimpulkan bahwa antarmuka Sistem Informasi *Logbook* Koas di Universitas XYZ berhasil membantu mahasiswa dan dosen dalam penilaian dan pencatatan *logbook*. Pengujian sistem menunjukkan bahwa pengguna puas dengan

tampilan yang mudah digunakan dan tidak ada *error* yang menghambat penggunaan [26].

2.11.5. Pengembangan Sistem Informasi *To Do List* Berbasis Website dalam Meninjau Kegiatan Mahasiswa Undiksha

Penelitian ini dilakukan oleh Rihana Santhi, Aprilia Monica Sari, Dewa Ketut Alit Maha Putra, I Gede Ciptha Aphila Diptha, Kadek Pebriawan, Muhamad Bintang Haekal dan Gede Surya Mahendra pada tahun 2022 di Universitas Pendidikan Ganesha Singaraja. Penelitian ini bertujuan untuk membuat sebuah sistem informasi yang dapat meninjau aktivitas mahasiswa Undiksha, dengan harapan dapat meningkatkan produktivitas mahasiswa dalam menjalani proses pembelajaran. Sistem ini dikembangkan berbasis website dengan menggunakan metode Waterfall. Pada hasil penelitian ini, Sistem Informasi *To Do List* Berbasis Web telah diimplementasikan dengan baik dan mampu meninjau kegiatan mahasiswa Undiksha menjadi lebih mudah. Para Mahasiswa dapat melakukan penambahan tugas, mengecek tugas, dan mengedit tugas. Diharapkan sistem informasi *To Do List* ini dapat membantu para mahasiswa dalam mengerjakan tugas tugasnya [6].

2.11.6. Pengembangan Aplikasi *To Do List* dalam Meningkatkan Produktivitas Mahasiswa Universitas Brawijaya Pada Masa Pandemi

Penelitian ini dilakukan oleh Muhammad Ilyas Abdullah di Universitas Brawijaya pada masa pandemi. Menurut Abdullah aplikasi *To Do List* adalah aplikasi yang digunakan untuk membantu seorang individu dalam mengingat suatu hal, mengatur waktu, serta merencanakan suatu kegiatan. Aplikasi *To Do List* sendiri menjadi aplikasi yang sangat dibutuhkan khususnya pada masa pandemi COVID-19. Pengembangan aplikasi ini menggunakan metode waterfall, pada tahap analisis kebutuhan, Abdullah melakukan wawancara secara teratur

pada 25 mahasiswa Universitas Brawijaya yang akan menjadi sampel terkait aplikasi *To Do List* ini. Pengujian pada penelitian ini menggunakan metode *Black Box Testing* yang fokus terhadap pengujian fungsionalitas. Hasil dari pengujian menunjukkan bahwa aplikasi berjalan dengan normal tanpa adanya bug. Meskipun begitu, penelitian lanjutan dengan jumlah sampel yang lebih besar serta alat instrument yang lebih valid untuk membuktikan efektivitas dari penggunaan aplikasi *To Do List* [27].

2.11.7. Perancangan Aplikasi *ToDo List* Berbasis Android Menggunakan *Flutter SDK* dan *Database SQLite*

Penelitian ini dilakukan oleh Rifky Azmi dan Rahmawati pada tahun 2021 yang memiliki tujuan untuk memaksimalkan manfaat ponsel sebagai sebuah alat pencatatan agenda penjadwalan yang akan menampilkan notifikasi pada waktu yang telah ditentukan. Sistem aplikasi ini menggunakan Android Studio dan Visual Studio Code sebagai *Integrated Development Environment (IDE)* dan *database (SQLite)*. Hasil akhir penelitian ini berupa aplikasi Android bernama *mytodo* yang dapat memanajemen aktivitas dilengkapi dengan fitur *push notification* pada saat batas waktu dan satu jam sebelum batas waktu yang telah ditentukan. Menurut Azmi dan Rahmawati, aplikasi *mytodo* ini dapat menjadi alternatif dalam kegiatan sehari-hari sebagai asisten pengingat yang dibutuhkan pengguna *smartphone* Android, dalam mengatur penjadwalan agenda kegiatan, setelah pengujian dilakukan, aplikasi dapat berjalan dengan baik di perangkat *smartphone* Android, tanpa menggunakan koneksi internet [28].

Untuk memberikan pemahaman yang lebih mendalam mengenai fitur-fitur yang ada dalam berbagai penelitian terkait yang telah dibahas sebelumnya, serta untuk menunjukkan kontribusi penelitian yang sedang dilakukan dalam skripsi ini, diperlukan perbandingan yang komprehensif. Perbandingan ini membantu dalam mengidentifikasi keunggulan dan kekurangan dari setiap penelitian, serta menyoroti inovasi pada penelitian ini. Tabel berikut ini menyajikan perbandingan fitur dari tujuh penelitian terkait yang telah dijelaskan

sebelumnya dengan penelitian yang sedang dilakukan dalam skripsi ini. Adapun perbandingan tersebut dapat dilihat pada Tabel 2.3.

Tabel 2.3. Perbandingan Penelitian Terkait

No	Judul	Tahun	Metode / Tools	Hasil
1	<i>Impact of Gamification on Student Engagement in Graduate Medical Studies</i>	2018	MANOVA (Multivariate Analysis of Variance) dan Student Engagement Survey (SES)	Analisis dapat pembelajaran gamifikasi siswa yang berdampak positif antara Periode Waktu 1 dan 2, diikuti oleh penurunan antara Periode Waktu 2 dan 3, menunjukkan fluktuasi tingkat keterlibatan siswa.
2	<i>Effectiveness of Innovative Gamified Learning Among Undergraduate Medical Student</i>	2019	Perbandingan Pembelajaran Kuasi-Eksperimental dan <i>Electrocardiogram (ECG) Interpretation</i>	Hasil perbandingan antara pembelajaran gamified (pada platform GaMED) dengan e-learning konvensional (pada platform MOODLE alias SMART2UMS) menunjukkan bahwa pembelajaran <i>gamified</i> lebih efektif dalam meningkatkan akurasi diagnostik ECG.
3	Pengembangan Aplikasi <i>To Do List</i> dalam Meningkatkan Produktivitas Mahasiswa Universitas Brawijaya pada Masa Pandemi	2021	<i>Waterfall, Black Box Testing</i> dan metode <i>Kotlin</i>	Aplikasi <i>To-Do List</i> dengan hasil pengujian <i>Black Box</i> yang berjalan normal tanpa ada <i>bug</i> .

Tabel 2.3. Perbandingan Penelitian Terkait (lanjutan)

No	Judul	Tahun	Metode / Tools	Hasil
4	Perancangan Aplikasi <i>ToDo List</i> Berbasis <i>Android</i> Menggunakan <i>Flutter SDK</i> dan <i>Database SQLite</i>	2021	<i>Grounded Research</i> dan <i>framework Flutter</i>	Aplikasi <i>mytodo</i> yang berjalan dengan baik
5	Pembangunan Tampilan Antarmuka Sistem Informasi <i>Logbook Koas</i> Program Studi Pendidikan Dokter Fakultas Kedokteran Universitas XYZ	2022	<i>Framework Flutter</i>	Antarmuka Sistem Informasi <i>Logbook Koas</i> yang berjalan dengan baik tanpa adanya <i>bug</i> .
6	Pengembangan Sistem Informasi <i>To Do List</i> Berbasis Website dalam Meninjau Kegiatan Mahasiswa Undiksha	2022	Metode <i>Waterfall</i> , HTML, PHP dan MySQL	Sistem Informasi <i>To Do List</i> Berbasis Web yang telah diimplementasikan dengan baik.
7	Analisis dan Perancangan <i>User Interface</i> dan <i>User Experience</i> Aplikasi <i>Mobile E-Learning</i> Berbasis Gamifikasi dengan Metode <i>Design Thinking</i>	2023	<i>Design Thinking</i> , <i>System Usability Scale</i> (SUS) dan <i>User Experience Questionnaire</i> (UEQ)	UI/UX Aplikasi <i>E-Learning</i> berbasis gamifikasi dengan nilai SUS 82 dan nilai UEQ di atas 1.5

Tabel 2.3. Perbandingan Penelitian Terkait (lanjutan)

No	Judul	Tahun	Metode / Tools	Hasil
8	Pengembangan Aplikasi <i>To-Do List</i> pada Mahasiswa Pendidikan Profesi Dokter Berbasis Gamifikasi	2024	Metode <i>Scrumban</i> , <i>framework</i> Laravel, <i>framework</i> Flutter dan <i>User Experience Questionnaire</i>	Aplikasi <i>To-Do List</i> (Gagas) yang berjalan dengan baik tanpa ada <i>bug</i> .

3.3. Alat dan Bahan Penelitian

3.3.1. Alat Penelitian

Adapun alat-alat yang digunakan dalam penelitian adalah sebagai berikut:

Tabel 3.2. Alat Penelitian

No	Nama Alat	Spesifikasi	Deskripsi
1	Laptop	Intel(R) Core i5-2520M @ 2.50GHz 2.50 GHz, RAM 8GB, Windows 10 OS.	Perangkat keras yang digunakan sebagai <i>compiler</i> dalam pemrograman
2	<i>Smartphone Android</i>	Realme C2, Android 9, Octa-Core 2.0GHz, RAM 3GB	Perangkat keras sebagai tempat melakukan pengujian aplikasi.
3	Visual Studio Code	Versi 1.88.1	<i>Integrated Development Environment (IDE)</i> untuk mengembangkan aplikasi <i>mobile</i> dan web.
4	Flutter SDK	Versi 3.10.0	<i>Software Development Kit</i> yang diperlukan untuk mengembangkan aplikasi <i>mobile</i> menggunakan bahasa pemrograman Dart.
5	Dart SDK	Versi 3.0.0	Bahasa pemrograman yang digunakan untuk mengembangkan aplikasi <i>mobile</i> .
6	Figma	<i>Online Figma</i>	Perangkat lunak untuk merancang antarmuka aplikasi.
7	Trello	<i>Online Trello</i>	<i>Tools</i> yang digunakan sebagai <i>Scrum Board</i> untuk memvisualisasi alur kerja.

3.3.2. Bahan Penelitian

Bahan yang digunakan dalam penelitian ini adalah klasifikasi dan daftar tugas Mahasiswa Pendidikan Profesi Dokter dari setiap stase selama menjalankan tahap profesi (co-assistant).

3.4. Tahapan Penelitian

Penelitian ini dijalankan dengan beberapa tahapan mengikuti model pengembangan perangkat lunak *Scrumban*. Berikut merupakan gambar dari tahapan pengembangan menggunakan metode *Scrumban*.



Gambar 3.1. Tahapan pada Metode *Scrumban* [29]

Tahapan pada metode *Scrumban* dimulai dengan tahap *Goals* yang bertujuan untuk mendefinisikan tujuan proyek secara luas dan menyeluruh. Tahap *Story Queue* berfungsi untuk memecahkan tujuan yang telah didefinisikan pada tahap *Goals* menjadi beberapa *story* atau *user story*. Tahap *Analysis* akan menganalisis dan memilih *user story* mana yang akan dikembangkan menjadi fitur-fitur pada proyek. Tahap *Development* merupakan tahap pengembangan aplikasi berdasarkan *story queue* yang telah dipilih pada tahap sebelumnya. Tahap *Test* bertujuan untuk menguji apakah *story queue* yang telah dikembangkan telah berjalan dengan baik. Tahap *Deployment* merupakan hasil dari semua *story queue* yang telah berhasil diuji dan digabungkan menjadi suatu aplikasi yang utuh. Tahap *Done* adalah tahap di mana semua tujuan pada proyek telah selesai. Penjelasan dari masing-masing tahapan pada metode *Scrumban* adalah sebagai berikut:

3.4.1. *Goals*

Tahap ini bertujuan untuk mendefinisikan tujuan proyek secara luas dan menyeluruh. Tujuan (*Goals*) tersebut dapat diperoleh dengan beberapa cara seperti *Requirement Gathering*. Penentuan kebutuhan sangat penting dilakukan untuk menentukan seperti apa aplikasi yang akan dikembangkan nantinya, dalam hal ini kebutuhan yang diperoleh bisa didapat dari riset atau wawancara terhadap Mahasiswa Pendidikan Profesi Dokter yang sedang menjalani masa profesi (*co-assistant*).

Adapun beberapa tujuan (*Goals*) pada proyek ini adalah sebagai berikut:

1. Memberikan kemampuan kepada mahasiswa koas untuk merencanakan dan mengelola waktu mereka dengan lebih baik, termasuk menetapkan tenggat waktu dan prioritas untuk tugas-tugas yang harus diselesaikan.
2. Membantu mahasiswa koas meningkatkan produktivitas mereka dengan memberikan aplikasi yang efektif untuk mengelola tugas dan tanggung jawab sehari-hari mereka.
3. Memungkinkan mahasiswa koas untuk melacak kemajuan mereka dalam menyelesaikan tugas dan kegiatan, sehingga mereka dapat mengidentifikasi area yang perlu ditingkatkan.
4. Memberikan pengingat dan notifikasi yang teratur untuk tugas yang harus diselesaikan agar mahasiswa koas tidak melewatkan tenggat waktu penting.
5. Membantu mengurangi stres dan kecemasan dengan membantu mahasiswa mengatur tugas-tugas mereka dengan baik, sehingga mereka dapat menjaga keseimbangan antara studi dan kehidupan pribadi.

3.4.2. *Story Queue*

Tahap ini bertujuan untuk memecahkan tujuan (*goals*) yang telah di definisikan pada tahap *Goals* menghasilkan *User Queue*. *User queue* adalah kumpulan dari *User Story* yang belum dikerjakan yang kemudian akan menjadi pedoman dalam menentukan *product backlog* seperti yang diterapkan dalam metode *Scrum* atau pun *Scrumban*.

Tabel 3.3. *Story Queue*

No	User	User Story
1	Mahasiswa Koas	Sebagai seorang mahasiswa koas, saya ingin dapat membuat daftar tugas harian, mingguan, atau per stase dalam aplikasi untuk mengatur jadwal studi saya.
		Sebagai seorang mahasiswa koas, saya ingin dapat memprioritaskan tugas berdasarkan tingkat pentingnya atau urgensi, sehingga saya tahu mana yang harus diselesaikan terlebih dahulu.
		Sebagai seorang mahasiswa koas, saya ingin dapat menandai tugas yang sudah selesai dalam aplikasi agar saya dapat melihat kemajuan saya.
		Sebagai seorang mahasiswa koas, saya ingin aplikasi memberikan pengingat atau notifikasi untuk tugas yang akan datang atau tenggat waktu yang mendekat.
		Sebagai seorang mahasiswa koas, saya ingin melihat riwayat tugas yang telah saya selesaikan dalam aplikasi agar saya dapat memantau kemajuan studi saya sepanjang waktu.
		Sebagai seorang mahasiswa koas, saya ingin menerima poin atau penghargaan virtual setiap kali saya menyelesaikan tugas, yang dapat digunakan untuk tujuan gamifikasi.
		Sebagai seorang mahasiswa koas, saya ingin aplikasi memiliki sistem peringkat yang memungkinkan saya bersaing dengan rekan mahasiswa lainnya.
		2
Sebagai seorang dokter pembimbing, saya ingin aplikasi dapat membuat tugas yang akan dikerjakan oleh mahasiswa-mahasiswa koas berdasarkan stase yang ditempati.		
Sebagai seorang dokter pembimbing, saya ingin memiliki akses ke daftar tugas stase mahasiswa koas saya melalui aplikasi ini, sehingga saya dapat memantau kemajuan mereka.		

Tabel 3.3. *Story Queue* (lanjutan)

No	User	User Story
2	Dokter Pembimbing	Sebagai seorang dokter pembimbing, saya ingin memiliki opsi untuk menilai tugas yang telah diselesaikan oleh mahasiswa koas saya dalam aplikasi ini.

3.4.3. Analysis

Tahap ini akan menganalisis dan memilih *user story* mana yang akan dikembangkan menjadi fitur-fitur pada proyek dan akan dikelompokkan berdasarkan kebutuhan fungsional dan kebutuhan non-fungsional. *User story* atau *product backlog* tersebut juga akan dianalisis berdasarkan prioritas setiap *backlog* yang ada, sehingga *backlog* yang memiliki prioritas utama akan memasuki kolom *Work On Wait* atau *To Do* seperti yang diterapkan dalam metode *Kanban*.

a. Kebutuhan Fungsional

Kebutuhan fungsional dari sistem adalah sebagai berikut:

Tabel 3.4. Kebutuhan Fungsional Aplikasi

No	Fitur	Deskripsi
1	Autentikasi	Pada aplikasi menyediakan proses <i>login</i> pengguna menggunakan email dan kata sandi. Selain itu aplikasi juga menyediakan proses <i>logout</i> untuk pengguna.
2	Melihat Daftar Tugas	Pengguna, baik mahasiswa koas ataupun dokter pembimbing dapat melihat daftar tugas yang telah mereka buat. Daftar tugas ini mencakup tugas-tugas yang masih tertunda maupun yang sudah selesai.
3	Membuat Tugas	Pengguna dapat membuat tugas baru dalam aplikasi. Pengguna dapat memberikan judul, deskripsi, tenggat waktu, stase, dan informasi lainnya terkait tugas yang akan mereka kerjakan.
4	Melihat Riwayat Tugas	Pengguna dapat melihat riwayat tugas yang telah mereka selesaikan. Pengguna dapat memeriksa tugas-tugas yang sudah ditandai sebagai selesai serta detailnya.

Tabel 3.4. Kebutuhan Fungsional Aplikasi (lanjutan)

No	Fitur	Deskripsi
5	Mendapatkan Notifikasi	Pengguna dapat menerima pemberitahuan atau notifikasi dalam aplikasi. Notifikasi dapat berupa pengingat tenggat waktu tugas, pemberitahuan tentang perubahan dalam tugas, atau informasi lain yang relevan.
6	Melihat Peringkat	Peringkat adalah fitur yang menampilkan peringkat atau ranking pengguna, seperti mahasiswa koas, berdasarkan pencapaian mereka dalam menggunakan aplikasi. Ini dapat melibatkan pemberian poin atau penghargaan kepada pengguna yang telah menyelesaikan banyak tugas atau mencapai tujuan tertentu dalam permainan gamifikasi.
7	Mengelola Stase	Stase adalah unit pembelajaran atau rotasi dalam program profesi dokter. Fitur ini memungkinkan dokter pembimbing untuk mengelola stase-stase yang terdiri dari para mahasiswa koas. Dokter pembimbing juga dapat mengatur stase-stase ini sesuai dengan jadwal, dan menambahkan tugas kepada para mahasiswa koas.
8	Melihat Status Tugas Mahasiswa	Fitur ini memberikan visibilitas kepada dokter pembimbing untuk melihat status tugas-tugas yang telah diberikan kepada mahasiswa koas. Dokter pembimbing dapat melihat apakah tugas-tugas tersebut sudah dikerjakan atau belum, serta melacak kemajuan mahasiswa koas dalam program mereka.

b. Kebutuhan Non-Fungsional

Kebutuhan non-fungsional dari sistem adalah sebagai berikut:

Tabel 3.5. Kebutuhan Non-Fungsional Aplikasi

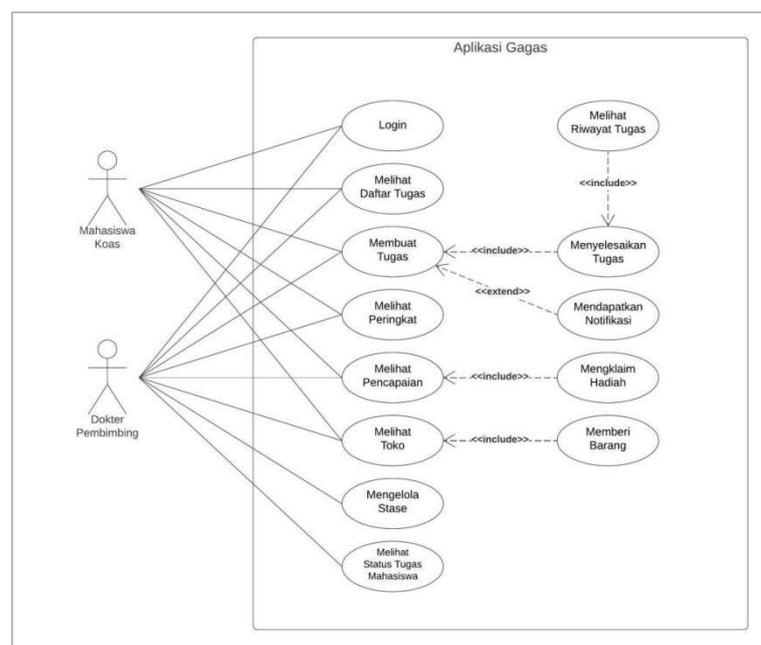
No	Aspek	Deskripsi
1	Kinerja (<i>Performance</i>)	Waktu Respons: Aplikasi harus merespon dengan cepat terhadap tindakan pengguna, seperti menambahkan atau mengedit tugas.

Tabel 3.5. Kebutuhan Non-Fungsional Aplikasi (lanjutan)

No	Aspek	Deskripsi
1	Kinerja (<i>Performance</i>)	Skalabilitas: Aplikasi harus mampu menangani jumlah pengguna yang berpotensi besar dan jumlah tugas yang signifikan tanpa mengalami penurunan kinerja yang signifikan.
2	Kegunaan (<i>Usability</i>)	Antarmuka Pengguna yang Ramah Pengguna: Memastikan antarmuka pengguna mudah digunakan dan intuitif. Pengguna harus dapat dengan mudah mengakses dan menggunakan fitur-fitur utama aplikasi.
3	Kinerja Perangkat Khusus	Optimalisasi Perangkat Bergerak: Memastikan aplikasi berjalan dengan baik pada perangkat mobile yang umum digunakan oleh mahasiswa koas.

c. Use Case Diagram

Use case diagram merupakan model diagram UML yang menunjukkan interaksi antar sistem atau dengan pengguna dari sistem. Sehingga dengan melihat *use case diagram* kita dapat mengetahui bagaimana suatu sistem akan bekerja. *Use case diagram* dari sistem ini dapat dilihat pada Gambar 3.2.



Gambar 3.2. Use Case Diagram

3.4.4. *Development*

Pada tahap ini dilakukan proses pengkodean dari aplikasi yang akan dikembangkan. *User queue* yang telah dianalisis dan dipilih pada tahap sebelumnya akan ditempatkan pada kolom *Work In Progress* dan dikerjakan dengan tetap memberikan limit atau batasan dalam jumlah *Work In Progress* yang dikerjakan. Menurut Paul dan Rahman pada jurnalnya yang berjudul “*Study on Agile Management in Construction Project Using Scrumban Methodology*”, metode *Scrumban* memiliki beberapa prinsip dalam proses pengembangannya, antara lain sebagai berikut:

1. *Visualize The Workflow*

Ini adalah salah satu tools yang diambil dari Kanban dan diterapkan pada *Scrumban*. Memvisualisasikan alur kerja pada sebuah *Scrumban Board* dari awal hingga akhir proses pengembangan. Dalam penelitian ini sendiri *Scrumban board* akan digambarkan menggunakan *tools* bernama *Trello*.

2. *Pull Work*

Pada metode *Scrumban*, pekerjaan yang telah siap akan ditarik ke dalam antrian, hal tersebut tidak seperti *Scrum* yang semua pekerjaan harus diselesaikan dengan *sprint* dan ditugaskan dari awal.

3. *Limit Work In Progress (WIP)*

Prinsip yang paling penting dalam *Scrumban* adalah menerapkan pembatasan dalam pekerjaan yang sedang berjalan pada tiap kapasitas di tim.

4. *Make Team Rules Explicit*

Dalam *scrum* setiap tim terorganisir sendiri dan mereka akan bekerja dalam mengordinasikan diri mereka sendiri, namun pada kenyataannya selalu ada kesenjangan yang terjadi pada tim, bagaimana mereka mengatur diri sendiri dan bekerja.

5. *Planning Meetings*

Tidak seperti *Scrum*, *Scrumban* mempunyai perencanaan yang singkat untuk memperbarui antrian *backlog* jika diperlukan.

6. *Review, Retrospectives and Daily Stand-up Meetings*

Dengan memberikan *feedback* secara langsung dari *product owner* atau *stakeholder* dengan tim seperti *managers* dan *customers*.

7. *Metrics and Optional Estimations*

Di dalam scrum diperkirakan menggunakan metrik seperti poin *story* dan jumlah tugas yang diambil dan dimasukkan ke dalam sprint berdasarkan kecepatan rata-rata tim dalam beberapa *sprint* terakhir [21].

3.4.5. *Test*

Pengujian atau *test* bertujuan untuk menguji apakah *story queue* yang telah dikembangkan telah berjalan dengan baik. Tahap pengujian ini dilakukan menggunakan metode *Unit Testing* dan *Usability Testing*. Pengujian dilakukan untuk menemukan kesalahan-kesalahan yang mungkin terjadi sehingga dapat dilakukan perbaikan.

3.4.6. *Deployment*

Tahap ini merupakan hasil dari semua *story queue* yang telah berhasil diuji dan digabungkan menjadi suatu aplikasi yang utuh. Pada tahap ini dilakukan dengan tujuan memudahkan penggunaan aplikasi dan kustomisasi *User Interface* (UI) aplikasi. Aplikasi akan di-*deploy* dan siap untuk digunakan user sebagai suatu *prototype* yang nyata.

3.4.7. *Done*

Tahap ini adalah tahap di mana semua tujuan pada proyek telah selesai serta akan dikumpulkan semua data dan dokumentasi selama pengembangan. Pada tahap ini juga terdapat penyusunan laporan hasil dari penelitian mengenai Perancangan Aplikasi *To-Do List* pada Mahasiswa Pendidikan Profesi Dokter Berbasis Gamifikasi. Hasil temuan yang ada kemudian digunakan sebagai skripsi pada Universitas Lampung.

IV. KESIMPULAN DAN SARAN

5.1. Kesimpulan

Kesimpulan yang diperoleh berdasarkan hasil dari penelitian adalah sebagai berikut:

1. Penelitian yang dilakukan telah mengembangkan aplikasi berbasis android yang dapat membantu mahasiswa koas dan dosen untuk dapat mengelola tugas stase dan tugas mandiri dengan menerapkan metode gamifikasi.
2. Pada proses pengembangan aplikasi *to-do list* (Gagas) menggunakan metode *Scrumban* terdapat 74 *backlog* yang merupakan hasil dari *breakdown* dari 11 *user story* pada *story queue* dan pengerjaannya dilakukan sebanyak 16 *sprint* dengan penyesuaian *Scrumban Board* dengan waktu pengembangan selama 195 hari.
3. Hasil pengujian menggunakan metode *Unit Testing* dengan total 9 skenario dan *Black Box Testing* dengan total 54 skenario pengujian didapatkan bahwa aplikasi berhasil berjalan sesuai dengan kebutuhan fungsional serta kebutuhan non-fungsional yang telah didefinisikan. Berdasarkan dengan tingkat persentase 100% untuk semua skenario, aplikasi ini berhasil dikembangkan dan berjalan dengan baik.
4. Hasil pengujian *Usability Testing* menggunakan UEQ memiliki nilai untuk daya tarik, efisiensi, ketepatan, stimulasi, dan kebaruan yang rata-rata berada di atas 1 dan nilai untuk kejelasan berada di atas 0.8. Dari grafik yang ditampilkan, lima dari enam skala berada di area hijau, sementara satu skala berada di area jingga, yang mengindikasikan bahwa hasilnya masuk dalam kategori baik.

5.2. Saran

Saran yang dapat diberikan untuk melanjutkan penelitian ini berdasarkan hasil penelitian yang telah dilakukan adalah sebagai berikut:

1. Menambahkan fitur agar dosen dapat memilah mahasiswa koas tertentu saja yang mendapatkan tugas stase, sehingga tidak perlu menyertakan semua anggota kelompok
2. Menambahkan fitur berupa *upload file* yang bisa dilampirkan pada tugas stase maupun tugas mandiri.
3. Menambahkan fitur misi harian, mingguan, dan bulanan untuk dapat memicu mahasiswa secara terus menerus menggunakan aplikasi secara rutin.
4. Menambahkan fitur yang berkaitan dengan *artificial intelligence* (AI) seperti konfirmasi menyelesaikan tugas dengan *face recognition*, rekomendasi tugas berdasarkan kebiasaan pengguna ataupun pengingat cerdas.
5. Menambahkan *role* admin yang dapat menjalankan fitur-fitur di luar fitur tugas seperti menukarkan hadiah.
6. Melakukan pengembangan untuk versi iOS sehingga dapat menjangkau mahasiswa koas dan dosen yang menggunakan perangkat iPhone dan iPad.

DAFTAR PUSTAKA

- [1] Pemerintah Republik Indonesia, “Undang-Undang Republik Indonesia Nomor 12 Tahun 2012 Tentang Pendidikan Tinggi”, Jakarta: Sekretariat Negara, 2012.
- [2] Badan Legislasi DPR RI, “Rancangan Undang Undang Tentang Pendidikan Kedokteran”, Jakarta, 2021.
- [3] A. A. Karina, “Analisis Dan Perancangan User Interface Dan User Experience Aplikasi Mobile E-Learning Berbasis Gamifikasi Dengan Metode Design Thinking (Studi Kasus : Program Profesi Dokter Universitas Lampung)”, Universitas Lampung, Lampung, 2023.
- [4] C. N. Hanafi dan Y. Widjaja, “Kemampuan manajemen waktu mahasiswa tahap profesi Fakultas Kedokteran Universitas Tarumanagara”, *Tarumanagara Medical Journal*, vol. 3, no. 1, hal. 18-28, 2021, [Online]. Tersedia di: <https://doi.org/10.24912/tmj.v3i2.11740>
- [5] N. Ftriah, “Manajemen Waktu Belajar Mahasiswa Keperawatan Dalam Melaksanakan Metode Problem Based Learning (PBL) di Universitas Islam Negeri Syarif Hidayatullah Jakarta”, Universitas Islam Negeri Syarif Hidayatullah, Jakarta, 2014.
- [6] T. Santhi, dkk, “Pengembangan Sistem Informasi To Do List Berbasis Website dalam Meninjau Kegiatan Mahasiswa Undiksha”, *Jurnal Teknologi Ilmu Komputer*, vol. 1, no. 1, hal. 30-36, 2022.
- [7] A. C. T. Klock, dkk, “Gamification in e-Learning Systems: A Conceptual Model to Engage Students and Its Application in an Adaptive e-Learning System”, *Springer International Publishing Switzerland*, vol. 9192, hal.

- 595–607, 2015, [Online]. Tersedia di: https://doi.org/10.1007/978-3-319-20609-7_56.
- [8] I. E. C. Putri, “Wewenang dan Tanggung Jawab Co-Ass dalam Pelayanan Kesehatan Terhadap Pasien di Rumah Sakit”, Universitas Negeri Semarang, Semarang, 2017.
- [9] R. Akbar dan T. Ridwan, “Perancangan Aplikasi Mobile To-Do List Sebagai Platform Delegasi Tugas Secara Remote pada Masa Pandemic Di Pt. Pupuk Kujang Cikampek”, *Jurnal INTEGRATED (Information Technology and Vocational Education)*, vol. 3, no. 1, hal. 9-16, 2021.
- [10] A. M. Toda, dkk, “User-Centered Gamification for E-Learning Systems: A Quantitative and Qualitative Analysis of its Application”, *Interacting with Computers*, vol. 31, no. 5, hal. 425–445, 2019, [Online]. Tersedia di: <https://doi.org/10.1186/s40561-019-0106-1>.
- [11] H. Jusuf, “Penggunaan Gamifikasi dalam Proses Pembelajaran”, *Jurnal TICOM*, vol. 5, no. 1, hal. 1–6, 2016.
- [12] D. Strmečki, A. Bernik dan D. Radošević, “Gamification in e-learning: Introducing gamified design elements into e-learning systems”, *Journal of Computer Science*, vol. 11, no. 12, hal. 1108–1117, 2015, [Online]. Tersedia di: <https://doi.org/10.3844/jcssp.2015.1108.1117>.
- [13] Android, “What is Android: The platform changing what mobile can do”, [Online]. Tersedia di: <https://www.android.com/what-is-android> (diakses 28 Agustus 2023).
- [14] Flutter, “Flutter Documentation”, [Online]. Tersedia di: <https://docs.flutter.dev> (diakses 22 Agustus 2023).
- [15] Freitas, Ed. “Flutter Succinctly”, [Online]. Tersedia di: <https://www.syncfusion.com/succinctly-free-ebooks/flutter-succinctly> (diakses 15 Mei 2024).
- [16] Laravel, “Laravel Documentation”, [Online]. Tersedia di: <https://laravel.com/docs/11.x> (diakses 27 Mei 2024).
- [17] Y. Yudhanto dan H. A. Prasetyo, *Mudah Menguasai framework Laravel*, Jakarta: PT Alex Media Komputindo, 2019.

- [18] A. R. Fajri, “Penerapan Design Pattern MVVM dan Clean Architecture Pada Pengembangan Aplikasi Android (Studi Kasus: Aplikasi Agree)”, Universitas Islam Indonesia, Yogyakarta, 2022
- [19] W. Sheikh dan N. Sheikh, “A Model-View-ViewModel (MVVM) Application framework for Hearing Impairment Diagnosis”, Preprint arXiv, arXiv:1911.08289, 2019.
- [20] N. Lutfiani, dkk, “Inovasi Manajemen Proyek I-Learning Menggunakan Metode Agile Scrumban”, *InfoTekJar: Jurnal Nasional Informatika dan Teknologi Jaringan*, vol. 5, no. 1, hal. 96-101, 2020, [Online]. Tersedia di: <https://doi.org/10.30743/infotekjar.v5i1.2848>.
- [21] A. J. Paul dan S. K. Rahman, “Study on Agile Management in Construction Project Using Scrumban Methodology”, *International Research Journal of Engineering and Technology (IRJET)*, vol. 5, no. 11, hal. 774–777, 2018, [Online]. Tersedia di: <https://www.irjet.net/archives/V5/i11/IRJET-V5I11148.pdf>.
- [22] A. G. Hartzani, “Evaluasi User Experience pada Dompot Digital Ovo Menggunakan User Experience Questionnaire (UEQ)”, Universitas Islam Negeri Syarif Hidayatullah, Jakarta, 2021.
- [23] M. Schrepp, “User Experience Questionnaire Handbook”, *UEQ Online*, ver. 11, 2023, [Online]. Tersedia di: <https://www.ueq-online.org/Material/Handbook.pdf>.
- [24] M. H. Ohn, K. M. Ohn, U. D. Souza, S. Yusof, dan Z. Ariffin, “Effectiveness of innovative gamified learning among undergraduate medical students”, *Journal of Physics: Conference Series, Institute of Physics Publishing*, vol. 1358, 2019, [Online]. Tersedia di: <https://doi.org/10.1088/1742-6596/1358/1/012060>.
- [25] R. Andriamiarisoa, “Impact of Gamification on Student Engagement in Graduate Medical Studies”, Walden University, Minneapolis, [Online]. Tersedia di: <https://scholarworks.waldenu.edu/dissertations>.
- [26] A. A. P. Mathias, “Pembangunan Tampilan Antarmuka Sistem Informasi Logbook Koas Program Studi Pendidikan Dokter Fakultas Kedokteran Universitas XYZ”, Universitas Atma Jaya Yogyakarta, Yogyakarta, 2023.

- [27] M. I. Abdullah, “Pengembangan Aplikasi To Do List dalam Meningkatkan Produktivitas Mahasiswa Universitas Brawijaya Pada Masa Pandemi”, [Online]. Tersedia di: <https://doi.org/10.1038/s41562-019-0672-9/>.
- [28] R. Azmi dan Rahmawati, “Perancangan Aplikasi *ToDo List* Berbasis *Android* Menggunakan *Flutter SDK* dan *Database SQLite*”, dalam Seminar Nasional Riset dan Teknologi (SEMNAS RISTEK), hal. 567-572, Jakarta, 2021.
- [29] Yodiz, “Scrumban - An amalgamation of Scrum and Kanban”, 2015, [Online]. Tersedia di: <https://www.yodiz.com/blog/scrumban-an-amalgamation-of-scrum-and-kanban> (diakses 31 Agustus 2023).