

BAB 3 METODOLOGI PENELITIAN

3.1 Metodologi Penelitian

1. Studi literatur

Tahap awal penelitian ini difokuskan pada studi literatur yang komprehensif. Tujuannya adalah untuk mengumpulkan dan menganalisis informasi dan pengetahuan yang relevan dengan topik penelitian. Sumber utama yang digunakan adalah sumber tertulis terpercaya seperti buku dan jurnal ilmiah. Melalui studi literatur ini, diharapkan dapat memahami metode dan teori-teori yang mendasari penelitian, serta mengidentifikasi permasalahan yang ingin dikaji.

2. Perancangan aplikasi dan antarmuka

Perancangan aplikasi meliputi 3 hal, yaitu perancangan *user interface*, perancangan *database*, dan pembuatan *flowchart*. Aplikasi ini dibuat berdasarkan *framework* gamifikasi bernama Octalysis.

3. Implementasi aplikasi

Aplikasi berbasis web ini dibuat dengan mengubah hasil desain yang telah dibuat selama proses perancangan aplikasi. Aplikasi ini dibuat menggunakan *library* ReactJS dan bahasa pemrograman TypeScript.

4. Pengujian aplikasi

Metode evaluasi aplikasi ini menggunakan *black box testing* yang melibatkan partisipasi pengguna langsung. Pengguna akan diminta untuk menguji aplikasi dan memberikan umpan balik melalui survei pada Google Form yang dirancang dengan pernyataan yang relevan dengan aplikasi. Tujuan utama pengujian ini adalah untuk mendapatkan masukan berharga dari pengguna dan memastikan fungsionalitas aplikasi yang optimal.

5. Evaluasi

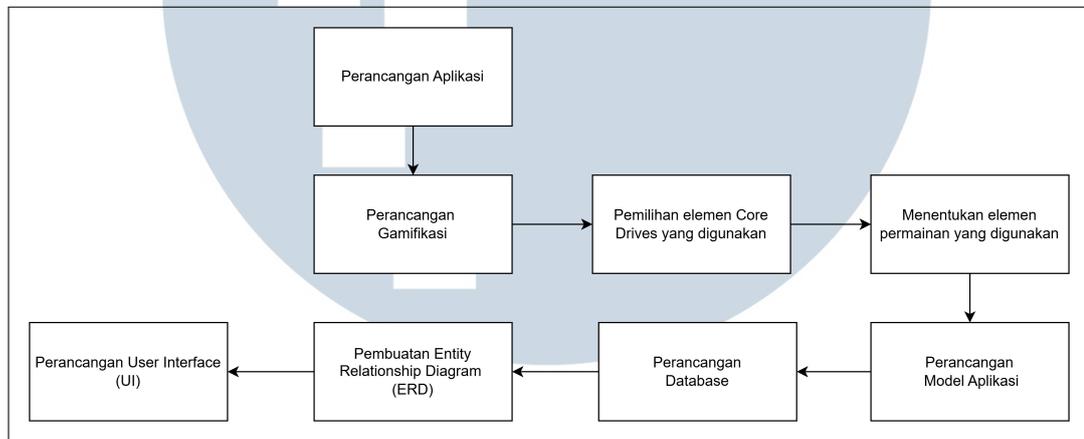
Pada tahapan evaluasi, hasil survei yang telah dilakukan pada beberapa pengguna akan dianalisis dan di evaluasi menggunakan skala likert yang bertujuan untuk mengukur nilai *Perceived Usefulness* dan *Perceived Ease*

of Use yang ada pada model *Technology Acceptance Model* (TAM). Hal ini berguna untuk mengetahui persentase kepercayaan pengguna dan apakah aplikasi ini mudah untuk digunakan oleh siapa pun.

6. Penulisan laporan

Setelah semua proses penelitian telah dilakukan, hasil penelitian akan ditulis ke dalam laporan skripsi berdasarkan hasil yang telah didapat.

3.2 Perancangan Aplikasi



Gambar 3.1. Flowchart perancangan aplikasi

Proses perancangan aplikasi pembelajaran bahasa pemrograman JavaScript ini akan dilakukan dalam beberapa tahapan, yaitu dimulai dengan perancangan gamifikasi, pembuatan model aplikasi, pembuatan flowchart sistem, pembuatan *Entity Relationship Diagram* (ERD), dan perancangan antarmuka. Setiap proses akan dijelaskan secara mendalam.

3.2.1 Perancangan Gamifikasi

Dalam perancangan aplikasi ini, digunakan salah satu *framework* gamifikasi yang dikembangkan oleh seorang peneliti bernama Yu-Kai Chou bernama Octalysis.

Pada penelitian ini, hanya empat elemen dari *framework* Octalysis yang digunakan karena berdasarkan studi kasus penggunaan Octalysis yang ditulis oleh Yu-Kai Chou[28], tidak semua elemen *Core Drives* wajib digunakan dan

hanya perlu menyesuaikan dengan tipe pengguna dan memahami motivasi mereka. Sehingga, ke empat elemen yang digunakan pada penelitian ini, antara lain:

1. *Development & Accomplishment*

Elemen ini berfokus pada keinginan pemain untuk belajar dan berkembang. Hal ini bisa dicapai dengan memberikan poin, dan *experience* sebagai hadiah ketika pemain berhasil menyelesaikan setiap modul dari sebuah kelas sehingga pemain termotivasi untuk terus melanjutkan ke modul-modul dan kelas yang ada.

2. *Scarcity & Impatience*

Elemen ini berfokus pada keinginan untuk merasakan hal yang baru dan sesuatu yang tidak bisa dimiliki oleh pemain. Perasaan tidak dapat memperoleh sesuatu akan memotivasi pemain untuk memikirkannya sepanjang hari. Hal ini dapat dicapai dengan memberikan durasi pada setiap tantangan (*daily challenge*) yang diikuti pemain.

3. *Unpredictability & Curiosity*

Elemen ini memiliki tujuan untuk membuat pemain ingin mengetahui apa yang akan terjadi selanjutnya, sehingga otak dari pemain akan terus berpikir akan hal tersebut. Pada aplikasi ini, nantinya pemain dapat mengakses tantangan harian (*daily challenge*) tetapi pemain tidak akan mengetahui tingkat kesusahan, topik tantangan, dan durasi pengerjaan tantangan tersebut karena tantangan akan dipilih secara acak dari *database*.

4. *Loss & Avoidance*

Elemen ini berfokus memberikan dorongan pada pemain untuk dapat menghindari sesuatu yang buruk. Contohnya, menghindari adanya kemungkinan kehilangan *progress* dari aktivitas yang telah dilakukan sebelumnya. Pada aplikasi ini, setiap pemain yang tidak mengerjakan tantangan harian (*daily challenge*) dalam waktu 24 jam, maka mereka akan kehilangan *streaks* yang mereka miliki.

Setelah dilakukan pemilihan elemen *Core Drives*, kemudian dilakukan pemilihan elemen permainan yang akan diterapkan pada aplikasi ini. Total ada 8 elemen permainan yang di implementasi pada aplikasi ini, antara lain:

1. Level

Elemen ini merupakan elemen permainan dasar. Level akan bergantung pada jumlah *experience* (XP) yang terkumpul oleh masing-masing pengguna. Semakin tinggi level pengguna, maka jumlah *experience* yang menjadi syarat untuk naik ke level selanjutnya semakin tinggi juga. Sekarang, level paling tinggi yang bisa dicapai oleh seorang pemain adalah level 50. Jumlah *experience* yang diperlukan setiap level akan naik sebanyak 200 *experience*. Rincian level dan jumlah *experience* dapat dilihat pada tabel 3.1.

Tabel 3.1. Sistem Level

Level	Jumlah <i>Experience</i>
1	0 XP
2	200 XP
3	400 XP
4	600 XP
5	800 XP
6	1000 XP
7	1200 XP
8	1400 XP
9	1600 XP
10	1800 XP
11	2000 XP
12	2200 XP
13	2400 XP
14	2600 XP
15	2800 XP
16	3000 XP
17	3200 XP
18	3400 XP
19	3600 XP
20	3800 XP
21	4000 XP
22	4200 XP
Lanjut pada halaman berikutnya	

Tabel 3.1 Interval Nilai dan Kategori (lanjutan)

Level	Jumlah <i>Experience</i>
23	4400 XP
24	4600 XP
25	4800 XP
26	5000 XP
27	5200 XP
28	5400 XP
29	5600 XP
30	5800 XP
31	6000 XP
32	6200 XP
33	6400 XP
34	6600 XP
35	6800 XP
36	7000 XP
37	7200 XP
38	7400 XP
39	7600 XP
40	7800 XP
41	8000 XP
42	8200 XP
43	8400 XP
44	8600 XP
45	8800 XP
46	9000 XP
47	9200 XP
48	9400 XP
49	9600 XP
50	9800 XP

2. *Points*

Points merupakan elemen permainan yang juga diimplementasikan pada aplikasi ini. Jumlah *points* yang dimiliki oleh pengguna dengan cara mereka

menyelesaikan suatu modul pembelajaran, mendapatkan *achievements*, atau menyelesaikan *daily challenge*. Nantinya, *points* akan berguna untuk digunakan pada elemen permainan selanjutnya, yaitu *leaderboard*.

3. *Leaderboard*

Elemen permainan satu ini merupakan elemen yang memperlihatkan peringkat semua pengguna berdasarkan jumlah *points* yang dimiliki setiap pengguna. Nantinya, pada *leaderboard* semua pengguna akan ditampilkan dalam bentuk tabel beserta dengan akumulasi jumlah *points*.

4. *Achievements*

Achievements merupakan elemen permainan berupa penghargaan ketika pengguna berhasil mencapai sesuatu. Setiap *achievements* yang ada, memiliki syarat tersendiri untuk bisa didapatkan oleh pengguna. Setiap *achievements* memiliki hadiah berupa penambahan *points* dan *experience*. Rincian dari setiap *achievements* dapat dilihat pada tabel 3.2.

Tabel 3.2. Tabel Achievements

Nama	Syarat	Points	Experiences
Hello, World	Menyelesaikan materi pembelajaran berjudul "Pengenalan JavaScript"	120 <i>points</i>	200 XP
JavaScript Wizard	Menyelesaikan materi pembelajaran berjudul "Pengembangan Web dengan ReactJS"	415 <i>points</i>	1000 XP
Lanjut pada halaman berikutnya			

Tabel 3.2 Tabel Achievements (lanjutan)

Nama	Syarat	Points	Experiences
Code Connoisseur	Menyelesaikan materi pembelajaran berjudul "Best Practices pada JavaScript"	275 <i>points</i>	300 XP
Challenge Accepted	Menyelesaikan <i>daily challenge</i> untuk pertama kalinya	75 <i>points</i>	100 XP
Daily Challenger	Memiliki <i>streaks</i> berjumlah 5	300 <i>points</i>	500 XP
Challenge the Titan	Memiliki <i>streaks</i> berjumlah 15	1000 <i>points</i>	2000 XP
Challenge Enthusiast	Menyelesaikan total 10 <i>daily challenge</i>	750 <i>points</i>	1500 XP

5. *Progress tracking*

Progress tracking merupakan salah satu elemen permainan yang berguna untuk melihat kemajuan pengguna pada pengerjaan aktivitas tertentu. Pada aplikasi JStudy, elemen ini digunakan untuk melihat kemajuan pembelajaran pengguna akan suatu modul pembelajaran yang sedang diikuti dengan menampilkan persentase penyelesaian modul.

6. *Rank*

Elemen permainan satu ini merupakan elemen untuk menilai kemampuan pengguna. Semakin tinggi *rank* seorang pengguna, maka pengguna dianggap memiliki performa belajar yang bagus dan pengetahuan yang lebih luas mengenai materi pembelajaran yang telah dipelajari. *Rank* pengguna akan

dipengaruhi oleh level yang dimiliki pada saat itu. Semakin tinggi level, maka semakin tinggi juga *rank* yang dicapai oleh pengguna. Rincian *rank* dapat dilihat pada tabel 3.3.

Tabel 3.3. Tabel Rank

Level	Rank
1 - 9	Beginner
10 - 29	Intermediatee
30 - 49	Advanced
50	Master

7. Challenge

Challenge merupakan salah satu elemen permainan yang berguna untuk melatih dan menguji kemampuan pengguna terhadap materi pembelajaran yang telah mereka pelajari. *Challenge* pada aplikasi ini hanya dapat diakses sebanyak sekali dalam sehari dan akan berpengaruh pada elemen permainan bernama *streaks*. Setiap *challenge* yang disediakan akan menyesuaikan dengan materi yang ada pada aplikasi pembelajaran.

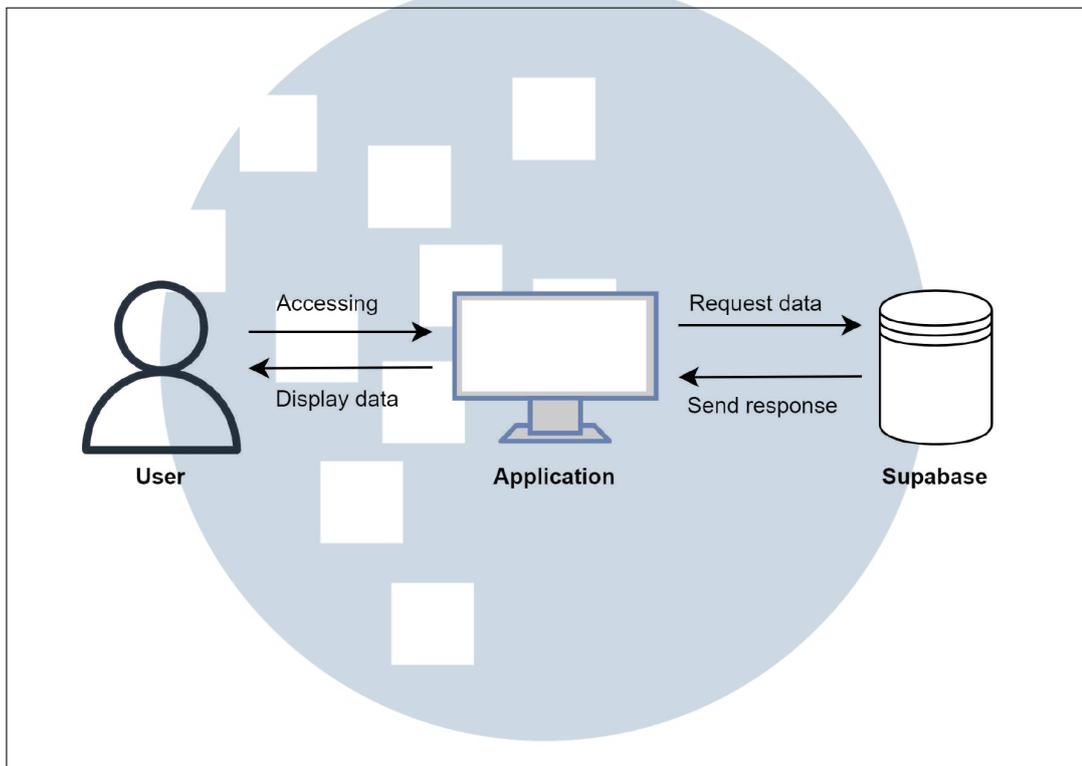
8. Streak

Streak merupakan elemen permainan di mana elemen ini akan menghitung berapa kali pengguna berhasil menyelesaikan tantangan secara berturut-turut. Jumlah *streak* dapat di-*reset* menjadi 0 ketika pengguna gagal menyelesaikan tantangan atau tidak mengerjakan tantangan dalam kurun waktu 24 jam.

3.2.2 Model Aplikasi

Pada aplikasi pembelajaran JavaScript ini, akan ada skenario di mana pengguna melakukan aktivitas pada aplikasi, lalu aplikasi akan langsung mengirimkan suatu *request* kepada server *database* yang digunakan, yaitu Supabase. Setelah *request* tersebut diterima oleh *server*, nanti data-data yang didapatkan akan dikirim kembali ke aplikasi dalam bentuk *response* yang kemudian akan ditampilkan pada aplikasi sehingga dapat dilihat pengguna seperti level, *rank*,

poin, *achievements*, dan kelas. Skenario ini dapat diilustrasikan seperti pada Gambar 3.2.



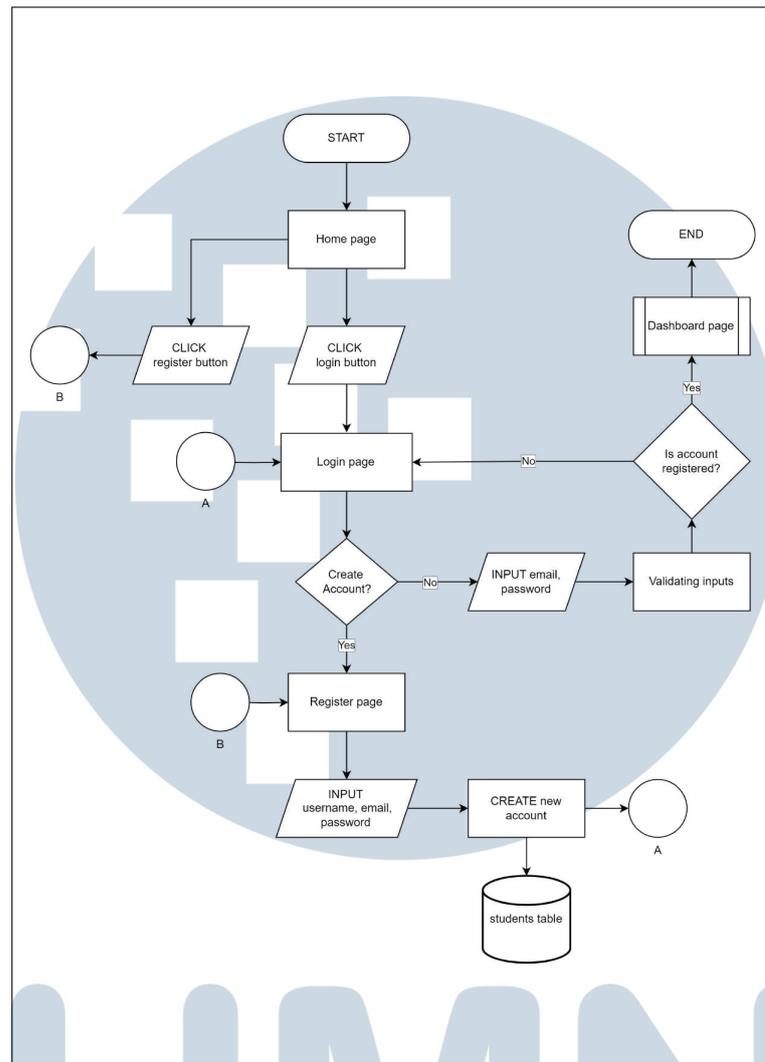
Gambar 3.2. Model aplikasi pembelajaran bahasa pemrograman JavaScript

3.2.3 Flowchart

Tujuan dari desain *flowchart* adalah untuk menjelaskan alur kerja aplikasi JStudy sebagai media pembelajaran JavaScript yang menggunakan gamifikasi.

A. Flowchart halaman Login dan Register

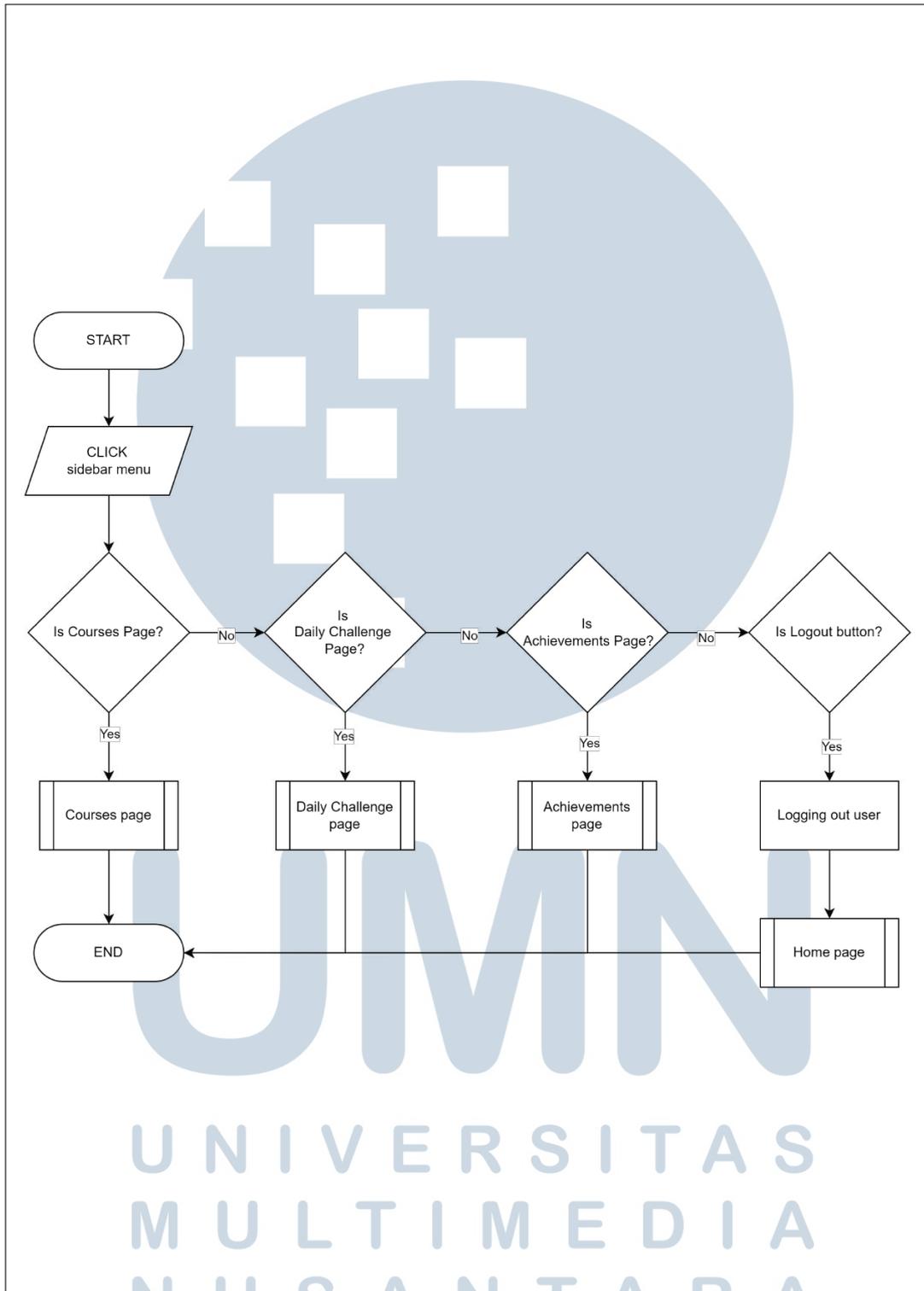
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.3. Flowchart halaman Login dan Register

Flowchart utama dari aplikasi JStudy ditunjukkan pada Gambar 3.3. Saat membuka aplikasi, ada dua tombol: "Masuk" dan "Daftar". Jika menekan tombol "Masuk", pengguna akan diarahkan ke halaman *login*. Di sana, mereka diminta untuk memasukkan *email* dan *password* mereka untuk memastikan bahwa data yang mereka masukkan benar. Jika data yang mereka masukkan benar, mereka akan dapat masuk ke halaman *dashboard*. Jika data yang mereka masukkan benar, prosesnya dapat dilanjutkan.

B. Flowchart halaman Dashboard

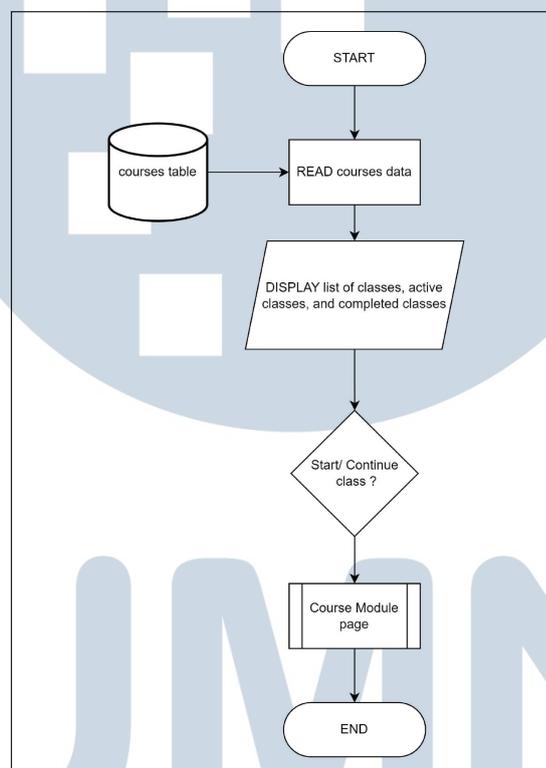


Gambar 3.4. Flowchart halaman Dashboard

Flowchart dari halaman *Dashboard* dapat dilihat pada Gambar 3.4. Pada *Dashboard* terdapat *sidebar* yang mengarahkan ke halaman lain, seperti

Courses, *Daily Challenge*, dan *Achievements*. Selain memungkinkan navigasi ke halaman lain, ada juga tombol *logout* yang ketika ditekan maka pengguna akan keluar dari akun nya sehingga mengharuskan pengguna untuk *login*, alurnya dilihat pada Gambar 3.3. Ketika setiap navigasi ditekan oleh pengguna, pengguna akan diarahkan ke halaman yang sesuai. Contohnya, ketika pengguna menekan navigasi "Courses" maka pengguna akan diarahkan ke halaman Courses yang *flowchart* seperti pada Gambar 3.5.

C. Flowchart halaman Courses

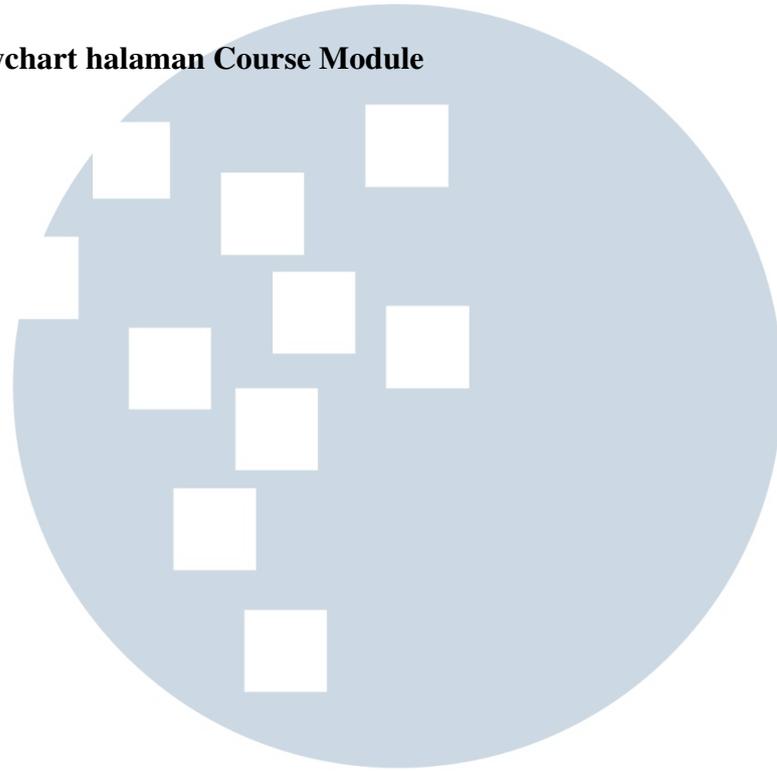


Gambar 3.5. Flowchart halaman Courses

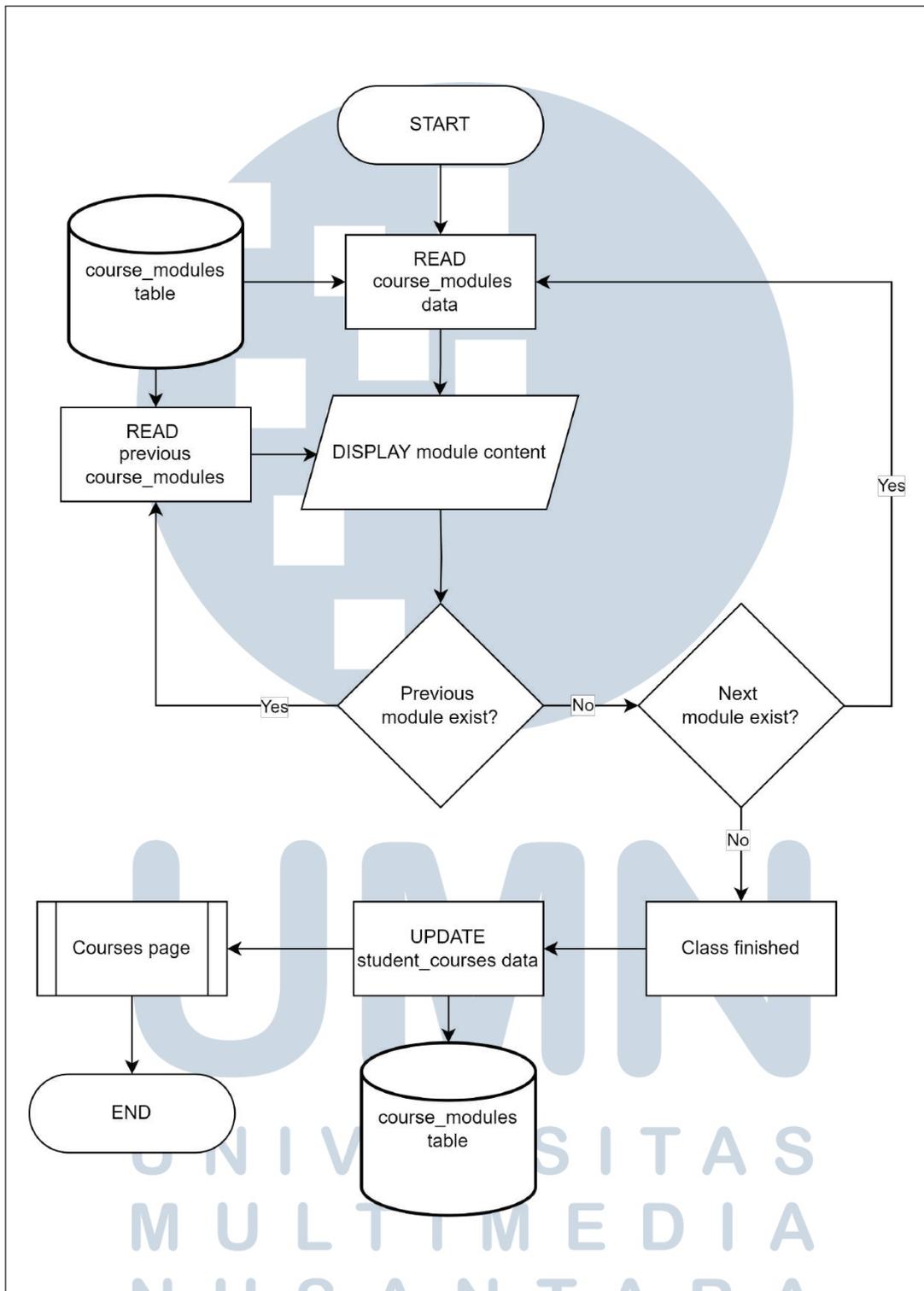
Flowchart dari halaman *Courses* ditunjukkan pada Gambar 3.5. Ketika pengguna memasuki halaman ini, pertama-tama akan terjadi pengambilan data *courses* yang didapat dari tabel *courses* pada *database*. Setelah pengambilan data selesai dilakukan, akan ditampilkan daftar kelas yang tersedia, kelas aktif (kelas yang telah dimulai oleh pengguna), dan kelas selesai (kelas yang telah diselesaikan oleh pengguna). Saat pengguna menekan tombol "Mulai kelas" pada daftar kelas, pengguna akan memasuki halaman *Course Module* yang alurnya dapat dilihat pada Gambar 3.6. Hal

yang sama juga terjadi ketika pengguna menekan tombol "Lanjut belajar" pada daftar kelas aktif.

D. Flowchart halaman Course Module



UMMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA



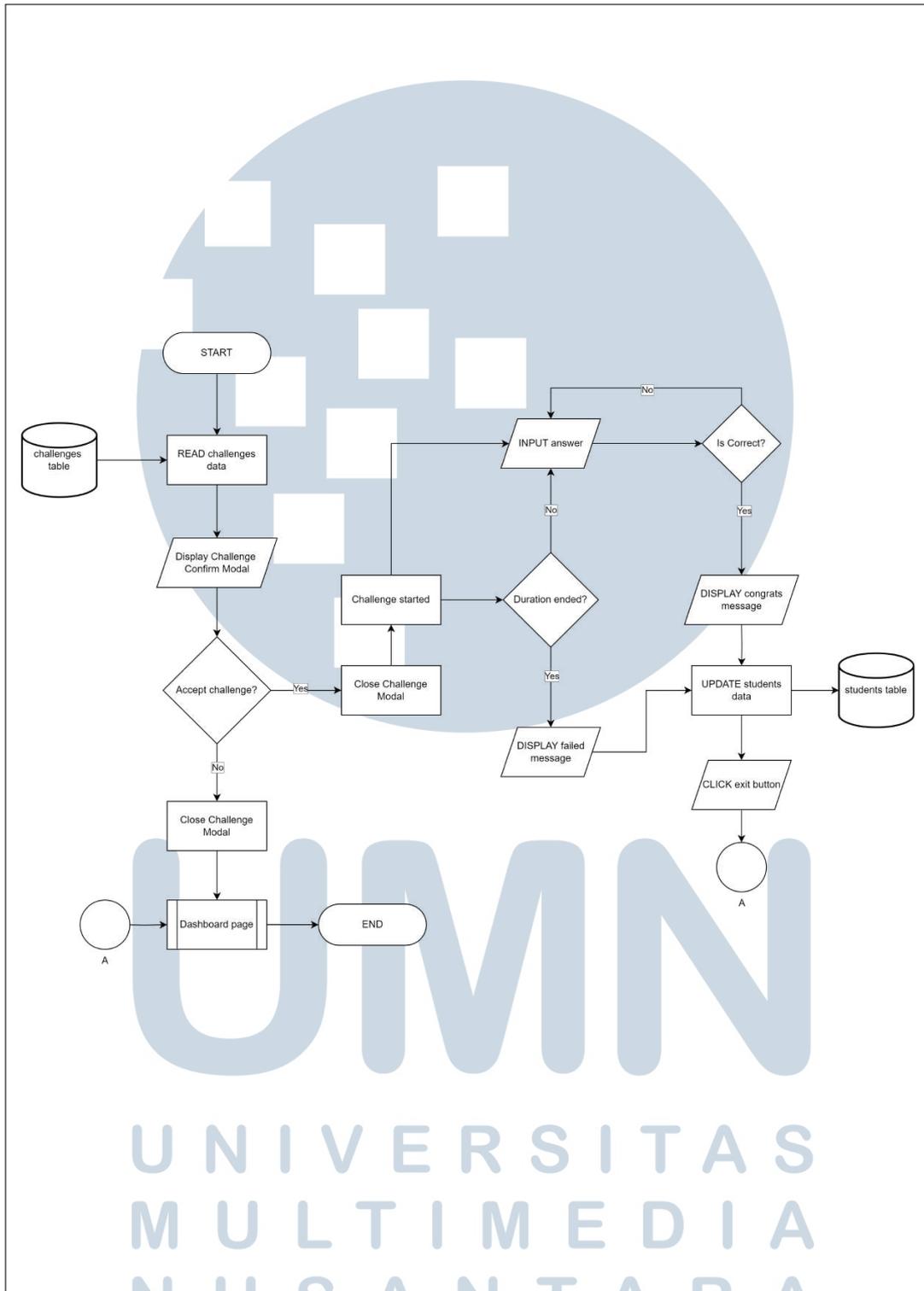
Gambar 3.6. Flowchart halaman Course Module

Flowchart dari halaman *Course Module* ditunjukkan pada Gambar 3.6. Ketika memasuki halaman *Course Module*, data dari modul akan diambil

terlebih dahulu dari *database* baru ditampilkan. Pada halaman ini terdapat navigasi ke modul sebelumnya dan modul selanjutnya. Jika terdapat modul sebelumnya, maka akan diambil data dari modul tersebut, hal serupa juga terjadi ketika terdapat modul selanjutnya. Ketika pengguna sudah sampai dimodul terakhir, maka tombol akan berubah menjadi "Selesai" yang ketika ditekan akan melakukan *update* terhadap data *student_courses* lalu mengembalikan pengguna ke halaman Courses yang alurnya bisa dilihat pada Gambar 3.5.

E. Flowchart halaman Daily Challenge



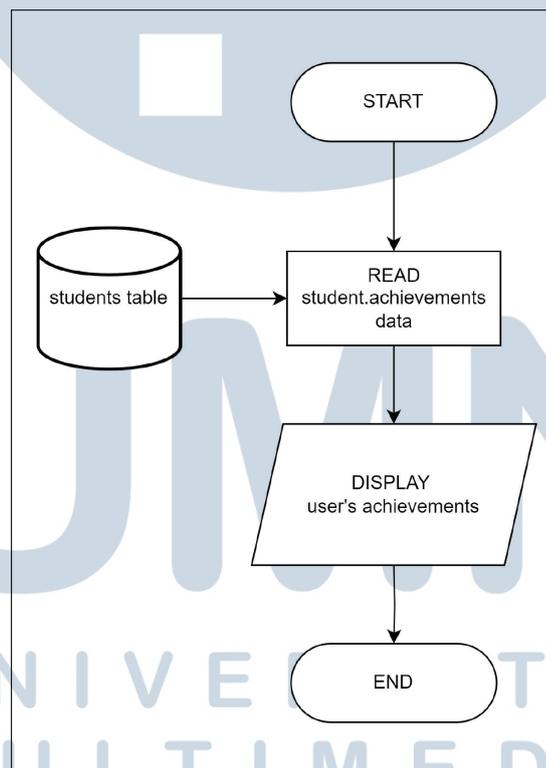


Gambar 3.7. Flowchart halaman Daily Challenge

Flowchart dari halaman *Daily Challenge* ditunjukkan pada Gambar 3.7. Halaman ini merupakan halaman di mana pengguna harus mengerjakan

soal tantangan yang diberikan secara acak dengan tingkat kesusahan dan topik soal yang juga didapatkan secara acak. Ketika pengguna memasuki halaman ini, akan diambil data *challenges* terlebih dahulu, setelah itu akan ditampilkan *confirmation modal* yang menampilkan data dari tantangan yang diberikan, seperti durasi, tingkat kesulitan, dan topik soal. Ketika pengguna memilih untuk menerima tantangan, maka soal akan ditampilkan dan durasi akan berjalan. Ketika pengguna memasukkan jawabannya, maka jawaban tersebut akan di cek terlebih dahulu, jika salah, maka pengguna harus terus memasukkan jawaban hingga benar atau durasi habis. Ketika jawaban benar, maka durasi akan dihentikan dan sebuah pesan selamat akan ditampilkan. Jika durasi habis, dan pengguna masih belum dapat menjawab dengan benar, akan ditampilkan pesan bahwa pengguna gagal menyelesaikan tantangan.

F. Flowchart halaman Achievements

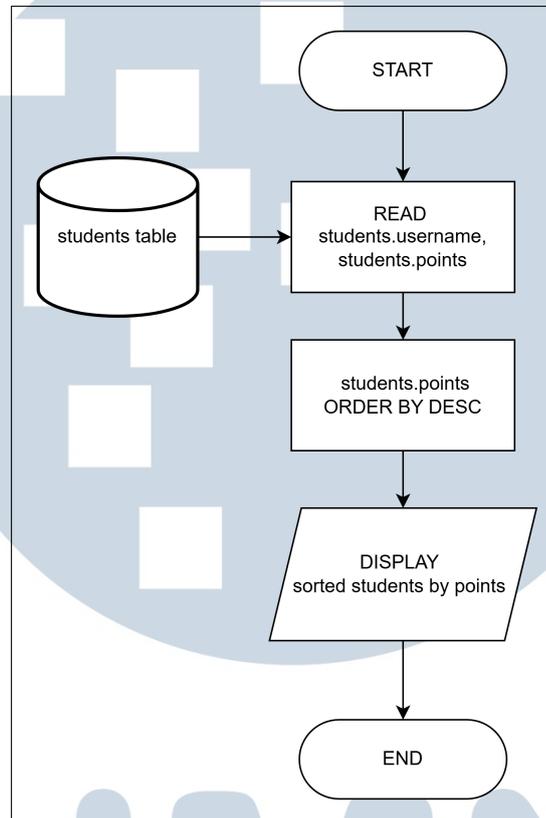


Gambar 3.8. Flowchart halaman Achievements

Flowchart dari halaman *Achievements* ditunjukkan pada Gambar 3.8. Pada saat pengguna membuka halaman ini, akan terjadi pengambilan data *students* dari *database* terlebih dahulu untuk mendapatkan data *achievements* yang

dimiliki oleh pengguna tersebut. Setelah data didapatkan, maka akan ditampilkan pada halaman ini.

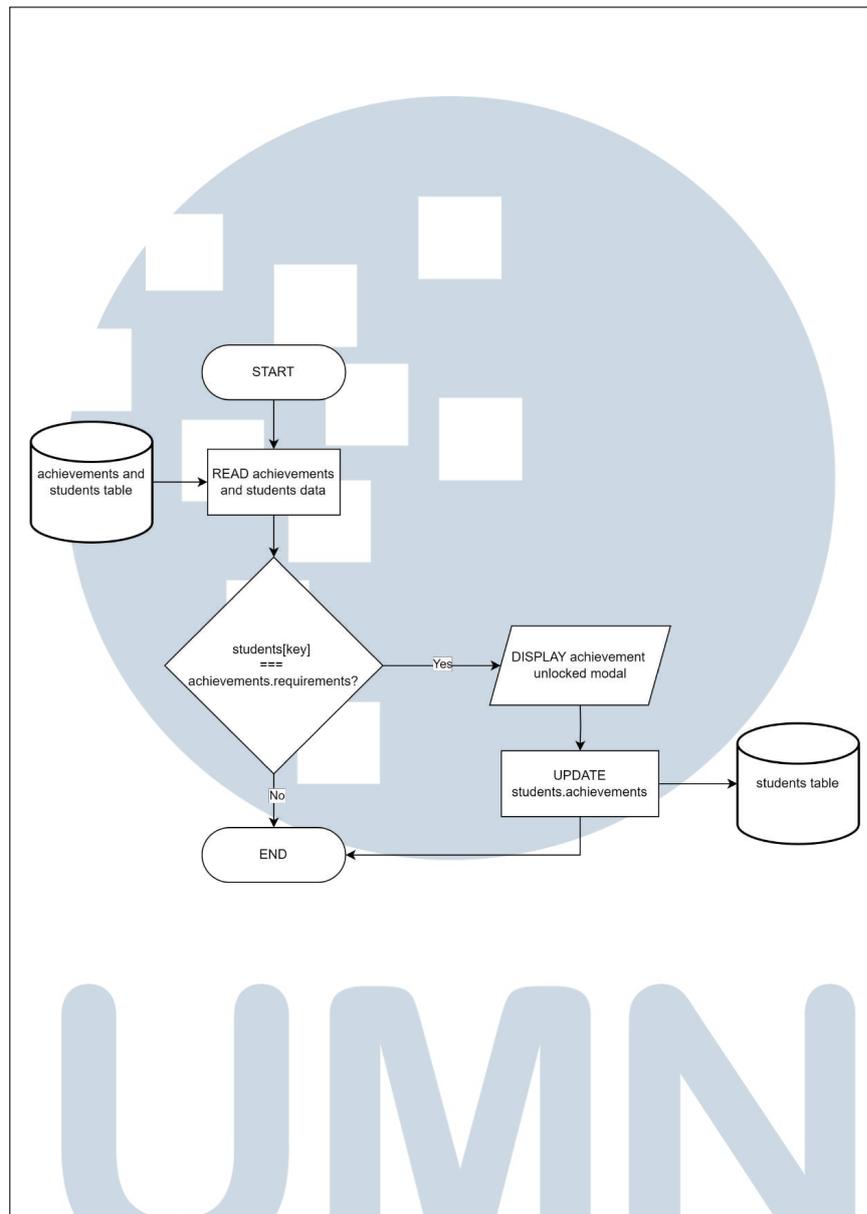
G. Flowchart halaman Leaderboard



Gambar 3.9. Flowchart halaman Leaderboard

Flowchart halaman *Leaderboard* ditunjukkan pada Gambar 3.9. Pada saat pengguna membuka halaman ini, akan terjadi proses pengambilan data tabel *students* dari *database*, kemudian dilakukan proses *sorting* secara *descending* pada data tersebut untuk menampilkan urutan pengguna dengan poin paling tinggi hingga paling rendah.

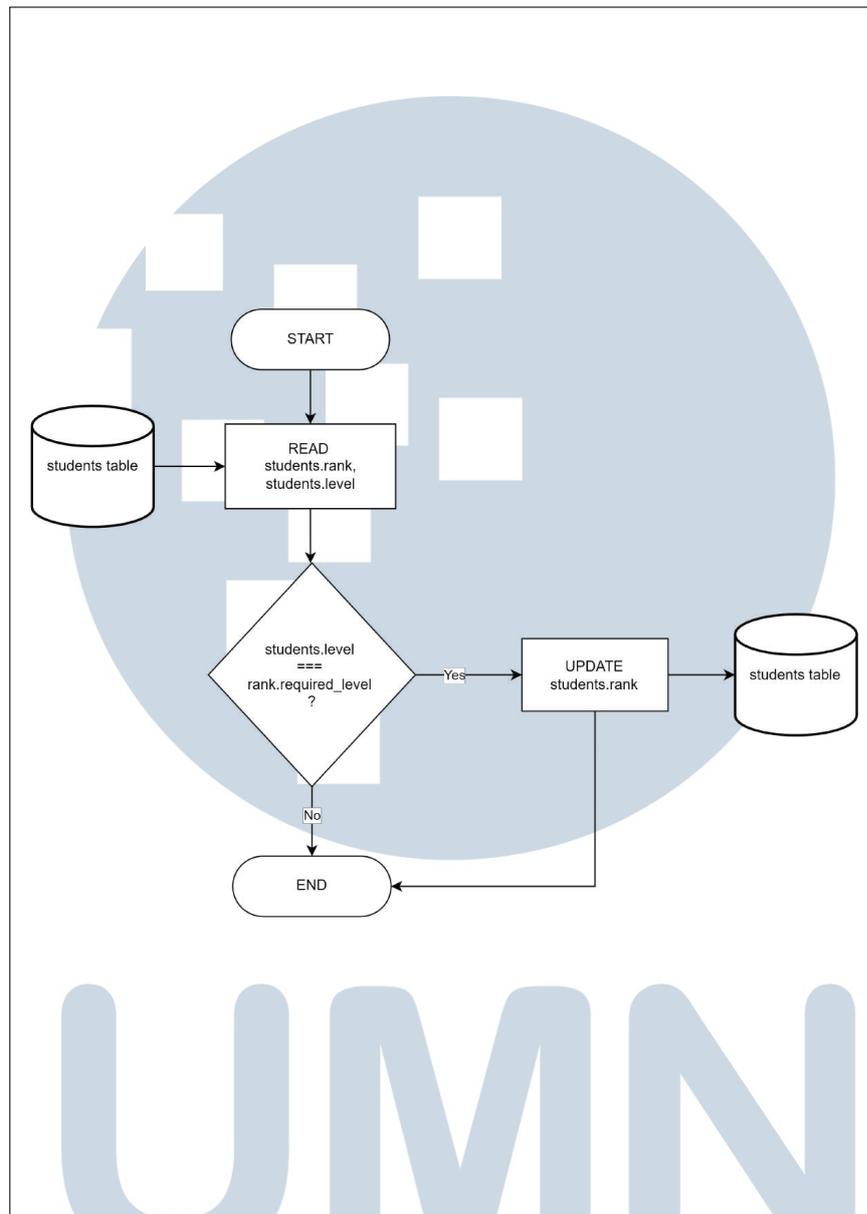
H. Flowchart sistem Achievement



Gambar 3.10. Flowchart elemen gamifikasi Achievement

Flowchart elemen gamifikasi *achievement* ditunjukkan pada Gambar 3.10. Aplikasi akan melakukan pengecekan terhadap data milik pengguna, dan melakukan perbandingan dengan syarat yang ditentukan pada setiap *achievement*. Jika data pengguna memenuhi syarat suatu *achievement*, maka akan ditampilkan pesan bahwa pengguna berhasil mendapatkan *achievement* baru, kemudian data pengguna pada *database* akan diperbarui.

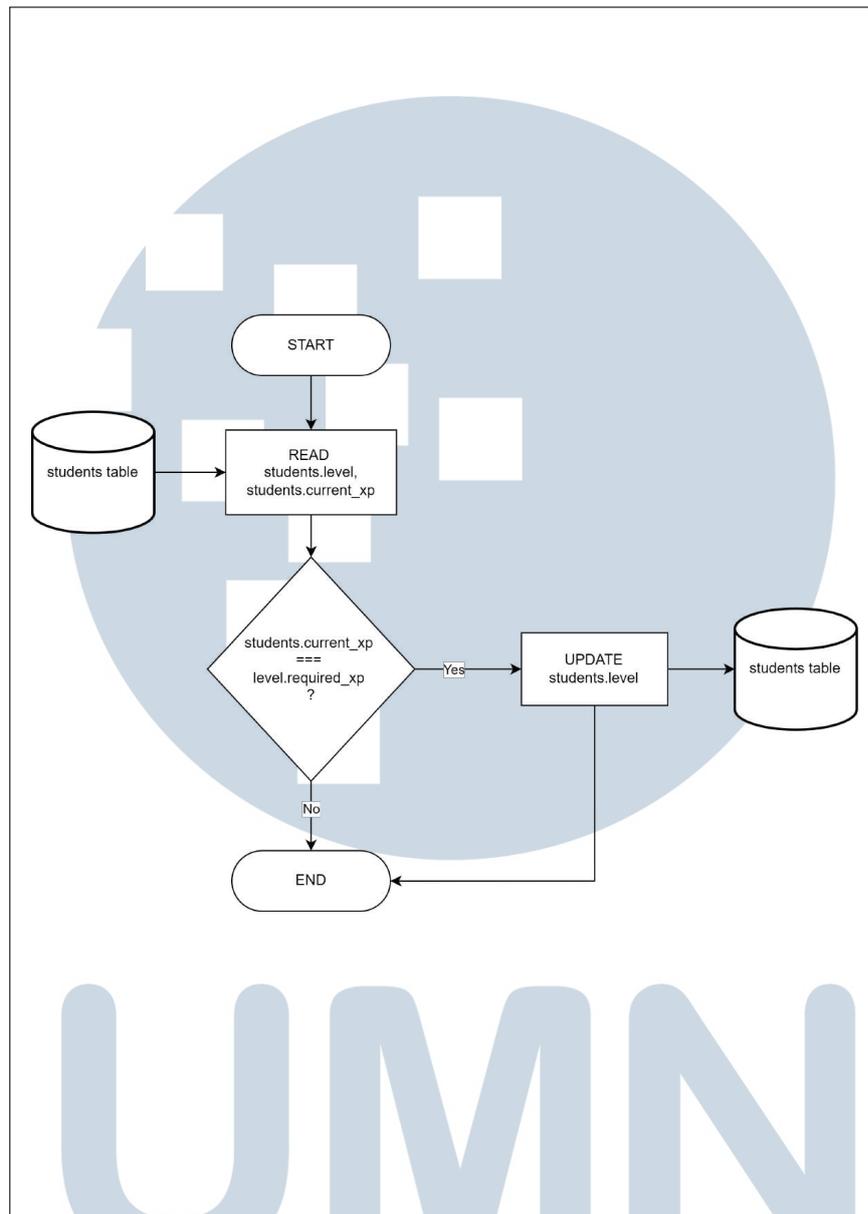
I. Flowchart sistem Rank



Gambar 3.11. Flowchart elemen gamifikasi Rank

Flowchart elemen gamifikasi *rank* ditunjukkan pada Gambar 3.11. Alur ini menjelaskan bagaimana *rank* pengguna dapat naik, contohnya dari *rank* "Beginner" menjadi "Advanced". Setiap *rank* memiliki ketentuan untuk bisa dicapai, yaitu berdasarkan level pengguna. Ketika level pengguna memenuhi syarat untuk memasuki *rank* baru, maka *rank* pengguna akan diperbarui di *database*.

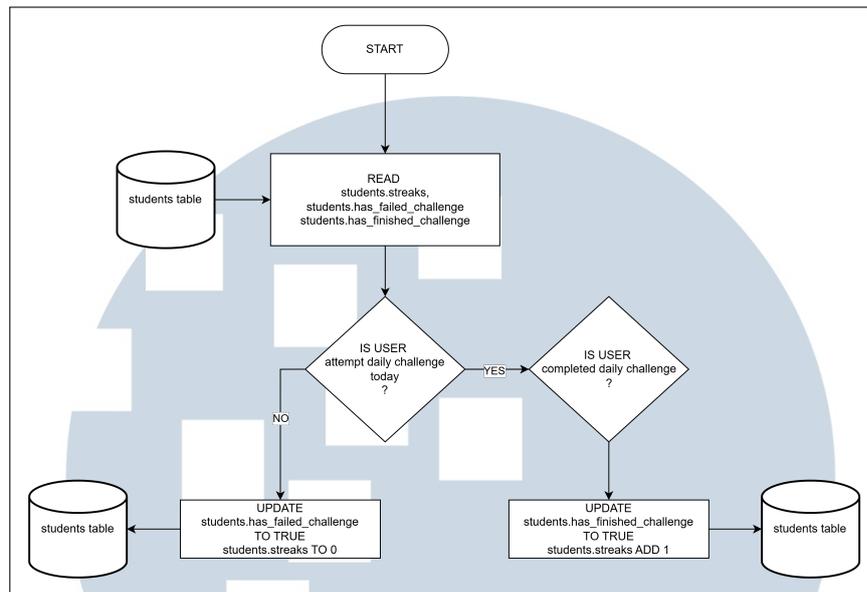
J. Flowchart sistem Level



Gambar 3.12. Flowchart elemen gamifikasi Level

Flowchart elemen gamifikasi *level* ditunjukkan pada Gambar 3.12. Alur ini menjelaskan bagaimana level pengguna dapat naik. Setiap level pada aplikasi ini memiliki syarat tersendiri untuk dapat memasuki level baru, yaitu berdasarkan jumlah *experience* atau XP yang dimiliki oleh pengguna. Ketika XP milik pengguna memenuhi syarat untuk naik level, maka level pengguna akan diperbarui di *database*.

K. Flowchart sistem Streaks

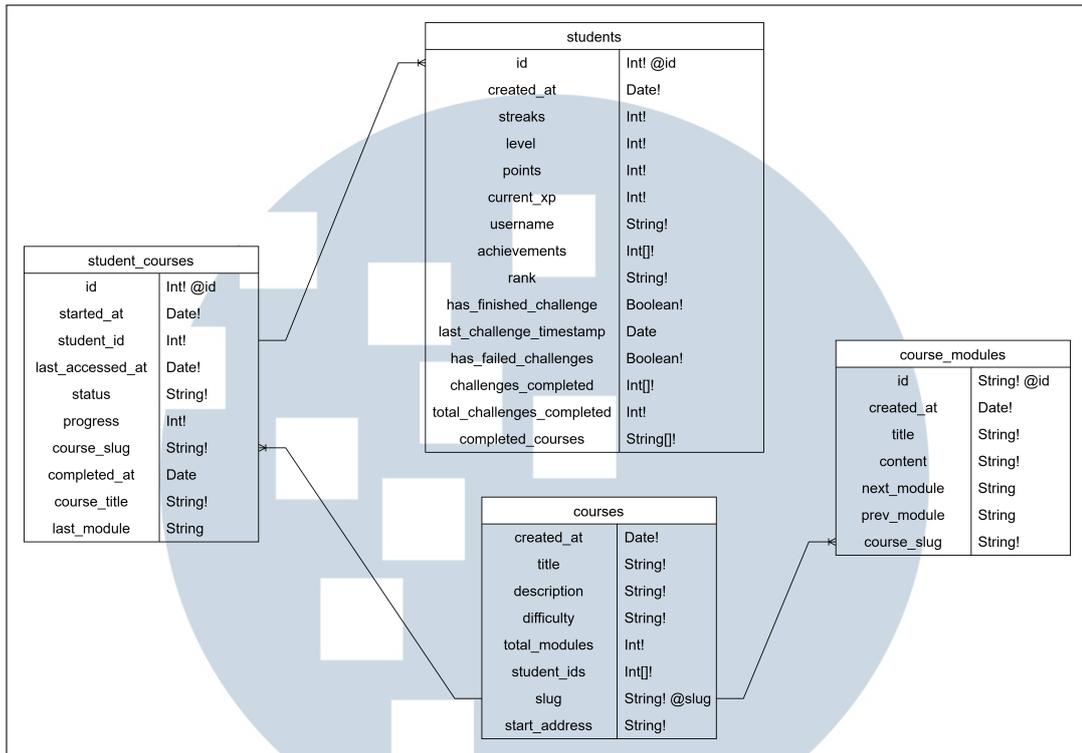


Gambar 3.13. Flowchart elemen gamifikasi Streaks

Flowchart elemen gamifikasi *streaks* ditunjukkan pada Gambar 3.13. Alur ini menjelaskan bagaimana jumlah *streaks* yang dimiliki oleh pengguna bisa bertambah dan bisa di-reset menjadi 0. Hal ini bergantung pada apakah pengguna berpartisipasi pada tantangan harian yang diberikan setiap harinya. Jika pengguna mengikuti dan berhasil menyelesaikan tantangan yang diberikan, maka jumlah *streaks* akan bertambah 1. Jika pengguna mengikuti tetapi gagal mengerjakan tantangan yang diberikan, maka jumlah *streaks* yang mereka miliki menjadi 0. Jumlah *streaks* juga bisa menjadi 0 ketika pengguna tidak mengerjakan tantangan dalam kurun waktu 24 jam.

3.2.4 Entity Relationship Diagram (ERD)

Gambar 3.14 menunjukkan rancangan *Entity Relationship Diagram* (ERD) aplikasi JStudy. Ini berguna untuk memberikan gambaran tentang skema *database* dan relasi dari setiap tabel pada *database* milik aplikasi JStudy. Pada ERD yang dirancang, terdapat total 4 tabel setelah dilakukan normalisasi sebanyak 2 kali (2NF), yaitu *students*, *student_courses*, *courses*, dan *course_modules* yang memiliki relasi nya masing-masing.



Gambar 3.14. Entity Relationship Diagram (ERD) JStudy

3.2.5 Perancangan Antarmuka

Perancangan antarmuka dibuat sebagai gambaran bagaimana tampilan dari aplikasi JStudy yang dibangun untuk penelitian ini nantinya. Perancangan antarmuka dilakukan menggunakan perangkat lunak bernama Figma yang memang dikhususkan untuk membuat rancangan antarmuka website maupun *mobile*.

A. Home

UNIVERSITAS
MULTIMEDIA
NUSANTARA

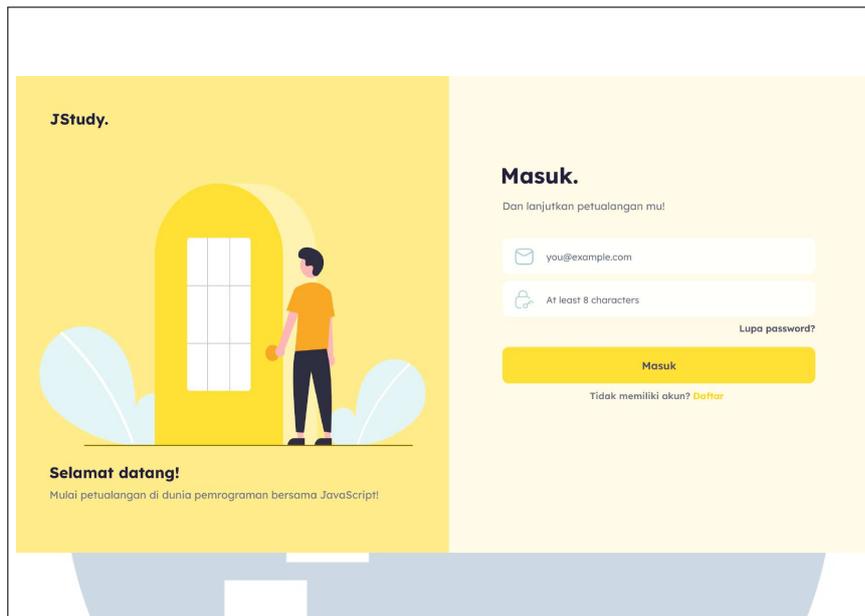


Gambar 3.15. Rancangan Antarmuka Halaman Home

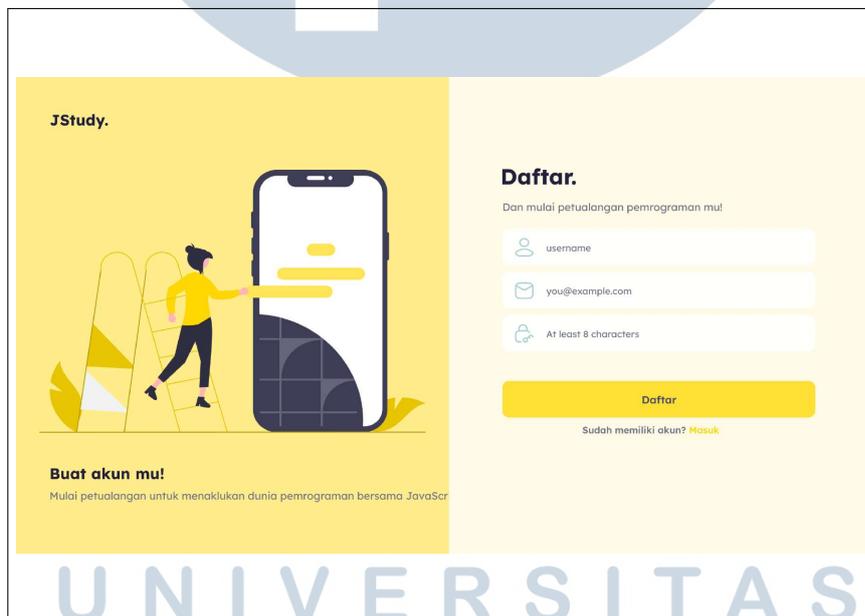
Gambar 3.15 menunjukkan halaman awal yang diperlihatkan ketika pengguna pertama kali memasuki website JStudy. Terdapat dua tombol dibagian kanan atas yang bertuliskan "Masuk" dan "Daftar". Ketika pengguna menekan tombol "Masuk" maka akan diarahkan ke halaman Login seperti pada Gambar 3.16. Jika pengguna menekan tombol "Daftar", maka akan diarahkan ke halaman Register seperti pada Gambar 3.17. Terdapat juga tombol "Masuk Kelas" yang berada ditengah halaman Home yang memiliki fungsi untuk mengarahkan pengguna langsung ke halaman Dashboard ketika pengguna telah melakukan proses *login*.

B. Authentication

UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.16. Rancangan Antarmuka Halaman Login

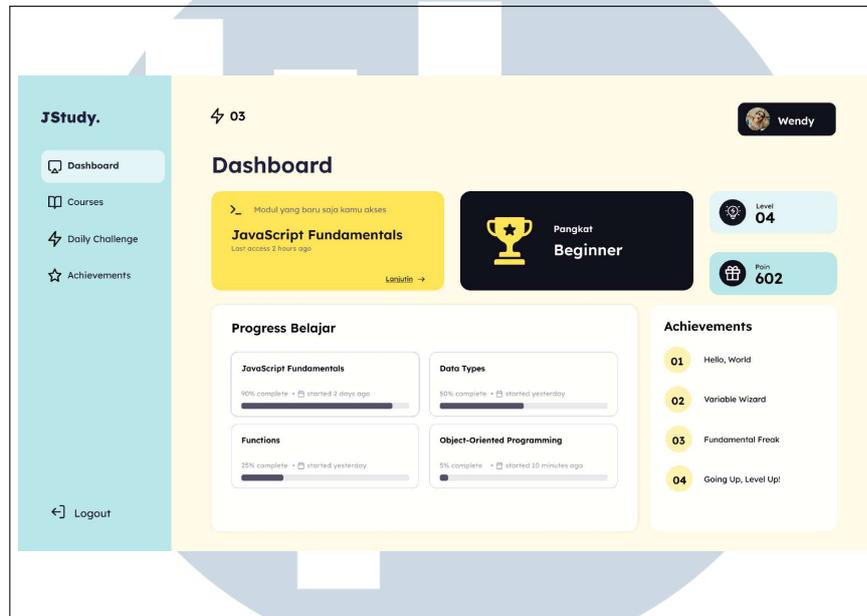


Gambar 3.17. Rancangan Antarmuka Halaman Register

Gambar 3.16 dan 3.17 merupakan halaman *authentication* yang berguna untuk pengguna melakukan proses pembuatan akun (*register*) atau masuk menggunakan akun yang mereka punya (*login*). Jika pengguna baru tidak melakukan kedua proses tersebut, maka pengguna tidak dapat mengakses

halaman Dashboard (Gambar 3.18) dan juga mengakses kelas yang ada pada aplikasi JStudy.

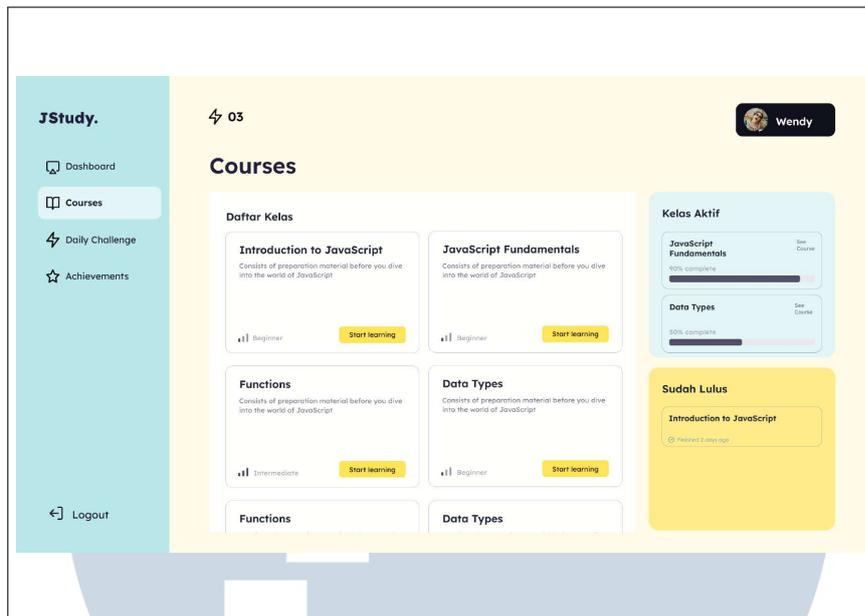
C. Dashboard



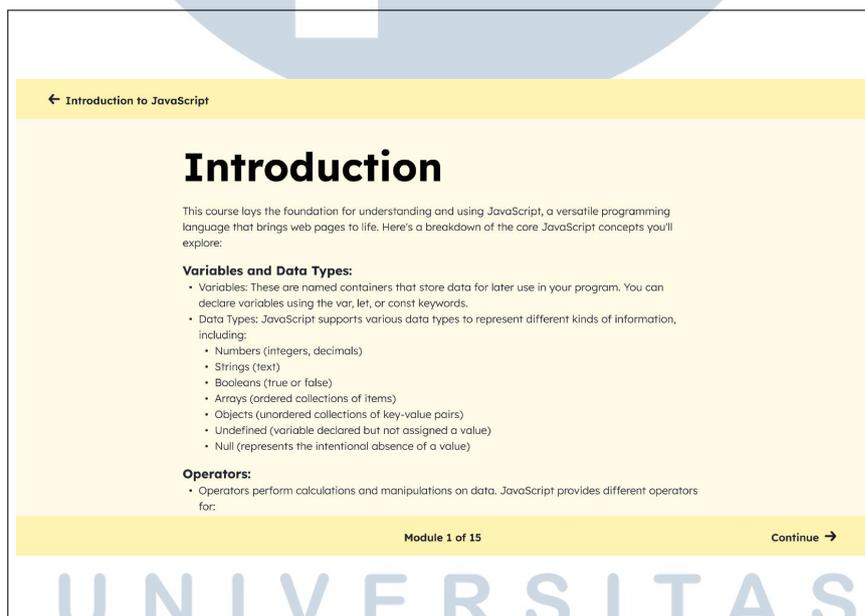
Gambar 3.18. Rancangan Antarmuka Halaman Dashboard

Gambar 3.18 merupakan halaman awal yang diperlihatkan ketika pengguna telah melakukan proses *login*. Halaman ini berisikan informasi umum milik pengguna, seperti modul yang telah diakses, *rank*, level, jumlah poin, progress belajar, dan *achievement*. Pada sisi kiri halaman juga terdapat *sidebar* yang memiliki beberapa menu navigasi dan masing-masing navigasi akan mengarahkan ke halaman berbeda ketika ditekan. Pada bagian kiri atas, terdapat logo petir dan angka, hal tersebut merupakan sistem gamifikasi bernama *streak* yang berguna untuk menghitung berapa kali pengguna berhasil dalam menyelesaikan *daily challenge* yang ada setiap harinya. *Streak* akan di reset menjadi 0 ketika pengguna gagal menyelesaikan *daily challenge* atau tidak mengerjakan *daily challenge* dalam kurun waktu 24 jam.

D. Courses



Gambar 3.19. Rancangan Antarmuka Halaman Courses

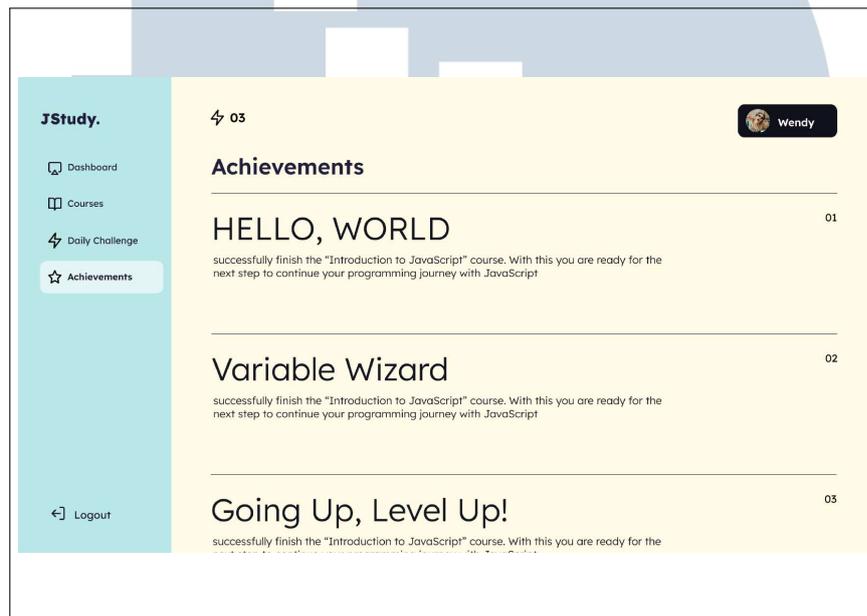


Gambar 3.20. Rancangan Antarmuka Halaman Course Module

Rancangan antarmuka halaman *Courses* dapat dilihat pada Gambar 3.19 yang berisikan sejumlah kelas dan juga kategori-kategori seperti daftar kelas, kelas aktif, dan sudah lulus. Daftar kelas merupakan daftar dari semua kelas yang tersedia pada aplikasi JStudy. Kelas aktif merupakan daftar kelas yang sedang diikuti oleh pengguna dan masih belum terselesaikan. Sudah lulus merupakan

daftar kelas yang telah diselesaikan oleh pengguna dan dianggap lulus. Ketika pengguna memulai atau melanjutkan kelas, pengguna akan diarahkan ke halaman Course Module seperti pada Gambar 3.20. Halaman Course Module merupakan halaman yang berisikan materi pembelajaran yang disediakan oleh aplikasi JStudy dengan materi berdasarkan kelas yang diikuti.

E. Achievements

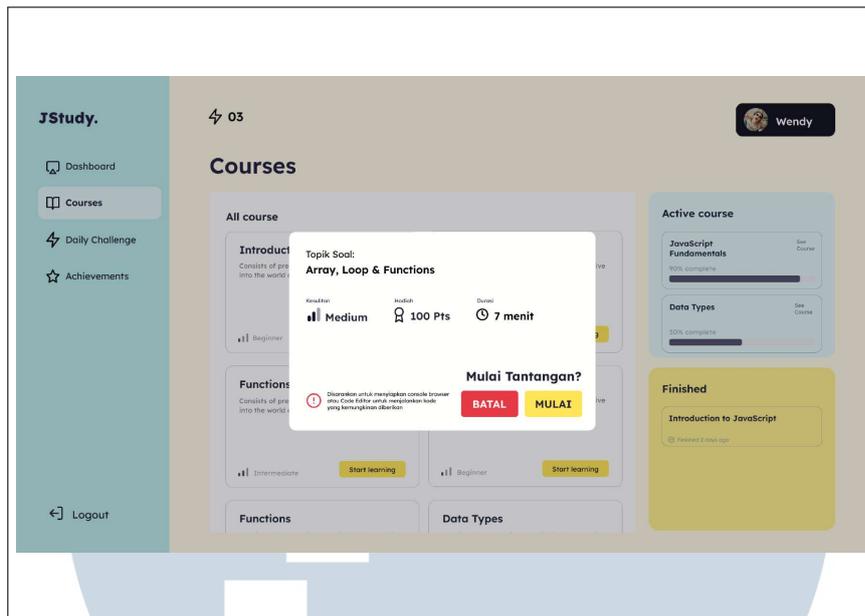


Gambar 3.21. Rancangan Antarmuka Halaman Achievements

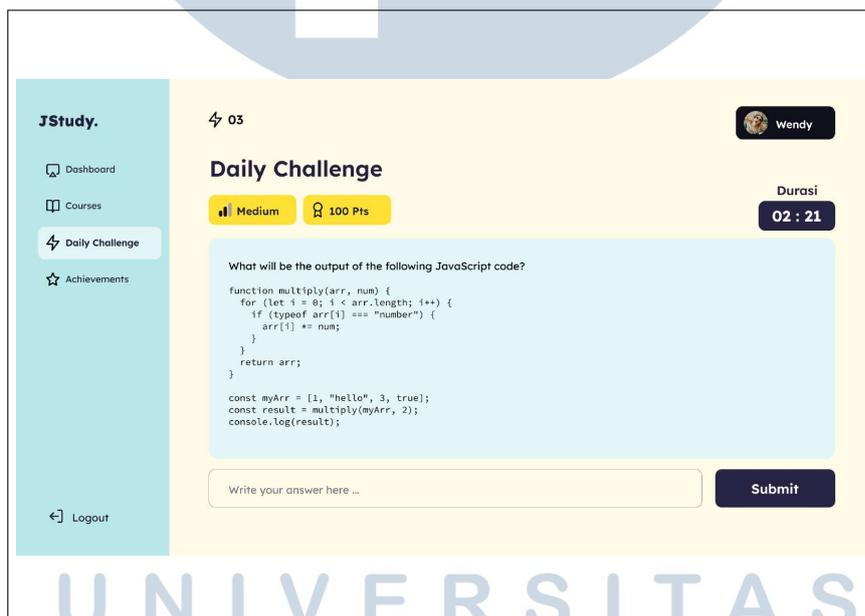
Rancangan antarmuka halaman *Achievements* ditunjukkan pada Gambar 3.21. Halaman ini berguna untuk menampilkan daftar penghargaan (*achievement*) yang berhasil didapatkan oleh pengguna. *Achievement* merupakan salah satu elemen gamifikasi yang diterapkan pada aplikasi JStudy.

F. Daily Challenges

UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.22. Rancangan Antarmuka Modal Konfirmasi Tantangan

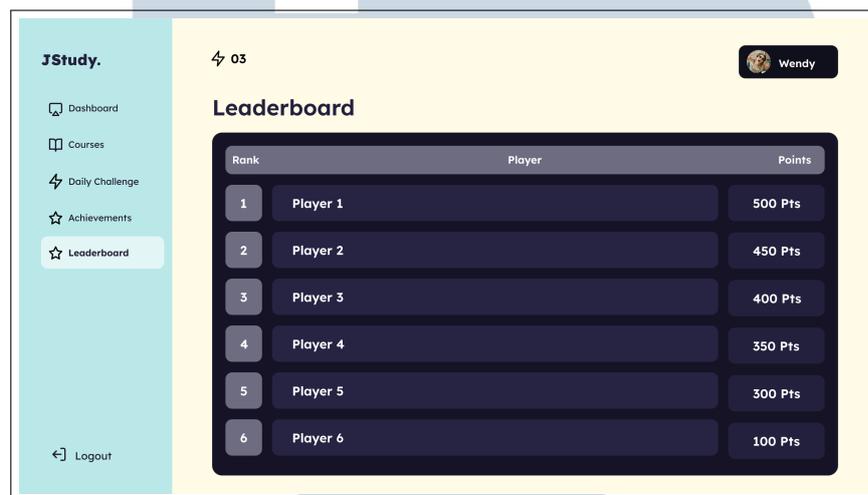


Gambar 3.23. Rancangan Antarmuka Halaman Daily Challenge

Rancangan antarmuka halaman *Daily Challenge* dapat dilihat pada Gambar 3.23 yang merupakan elemen gamifikasi lain yang diterapkan pada aplikasi JStudy. *Daily Challenge* merupakan tantangan yang ada setiap harinya dan pengguna dapat mengikuti tantangan ini, setiap pengguna yang berhasil menyelesaikan tantangan akan diberikan hadiah berupa poin

dan XP (*experiences*). Tantangan akan diberikan secara acak sehingga pengguna tidak akan mengetahui tingkat kesulitan dari tantangan yang akan dikerjakannya. Tetapi, pengguna bisa memutuskan untuk mengikuti tantangan tersebut atau tidak, karena ketika mengakses halaman ini, akan ada *pop-up* modal yang muncul seperti pada Gambar 3.22.

G. Leaderboard



Rank	Player	Points
1	Player 1	500 Pts
2	Player 2	450 Pts
3	Player 3	400 Pts
4	Player 4	350 Pts
5	Player 5	300 Pts
6	Player 6	100 Pts

Gambar 3.24. Rancangan Antarmuka Halaman Leaderboard

Gambar 3.24 merupakan rancangan antarmuka dari halaman *Leaderboard* yang berfungsi untuk menampilkan peringkat setiap pengguna berdasarkan akumulasi poin yang mereka miliki. Urutan ditampilkan dalam bentuk tabel yang menampilkan informasi seperti nama dan jumlah poin pengguna. Leaderboard sendiri merupakan salah satu elemen gamifikasi yang turut di implementasi pada aplikasi JSStudy.

UNIVERSITAS
MULTIMEDIA
NUSANTARA