

**Repetit: Sistem Asisten Belajar berbasis AI untuk Meningkatkan  
Retensi Pembelajaran Siswa**



Disusun Oleh:

Fathimah Az Zahra Sanjani	4611422057
Thoriq Ibrahim Farras	4611422074
Almaas Izdiyar Sant	4611422076

**PROPOSAL PENGEMBANGAN PERANGKAT LUNAK  
GEMASTIK 2024 DIVISI 8**

## **I. Judul>Nama Perangkat Lunak**

Repetit: Sistem Asisten Belajar Berbasis AI untuk Meningkatkan Retensi Pembelajaran Siswa.

## **II. Latar Belakang Ide Perangkat Lunak**

Proses belajar mandiri menjadi sangat penting bagi semua individu dalam memperdalam pemahaman terhadap suatu materi atau subjek [1]. Namun, tidak semua individu memiliki keterampilan dan metode belajar yang efektif untuk memaksimalkan potensi mereka. Banyak orang yang masih bergantung pada metode belajar konvensional seperti menghafal dan meng-highlight materi menggunakan stabilo, yang seringkali kurang optimal dalam membantu mereka memahami konsep-konsep yang kompleks [2]. Tantangan ini dialami oleh berbagai kalangan baik pelajar, mahasiswa, maupun profesional. Sehingga, dibutuhkan solusi yang dapat menjembatani kesenjangan dalam mengakses metode belajar yang efektif.

*Active Recall* adalah metode belajar yang mengharuskan pelajar memanggil dan merekonstruksi informasi dari memori secara aktif [3]. Memanfaatkan teknik *Active Recall* mampu meningkatkan performa belajar, efisiensi akademik dan retensi pengetahuan yang lebih baik dibandingkan menggunakan metode tradisional [4]. *Active recall* mendorong keterlibatan aktif siswa dengan memanggil kembali informasi dari memori, tidak hanya menerima informasi secara pasif seperti menghafal dengan meng-highlight atau membaca berulang catatan [5]. Sebaliknya, teknik ini mengharuskan siswa mengingat kembali pengetahuan yang telah dipelajari sebelumnya secara aktif, misalnya menggunakan *flashcard* atau peta konsep [6]. Namun, penerapannya seringkali terkendala karena dibutuhkan usaha dan waktu yang lebih besar dalam membuat bahan dan melakukan *Active Recall* secara mandiri.

Untuk memaksimalkan potensinya, metode *Active Recall* dapat dikombinasikan dengan *Spaced Repetition Learning* (SRL) yang memanfaatkan pengulangan materi dengan interval waktu yang semakin melebar [7]. SRL mengoptimalkan daya ingat jangka panjang dengan mengikuti prinsip dasar neurosains bahwa pengulangan berkala jauh lebih efektif dibanding menghafal semuanya sekaligus [8]. Dengan mengombinasikan keduanya, pelajar tidak hanya mengingat kembali informasi secara aktif, tetapi juga pada selang waktu tertentu yang memperkuat retensi dan asosiasi pengetahuan dalam memori jangka panjang. Riset membuktikan kombinasi ini 2-3 kali lebih efektif dibanding metode konvensional dalam mengingat konsep-konsep kompleks untuk jangka panjang[9]. Terutama saat waktu terbatas, memprioritaskan *active recall* dan *SRL* terhadap materi lama terbukti lebih optimal ketimbang berusaha memaksa memahami banyak hal baru sekaligus [10].

Salah satu Tujuan Pembangunan Berkelanjutan (SDGs) yang harus dicapai adalah nomor 4, yaitu menjamin pendidikan yang inklusif dan berkualitas serta merata bagi seluruh masyarakat [11], [12]. Untuk mewujudkan hal tersebut, pemanfaatan teknologi kecerdasan buatan (AI) dapat menjadi kunci dalam meningkatkan kualitas dan pemerataan pelayanan publik di bidang pendidikan di Indonesia [13]. Dengan kemampuannya untuk menjadikan platform belajar interaktif yang adaptif, AI mampu memfasilitasi metode belajar aktif dengan menyesuaikan konten, metode, dan kecepatan pembelajaran sesuai kebutuhan serta gaya belajar masing-masing individu [14]. Teknologi AI juga dapat berkontribusi dalam memperkuat sistem pendidikan nasional dengan memberikan akses yang merata terhadap metode belajar efektif, bahan ajar berkualitas, penilaian esai secara otomatis, serta bimbingan pembelajaran yang dipersonalisasi [15] bagi seluruh masyarakat Indonesia di berbagai pelosok negeri.

Dengan mengintegrasikan teknik *Active Recall*, SRL, dan teknologi AI, Repetit menyediakan platform belajar interaktif dan adaptif yang memandu individu mengadopsi metode belajar aktif seperti menjawab pertanyaan melalui *flashcard* hingga menguji pemahaman secara mandiri. Repetit juga menyediakan akses mentoring yang dapat dilakukan setiap individu sebagai bahan evaluasi untuk mendukung proses belajar mandiri sekaligus interaksi dengan ahli-ahli yang terqualifikasi baik untuk memberikan mentoring. Aplikasi Repetit ditujukan untuk memfasilitasi proses belajar mandiri dengan menggunakan metode *Active Recall* dan SRL serta mendukung proses pembelajaran individu secara mandiri, bukan untuk menyediakan sumber belajar pada tiap-tiap individu.

### **III. Tujuan dan Manfaat Dikembangkannya Perangkat Lunak**

#### **a. Tujuan**

Tujuan dari pengembangan perangkat lunak ini adalah:

- 1) Menyediakan platform belajar interaktif dan adaptif yang memfasilitasi penerapan teknik *Active Recall* dan SRL dalam proses pembelajaran mandiri bagi mahasiswa, demi meningkatkan retensi pembelajaran.
- 2) Mengintegrasikan teknologi AI untuk menciptakan pertanyaan dari modul yang diinput pengguna dan melakukan penilaian jawaban otomatis untuk mengevaluasi pemahaman serta kinerja pembelajar secara efisien dan akurat.
- 3) Menciptakan perangkat lunak yang mampu memberikan bimbingan, umpan balik, dan evaluasi yang handal serta akurat dalam proses belajar mandiri dengan metode *Active Recall* dan SRL, sehingga mendapat respon baik dari pengguna dengan indikator peningkatan motivasi dan efektivitas belajar.

#### **b. Manfaat**

Manfaat yang diharapkan dari perangkat lunak ini adalah membantu mahasiswa mengoptimalkan potensi akademis melalui proses belajar mandiri dengan teknik *Active Recall* dan SRL, meningkatkan performa dan retensi pengetahuan dengan metode belajar

yang lebih baik dibanding metode tradisional, memberikan akses metode belajar efektif, penilaian esai secara otomatis, dan *mentoring* yang dilakukan oleh orang yang berkompeten bagi individu dalam upaya peningkatan pelayanan pendidikan dan sumber daya manusia yang unggul.

#### IV. Batasan Perangkat Lunak yang Dikembangkan

1. Perangkat lunak ditujukan untuk mahasiswa perguruan tinggi sebagai pengguna utama, dan berfokus pada memfasilitasi proses belajar mandiri dengan menggunakan metode *Active Recall* dan SRL melalui platform belajar interaktif.
2. Perangkat lunak ini mendukung proses pembelajaran individu secara mandiri, bukan untuk menyediakan sumber belajar tiap-tiap individu.

#### V. Metodologi Pengembangan Perangkat Lunak

Implementasi perangkat lunak ini berpegang teguh kepada prinsip-prinsip *Agile Development*. Penerapan utama dari prinsip-prinsip tersebut adalah pelaksanaan *scrum*, di mana tiap anggota tim mendapatkan tugas dan mengerjakannya dalam periode-periode waktu terdedikasi yang disebut *sprint*. Selain itu, *scrum meeting* dilakukan setelah setiap *sprint* untuk mengecek kemajuan setiap anggota, status tugas yang dikerjakan, dan pembagian tugas untuk sesi *sprint* selanjutnya.

Dengan mempertimbangkan waktu yang terbatas, implementasi pembuatan sistem dilakukan dengan mengadopsi kerangka kerja *rapid app development* (RAD). RAD merupakan sebuah metodologi pengembangan perangkat lunak yang mengutamakan pengembangan dan iterasi cepat serta *feedback* pengguna pada setiap tahapan. Beberapa prinsip RAD yang diterapkan pada pengembangan aplikasi Repetit adalah:

##### a. *Prototyping*

RAD mengutamakan pembuatan prototipe dari mulai prototipe yang memiliki fungsi minimal atau dasar, hingga prototipe yang memiliki semua fitur yang direncanakan.

##### b. Keterlibatan Pengguna

RAD mengutamakan keterlibatan pengguna dalam proses pengembangan. Dalam metode *Waterfall*, pengguna umumnya memberi *feedback* di akhir, yang terkadang ada beberapa komponen/fitur penting yang tidak terpikirkan sebelumnya. Hal ini mengakibatkan proses pengembangan harus diulangi dari awal. Dalam RAD, dengan adanya keterlibatan pengguna yang ekstensif pada setiap, tim pengembang dapat menantisipasi perubahan atau penambahan fitur yang sebelumnya tidak terpikirkan. Penggunaan *prototyping* dari awal dapat membantu pengguna memberi *feedback* yang lebih konkrit dibandingkan dengan jika pengguna hanya diberi konsep abstrak tentang aplikasinya.

Adapun penerapan prinsip-prinsip di atas pada tiap tahap pengembangan aplikasi Repetit sebagai berikut:

a. Perencanaan

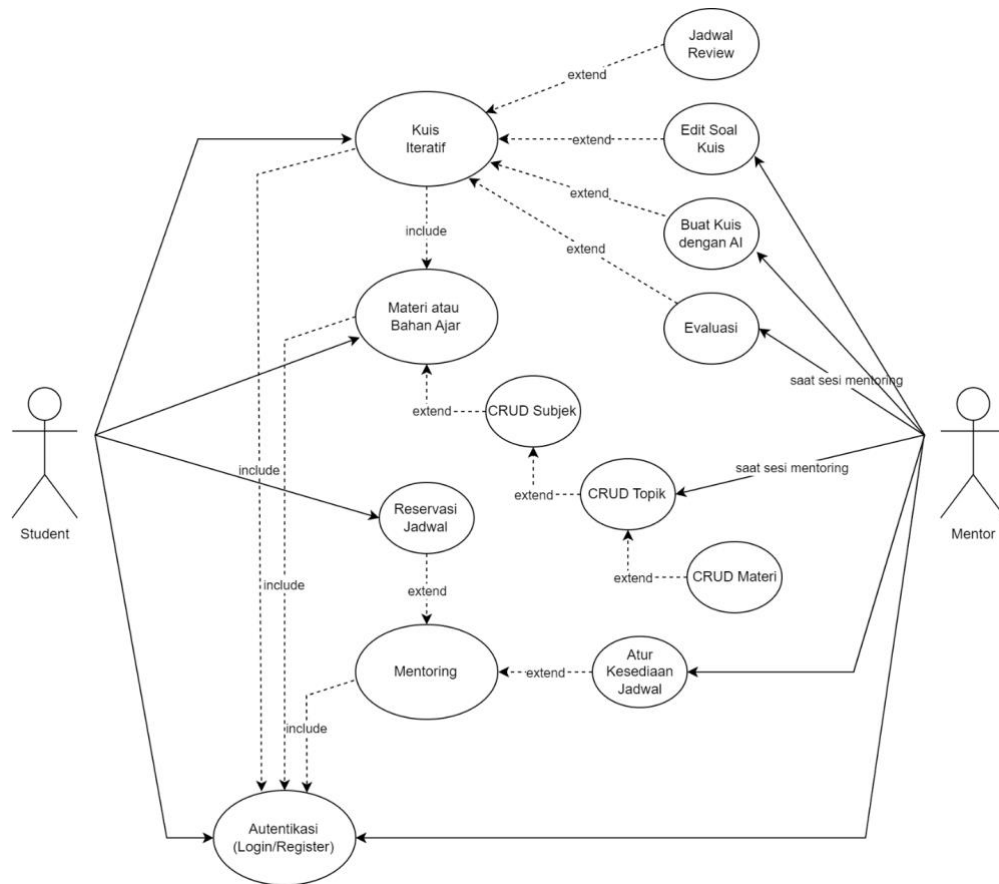
Perencanaan dimulai dengan melakukan survei kepada calon pengguna untuk menggali informasi terkait kebutuhan mereka dalam proses belajar mandiri. Survei dilakukan terhadap 26 mahasiswa UNNES dari berbagai program studi dan tahun angkatan. Survei ini mencakup identifikasi gaya belajar yang paling efektif bagi setiap individu serta eksplorasi perangkat lunak pembelajaran yang telah mereka gunakan sebelumnya, baik kelebihan maupun kekurangannya. Selain itu, survei mengidentifikasi fitur-fitur dan fungsionalitas yang diharapkan dari perangkat lunak belajar mandiri yang akan dikembangkan. Hasil survei ini menjadi dasar penting untuk menentukan ruang lingkup, persyaratan fungsional dan non-fungsional, serta batasan dari perangkat lunak yang akan dibangun.

b. Analisis Kebutuhan

Setelah melakukan survei untuk menggali kebutuhan pengguna, tahap selanjutnya merupakan analisis kebutuhan dengan mengolah dan menganalisis secara mendalam hasil survei untuk mengidentifikasi serta mendokumentasikan secara rinci kebutuhan fungsional dan non-fungsional perangkat lunak. Kegiatan yang dilakukan mencakup mempelajari data survei, melakukan wawancara dan diskusi lanjutan, menentukan prioritas kebutuhan, mendefinisikan persyaratan fungsional seperti fitur, alur kerja, integrasi sistem, serta mengidentifikasi persyaratan non-fungsional terkait performa, keamanan, kehandalan, skalabilitas, dan aspek kualitas lainnya, yang keseluruhannya dituangkan dalam dokumentasi spesifikasi kebutuhan perangkat lunak secara lengkap dan rinci sebagai acuan utama bagi pengembang.

c. Perancangan

Pada tahap ini, akan dimulai perancangan mengenai desain dan arsitektur sistem yang akan digunakan sesuai kebutuhan yang telah dianalisis. Aktivitas ini mencakup pembuatan flowchart sistem, pendefinisian komponen perangkat lunak, dan desain basis data. Perancangan juga melibatkan pembuatan *low fidelity* menggunakan Excalidraw, serta *mid fidelity* dan *high fidelity* menggunakan Figma, demi memastikan antarmuka yang *user-friendly* dan intuitif.



Gambar 1. Diagram *use-case* untuk Repetit.



Pembuatan algoritma penjadwalan *flashcard* dan algoritma penilaian jawaban otomatis dengan teknik *Natural Language Processing* (NLP) akan dikembangkan menggunakan bahasa pemrograman Python. Aplikasi web akan dibangun menggunakan *framework* Next.js dengan JavaScript untuk *frontend*. Infrastruktur *cloud* akan dibangun menggunakan *Google Cloud Platform* (GCP) untuk autentikasi dan penyimpanan data. Setelah semua unit siap, komponen akan diintegrasikan untuk memastikan aplikasi bekerja secara utuh dan menyeluruh.

e. Pengujian Sistem

Pengujian Sistem Dalam memastikan perangkat lunak berjalan dengan baik, pengujian akan dilakukan dalam 2 tahap:

1) Pengujian Internal

Pengujian internal akan dilakukan sebelum pengujian lapangan. Pengujian ini akan dilakukan oleh internal tim dengan menguji fitur-fitur utama aplikasi seperti penjadwalan *flashcard*, penilaian jawaban otomatis dengan NLP, rekomendasi pembelajaran adaptif, dan integrasi dengan *mentoring*. Aspek-aspek lain yang akan diuji mencakup kenyamanan antarmuka pengguna, pengalaman pengguna, performa aplikasi, serta kompatibilitas dengan berbagai perangkat.

2) Pengujian Pengguna

Pengujian pengguna akan menggunakan metode pengujian *black-box* yang melibatkan sekelompok mahasiswa untuk mencoba menggunakan aplikasi Repetit dan memberikan umpan balik. Pengguna akan diminta untuk melakukan tugas-tugas seperti menjawab *flashcard*, membuat ringkasan, mendapatkan rekomendasi pembelajaran, dan berinteraksi dengan fitur *mentoring*. Mereka akan menilai aspek-aspek seperti kegunaan aplikasi, kemudahan penggunaan, efektivitas metode belajar yang ditawarkan, serta kepuasan pengguna secara keseluruhan.

f. Evaluasi

Pada tahap ini, hasil dari pengujian internal dan pengujian pengguna akan dievaluasi secara mendalam. Evaluasi akan berfokus pada identifikasi masalah, *bug*, atau kekurangan dalam aplikasi, terutama yang terkait dengan fitur-fitur utama seperti penjadwalan *flashcard*, penilaian jawaban otomatis dengan NLP, rekomendasi pembelajaran adaptif, dan integrasi dengan *mentoring*. Umpan balik dari pengguna juga akan dipertimbangkan untuk perbaikan antarmuka dan pengalaman pengguna.

g. Peluncuran

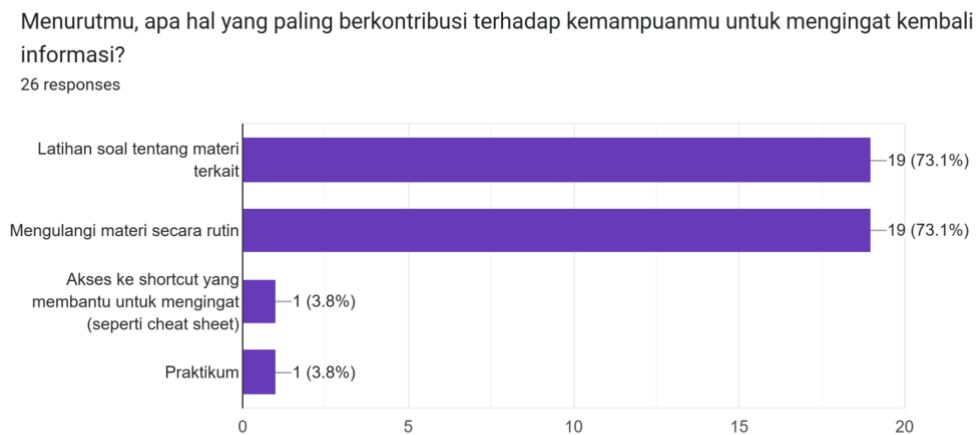
Setelah evaluasi selesai dan tidak ada masalah signifikan yang tersisa, aplikasi Repetit siap untuk diluncurkan kepada publik. Peluncuran akan dilakukan melalui



penyebaran aplikasi web. Kegiatan pemasaran dan promosi akan dilakukan untuk meningkatkan kesadaran pengguna tentang aplikasi baru ini dan mendorong adopsi yang lebih luas di kalangan mahasiswa. Hal ini dapat dilakukan melalui kampanye digital, kolaborasi dengan organisasi mahasiswa, serta kemitraan dengan perguruan tinggi untuk mempromosikan Repetit sebagai platform belajar mandiri yang efektif dengan metode terbukti seperti *Active Recall* dan *SRL*, didukung oleh teknologi AI dan fitur Mentoring.

## VI. Analisis Kebutuhan dan Desain Solusi Perangkat Lunak

### a. Analisis Kebutuhan Perangkat Lunak



Gambar 4. Hasil survei yang menanyakan kontributor terbesar dalam retensi informasi.

Pengembangan aplikasi “Repetit” bertujuan untuk menyediakan sistem asisten belajar berbasis AI dengan fitur *mentorship* guna meningkatkan retensi pembelajaran mahasiswa. Analisis ini melibatkan konsultasi dan survei kepada mahasiswa maupun pelajar serta identifikasi kebutuhan pengguna yang spesifik. Berikut merupakan identifikasi kebutuhan melalui analisis ini:

- 1) Penjadwalan Pengulangan Materi: Seperti yang ditunjukkan pada Gambar 4, banyak mahasiswa yang menanggapi bahwa pengulangan materi secara rutin adalah salah satu kontributor terbesar dalam retensi informasi. Maka dari itu aplikasi harus mampu menjadwalkan pengulangan materi secara otomatis menggunakan teknik *SRL* yang memastikan pengulangan materi yang telah dipelajari memperkuat retensi pengetahuan jangka panjang pengguna.
- 2) Pembuatan *flashcard* otomatis: Aplikasi harus mampu membuat *flashcard* uji pemahaman secara otomatis untuk setiap materi yang telah dipelajari pengguna. Pembuatan *flashcard* uji pemahaman otomatis ini menerapkan prinsip *active*

*recall*, mengharuskan pengguna memanggil kembali dan merekonstruksi informasi yang telah dipelajari secara aktif

- 3) Penilaian *flashcard* Otomatis: Aplikasi harus mampu melakukan penilaian secara otomatis terhadap *flashcard* yang telah dikerjakan oleh pengguna untuk mendapatkan umpan balik yang cepat terhadap kinerja mereka dan meningkatkan efektivitas proses belajar.
- 4) Antarmuka intuitif: Aplikasi harus memiliki antarmuka dengan ikon dan instruksi yang jelas serta tata letak yang intuitif untuk memastikan bahwa pengguna dapat dengan mudah mengakses dan menggunakan berbagai fitur yang tersedia tanpa kebingungan.
- 5) Fleksibilitas dalam Reservasi Sesi *Mentoring*: Aplikasi harus menyediakan fleksibilitas bagi siswa dalam mereservasi sesi mentoring dengan *mentor* yang kompeten. Fitur ini memungkinkan siswa untuk menjadwalkan sesi bimbingan berdasarkan ketersediaan waktu mereka dan *mentor*. Selain itu, tidak semua pelajar memiliki dana untuk melakukan reservasi dengan *mentor* yang pada umumnya memasang tarif per sesi.
- 6) Penggalangan dana dan pembagian insentif yang adil bagi para *mentor*: Aplikasi harus memiliki metode penggalangan dana untuk perawatan dan pengembangan di masa depan, serta sistem insentif yang adil bagi para *mentor* yang akan menjadi bagian penting dari aplikasi.

b. Desain Solusi Perangkat Lunak

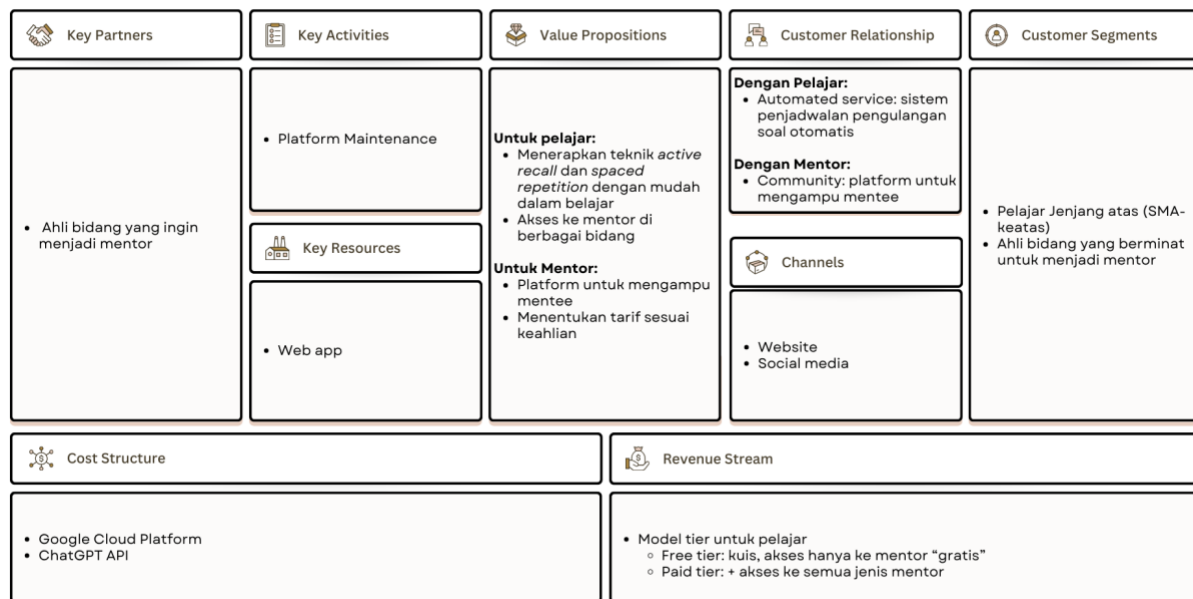
Berdasarkan analisis kebutuhan, berikut adalah desain solusi untuk perangkat lunak “Repetit”:

- 1) Modul Penjadwalan Pengulangan Materi: Aplikasi akan menggunakan teknik SRL untuk melakukan penjadwalan pengulangan materi oleh pengguna. Berdasarkan informasi ini, aplikasi akan secara dinamis menyesuaikan jadwal pengulangan berdasarkan kemajuan dan kinerja pengguna.
- 2) Modul Pembuatan *flashcard* Otomatis: Aplikasi akan mengombinasikan teknologi AI dan NLP untuk membuat berbagai jenis *flashcard* uji kompetensi untuk pengguna.
- 3) Modul Penilaian *flashcard* Otomatis: Aplikasi akan menggunakan teknologi AI dan NLP untuk melakukan penilaian otomatis terhadap *flashcard* uji yang telah dikerjakan oleh pengguna. Melalui informasi tersebut, aplikasi akan memberikan umpan balik secara langsung untuk mengevaluasi pemahaman pengguna selama masa pembelajaran.
- 4) Antarmuka Pengguna: Aplikasi ini akan memiliki desain antarmuka yang intuitif dan mudah digunakan dengan menampilkan instruksi yang jelas, visualisasi yang menarik, dan navigasi yang sederhana.

- 5) Fleksibilitas dalam Reservasi Sesi *Mentoring*: Aplikasi akan menyediakan reservasi sesi *mentoring* yang fleksibel. Melalui ini, aplikasi akan menyediakan *mentor* yang kompeten untuk mendapatkan konsultasi dan bimbingan yang dipersonalisasi oleh pengguna. Selain itu, aplikasi akan menyediakan akses kepada mentor-mentor tanpa biaya agar semua pelajar dapat terfasilitasi kebutuhannya.
- 6) Penggalangan dana dan pembagian insentif yang adil bagi para *mentor*: Aplikasi akan menerapkan sistem langganan, di mana pelajar dapat berlangganan untuk menghilangkan iklan, serta melakukan *mentoring* dengan mentor-mentor berbayar. Dana yang didapatkan dari pelajar nantinya akan digunakan untuk perawatan dan pengembangan lanjutan serta memberikan insentif bagi para mentor gratis. Bagi para mentor berbayar, mereka dapat memasang tarif per sesi sesuai tarif pasaran. Aplikasi mengambil 10% dari tarif yang dipasang mentor untuk perawatan dan pengembangan lanjutan serta memberikan insentif bagi para mentor gratis. Secara detilnya dapat dilihat pada *business model canvas* pada Gambar Y.

### Business Model Canvas v0.1

Repetit: Sistem Asisten Belajar berbasis AI untuk Meningkatkan Retensi Pembelajaran Siswa

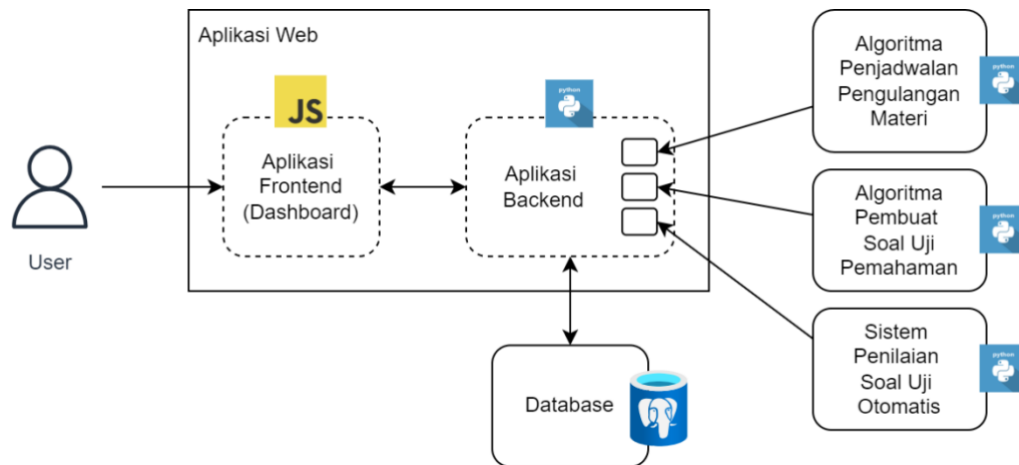


Gambar Y. *Business Model Canvas* Aplikasi Repetit.

## VII. Implementasi Perangkat Lunak

### a. Arsitektur Aplikasi

Arsitektur yang digunakan pada Repetit ditunjukkan pada Gambar 3.



Gambar 5. Arsitektur Sistem Repetit.

#### 1) JavaScript

JavaScript adalah bahasa pemrograman yang digunakan dalam pengembangan aplikasi web "Repetit". Bahasa ini akan digunakan untuk mengimplementasikan logika sisi klien, mengelola interaksi dengan pengguna, dan memastikan bahwa antarmuka pengguna berfungsi dengan baik di berbagai perangkat.

#### 2) Python

Python merupakan bahasa pemrograman yang akan digunakan untuk pengembangan algoritma pemrosesan data dan kecerdasan buatan (AI) dalam aplikasi "Repetit". Python memungkinkan untuk tugas-tugas seperti analisis data, pemrosesan bahasa alami (NLP), dan pembelajaran mesin yang penting untuk menyediakan fitur-fitur seperti penilaian otomatis, rekomendasi pembelajaran adaptif, dan personalisasi konten.

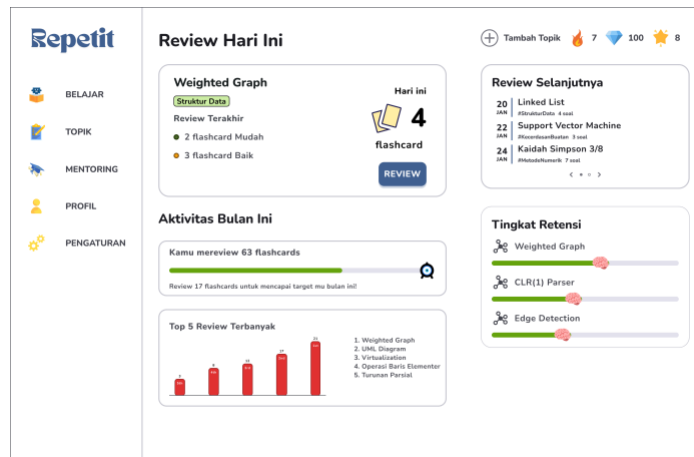
#### 3) Google Cloud Platform

Google Cloud Platform akan digunakan untuk infrastruktur cloud aplikasi "Repetit". GCP menyediakan berbagai layanan cloud yang memungkinkan aplikasi untuk skala dan bekerja dengan efisien. Aplikasi "Repetit" akan menggunakan GCP untuk menyimpan data pengguna dan informasi yang berkaitan.

## VIII. Hasil dan Pembahasan

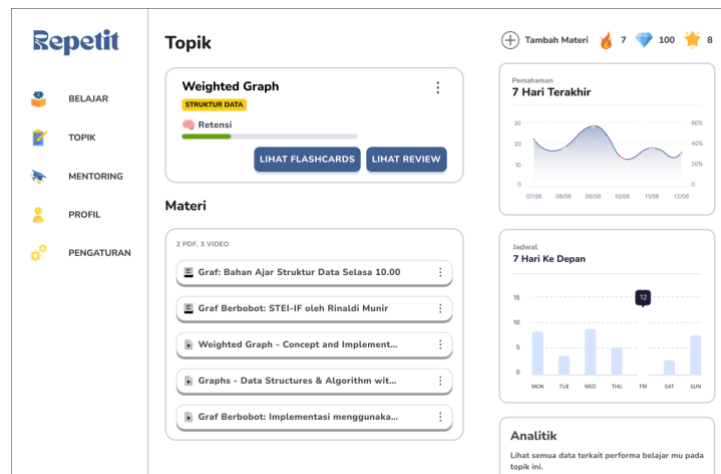
### a. Hasil Implementasi

Implementasi aplikasi Repetit telah berhasil dilaksanakan sesuai dengan rencana pengembangan seperti yang sudah diuraikan pada bagian II. Seluruh fitur utama sesuai hasil analisis kebutuhan telah terintegrasi dengan baik. Gambar 6 menunjukkan tampilan utama aplikasi Repetit.



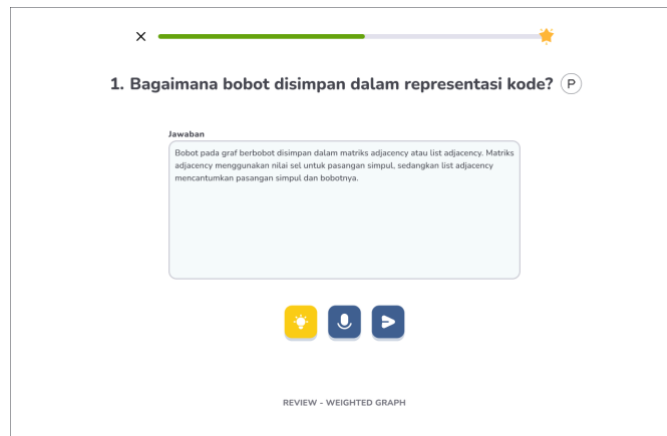
Gambar 6. Tampilan Utama Aplikasi Repetit

Melalui tampilan utama aplikasi ini seperti di Gambar 6, pengguna dapat melihat *flashcard* yang harus diselesaikan hari ini dan mengerjakan *flashcard* tersebut. Pengguna juga dapat melihat jadwal *review flashcard* selanjutnya yang berada pada bagian kanan di Gambar 6. Adapun penambahan materi belajar dapat dilakukan pada halaman Topik seperti yang terlihat pada Gambar 7.



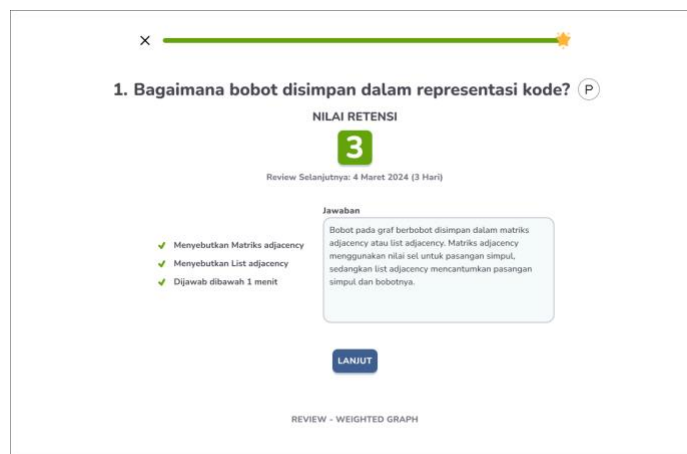
Gambar 7. Halaman Topik Aplikasi Repetit

Pengguna dapat mengerjakan *flashcard* dengan tampilan seperti yang ditunjukkan pada Gambar 8.



Gambar 8. Tampilan Mengerjakan Flashcard Aplikasi Repetit

Setelah mengirimkan jawaban, pengguna akan mendapatkan penilaian otomatis atas jawaban sebelumnya sebagai tolak ukur tingkat retensi pengguna. Pengguna juga akan langsung mendapatkan jadwal untuk *me-review flashcard* selanjutnya yang tampilannya dapat dilihat pada Gambar 9.



Gambar 9. Tampilan Penilaian Otomatis dan Jadwal Review Selanjutnya

## b. Pengujian Sistem

### 1) Pengujian Internal

Pengujian internal dilakukan oleh tim pengembang untuk memastikan stabilitas dan fungsionalitas aplikasi sebelum diuji oleh pengguna eksternal. Berikut adalah hasil setelah dilakukan pengujian internal: fitur penjadwalan pengulangan materi dapat berfungsi sesuai dengan metode SRL, memastikan materi diulang pada interval waktu yang optimal untuk meningkatkan retensi pembelajaran. Adapun fitur pembuatan *flashcard* otomatis mampu menghasilkan *flashcard* yang relevan dan lengkap sesuai dengan materi yang dipelajari, mendukung penggunaan metode *Active Recall*. Selain itu, fitur penilaian otomatis

menggunakan NLP untuk menilai jawaban pengguna, memberikan umpan balik yang konstruktif. Fitur Mentoring juga telah memungkinkan bagi pengguna untuk menjadwalkan sesi mentoring sesuai ketersediaan mentor dan waktu mereka. Adapun antarmuka intuitif juga telah dimiliki dengan *User Interface* (UI) yang responsif dan mudah dinavigasi

## 2) Pengujian Pengguna

Pengujian menggunakan metode *black-box testing* dilakukan oleh pengguna eksternal untuk mengevaluasi fungsionalitas aplikasi tanpa melihat kode sumber. Pengguna diminta untuk menggunakan aplikasi Repetit secara menyeluruh, termasuk mencoba semua fitur utama seperti penjadwalan pengulangan materi, pembuatan *flashcard* otomatis, penilaian otomatis, dan reservasi mentor. Hasil pengujian aplikasi Repetit menggunakan metode *black-box* dapat dilihat pada Tabel I.

**TABEL I**  
HASIL PENGUJIAN BLACK-BOX APLIKASI REPETIT

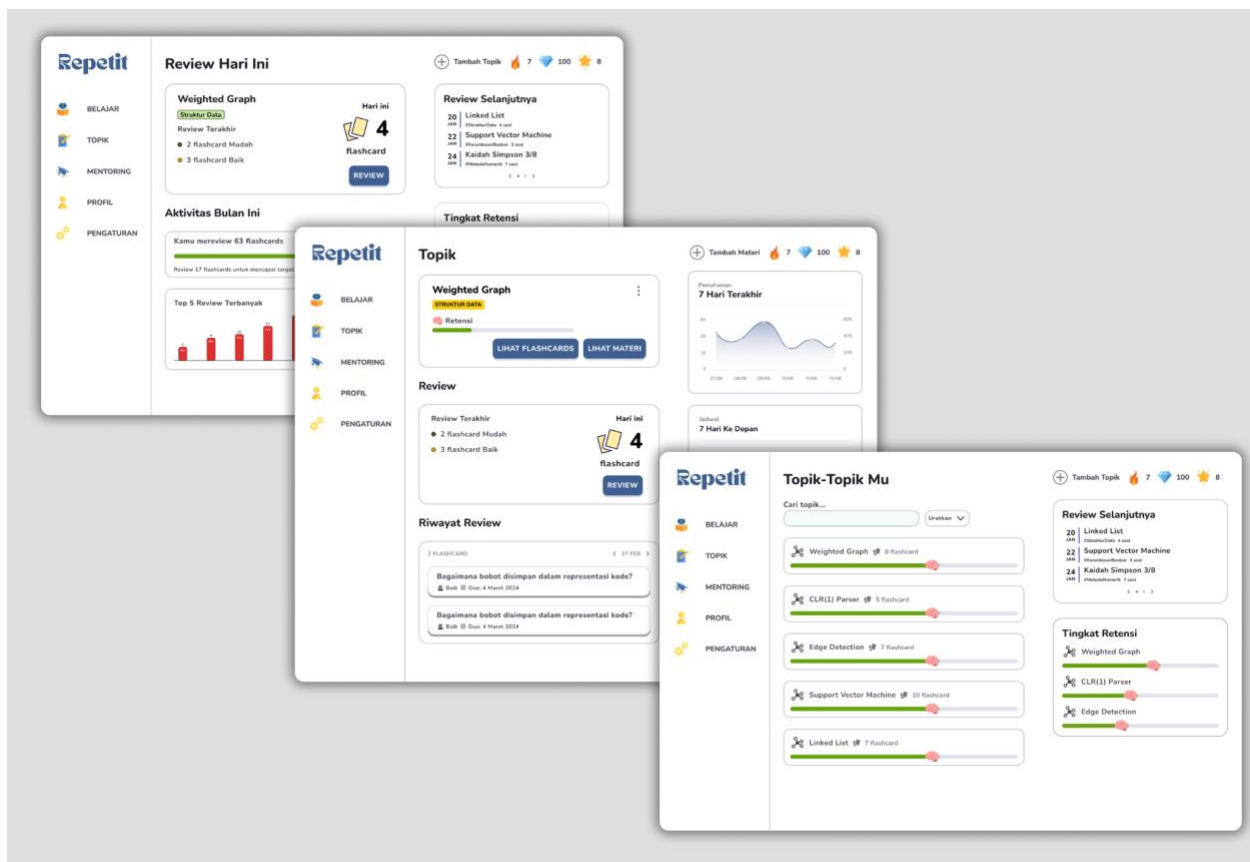
No	Aktivitas Pengujian	Ekspektasi	Actual Result	Perbaikan
1	Mencari “Repetit” di browser	Pengguna memasuki halaman utama aplikasi Repetit	Memasuki halaman utama	Baik
2	Login menggunakan akun google	Pengguna login menggunakan akun Google	Berhasil login	Baik
3	Menambahkan topik	Pengguna diarahkan pada pop-up yang dapat diinput judul dan tag topik	Berhasil menambahkan topik	Baik
4	Cek halaman Topik	Pengguna diarahkan halaman Topik yang berisi list topik yang sudah diinput	Halaman Topik berhasil ditampilkan	Baik
5	Menambahkan materi di halaman Topik	Pengguna diarahkan ke pop-up yang dapat diinput materi dari topik.	Berhasil menambahkan materi	Baik
6	Membuat flashcards otomatis	Pengguna dapat membuat flashcard dari materi yang sudah diinput sebelumnya	Berhasil membuat flashcard	Baik
7	Mengerjakan flashcard	Pengguna dapat mengerjakan flashcard yang dibuat oleh AI	Berhasil mengerjakan Flashcard	Baik
8	Mengirimkan jawaban Flashcard	Pengguna dapat mengirimkan jawaban atas flashcard yang dibuat	Berhasil mengirim jawaban	Baik
9	Mendapatkan hasil penilaian jawaban otomatis	Aplikasi memberikan penilaian jawaban pengguna	Berhasil memberikan penilaian	Baik
10	Mendapatkan jadwal pengulangan kembali	Aplikasi memberikan jadwal pengulangan berdasarkan retensi pengguna	Berhasil memberikan jadwal	Baik
11	Mendapatkan reward setelah pengerjaan flashcard	Pengguna berhasil mendapatkan reward setelah mengerjakan flashcard	Berhasil mendapat reward	Baik
12	Cek halaman Mentoring	Pengguna diarahkan halaman Mentoring	Halaman Mentoring berhasil ditampilkan	Baik
13	Memilih mentor untuk konsultasi.	Pengguna dapat memesan sesi mentoring pada mentor yang tersedia	Berhasil memesan sesi mentoring	Baik

Berdasarkan hasil uji internal dan uji eksternal menggunakan metode *black-box*, Aplikasi Repetit telah mampu untuk diluncurkan sesuai dengan perancangan yang telah direncanakan. Maka dapat disimpulkan bahwa aplikasi Repetit dinyatakan baik untuk dapat digunakan.

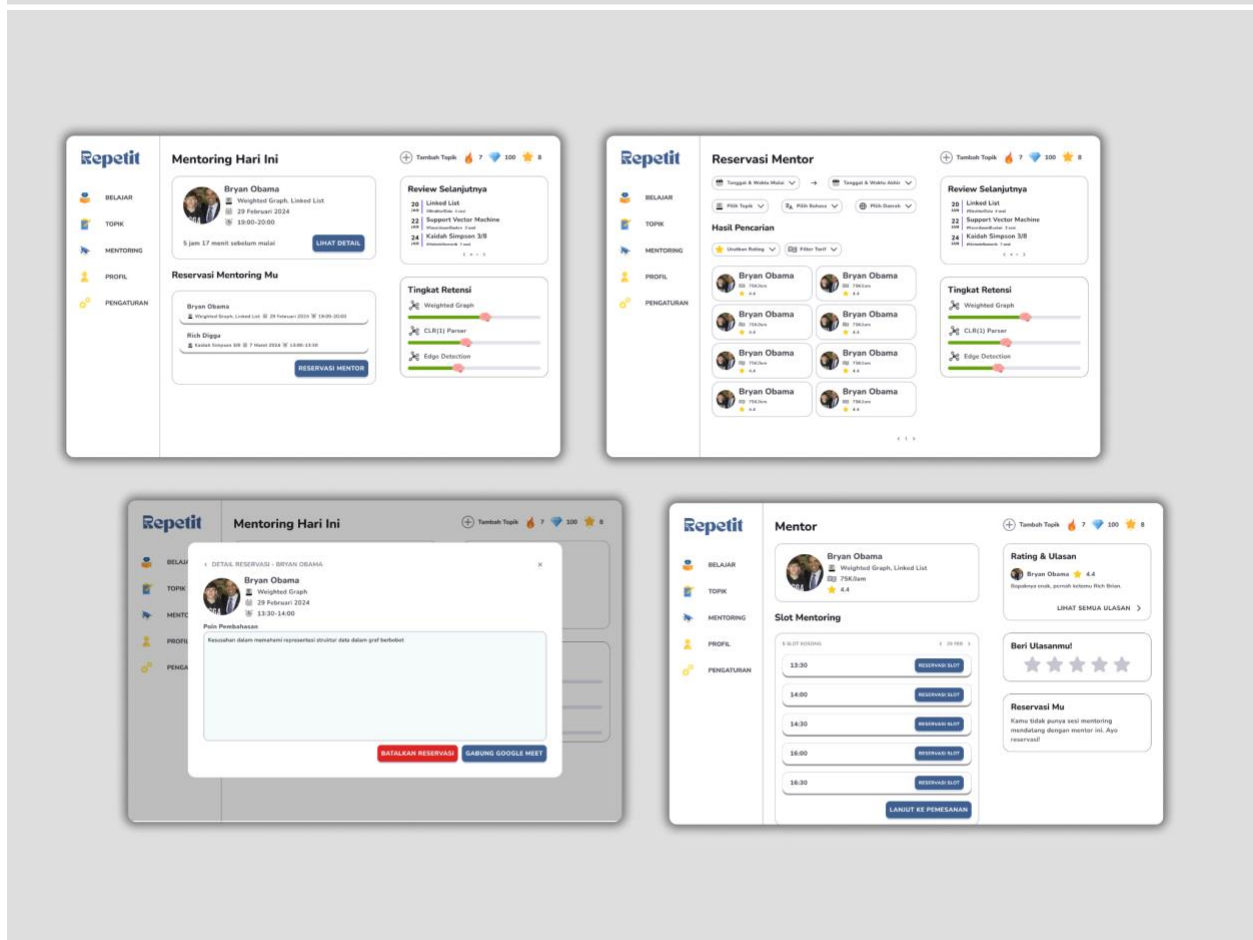
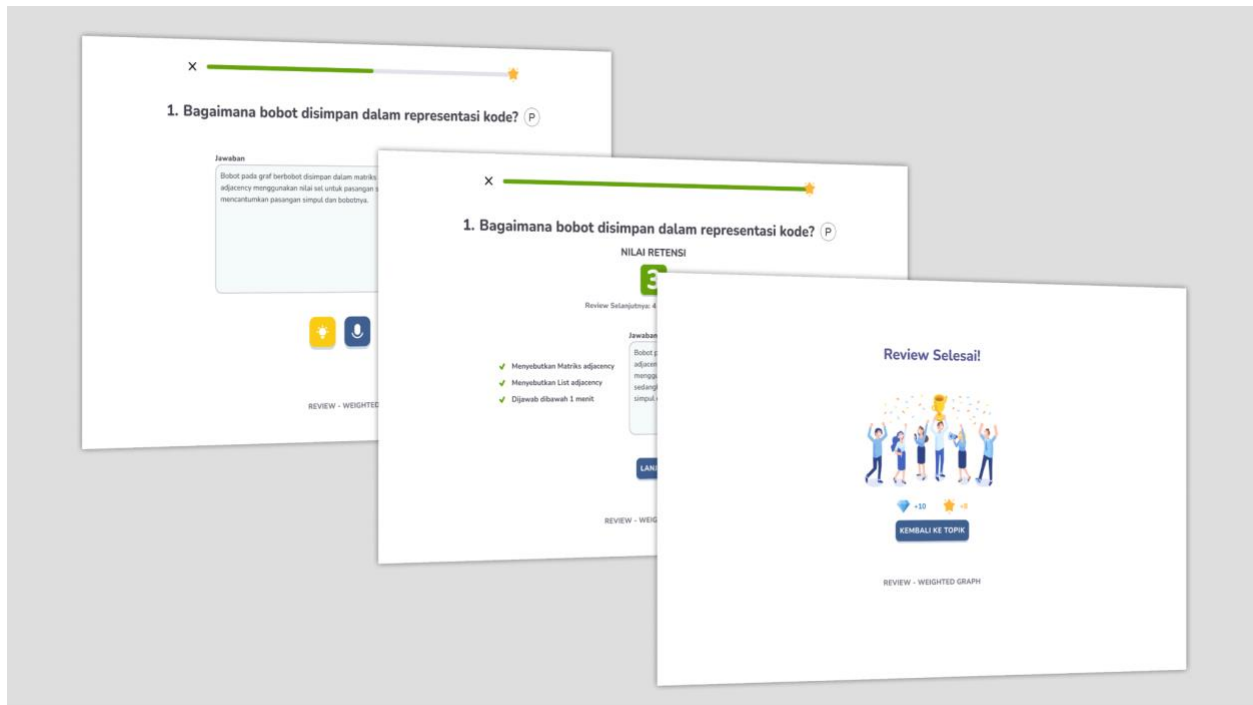
## IX. Kesimpulan

Pengembangan aplikasi Repetit sebagai sistem asisten belajar berbasis AI berhasil mencapai tujuannya dalam meningkatkan retensi pembelajaran siswa melalui penerapan metode *Active Recall* dan SRL. Hasil pengujian internal dan eksternal menunjukkan bahwa aplikasi ini tidak hanya berfungsi sesuai dengan perancangan awal, tetapi juga memberikan manfaat nyata bagi pengguna, terutama dalam hal peningkatan retensi pengetahuan dan kemudahan penggunaan. Dengan keberhasilan implementasi dan pengujian ini, aplikasi Repetit siap diluncurkan dan diharapkan dapat memberikan kontribusi yang signifikan dalam bidang pendidikan, terutama dalam memfasilitasi akses terhadap metode belajar yang lebih efektif dan inovatif.

## X. Screenshot Mockup





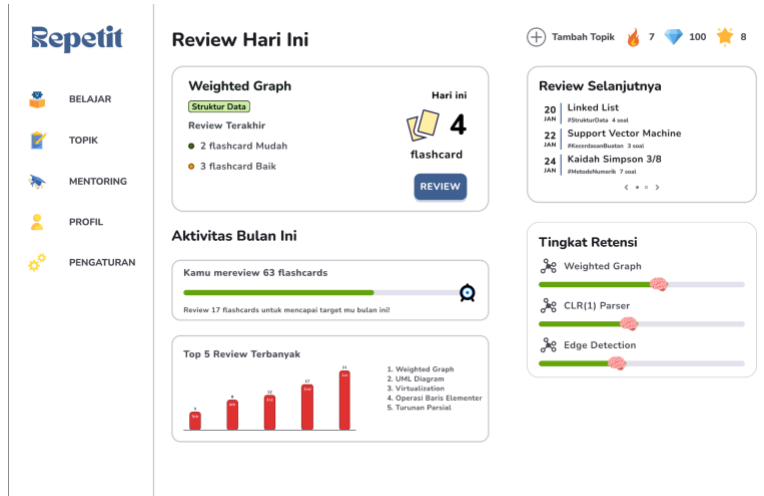


Gambar 6, 7, & 8. Mockup aplikasi Repetit.

## XI. Dokumentasi Penggunaan

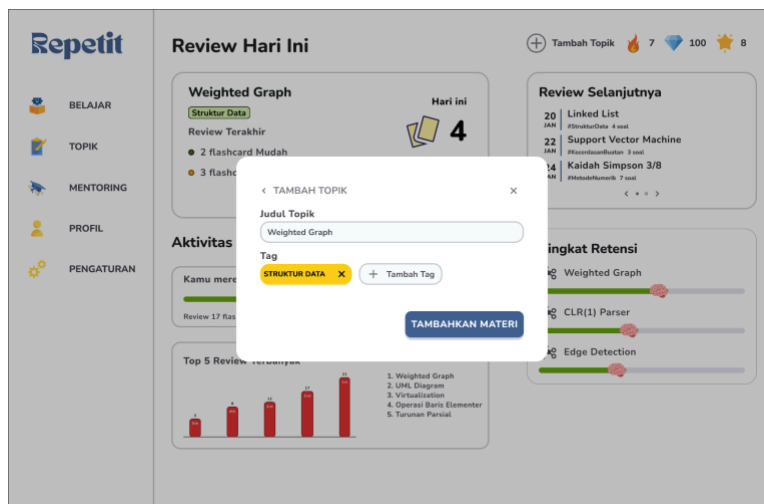
### Menambahkan Topik

1. Klik **Tambah Topik** di bagian atas kanan.



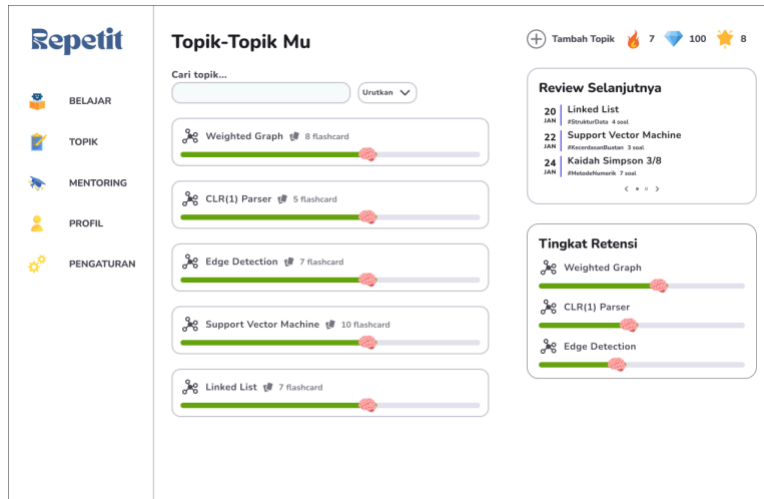
Gambar 9. Laman belajar (laman *home*).

2. Masukkan judul topik dan tag jika ada.



Gambar 10. *Popup* tambah topik.

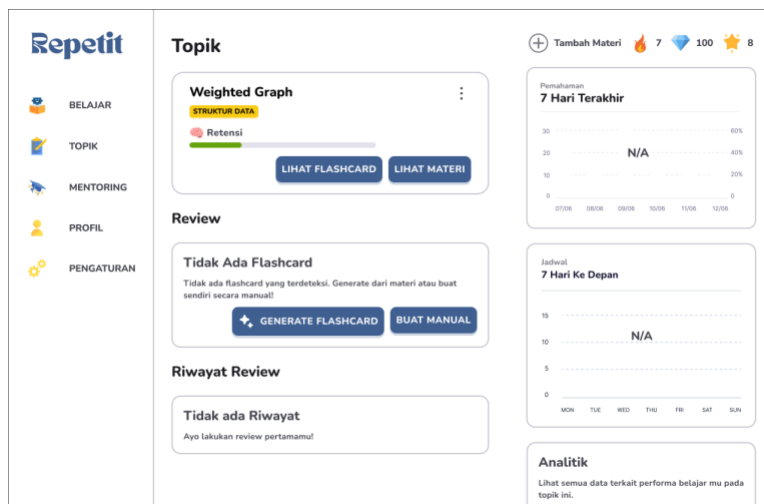
3. Topik dapat dilihat di laman Topik, yang dapat diakses melalui tombol **Topik** di *sidebar*.



Gambar 11. Laman kumpulan topik.

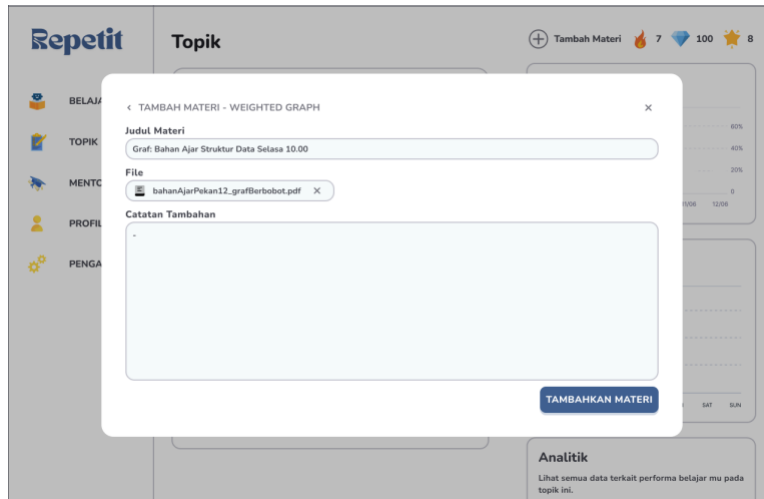
### Men-generate Flashcard Dari Materi

1. Navigasi ke salah satu topik.
2. Klik **Tambah Materi** di bagian atas kanan.



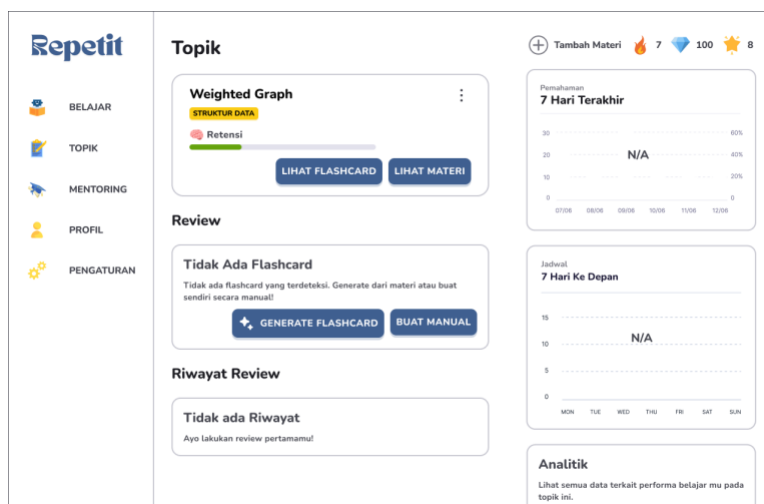
Gambar 12. Laman topik.

3. Masukkan judul materi, file materi, dan catatan tambahan mengenai materi jika ada, lalu klik **Tambahkan Materi**. Pengguna dianjurkan untuk menambahkan 2 atau lebih bahan materi agar sistem dapat menghasilkan *flashcard* yang berkualitas.



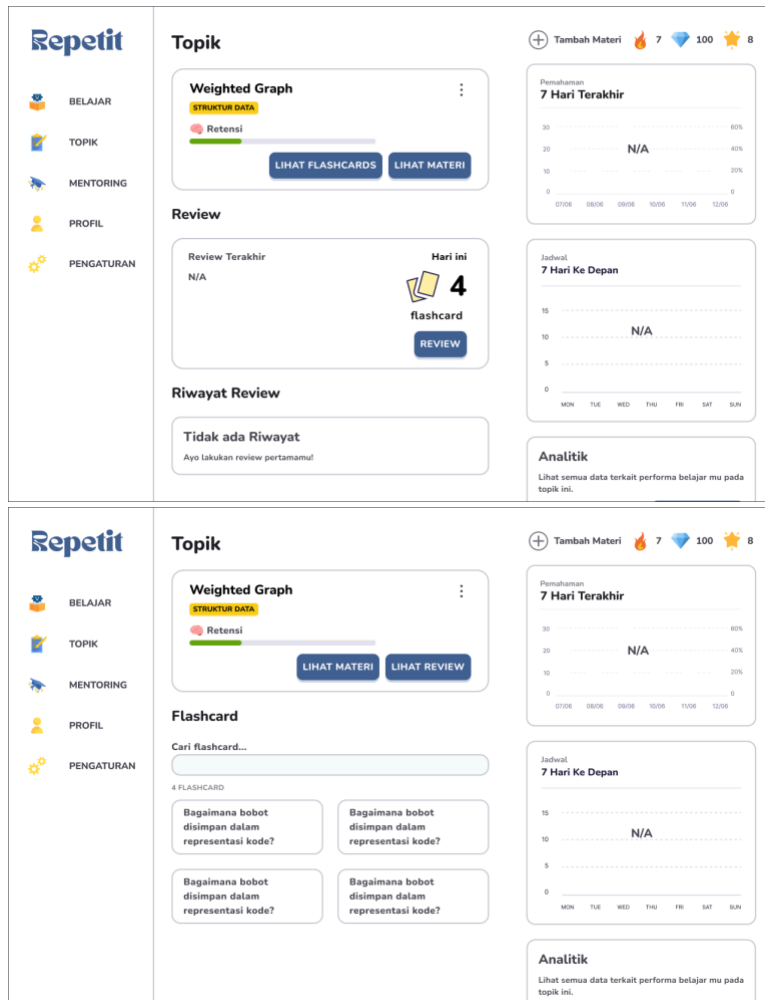
Gambar 13. *Popup* tambah materi.

4. Klik **Generate Flashcard** untuk men-*generate flashcard*, tunggu beberapa saat.



Gambar 14. Laman topik.

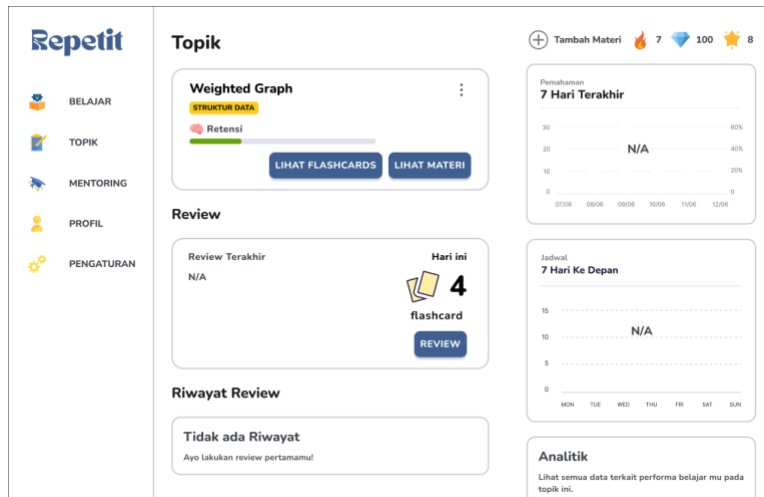
5. Flashcard sudah terbuat dan bisa langsung direview. Untuk melihat semua flashcard yang dihasilkan, klik **Lihat Flashcard**.



Gambar 15 & 16. Laman topik dan laman topik bagian *flashcard*.

### Melakukan *Review*

1. Navigasi ke salah satu topik.
2. Di bawah *heading* Review, klik **Review**.



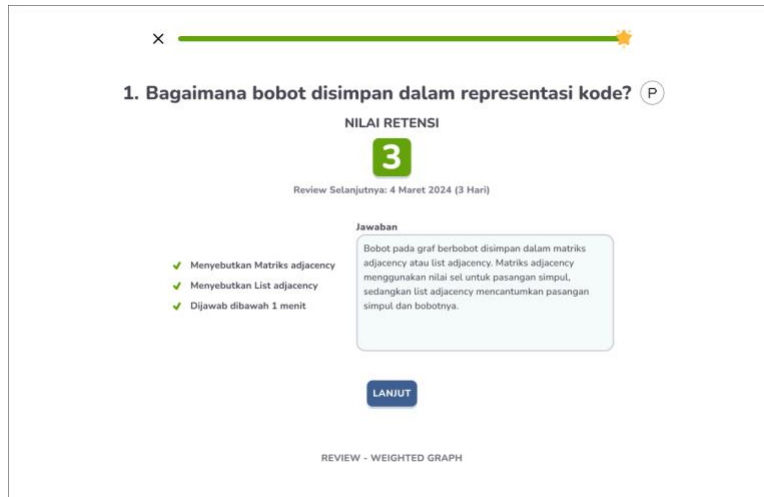
Gambar 17. Laman topik.

3. Jawab pertanyaan dengan cara mengetik di kotak teks atau klik tombol **Microphone**.



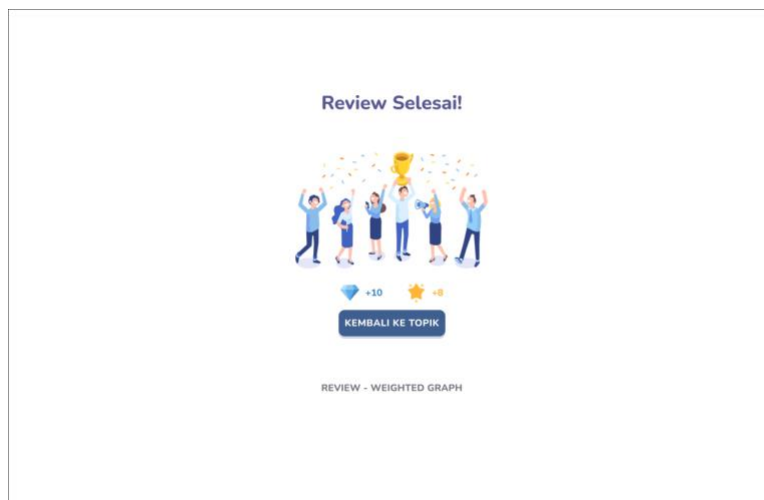
Gambar 18. Laman *review flashcard*.

4. Jika sudah puas dengan jawaban, klik tombol **Kirim**.
5. Pengguna akan menerima feedback dari jawaban yang diberi, termasuk nilai retensi, jadwal *review* selanjutnya, dan kriteria jawaban benar mana saja yang terpenuhi. Klik **Lanjut**.



Gambar 19. Laman evaluasi *flashcard*.

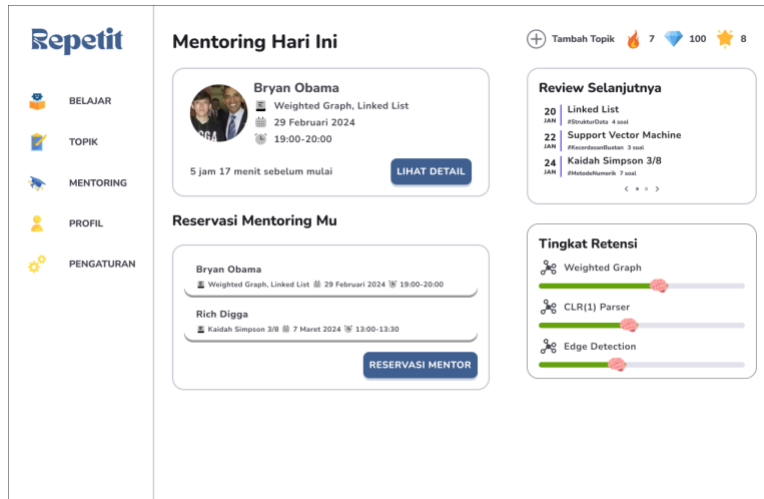
## 6. *Review* selesai.



Gambar 20. Laman selesai *review*.

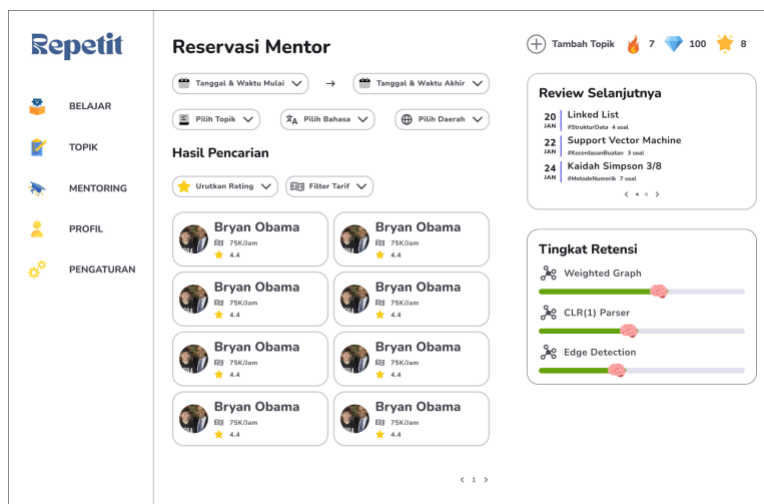
## Melakukan Reservasi Sesi Mentoring

1. Klik **Mentoring** pada *sidebar*.
2. Di bawah *heading* Reservasi Mentoring Mu, klik **Reservasi Mentor**.



Gambar 21. Laman *mentoring*.

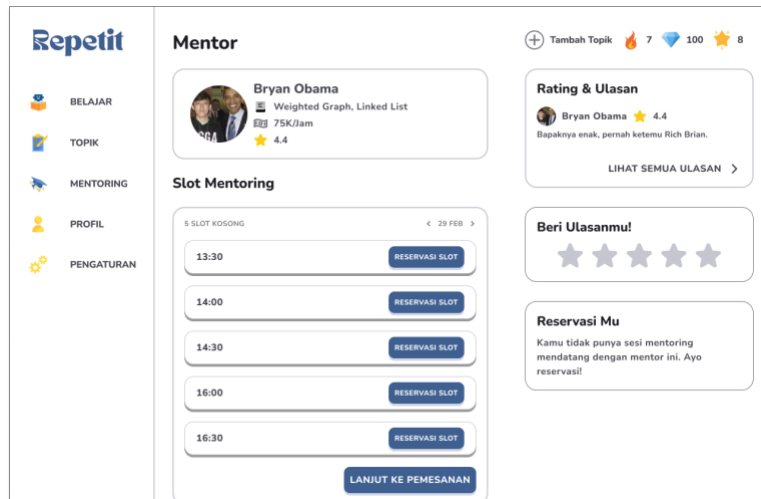
3. Cari mentor menggunakan elemen form yang diberikan yaitu **tanggal dan waktu mulai, tanggal dan waktu akhir, topik, bahasa, dan daerah**. Hasil pencarian akan muncul di bawah *heading* Hasil Pencarian. Klik salah satu mentor untuk melanjutkan reservasi.



Gambar 22. Laman pencarian *mentor*.

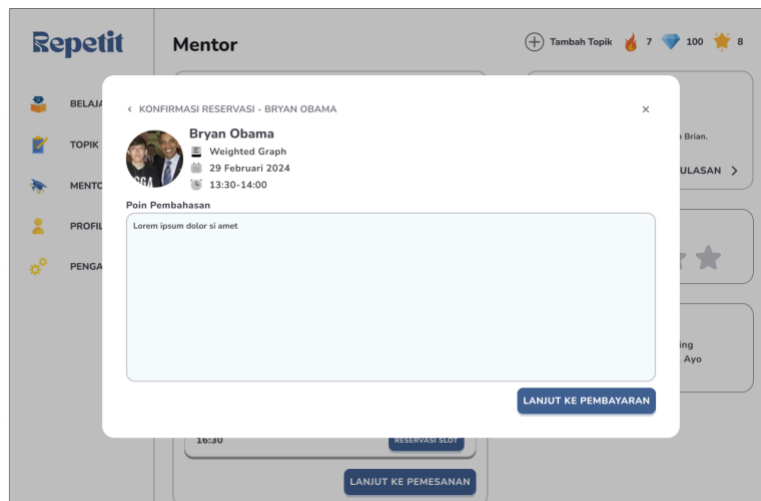
4. Di bawah heading Slot Mentoring, klik **Reservasi Slot** untuk melakukan reservasi pada slot yang tersedia. Pengguna dapat melakukan reservasi pada beberapa slot. Klik **Lanjut ke Pemesanan**.





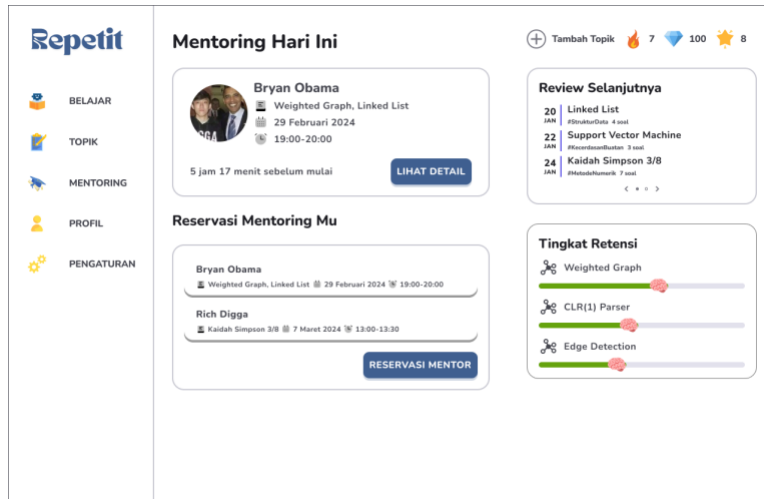
Gambar 23. Laman profil *mentor*.

5. Isi kotak teks dengan poin-poin yang ingin dibahas pada saat sesi mentoring nanti. Klik **Lanjut ke Pembayaran**.



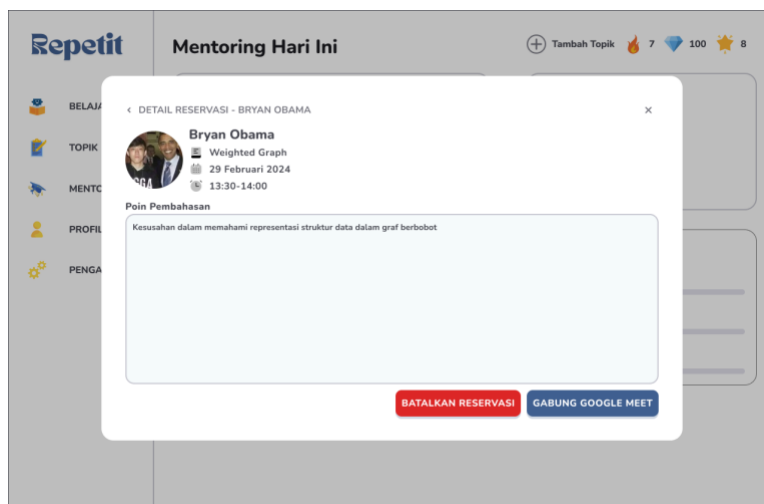
Gambar 24. *Popup* konfirmasi reservasi *mentoring*.

6. Pengguna akan diberi pilihan metode pembayaran, di mana setiap metode pembayaran akan (1) me-redirect pengguna ke aplikasi metode pembayaran, atau (2) memberi nomor *virtual account* untuk dapat melakukan pembayaran.
7. Setelah pembayaran selesai, sesi akan muncul di laman **Mentoring**. Klik **Lihat Detail**.



Gambar 25. Laman *mentoring*.

8. Saat sudah waktunya, klik **Gabung Google Meet**. Pengguna akan *diredirect* ke ruangan Google Meet dengan mentor.



Gambar 26. *Popup detail reservasi mentoring*.

## REFERENSI

- [1] E. Germani, E. Fromont, and C. Maumet, "On the benefits of self-taught learning for brain decoding," *Gigascience*, vol. 12, 2023, doi: 10.1093/gigascience/giad029.
- [2] Q. Gu, F. Ahmad, and T. Sumner, "Improving conceptual learning through customized knowledge visualization," in *3rd International Conference on Knowledge Discovery and Data Mining, WKDD 2010*, 2010. doi: 10.1109/WKDD.2010.68.
- [3] J. Yao, Z. Dou, J. Y. Nie, and J. R. Wen, "Looking Back on the Past: Active Learning with Historical Evaluation Results," *IEEE Trans Knowl Data Eng*, 2020, doi: 10.1109/TKDE.2020.3045816.

- [4] A. M. Abbas, T. Hamid, C. Iwendi, F. Morrissey, and A. Garg, "Improving Learning Effectiveness by Leveraging Spaced Repetition (SR)," in *Lecture Notes in Electrical Engineering*, 2023. doi: 10.1007/978-981-99-1051-9\_10.
- [5] J. Xu *et al.*, "Active recall strategies associated with academic achievement in young adults: A systematic review," *Journal of Affective Disorders*, vol. 354, 2024. doi: 10.1016/j.jad.2024.03.010.
- [6] J. M. Ortega-Tudela, M. T. Lechuga, and C. J. Gómez-Ariza, "A specific benefit of retrieval-based concept mapping to enhance learning from texts," *Instr Sci*, vol. 47, no. 2, 2019, doi: 10.1007/s11251-018-9476-y.
- [7] P. Musfeld, A. S. Souza, and K. Oberauer, "Repetition learning is neither a continuous nor an implicit process," *Proc Natl Acad Sci U S A*, vol. 120, no. 16, 2023, doi: 10.1073/pnas.2218042120.
- [8] M. Caffrey and S. Commins, "Preservation of long-term memory in older adults using a spaced learning paradigm," *Eur J Ageing*, vol. 20, no. 1, 2023, doi: 10.1007/s10433-023-00750-5.
- [9] J. Smith *et al.*, "Placeholder Text: A Study," *Citation Styles*, vol. 3, Jul. 2021, doi: 10.10/X.
- [10] A. I. Choe, S. Woodard, B. M. Thompson, V. Walter, J. S. Fotos, and C. J. Kasales, "Spaced Education: Randomized Trial Comparing Learning Efficiency of the Adaptive Versus Nonadaptive Spaced Education Systems Among Radiology Residents," *Journal of the American College of Radiology*, vol. 19, no. 6, 2022, doi: 10.1016/j.jacr.2022.03.010.
- [11] P. T. M. Marope, "Quality and development-relevant education and learning: Setting the stage for the Education 2030 Agenda," *Prospects*, vol. 46, no. 1, 2016. doi: 10.1007/s11125-016-9387-0.
- [12] shiohira kelly, "Understanding the impact of artificial intelligence on skills development," *United Nations Educational, Scientific and Cultural Organization*, 2021.
- [13] R. F. Kizilcec, "To Advance AI Use in Education, Focus on Understanding Educators," *International Journal of Artificial Intelligence in Education*, vol. 34, no. 1, 2024. doi: 10.1007/s40593-023-00351-4.
- [14] O. Hamal, N. E. El Faddouli, M. H. Alaoui Harouni, and J. Lu, "Artificial Intelligent in Education," *Sustainability (Switzerland)*, vol. 14, no. 5, 2022, doi: 10.3390/su14052862.
- [15] J. Su, J. Ye, L. Nie, Y. Cao, and Y. Chen, "Optimizing Spaced Repetition Schedule by Capturing the Dynamics of Memory," *IEEE Trans Knowl Data Eng*, vol. 35, no. 10, 2023, doi: 10.1109/TKDE.2023.3251721.