

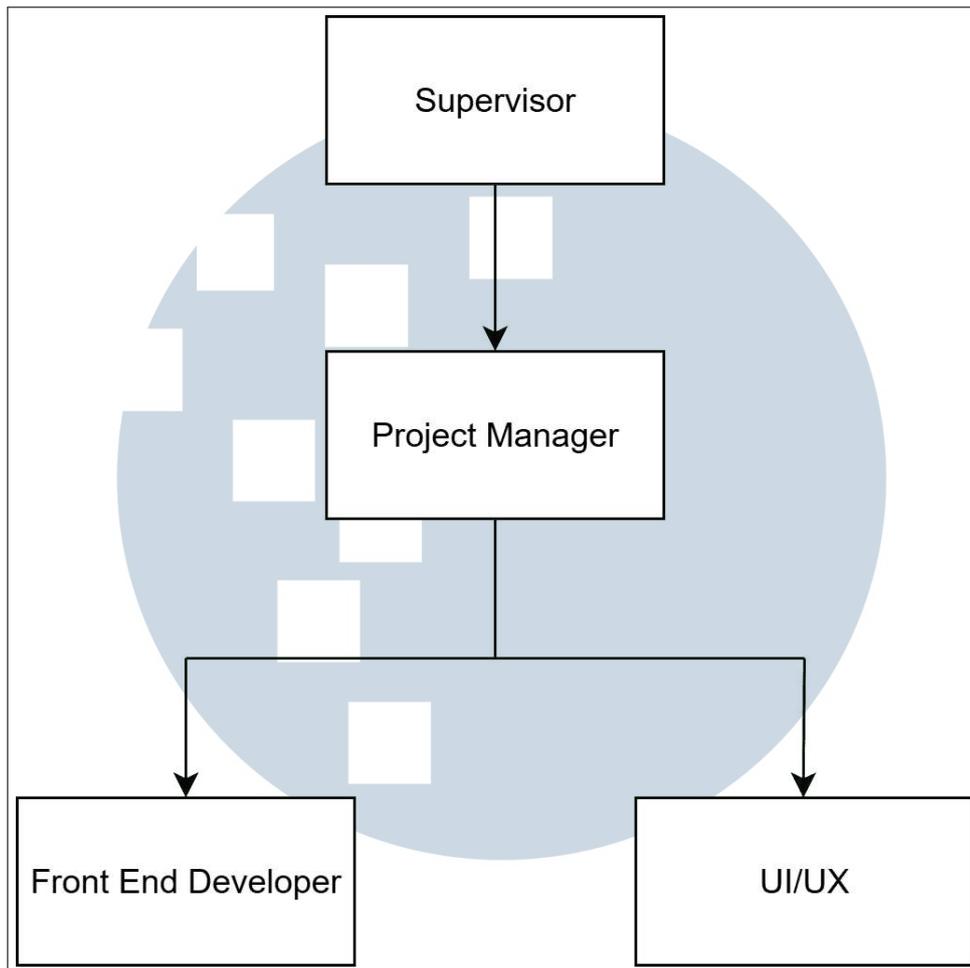
## BAB 3

### PELAKSANAAN KERJA MAGANG

#### 3.1 Kedudukan dan Koordinasi

Selama proses kerja magang di PT Flow Byte Digital, kedudukan yang ditempati adalah *Frontend Developer* yang berada pada divisi Teknologi. Divisi Teknologi memiliki 3 anggota, dengan 1 anggota sebagai *Project Manager* yang bertanggung jawab mengatur jalannya sebuah proyek yang sedang dikerjakan sementara dua anggota lainnya berfokus pada pengembangan antarmuka pengguna (*user interface*) menggunakan Vue.js Proyek dikembangkan menggunakan Visual Studio Code (VSCode) sebagai *Integrated Development Environment (IDE)* utama. Selama proses pengerjaan proyek, komunikasi dan koordinasi tim dilakukan melalui *platform* Discord dan *website* ClickUp. *Daily meeting* diadakan setiap hari pada pukul 21.00 WIB untuk membahas perkembangan proyek LMS Flowcamp dan memberikan *feedback* terhadap hasil kerja masing-masing anggota tim.





Gambar 3.1. Koordinasi Tugas Magang

Gambar 3.1 menggambarkan alur koordinasi dalam proses pemberian tugas magang pada sebuah proyek. Proses dimulai dari *Supervisor* yang berperan sebagai pembimbing utama, di mana ia terlebih dahulu berdiskusi dan berkoordinasi dengan *Project Manager* untuk membahas rencana kerja. Setelah mendapatkan arahan dari *Supervisor*, *Project Manager* akan mendistribusikan tugas dan memberikan instruksi secara langsung kepada tim teknis, yaitu Front End Developer dan tim UI/UX, sesuai dengan bidang dan tanggung jawab masing-masing.

### 3.2 Tugas yang Dilakukan

Selama pelaksanaan kegiatan magang di PT Flow Byte Digital, tugas-tugas yang diberikan berfokus pada pengembangan sisi *frontend* untuk sistem *Learning Management System* (LMS). Seluruh aktivitas magang diarahkan pada satu proyek utama, yaitu pengembangan sistem LMS bernama FlowCamp ID. Penugasan

diberikan langsung oleh *Project Manager* dan dilaksanakan sesuai dengan *timeline* proyek yang telah ditentukan. Ruang lingkup pekerjaan mencakup implementasi desain UI dari Figma ke dalam tampilan *website* menggunakan Vue.js dan Tailwind CSS, pengembangan komponen-komponen *reusable*, validasi *form*, pengelolaan *routing* menggunakan vue-router, serta penerapan *page protection*. Selain itu, dilakukan pula optimasi performa sisi *frontend* melalui teknik *lazy loading* dan *code splitting*. Proses pengembangan menggunakan Git sebagai alat *version control*, sedangkan koordinasi tim dilaksanakan melalui *daily stand-up meeting* yang dilakukan secara *online* menggunakan *platform* Discord.

### 3.3 Uraian Pelaksanaan Magang

Uraian pelaksanaan kerja magang selama Di PT Flow Byte Digital diuraikan seperti pada Tabel 3.1.

Tabel 3.1. Pekerjaan yang dilakukan tiap minggu

Minggu Ke -	Pekerjaan yang dilakukan
1	Melakukan pembelajaran dan <i>research</i> lebih lanjut terkait Vue.js dan Tailwind CSS, seperti struktur proyek Vue, penerapan komponen dan instalasi Tailwind CSS dalam proyek Vue, termasuk pengaturan file <code>tailwind.config.js</code> .
2	Mengerjakan UI/UX untuk <i>Landing Page Home</i> dan mengerjakan komponen <i>navbar footer</i> untuk <i>Landing Page Home</i> .

UIN  
UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

Minggu Ke -	Pekerjaan yang dilakukan
3	Mengerjakan halaman <i>Landing Page Free Class</i> , dimulai dari <i>slicing</i> tampilan ketika tidak ada kelas yang tersedia (menampilkan pesan “ <i>No Class Found</i> ”. Setelah itu, melanjutkan ke versi halaman saat data tersedia, termasuk membuat layout daftar kelas dengan filter kategori (All Program, Back-End, Front-End, dsb) dan tombol <i>See More</i> .
4	Mengerjakan halaman <i>Landing Page Bootcamp</i> , yang berisi informasi program <i>bootcamp</i> untuk <i>user</i> .
5	Melakukan <i>slicing</i> desain halaman <i>register</i> sesuai desain UI/UX yang telah dibuat.
6	Mengerjakan UI halaman <i>Forgot Password</i> Melakukan integrasi dengan halaman <i>register</i> agar data tetap sinkron.
7	Mengerjakan <i>Dashboard Overview</i> mentor, serta bagian <i>Ongoing Classes</i> yang menampilkan progres belajar setiap kelas dan mengerjakan komponen <i>Performance Students</i> dengan <i>dropdown</i> filter dan tabel nilai siswa berdasarkan kelas yang dipilih.
8	Melakukan <i>slicing</i> desain halaman <i>All Classes</i> pada <i>dashboard</i> mentor, termasuk <i>layout grid</i> kartu kelas, elemen gambar, judul, pengajar, level, dan <i>rating</i> .
9	Melakukan <i>slicing</i> halaman <i>Detail Kelas (Summary)</i> yang berisi tentang informasi nilai dan menambahkan komponen <i>rating chart</i> beserta <i>button</i> untuk <i>rating</i> dan membuat <i>layout</i> halaman <i>Review</i> yang akan menampilkan <i>comment</i> dari <i>user</i> .
10	Mengerjakan halaman <i>Detail Class Modul</i> , yang menampilkan daftar modul pembelajaran seperti <i>file</i> PPT, video rekaman, dan tugas yang berkaitan dengan topik tertentu.
11	Mengerjakan <i>pop up</i> untuk halaman <i>Detail Class Modul</i> yang digunakan untuk menambahkan materi pembelajaran. <i>Pop up</i> ini terdiri dari <i>dropdown</i> pemilihan topik, kolom deskripsi materi, dan input nama <i>file</i> atau <i>resource</i> .

Minggu Ke -	Pekerjaan yang dilakukan
12	Mengerjakan fitur <i>Preview List Modul</i> yang memungkinkan mentor untuk menambahkan materi, tugas, dan rekaman video ke dalam modul pembelajaran, serta pembuatan <i>form pop-up</i> untuk masing-masing jenis input (materi, tugas, dan video).
13	Mengerjakan fitur <i>Task Submission &amp; Grading</i> untuk mentor, yang digunakan untuk melihat dan menilai pengumpulan tugas dari siswa yang berisi halaman daftar tugas per kelas, tampilan tabel pengumpulan tugas dari tiap siswa lengkap dengan status ( <i>on time, overdue, ungraded</i> ), dan mengerjakan tampilan <i>pop-up</i> untuk memberikan <i>grade</i> untuk siswa.
14	Mengerjakan fitur <i>Student Progress Report</i> yang menampilkan detail performa siswa dalam mengikuti kelas tertentu. Fitur ini mencakup tampilan daftar tugas beserta nilai yang diperoleh, status kelulusan ( <i>Completed/On Going</i> ), serta total nilai secara keseluruhan dan mengerjakan komponen <i>pop-up</i> yang muncul saat mentor menekan salah satu siswa yang akan menampilkan status progres dan detail nilai.
15	Membantu mengintegrasikan bagian data agar logikanya sesuai dengan yang diharapkan menggunakan <i>local storage</i>
16	Melakukan <i>review</i> sederhana ke halaman yang sudah dibuat dan melakukan <i>resolve conflicts</i> di setiap <i>branch</i> Github yang masih belum <i>merge</i> .

- Pada minggu pertama kerja magang, *supervisor* memberikan penugasan berupa mempelajari Vue.js
- Pada minggu kedua hingga minggu ketiga, yang dilakukan pada kerja magang adalah melakukan *set up* pada *repository* di GitHub dan mulai mengerjakan *Landing Page*
- Pada minggu keempat hingga minggu kelima, yang dilakukan pada kerja magang mengerjakan UI untuk halaman *register* dan integrasi dengan halaman *Forgot Password*.
- Pada minggu keenam hingga minggu kelima belas, proses pengerjaan yang dilakukan adalah mengerjakan UI pada *dashboard* mentor beserta melakukan

integrasi data *dummy* pada halaman halaman yang dikerjakan.

- Pada minggu keenam belas, hal yang dikerjakan pada pelaksanaan magang adalah melakukan *review* terhadap apa yang telah dikerjakan.

### 3.4 Perancangan Sistem

#### 3.4.1 Teknologi yang Digunakan

*Website Flow Camp ID* dikembangkan menggunakan Vue.js 3 sebagai *frontend framework* yang bersifat progresif untuk membangun UI yang dinamis. Penggunaan komponen Vue.js memungkinkan pembuatan elemen UI yang dapat digunakan kembali, yang mengurangi redundansi dan menyederhanakan pemeliharaan serta skalabilitas di masa depan [5]. Selain itu, TailwindCSS sebagai *utility-first CSS framework* guna mempercepat proses *styling* secara efisien, di mana penggunaan Tailwind sebagai kerangka kerja CSS dapat mempersingkat waktu pemrosesan gaya CSS dan memberikan pengguna kebebasan untuk mendesain sesuai keinginan dengan semua fitur yang memudahkannya, seperti membuat *website* yang responsif [6]. Integrasi Tailwind CSS untuk desain responsif juga terbukti meningkatkan responsivitas sistem dan memastikan *platform* yang efisien serta menarik secara *visual* [7].

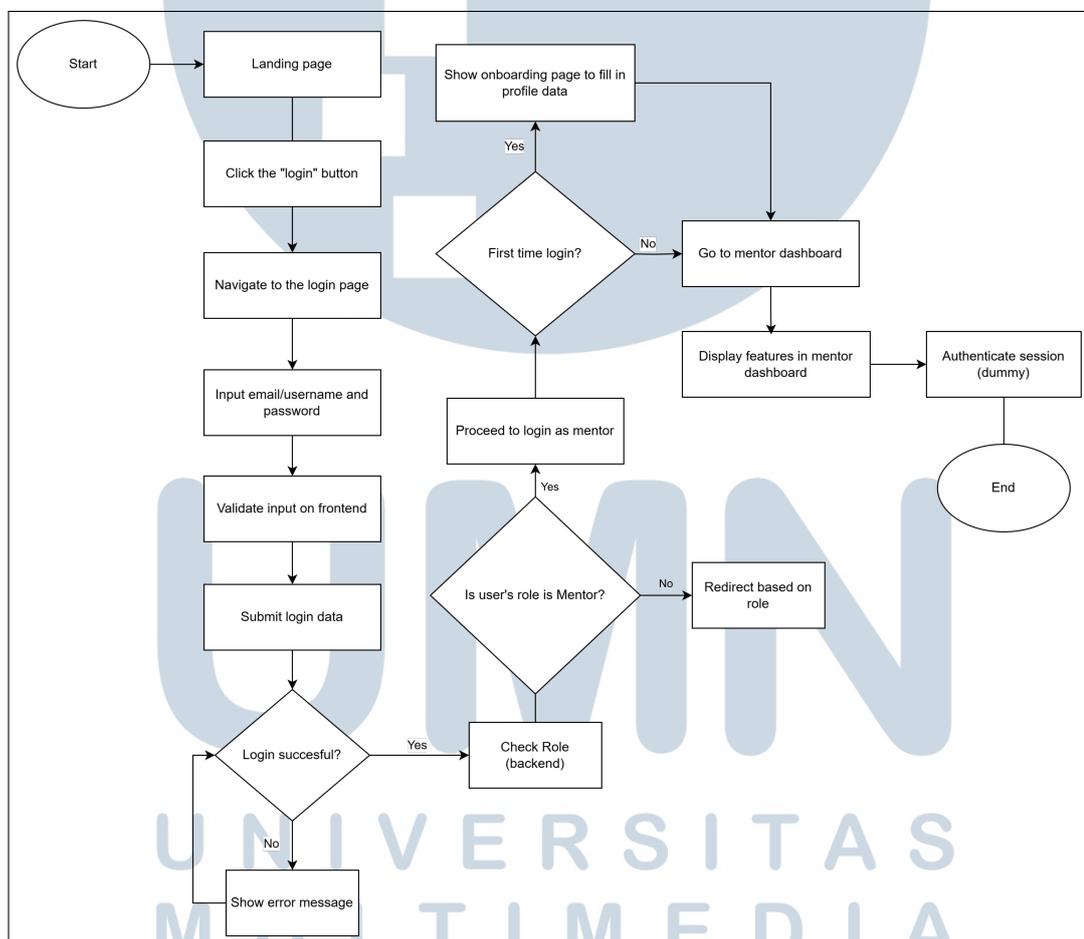
Proses *build* dan pengembangan untuk proyek ini dilakukan menggunakan Vite, sebuah *build tool modern* yang dikenal karena kecepatan dan keringkasannya. Sebagai sebuah *build tool* untuk pengembangan *front-end*, Vite dirancang untuk mengelola berbagai aktivitas teknis yang bertujuan meningkatkan alur kerja dan performa. Aktivitas ini mencakup konversi kode, penggabungan beberapa *file* menjadi satu, kompresi kode untuk mengurangi ukuran, serta otomatisasi prosedur pengujian [8].

Untuk penyimpanan data sementara di sisi klien, digunakan fitur *LocalStorage* guna meningkatkan kinerja dan responsivitas aplikasi. Pemanfaatan *LocalStorage* ini secara signifikan mengurangi ketergantungan pada permintaan server yang berulang, yang terbukti efektif dalam mempercepat proses aplikasi berbasis *web* [9]. Dengan demikian, data seperti preferensi atau status sesi dapat diakses kembali secara instan, yang secara langsung menciptakan pengalaman pengguna yang lebih lancar dan mulus [10].

### 3.4.2 Flowchart

Berikut adalah *flowchart* yang menggambarkan alur navigasi dan interaksi *user* pada *website* Flowcamp ID. *Flowchart* ini dirancang untuk memberikan *visual* tentang struktur dan fungsi utama *website*, sehingga dapat memudahkan pemahaman bagaimana pengguna dapat mengakses dan berinteraksi dengan berbagai fitur yang tersedia. Representasi *visual* ini membuat alur sistem menjadi lebih mudah dipahami, terutama karena proses yang berjalan di sistem *digital* seringkali tidak tampak secara eksplisit [11].

#### 1. Flowchart Website Login ke dashboard mentor



Gambar 3.2. Flowchart halaman Login ke dashboard Mentor

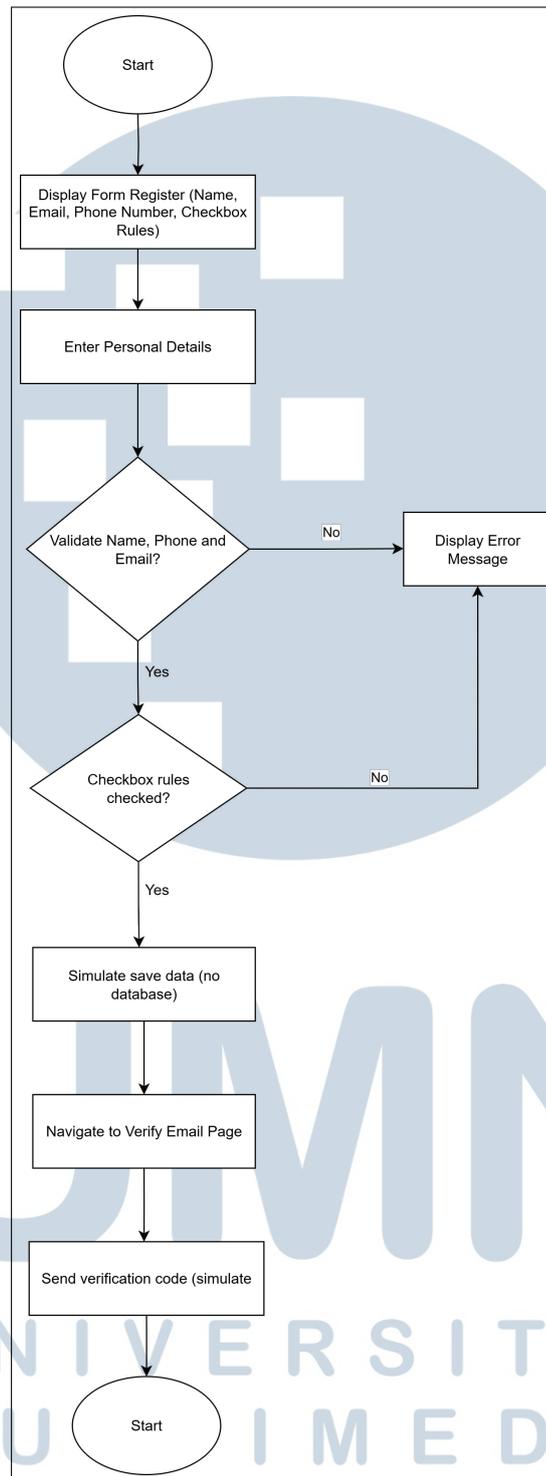
Pada halaman *login*, pengguna mengisi email atau *username* dan *password*. Setelah data dikirim, sistem akan melakukan validasi input di sisi *frontend*. Karena sistem belum terhubung dengan *backend*, proses autentikasi dan pengecekan peran (*role*) masih bersifat simulasi menggunakan data *dummy*.

Jika data dianggap valid, sistem akan mengecek peran pengguna. Jika *role* pengguna adalah sebagai mentor, pengguna akan diarahkan ke halaman *dashboard* mentor. Jika bukan mentor, sistem akan meng-*direct* pengguna ke halaman lain sesuai *role*.

Jika mentor *login* untuk pertama kali, sistem akan menampilkan halaman *onboarding* yang bertujuan untuk melengkapi data diri mentor. Setelah itu, mentor diarahkan ke halaman *dashboard* utama. Proses validasi akses masih dilakukan secara statis dengan memanfaatkan *localStorage* serta logika *role-based* di sisi *frontend*.

## 2. Flowchart Register



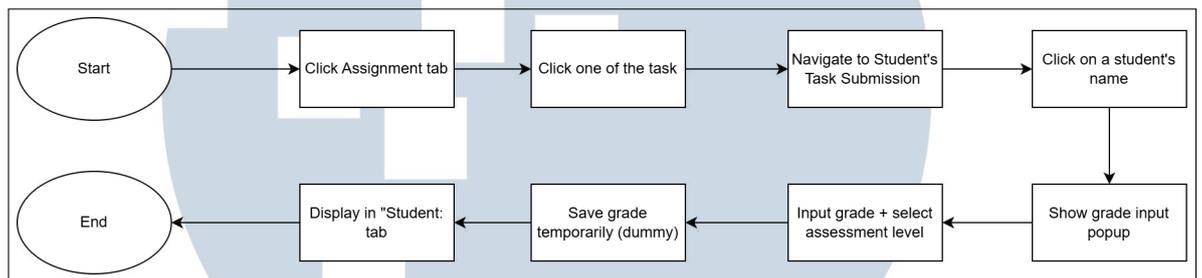


Gambar 3.3. *Flowchart* halaman *Register*

Gambar tersebut adalah *flow* halaman *register*. Pengguna mengakses halaman *register* ketika menekan *button Get Started* pada *Bootcamp Landing Page* LMS Flowcamp. Pada halaman *register*, pengguna dapat memasukkan email,

nomor telepon dan *password* dengan format yang benar dan mengisi *rules checkbox*. Setelah berhasil melakukan registrasi, pengguna akan dialihkan ke halaman verifikasi email untuk memasukkan kode verifikasi. Jika proses registrasi gagal, sistem akan menampilkan *error message* yang memberikan petunjuk untuk memperbaiki format pengisian registrasi atau memastikan *checkbox* telah dicentang.

### 3. Flowchart Submit Grade

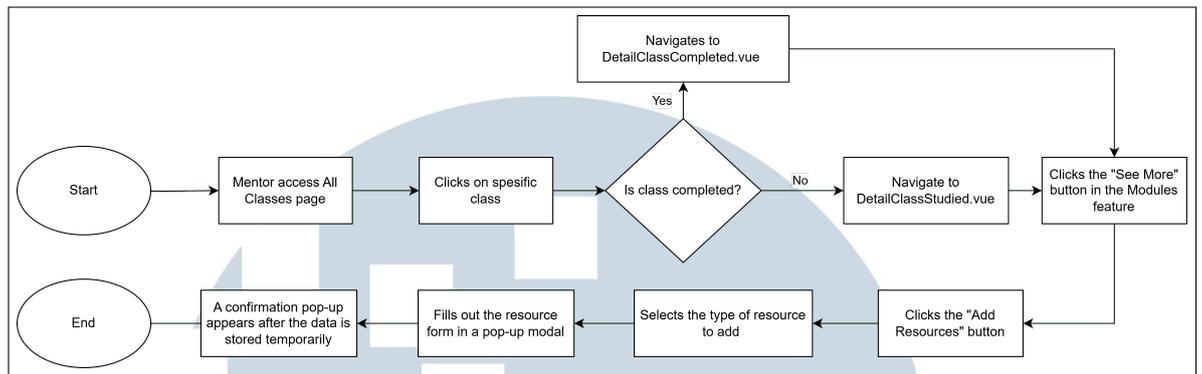


Gambar 3.4. Flowchart Submit Grade

Gambar 3.4 merupakan *flow* dari fitur pemberian nilai oleh mentor melalui halaman *Academy* pada *dashboard* Mentor. Proses diawali saat mentor berada di halaman *Academy* lalu memilih *tab Assignment* yang berisi daftar tugas dari berbagai kelas yang tersedia. Ketika mentor menekan salah satu daftar tugas, sistem akan mengarahkan ke halaman detail tugas berupa daftar pengumpulan tugas dari setiap siswa. Ketika mentor menekan salah satu nama siswa dari tabel tersebut, akan muncul *pop-up grade* yang menampilkan status pengumpulan tugas, input nilai manual, serta memilih kriteria pengerjaan siswa. Setelah itu, mentor dapat menekan *submit button* untuk menyimpan nilai tersebut. Nilai yang dimasukkan akan muncul pada *tab Student*, tepatnya pada kolom *Total Grades*.

### 4. Flowchart Add New Resources

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

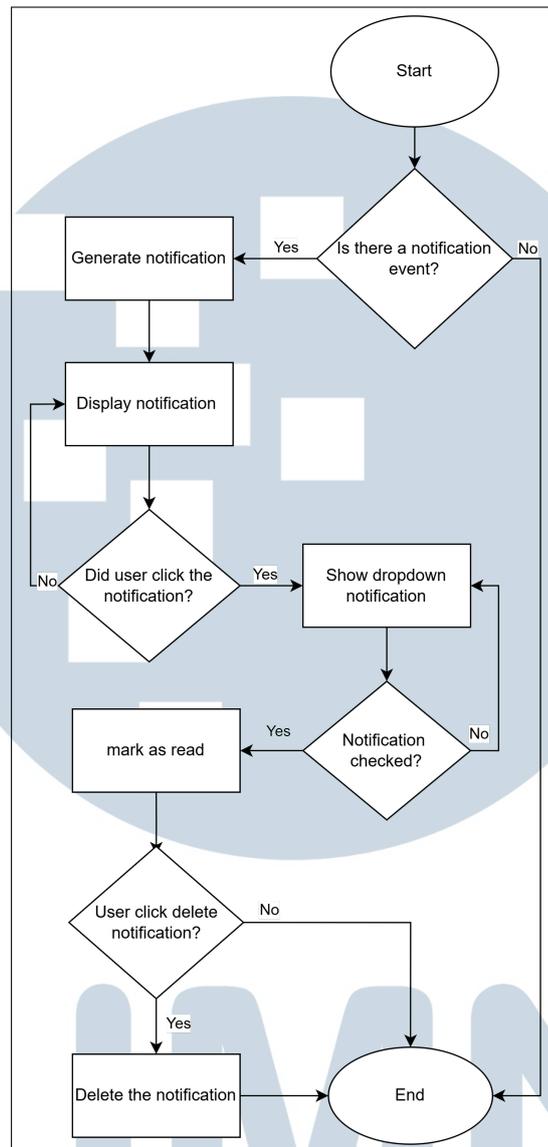


Gambar 3.5. *Flowchart Add New Resources*

Gambar 3.5 menggambarkan *flow* dari fitur penambahan materi baru oleh mentor melalui halaman *Detail Class*. Proses dimulai ketika mentor membuka halaman *All Classes*, lalu memilih salah satu kelas dan mentor akan diarahkan ke halaman detail kelas (*DetailClassCompleted.vue* atau *DetailClassStudied.vue*). Di halaman *Detail Class* terdapat fitur daftar modul. Untuk menambahkan materi baru, mentor menekan *button "See More"* di bagian modul, kemudian memilih opsi *"Add Resources"* dan memilih jenis materi yang akan ditambahkan, seperti PPT, video, atau bahkan *task* baru untuk siswa. Setelah memilih tipe materi, akan muncul *pop-up form input* sesuai jenis *resource* yang dipilih. Setelah materi berhasil disimpan, informasi materi yang baru ditambahkan akan muncul pada daftar modul yang ada di halaman kelas.

##### 5. *Flowchart* Notifikasi

  
 UNIVERSITAS  
 MULTIMEDIA  
 NUSANTARA



Gambar 3.6. Flowchart Notifikasi

Gambar 3.6 merupakan *flow* dari fitur notifikasi yang ditujukan untuk akun mentor pada *platform* LMS FlowCamp ID. Sistem ini dirancang untuk memberikan informasi terkait aktivitas siswa, khususnya mengenai status pengumpulan tugas. Proses diawali saat sistem mendeteksi *event* bahwa seorang siswa telah mengumpulkan tugas. Jika *event* tersebut terdeteksi, maka sistem akan membuat notifikasi baru yang disimpan ke dalam *localStorage*. Ikon lonceng notifikasi yang terdapat pada komponen *NavbarMentor.vue* kemudian akan menampilkan jumlah notifikasi yang belum dibaca. Ketika mentor menekan ikon lonceng tersebut, sistem akan menampilkan *dropdown* berisi daftar notifikasi yang diurutkan berdasarkan

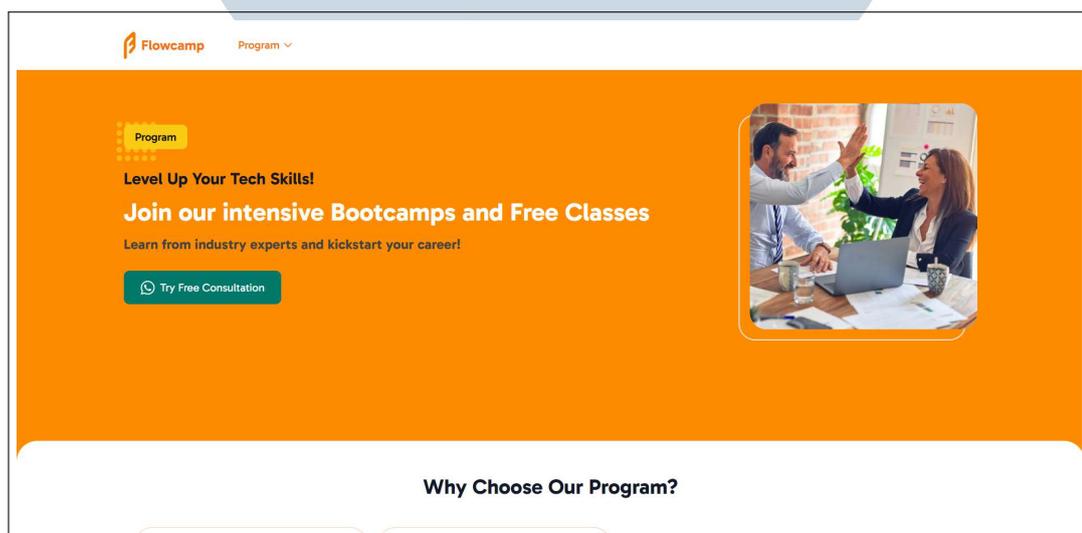
status baca dan waktu.

Jika mentor menekan salah satu notifikasi, maka sistem akan menjalankan fungsi `navigateToNotification()` untuk menandai notifikasi sebagai telah dibaca, memperbarui `localStorage`, menutup `dropdown`, serta menavigasi mentor ke halaman detail tugas terkait. Tersedia juga opsi untuk menghapus notifikasi, baik secara individual maupun seluruhnya. Apabila tidak terdapat notifikasi, sistem akan menampilkan `state` kosong berupa pesan “*No notifications*”.

### 3.4.3 Implementasi

Berikut merupakan hasil web Flowcamp yang telah dikembangkan:

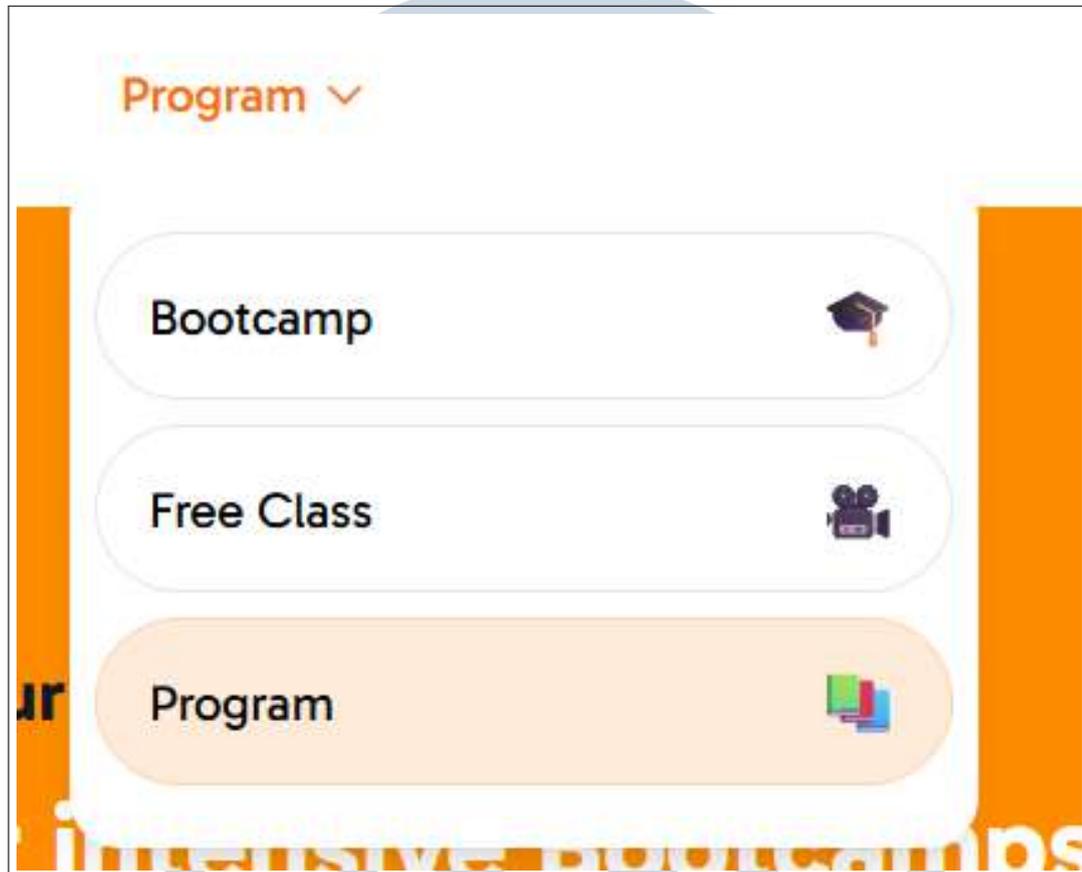
#### 1. *Landing Page Home*



Gambar 3.7. Banner section and navbar

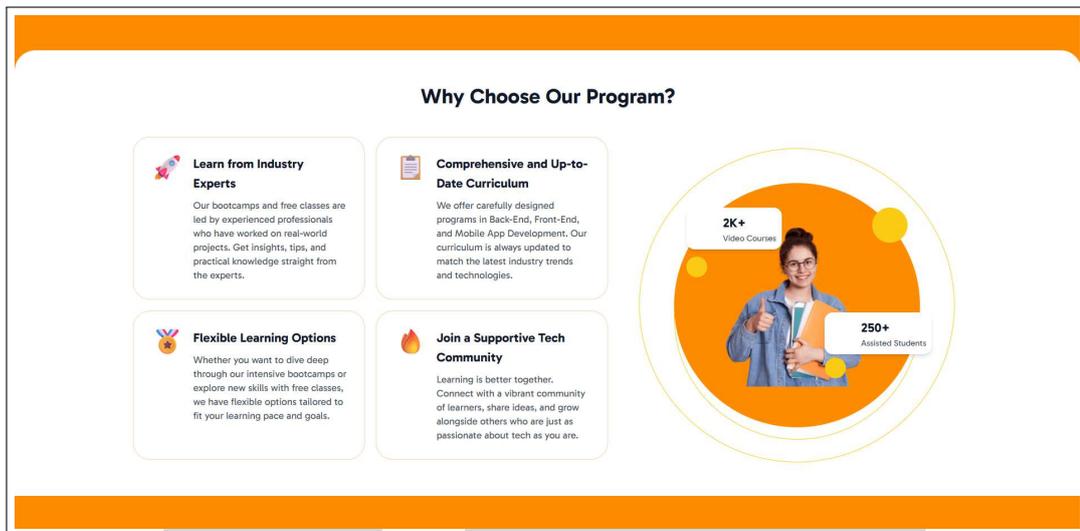
Gambar 3.7 menampilkan *section* pertama dari *landing page home* web Flowcamp yang dilengkapi dengan navbar yang berisi logo “Flowcamp” di sisi kiri dan menu navigasi *dropdown*. Di bawah navbar, terdapat *banner section* dengan *headline* “*Level Up Your Tech Skills! Join our intensive Bootcamps and Free Classes*” dan *sub-headline* “*Learn from industry experts and kickstart your career!*” serta adanya sebuah button CTA (*Call to Action*) berwarna hijau dengan teks “*Try Free Consultation*” yang berfungsi

untuk menghubungkan pengguna langsung dengan nomor kontak admin dari *website* saat diklik.



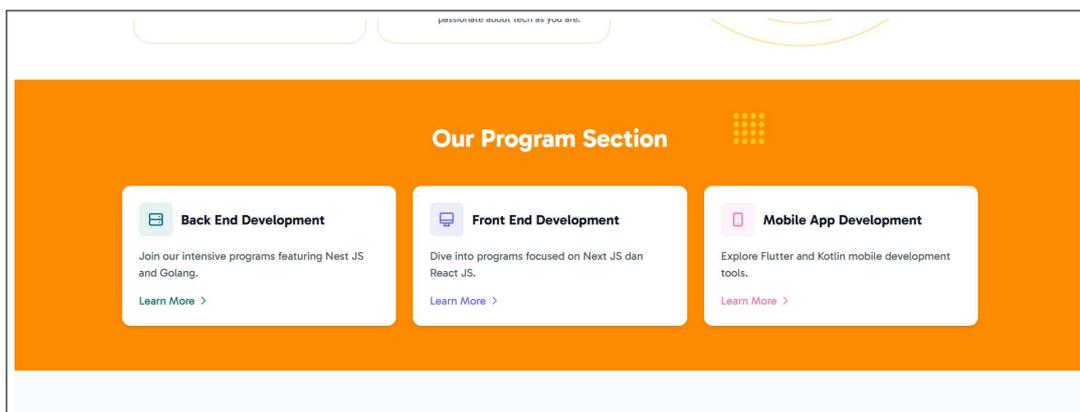
Gambar 3.8. *Dropdown Navbar*

Gambar 3.8 ini menunjukkan implementasi dropdown menu dari opsi "Program" yang terletak di *navbar* pada *landing page*. Ketika opsi "Program" di *navbar* ditekan, sebuah *dropdown* akan muncul menampilkan tiga pilihan utama yaitu *Bootcamp*, *Free Class*, dan *Program*. Setiap *item* dalam *dropdown* ini berfungsi sebagai navigasi, ketika salah satu pilihan seperti ditekan, pengguna akan diarahkan ke halaman yang sesuai dengan program tersebut. Sebagai indikator *visual* dari halaman yang sedang aktif, elemen *navbar* yang relevan akan berubah warna menjadi *orange*, menandakan bahwa pengguna sedang berada di halaman tersebut. Ini membantu pengguna memahami posisi mereka dalam struktur navigasi *website*.



Gambar 3.9. *Why Choose Our Program* section

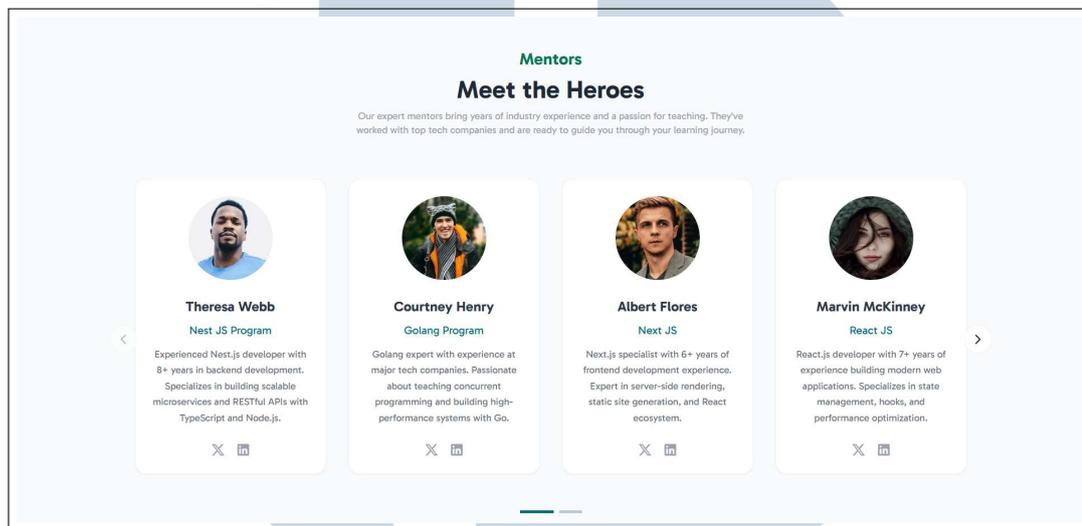
Gambar 3.9 ini menunjukkan implementasi *Why Choose Our Program Section*. *Section* ini bertujuan untuk memberikan informasi singkat namun meyakinkan mengenai keunggulan program pembelajaran yang ditawarkan oleh *platform*. Tampilan terdiri dari empat buah *card* informatif yang didesain dengan *icon* pendukung dan *text* yang mudah dipahami.



Gambar 3.10. *Program section*

Gambar 3.10 ini menunjukkan implementasi *section Program Section* yang berfungsi sebagai navigasi *visual* untuk memperkenalkan tiga materi utama yang ditawarkan oleh Flowcamp, yaitu *Back End Development*, *Front End Development*, dan *Mobile App Development*. Masing-masing jalur ditampilkan dalam bentuk *card* yang dilengkapi dengan ikon, judul program,

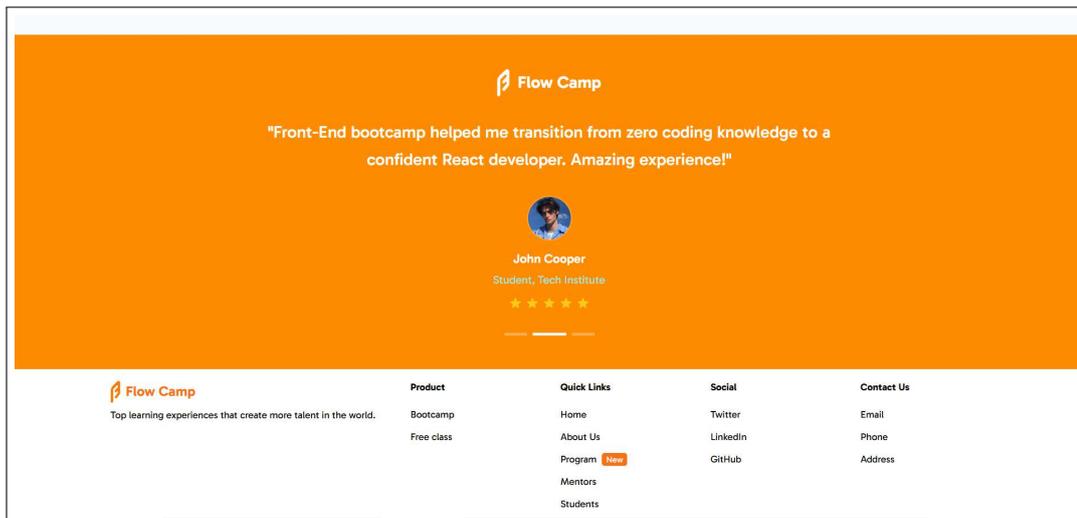
deskripsi singkat, serta button *Learn More*. Setiap elemen *card* ini memiliki fungsi interaktif yaitu ketika pengguna menekan salah satu *card* atau *button Learn More*, mereka akan diarahkan ke halaman *landing page Bootcamp* yang sesuai dengan kategori program yang dipilih.



Gambar 3.11. Mentor section

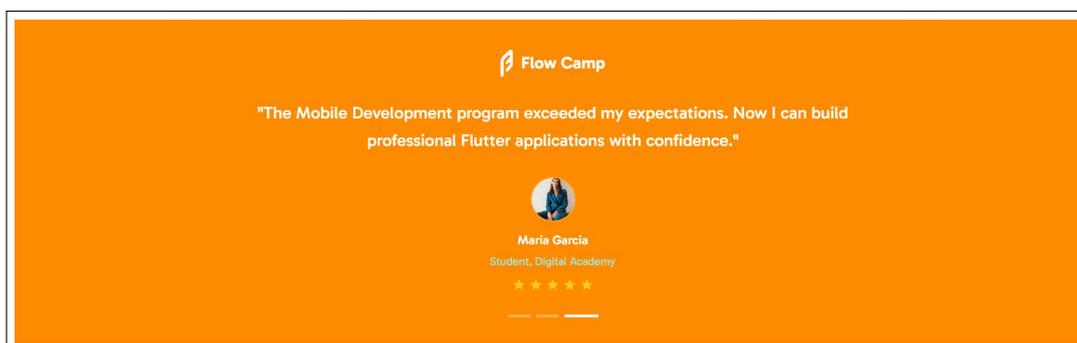
Gambar 3.11 ini menunjukkan tampilan *section Mentors* yang menampilkan daftar mentor yang terlibat dalam program pembelajaran. Setiap mentor ditampilkan dalam bentuk *card* yang berisi foto profil, nama, program yang diajar, serta deskripsi singkat mengenai latar belakang dan keahliannya.

Section ini menggunakan sistem *carousel* atau *slider*, yang memungkinkan pengguna menggeser daftar mentor secara horizontal. Navigasi dilakukan dengan tombol panah kiri dan kanan, serta indikator *pagination bar* di bagian bawah yang menunjukkan posisi pengguna dalam daftar mentor. Fitur ini memudahkan pengguna untuk melihat lebih banyak profil mentor tanpa harus membuka halaman baru sekaligus menjaga tampilan tetap rapi, efisien, dan interaktif.



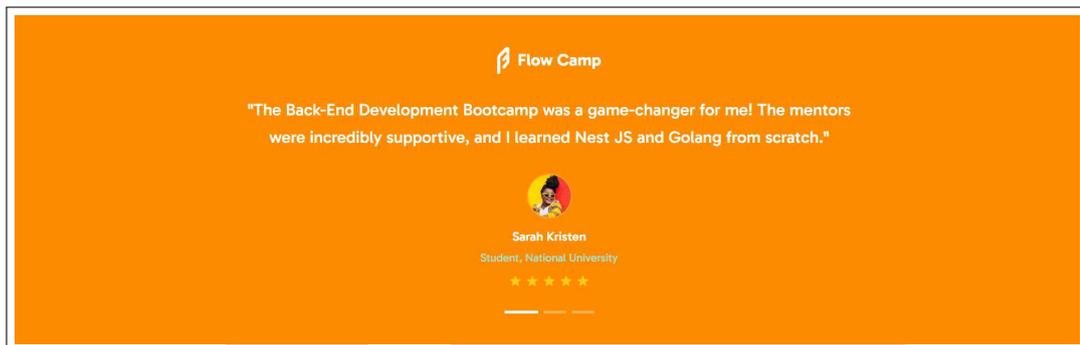
Gambar 3.12. Testimonial Section and Footer

Gambar 3.12 ini menunjukkan implementasi section *Testimoni* dan *Footer* di *Home Page*. Di bagian bawah halaman, terdapat *section Footer* yang berisi informasi tambahan seperti nama produk, *link* ke halaman penting (*Home, Academy, Free Class*), *help*, dan kontak admin. *Footer* ini bersifat statis dan berfungsi sebagai area penutup halaman yang menyediakan navigasi cepat ke bagian lain dari *platform*.



Gambar 3.13. Testimonial section slide 2

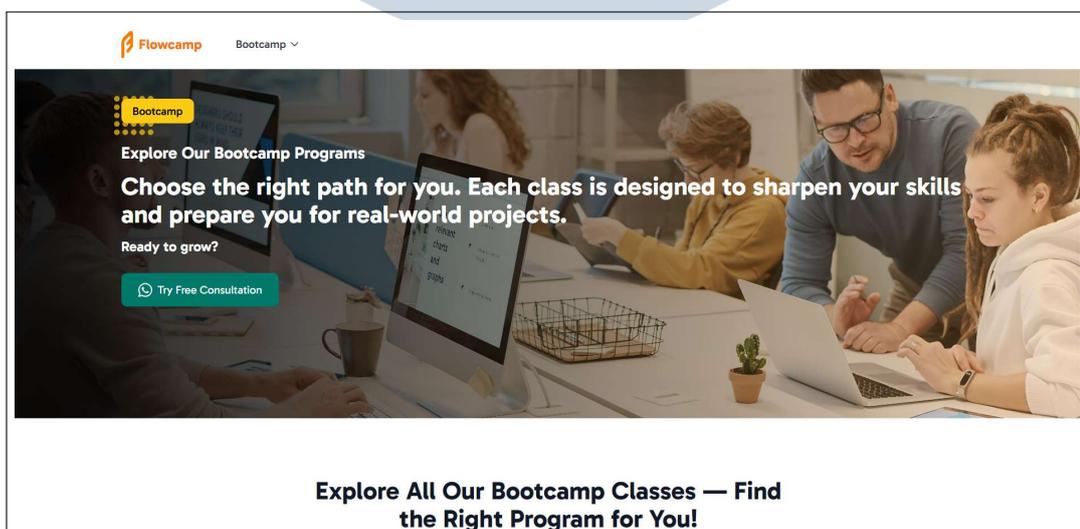
*Section* testimoni yang ditampilkan pada gambar 3.13 menunjukkan salah satu testimoni pengguna dalam bentuk *card* yang berisi ringkasan tentang pengalaman belajar, nama, program yang diikuti, dan *rating* bintang. Testimoni ditampilkan secara bergantian menggunakan komponen *carousel* atau *slider* otomatis.



Gambar 3.14. *Testimonial section slide 3*

Navigasi *carousel* ini dapat dilakukan secara manual menggunakan tombol panah kiri dan kanan, serta dilengkapi dengan *pagination indicator* berbentuk titik di bagian bawah yang ditunjukkan pada gambar 3.14. Selain navigasi manual, carousel ini juga menggunakan fitur *auto-play* yang secara otomatis mengganti isi card setiap 3 detik tanpa interaksi dari pengguna.

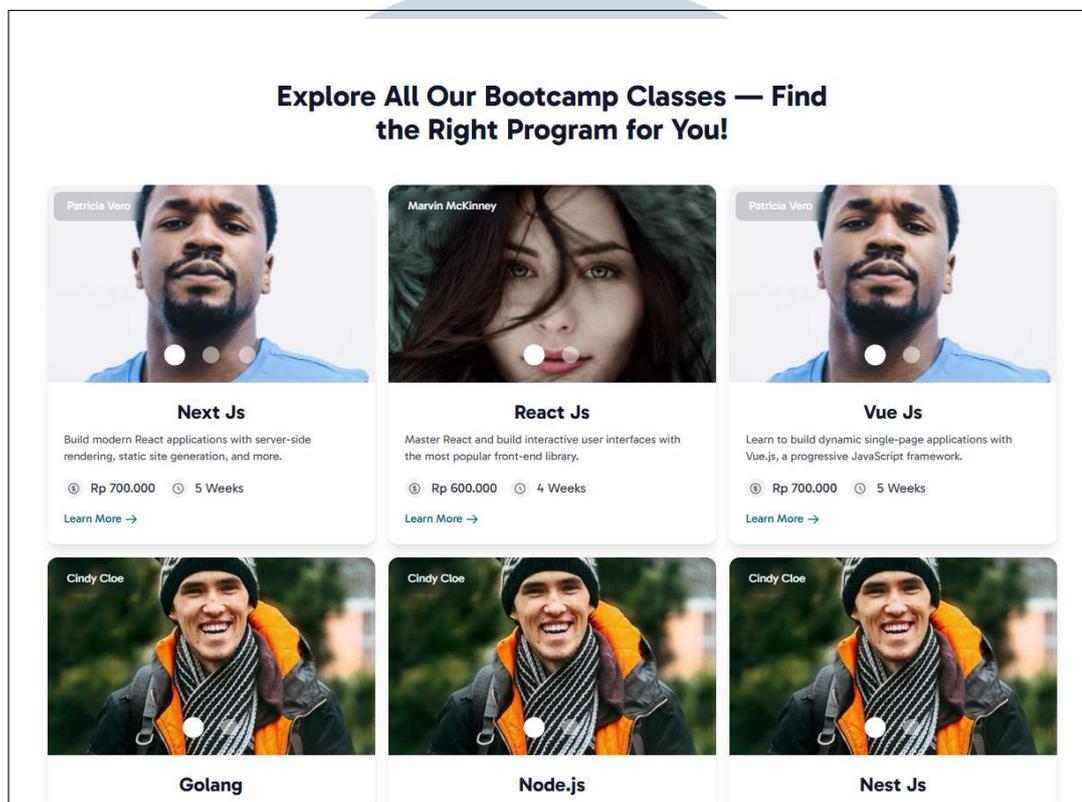
## 2. *Landing Page Bootcamp - See More*



Gambar 3.15. *Banner section*

Gambar 3.15 ini menunjukkan tampilan *Banner section* pada halaman lanjutan *page Bootcamp* (halaman hasil dari tombol "See More" pada *Landing Page Bootcamp*) yang menampilkan beberapa elemen utama berupa kategori ("Bootcamp"), judul pengantar, dan deskripsi singkat. Selain itu, terdapat tombol *Call to Action* bertuliskan "Try Free Consultation" yang

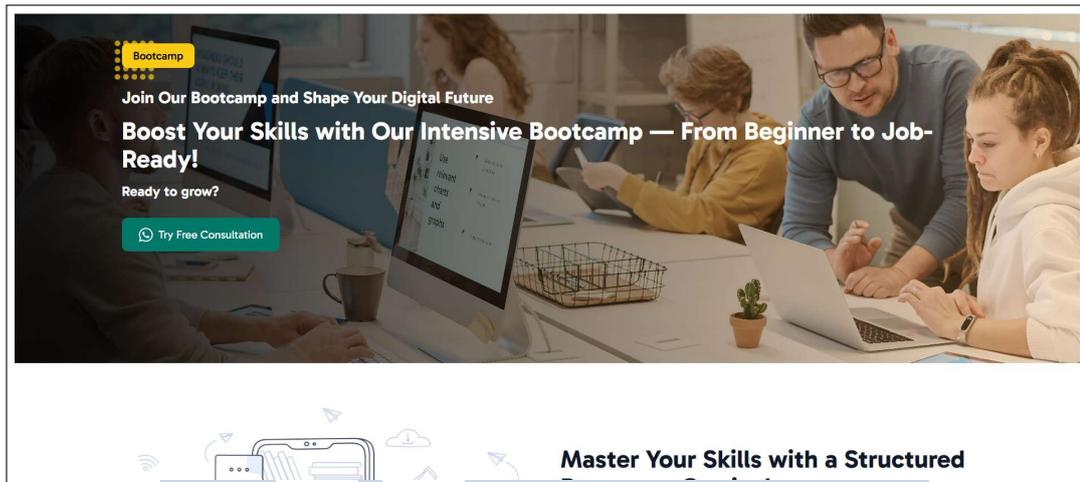
mengarahkan pengguna untuk mengambil langkah awal dalam mengikuti program.



Gambar 3.16. Content Section

Gambar 3.16 ini menunjukkan *section* konten yang berisi seluruh *program bootcamp* yang tersedia dalam bentuk kartu (*card layout*). Setiap kelas ditampilkan dalam sebuah *card* yang terdiri atas foto mentor, nama program (seperti Next.js, React.js, Vue.js, Golang, Node.js), deskripsi singkat materi yang dipelajari, harga, durasi kelas, dan tombol *Learn More*. Desain ini memudahkan pengguna membandingkan berbagai program yang ditawarkan, baik dari sisi teknologi, mentor, maupun detail teknis lainnya dan *section* ini dapat mengarahkan pengguna untuk memilih program sesuai minat dan kebutuhan mereka.

### 3. Learn More Page



Gambar 3.17. *Banner Section*

Gambar 3.17 ini menunjukkan tampilan *Banner Section* pada halaman *Bootcamp Learn More*, yaitu halaman detail program yang muncul setelah pengguna menekan tombol *Learn More* pada salah satu *card* kelas di halaman *Bootcamp See More*. Section ini menampilkan elemen teks yang memperkenalkan kelas *bootcamp* yang dipilih. Judul utama pada *section* ini berisi ajakan seperti “*Boost Your Skills with Our Intensive Bootcamp — From Beginner to Job-Ready!*” dan disertai *button Call to Action* bertuliskan “*Try Free Consultation*”. Teks pendukung lainnya menyampaikan bahwa program ini ditujukan untuk membantu peserta berkembang dari level dasar hingga siap kerja.

UMN  
UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

**Master Your Skills with a Structured Bootcamp Curriculum**

**Comprehensive Curriculum With Hands-On Experience**

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

- ✓ Lorem ipsum dolor sit amet, consectetur adipiscing elit.
- ✓ Lorem ipsum dolor sit amet, consectetur adipiscing elit.
- ✓ Lorem ipsum dolor sit amet, consectetur adipiscing elit.
- ✓ Lorem ipsum dolor sit amet, consectetur adipiscing elit.

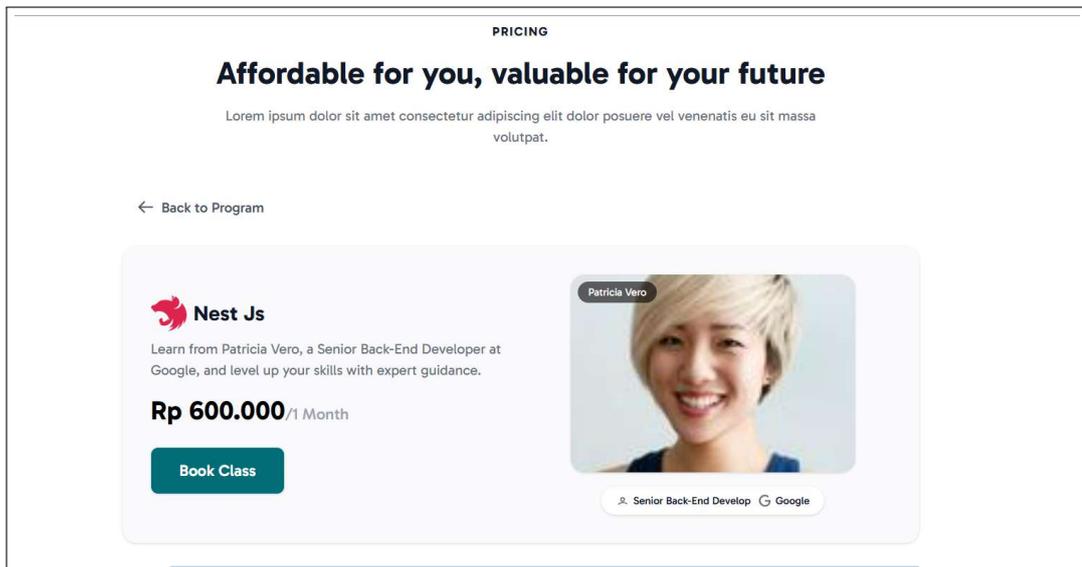
**What You'll Learn?**

01 WEEK	02 WEEK	03 WEEK	04 WEEK
<b>Introduction Nest JS</b>	<b>Building Rest APIs</b>	<b>Database Integration</b>	<b>Authentication</b>
<ol style="list-style-type: none"> <li>1. Understanding Node.js and TypeScript basics</li> <li>2. Setting up a new Nest JS project</li> <li>3. Exploring the project structure (modules, controllers, services)</li> </ol>	<ol style="list-style-type: none"> <li>1. Defining routes and handling HTTP requests</li> <li>2. Implementing CRUD operations</li> <li>3. Testing API endpoints with Postman</li> </ol>	<ol style="list-style-type: none"> <li>1. Introduction to databases SQL</li> <li>2. Connecting Nest JS to a PostgreSQL database using TypeORM</li> <li>3. Querying data</li> </ol>	<ol style="list-style-type: none"> <li>1. Implementing JWT (JSON Web Token) authentication</li> <li>2. Creating login and registration endpoints</li> <li>3. Protecting routes with guards</li> </ol>

Gambar 3.18. *Curriculum Overview Section*

Gambar 3.18 ini menunjukkan tampilan *Curriculum Overview Section* pada halaman *Bootcamp Learn More* yang menjelaskan tentang kurikulum yang akan dipelajari selama mengikuti program *bootcamp*. *Section* dilanjutkan dengan bagian “*What You’ll Learn?*” yang membagi kurikulum ke dalam empat minggu pembelajaran. Setiap minggu disajikan dalam bentuk *learning milestone card*, lengkap dengan label waktu (Week 01 sampai Week 04), judul materi utama, dan daftar submateri yang dipelajari pada minggu tersebut.

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA



Gambar 3.19. Pricing Section

Gambar 3.19 ini menunjukkan implementasi *Pricing Section* yang menampilkan informasi harga *bootcamp*, beserta detail mentor dan ajakan untuk bergabung ke dalam kelas. Komponen utama dari *section* ini adalah *course card* yang menampilkan nama kelas, harga kelas, deskripsi singkat tentang mentor, serta *button* “*Book Class*” untuk pendaftaran.

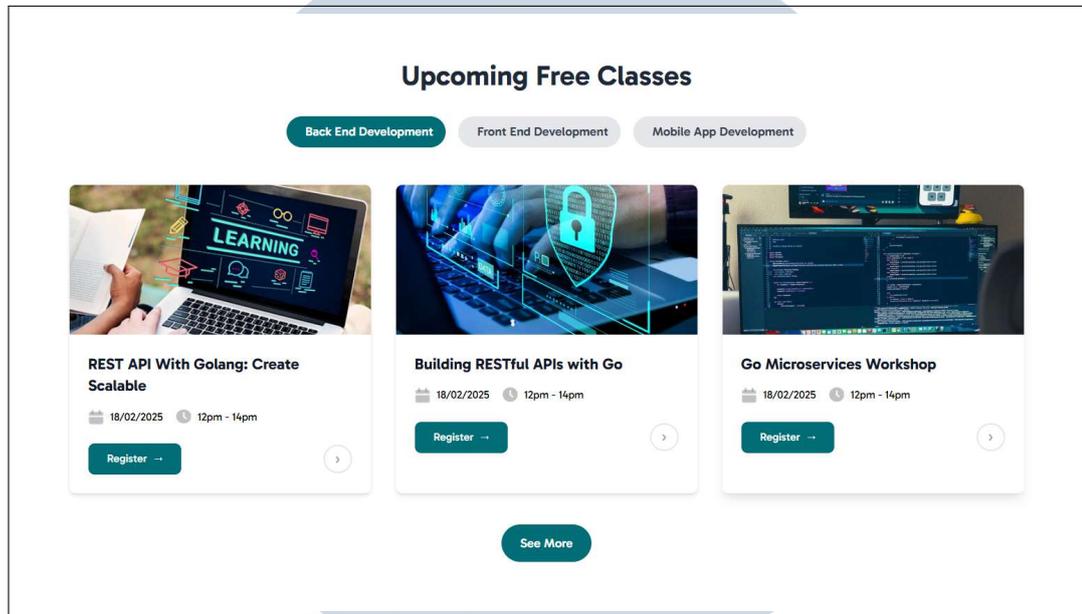
#### 4. Landing Page Free Class



Gambar 3.20. Banner section

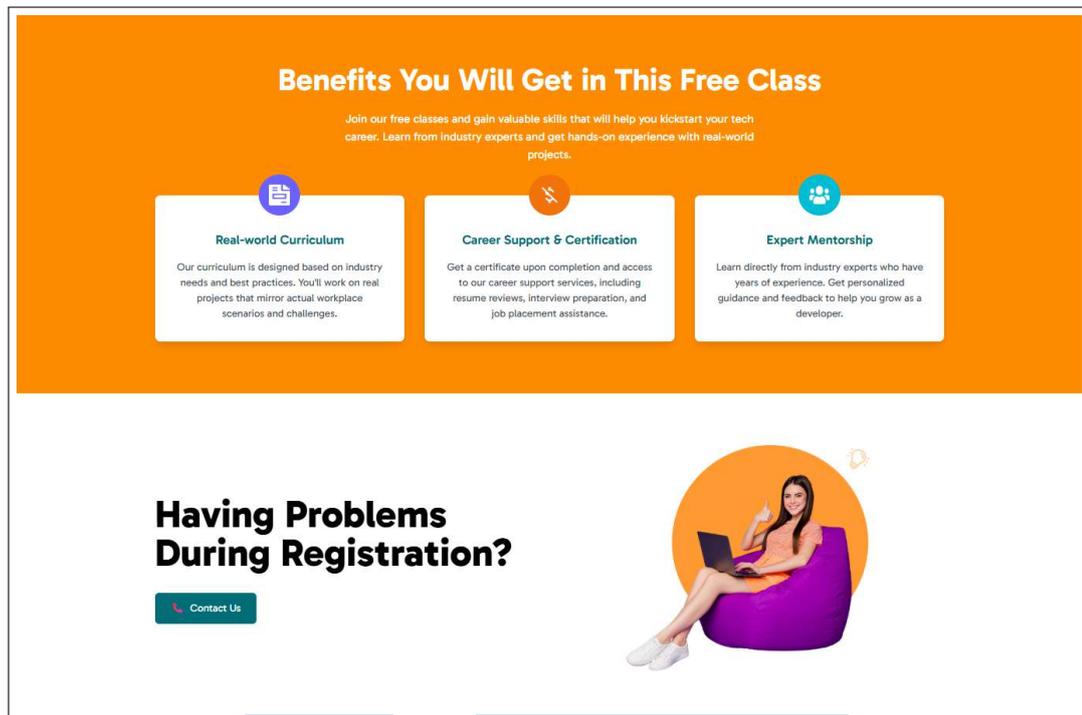
Gambar 3.20 menampilkan *Banner Section* dari halaman *Landing Page*

*Free Class* yang mempromosikan program *free class* yang juga terdapat *button Call to Action* bertuliskan “*Try Free Consultation*” yang mengarahkan pengguna untuk mengambil langkah awal dalam mengikuti program.



Gambar 3.21. *Classes Section*

Gambar 3.21 menampilkan *Upcoming Free Classes Section* yang berisi daftar kelas gratis yang akan diselenggarakan, lengkap dengan informasi singkat terkait jadwal, durasi, dan kategori kelas. Terdapat tiga *card kelas* yang ditampilkan secara horizontal dengan *button Register* pada setiap *cardnya* yang membantu pengguna untuk mendaftar ke kelas yang diminati. Tersedia juga *filter kategori* berupa tombol seperti “*Back End Development*”, “*Front End Development*”, dan “*Mobile App Development*” yang memungkinkan pengguna memfilter kelas berdasarkan bidang yang diinginkan. Di bagian bawah *section* ini, terdapat tombol *See More* yang mengarahkan pengguna ke halaman lanjutan berisi daftar lengkap program kelas gratis.

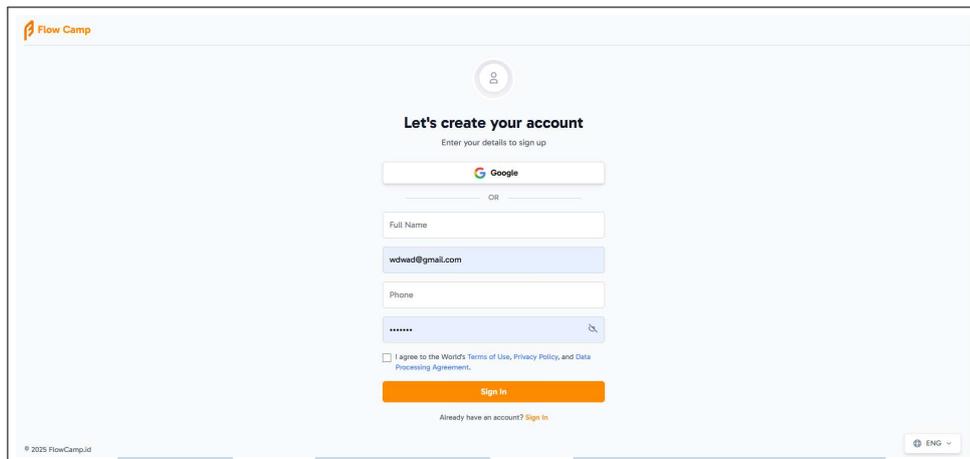


Gambar 3.22. *Benefits Section dan Help Section*

Gambar 3.22 menampilkan dua bagian utama pada halaman *Landing Page Free Class* yaitu *Benefits Section* dan *Help Section*. Pada bagian atas, terdapat *Benefits Section* yang menjelaskan tiga manfaat jika mengikuti program kelas gratis di FlowCamp. Masing-masing manfaat ditampilkan dalam bentuk *card* dengan ikon dan judul yang representatif. Di bawahnya terdapat *Help Section* yang menampilkan pesan bertanya “*Having Problems During Registration?*” dengan *button* “*Contact Us*” yang berfungsi sebagai jalur bantuan bagi pengguna jika mengalami kendala dalam proses pendaftaran.

##### 5. Halaman *Register*

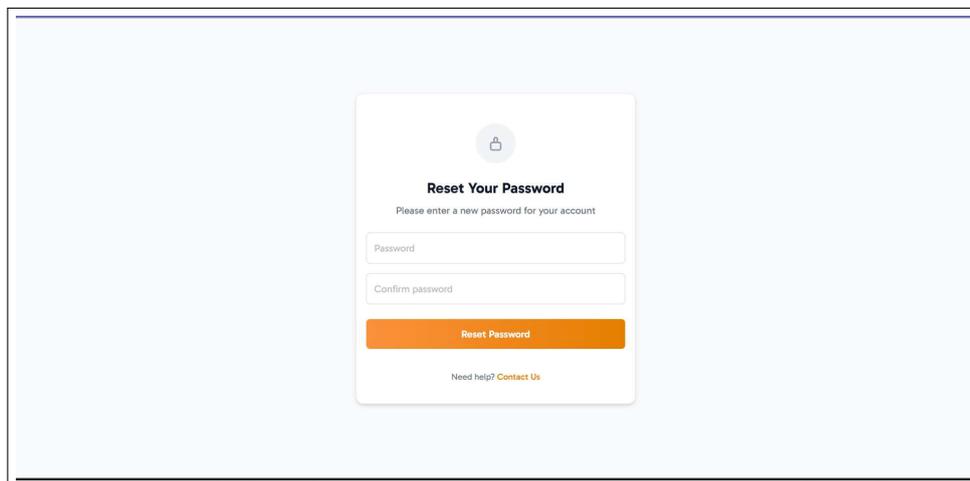
U N I V E R S I T A S  
M U L T I M E D I A  
N U S A N T A R A



Gambar 3.23. Halaman *Register*

Halaman *register* terlihat pada gambar 3.23 terdapat *form* yang berisikan email, nomor telepon dan *password*, pengguna harus memasukkan data tersebut untuk melakukan registrasi dan mencentang *agree to rules checkbox*. Ketika pengguna berhasil melakukan registrasi, sistem akan melakukan *redirect* ke halaman *dashboard* namun jika gagal akan menampilkan pesan *error*.

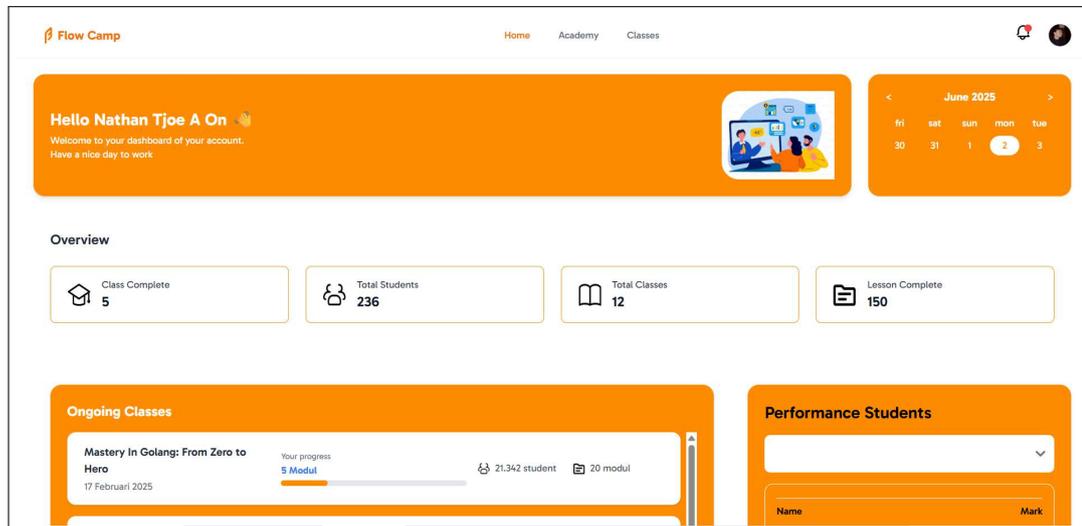
## 6. Halaman *Forgot Password*



Gambar 3.24. Halaman *Forgot Password*

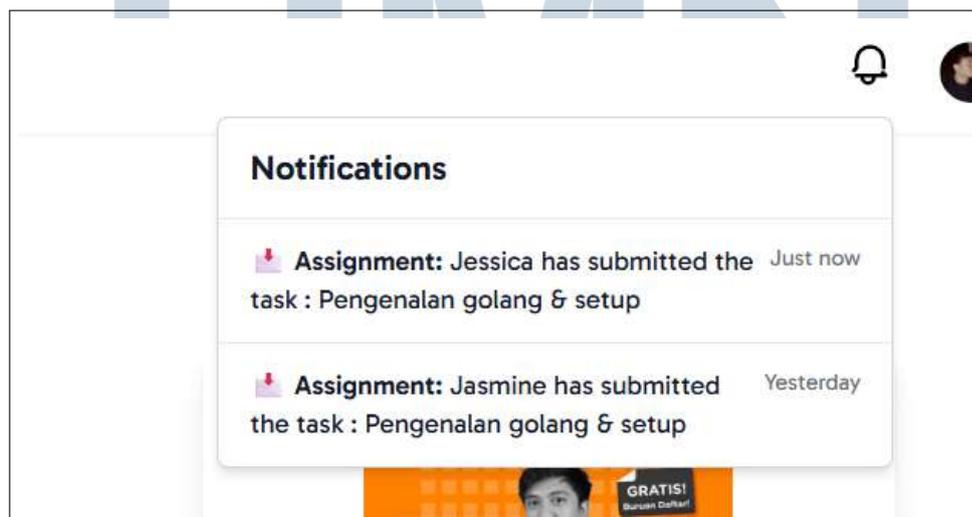
Gambar 3.24 menunjukkan implementasi fitur *Forgot Password* pada web Flowcamp. Saat pengguna menekan tombol '*Forgot Password*', halaman ini akan melakukan *redirect* ke halaman khusus untuk melakukan *reset password* akun mereka. Setelah proses *reset* berhasil, pengguna akan segera dialihkan ke halaman verifikasi.

## 7. Home pada *dashboard* Mentor



Gambar 3.25. *Home dashboard mentor + navbar*

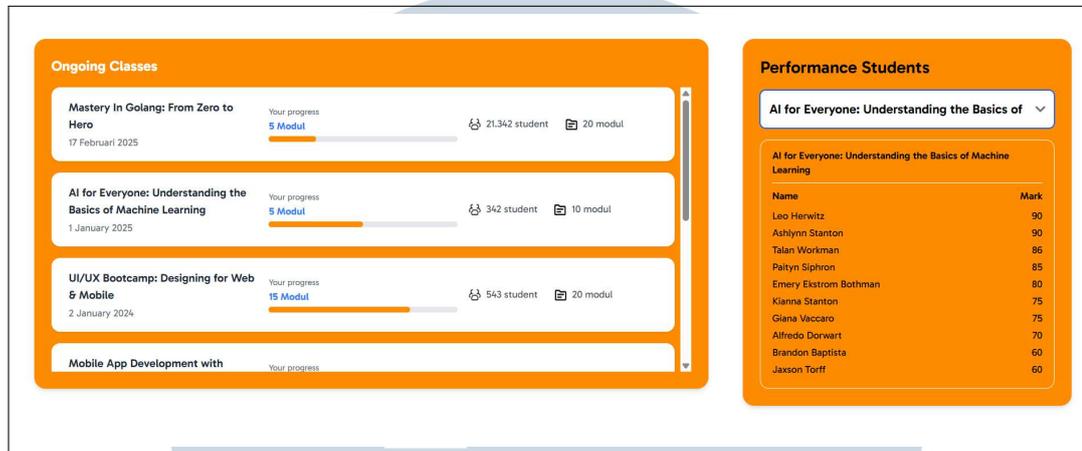
Gambar 3.25 adalah tampilan *home* dan *navbar* pada *dashboard* mentor. Pada bagian atas terdapat kalimat sapaan yang ditujukan kepada pengguna, adanya fitur kalender yang menampilkan tanggal secara *real-time*, fitur untuk menampilkan (*overview*) dan menampilkan detail mengenai aktivitas kelas yang sedang diampu oleh pengguna. Selain itu, terdapat fitur untuk menampilkan performa siswa beserta nilai mereka dalam suatu kelas.



Gambar 3.26. *Fitur Notifikasi Mentor*

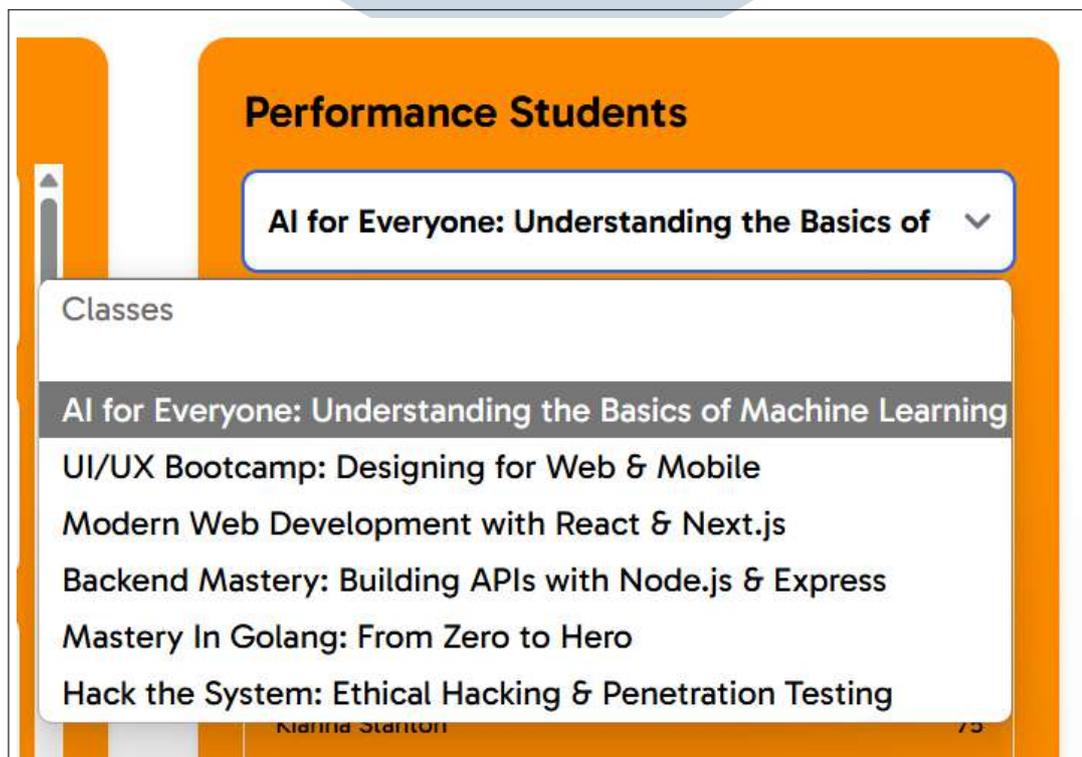
Gambar 3.26 menunjukkan implementasi fitur notifikasi *navbar dashboard*

mentor, yang akan muncul jika peserta melakukan aktivitas di kelas yang dia ambil.



Gambar 3.27. *Ongoing class and List Performance*

Gambar 3.27 menunjukkan tampilan fitur panel *ongoing classes and list performance*.

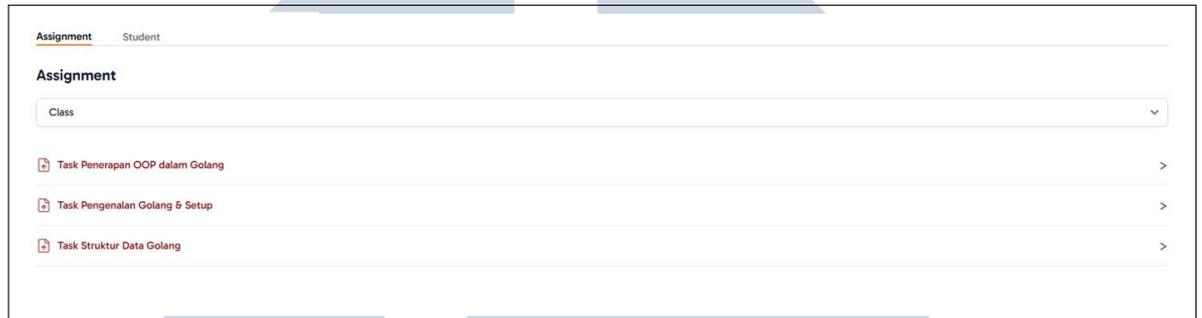


Gambar 3.28. *Dropdown Performance Student*

Gambar 3.28 menunjukkan komponen *dropdown* yang digunakan untuk

memilih kelas untuk melihat *performance* siswa yang mengambil kelas tersebut.

## 8. *Academy* dashboard mentor



Gambar 3.29. *Academy page dashboard mentor - Tab Assignment*

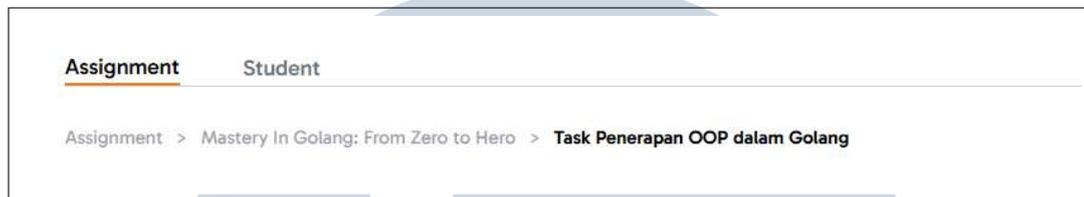
Gambar 3.29 menampilkan implementasi halaman *Academy* pada *dashboard* mentor. Pada halaman ini terdapat daftar tugas berdasarkan kelas yang dapat dipilih melalui *dropdown*. Ketika mentor menekan salah satu nama tugas pada list kelas di sebelah kiri, sistem akan mengarahkan ke halaman detail *assignment* kelas.

Student	Submission Date	Status	Grading Date	Grades
Alfonso Kenter	2 February 2025	ontime	3 February 2025	90
Emery Franci	5 February 2025	ontime	7 February 2025	-
Jasmine	1 February 2025	on time	3 February 2025	100
Justin Botosh	5 February 2025	overdue	7 February 2025	-
Paityn Press	1 February 2025	ontime	3 February 2025	80
Tatiana Passaquin...	3 February 2025	Overdue	3 February 2025	70

Gambar 3.30. *Task Students Submission*

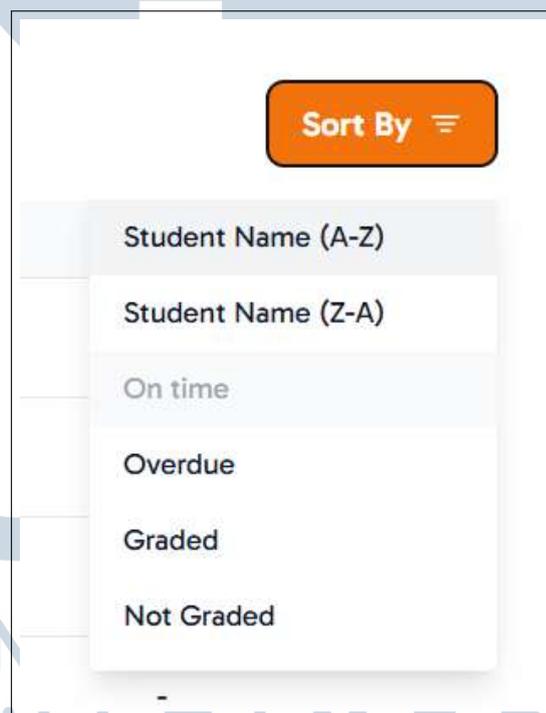
Halaman detail pada gambar 3.30 ini menampilkan data pengumpulan tugas oleh siswa dalam bentuk tabel, termasuk informasi nama siswa, tanggal pengumpulan, status pengumpulan (*on time*, atau *overdue*), tanggal penilaian,

dan nilai akhir. Data siswa juga dapat diurutkan melalui tombol *Sort By* yang tersedia di pojok kanan atas tabel.



Gambar 3.31. *Breadcrumbs*

Pada halaman ini juga terdapat fitur *breadcrumbs* yang ditunjukkan gambar 3.31 dengan indikator warna hitam ketika sedang aktif dan memudahkan pengguna untuk kembali ke halaman sebelumnya dengan cepat. Selain itu, tersedia juga navigasi berbentuk *tabs* yang terdiri dari *Assignment* dan *Student* untuk memisahkan tampilan berdasarkan fungsi yang berbeda.



Gambar 3.32. Filter Sort By

Gambar 3.32 menampilkan komponen *dropdown sorting* di halaman detail daftar pengumpulan tugas siswa di dashboard mentor. *Dropdown* ini muncul setelah pengguna menekan tombol *Sort By* dan berfungsi untuk mengurutkan data pada tabel tugas berdasarkan beberapa kriteria.

Alfonso Kenter On Time  
Submitted on 2 February 2025

**Review Criteria:**

- **Highly Accurate** → Exceeds expectations by meeting all standards exceptionally well in both steps and output.
- **Accurate** → Fully meets all standards in both steps and output.
- **Fairly Accurate** → Meets the standards, though there are some errors in steps or output.
- **Slightly Accurate** → Many errors in steps, input, and output, but some standards are met.
- **Inaccurate** → Steps, input, and output are completely incorrect.

No link submitted

**Submission Assessment**

Highly Accurate   Accurate   Fairly Accurate   Slightly Accurate   Inaccurate

**Grades**

90

Submit

Gambar 3.33. Student's grade

Gambar 3.33 menampilkan *pop-up* untuk memberikan nilai pada siswa ketika mentor menekan salah satu nama siswa pada daftar pengumpulan tugas. Bagian atas *pop-up* menampilkan informasi siswa seperti nama, status pengumpulan tugas, serta tanggal pengumpulan. Di bawahnya terdapat panel *Review Criteria* yang menjelaskan lima tingkatan penilaian mulai dari *Highly Accurate* hingga *Inaccurate*, lengkap dengan deskripsi standar penilaian masing-masing. Mentor dapat memilih salah satu kategori penilaian dari bagian *Submission Assessment*, serta mengisi nilai numerik secara manual pada kolom *Grades*. Setelah penilaian selesai, mentor dapat menekan *button Submit* untuk menyimpan hasil penilaian.

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

Assignment **Student**

Student

Class  Status Class

Student	Class	Modul	Total Grades	Status	Progress
Jasmine	Mastery In Golang: From Zero to Hero	10/10	100	Passed	100%
Paityn Press	UI/UX Bootcamp: Design Thinking	14/14	80	Passed	100%
Tatiana Passa	Mastery In Golang: From Zero to Hero	7/10	-	On Going	75%
Justin Botosh	Backend Mastery: Build REST API	12/16	-	Not passed	75%
Emery Franci	Backend Mastery: Build REST API	6/12	-	Not passed	50%

Show  Row < 1 >

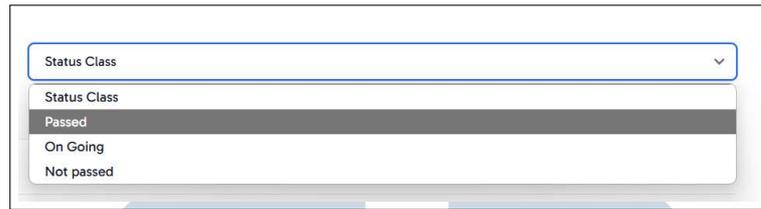
Gambar 3.34. Academy Tabs Student

Gambar 3.34 ini merupakan tampilan *tab Student* dalam halaman *Academy* yang fungsinya adalah untuk menampilkan daftar siswa yang mengikuti kelas, lengkap data kelas, dan nilai siswa. Setiap baris siswa dalam tabel ini bisa diklik, dan jika diklik akan menampilkan *pop-up* detail nilai. Di dalam *pop-up* tersebut, mentor dapat melihat semua nilai tugas siswa pada kelas tersebut, yang dikelompokkan berdasarkan status *Passed*, *On Going*, atau *Not Passed*.



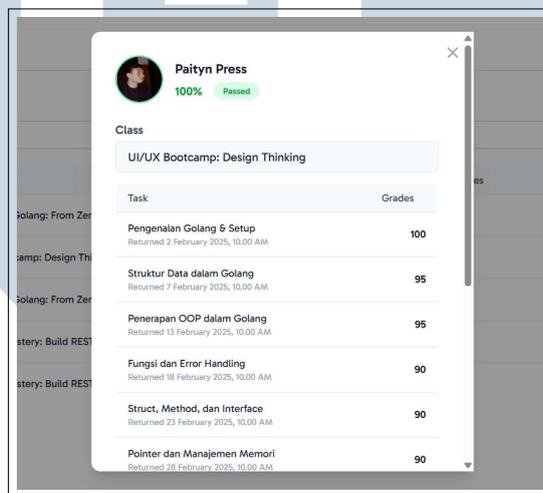
Gambar 3.35. Filter berdasarkan kelas yang diikuti

Gambar 3.35 menunjukkan komponen *dropdown* yang berfungsi untuk memfilter siswa berdasarkan kelas yang mereka ikuti.



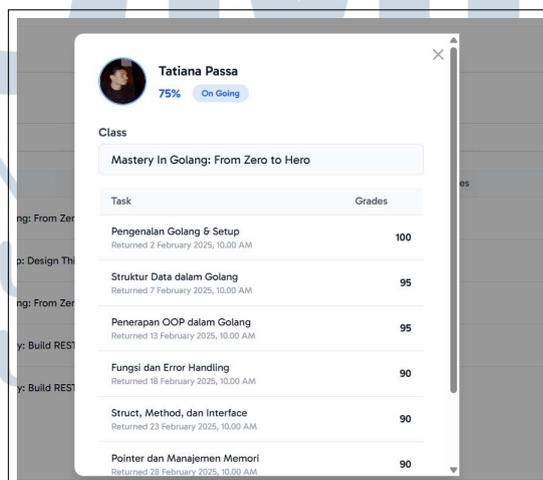
Gambar 3.36. Filter berdasarkan status kelulusan siswa

Gambar 3.36 menunjukkan komponen *dropdown* yang berfungsi untuk memfilter siswa berdasarkan status kelulusan siswa.



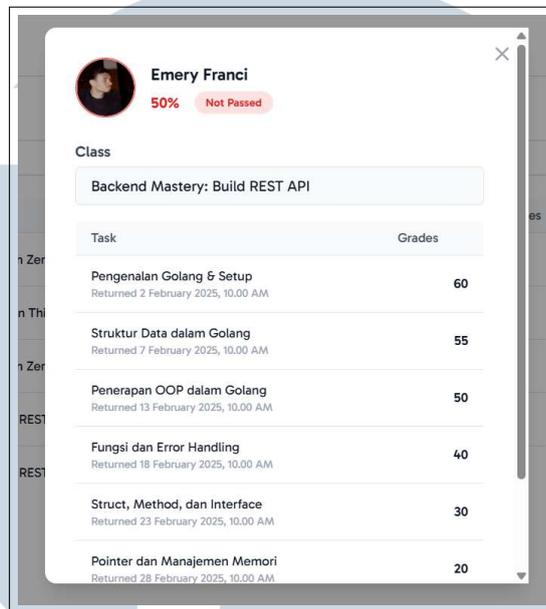
Gambar 3.37. *Popup* Detail Nilai *Passed*

Gambar 3.37 menunjukkan tampilan detail nilai siswa dengan status kelulusan *Passed*.



Gambar 3.38. *Popup* Detail Nilai *On Going*

Gambar 3.38 menunjukkan tampilan detail nilai siswa dengan status kelulusan *On Going*.

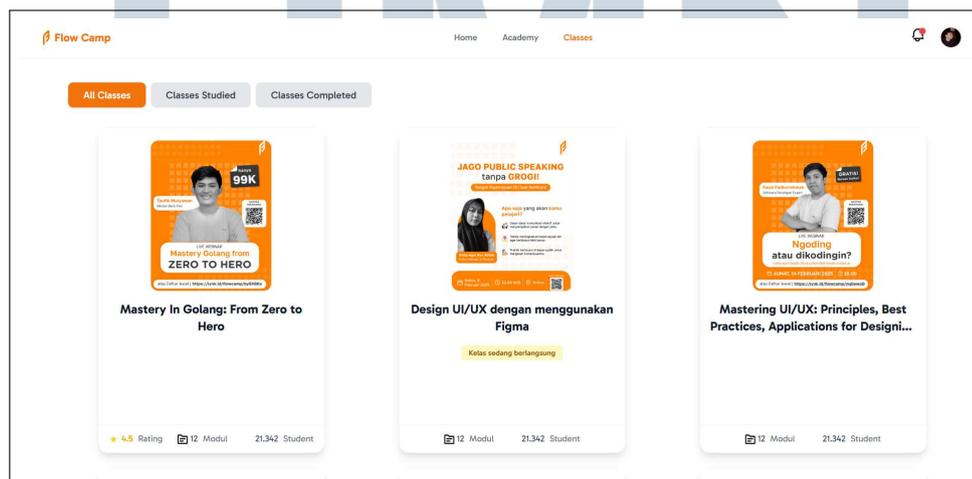


Task	Grades
Pengenalan Golang & Setup Returned 2 February 2025, 10.00 AM	60
Struktur Data dalam Golang Returned 7 February 2025, 10.00 AM	55
Penerapan OOP dalam Golang Returned 13 February 2025, 10.00 AM	50
Fungsi dan Error Handling Returned 18 February 2025, 10.00 AM	40
Struct, Method, dan Interface Returned 23 February 2025, 10.00 AM	30
Pointer dan Manajemen Memori Returned 28 February 2025, 10.00 AM	20

Gambar 3.39. *Popup* Detail Nilai *Not Passed*

Gambar 3.39 menunjukkan tampilan detail nilai siswa dengan status kelulusan *Not Passed*.

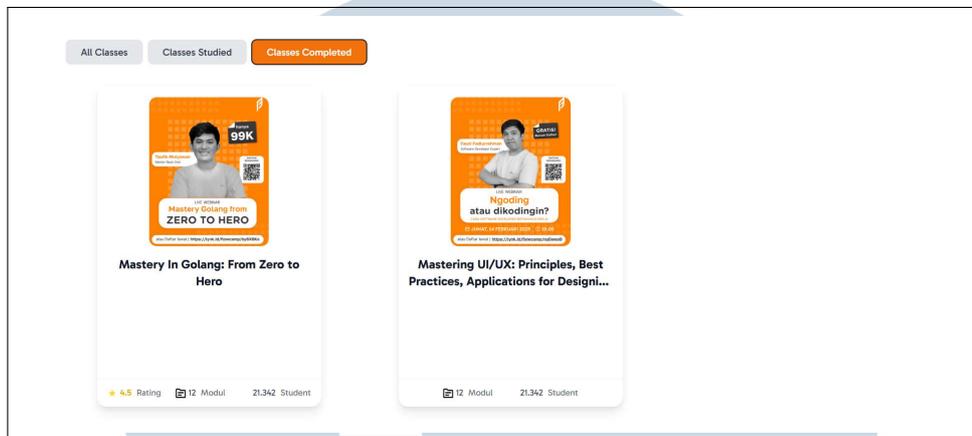
## 9. *Classes* di *dashboard mentor*



Gambar 3.40. All Classes

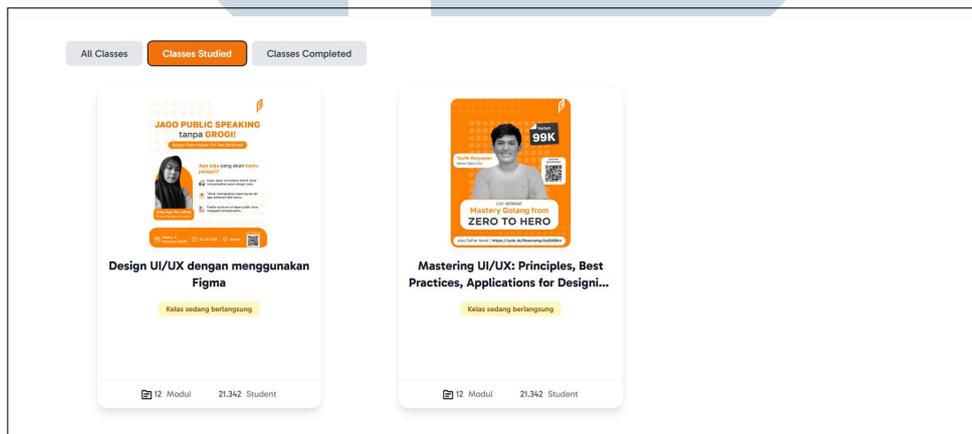
Gambar 3.40 adalah hasil dari implementasi *all classes*. Halaman ini menampilkan daftar kelas yang sedang diambil oleh mentor dan kelas yang

telah diselesaikan. Pada halaman ini juga tersedia tab navigasi untuk memfilter kelas berdasarkan status apakah kelasnya sudah selesai atau belum.



Gambar 3.41. Tab navigasi *class completed*

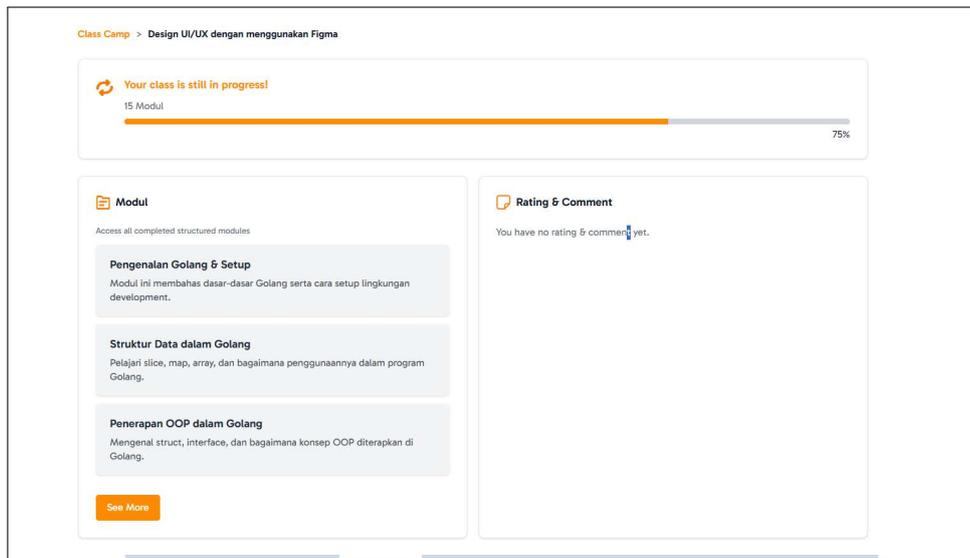
Gambar 3.41 menampilkan fitur *tab* navigasi filter *class completed*.



Gambar 3.42. Tab navigasi *class studied*

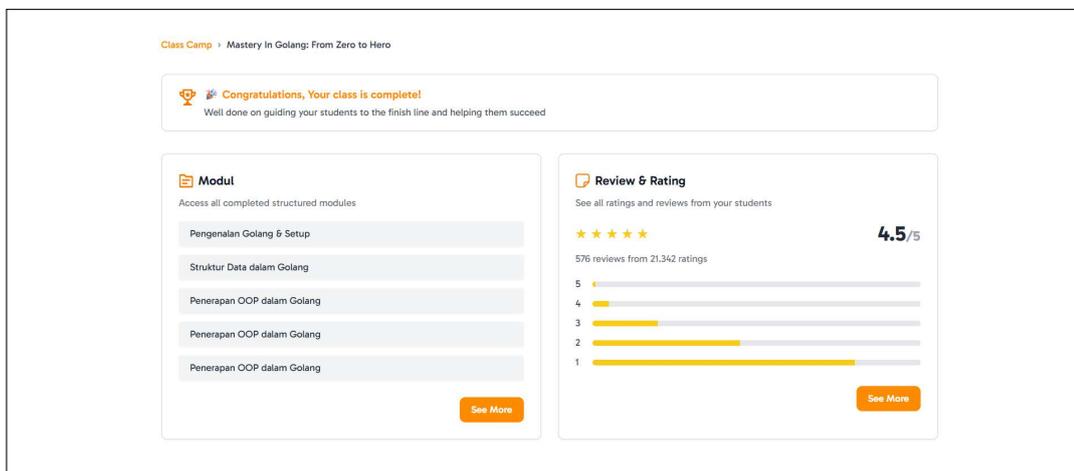
Gambar 3.42 menampilkan fitur *tab* navigasi filter *class studied* yang terdapat indikator visual pada kartu kelas ketika kelas sedang diambil.

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA



Gambar 3.43. Detail *Class Studied*

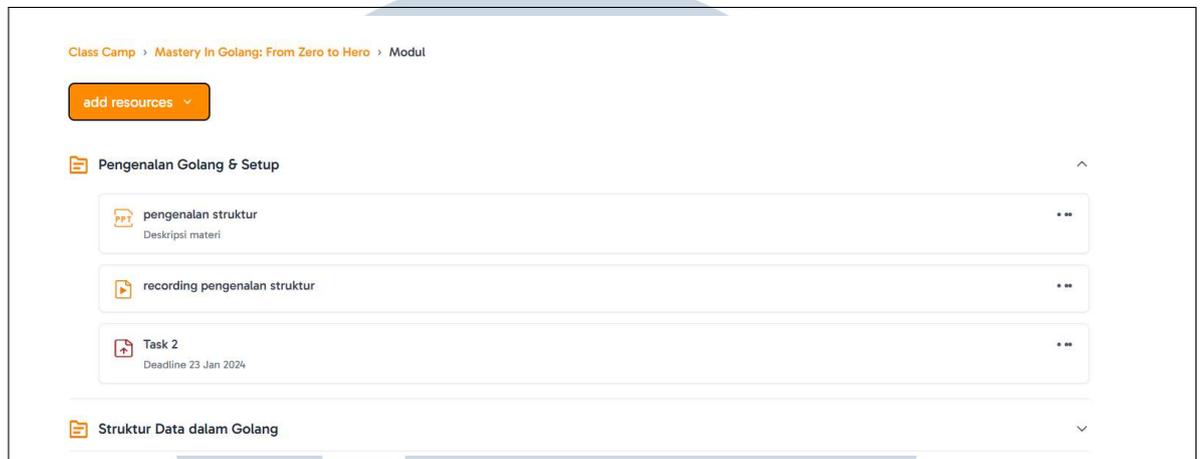
Gambar 3.43 menunjukkan tampilan halaman detail kelas untuk kelas yang masih berlangsung. Di bagian atas halaman, terdapat informasi status kelas yang menunjukkan bahwa kelas belum selesai, disertai dengan indikator progres berupa *progress bar* dan jumlah modul yang telah diselesaikan.



Gambar 3.44. *Class Completed*

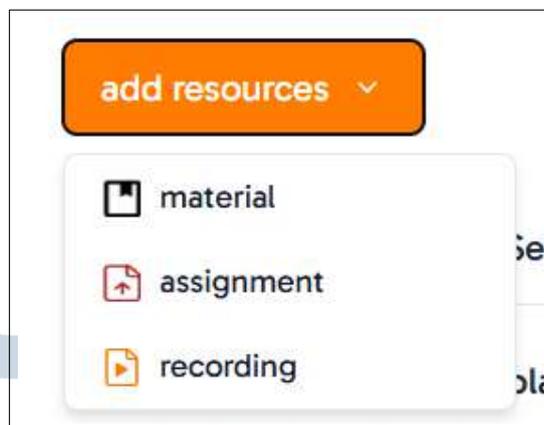
Gambar 3.44 menampilkan halaman detail kelas yang sudah selesai. Pada halaman ini daftar modul yang telah dipelajari, dilengkapi tombol *See More* untuk melihat detail modul dan terdapat bagian *Review and Rating* yang menampilkan penilaian dan *review* dari siswa. Pada bagian atas halaman

terdapat *breadcrumbs* berwarna abu-abu yang menunjukkan halaman aktif dan dapat digunakan untuk kembali ke halaman *All Classes*.



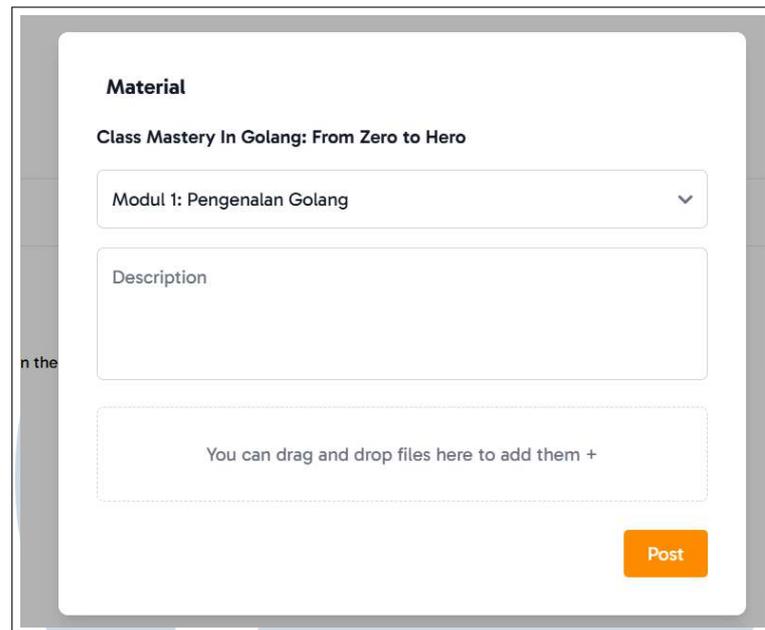
Gambar 3.45. Modul Detail

Gambar 3.45 menampilkan halaman detail modul dalam bentuk *dropdown collapsible*. Saat modul ditekan, *dropdown* akan terbuka dan menampilkan daftar materi yang termasuk dalam modul tersebut.



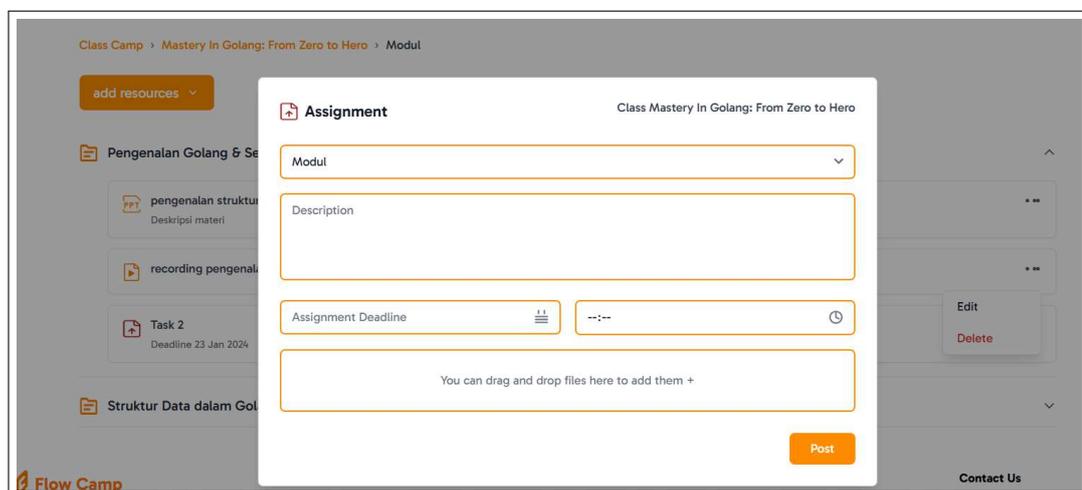
Gambar 3.46. *Dropdown add resources*

Di bagian atas halaman terdapat button “*Add Resources*” berwarna oranye, yang berfungsi untuk menambahkan materi baru ke dalam salah satu modul seperti yang ditampilkan pada gambar 3.46. Mentor dapat memilih jenis *resource* seperti *file*, video, atau tugas baru. Setelah mentor memilih salah satu opsi, sistem akan menampilkan *popup form* yang sesuai dengan jenis materi yang dipilih.



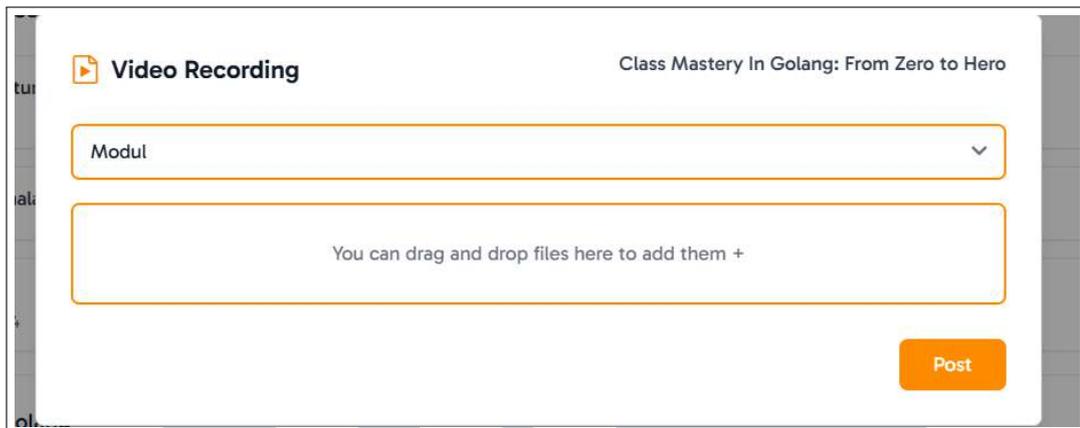
Gambar 3.47. *Material Modul Pop Up*

Gambar 3.47 menampilkan komponen *pop up* ketika memilih untuk menambahkan materi dalam file PPT.



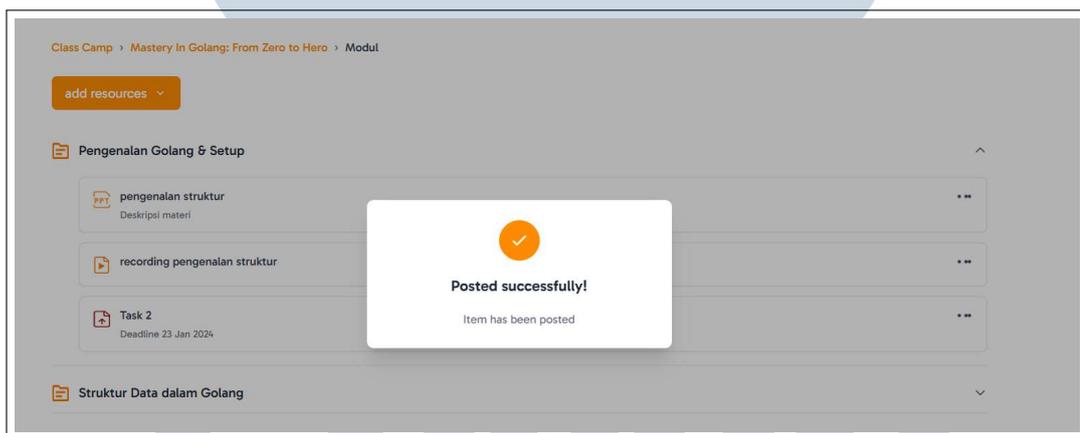
Gambar 3.48. *Assignment Modul Pop Up*

Gambar 3.48 menampilkan komponen *pop up* ketika memilih untuk menambahkan tugas untuk siswa.



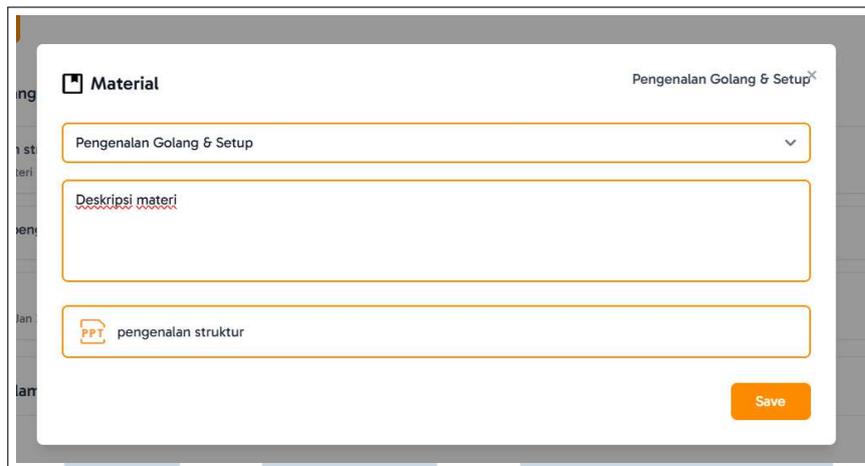
Gambar 3.49. Video Modul Pop Up

Gambar 3.49 menampilkan komponen *pop up* ketika memilih untuk menambahkan materi dalam bentuk video.



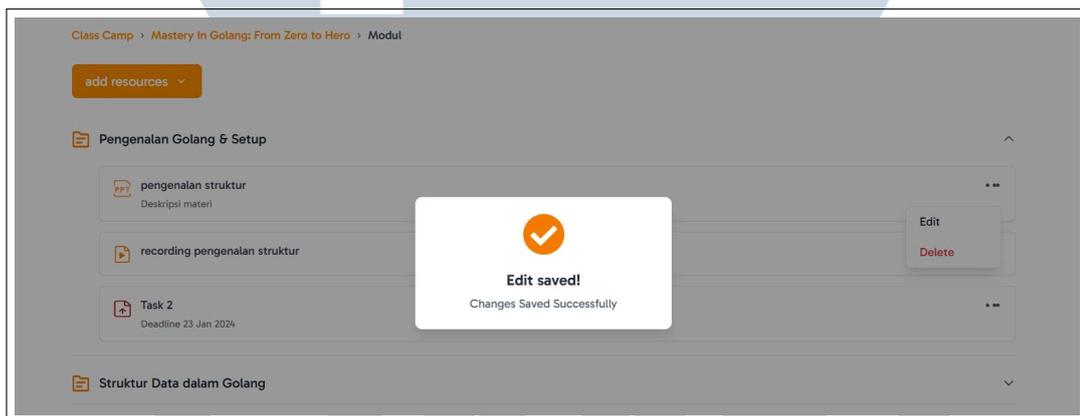
Gambar 3.50. Pop Up Posted

Gambar 3.50 menunjukkan bahwa setelah *form* diisi dan mentor mengklik *button* “Post”, sistem akan menyimpan data secara lokal dan menampilkan *popup* notifikasi yang menandakan bahwa materi berhasil ditambahkan. Setelah itu, materi tersebut langsung muncul pada daftar konten modul yang bersangkutan. Pada halaman detail modul, setiap item materi memiliki ikon tiga titik vertikal di sisi kanan. Ikon ini berfungsi sebagai menu “More Options”. Ketika ikon tersebut ditekan, akan muncul opsi tambahan seperti edit dan *delete*. Jika mentor memilih opsi edit, sistem akan menampilkan kembali *pop-up* sesuai dengan kategori materi yang ingin diedit.



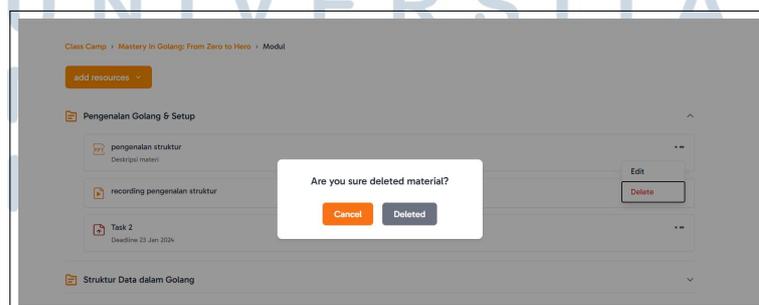
Gambar 3.51. Edit materi

Gambar 3.51 menampilkan *popup* materi yang ingin di edit oleh mentor.



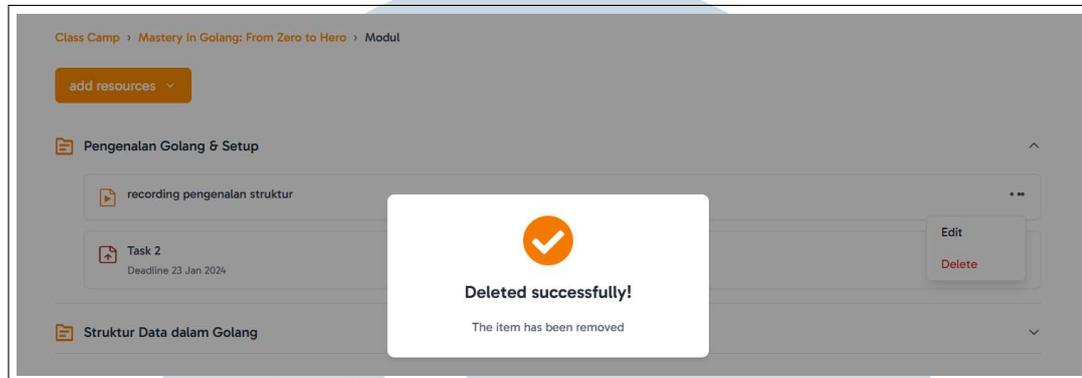
Gambar 3.52. *Edit saved popup*

Gambar 3.52 menunjukkan bahwa setelah mentor menyelesaikan proses pengeditan dan menekan *button save*, sistem akan menampilkan *pop-up* notifikasi yang menandakan bahwa perubahan berhasil disimpan.



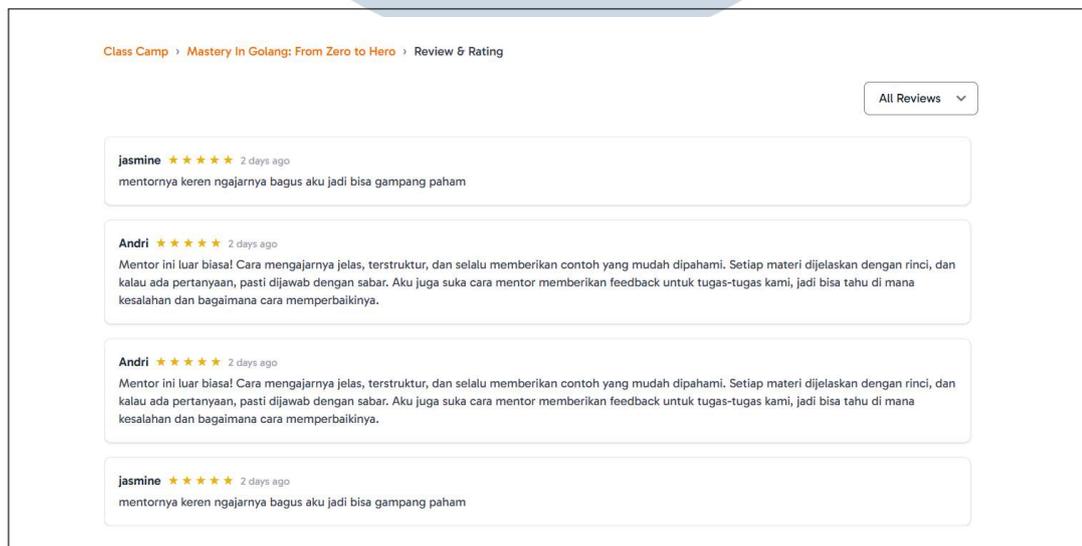
Gambar 3.53. *Pop Up Konfirmasi*

Ketika mentor menekan opsi *delete* pada menu *More Options*, sistem akan menampilkan *pop-up* konfirmasi seperti yang ditunjukkan pada gambar 3.53



Gambar 3.54. *Pop up Delete*

Gambar 3.54 menunjukkan jika mentor menekan *button Yes*, maka sistem akan menampilkan notifikasi bahwa materi berhasil dihapus, dan item tersebut akan otomatis hilang dari daftar modul.



Gambar 3.55. *See more Review and Rating*

Gambar 3.55 menampilkan halaman komentar dan *rating* bintang yang diberikan oleh siswa terhadap kelas yang telah mereka ikuti.

## 3.5 Kendala dan Solusi yang Ditemukan

### 3.5.1 Kendala

Kendala yang ditemukan selama proses kerja magang pada PT Flow Byte Digital adalah sebagai berikut:

- *Design UI/UX* yang seharusnya sudah *final* mengalami beberapa kali revisi dari *Project Manager* sehingga harus melakukan penyesuaian ulang terhadap *frontend* yang sudah di implementasikan ke dalam *web*.
- Di tengah proses pengembangan proyek, terjadi perubahan kebijakan terkait *framework styling* yang awalnya dirancang menggunakan *Bootstrap* diganti dengan *Tailwind CSS*.
- Tidak adanya tim *backend* yang terlibat dalam proyek ini sehingga proyek ini dikerjakan tanpa integrasi dengan API
- Pada bulan ketiga proyek, seluruh anggota tim UI/UX mengundurkan diri karena masa magangnya telah selesai sehingga beberapa fitur belum memiliki *design final*.

### 3.5.2 Solusi

Solusi yang dilakukan untuk menyelesaikan kendala yang ditemukan adalah sebagai berikut:

- Melakukan revisi *design* sesuai arahan dan terus berkoordinasi dengan *Project Manager* untuk memahami revisi secara tepat
- Melakukan adaptasi terhadap *Tailwind CSS* dan mengubah semua komponen yang sudah dibuat agar sesuai dengan *Tailwind CSS*.
- Menyesuaikan pengembangan proyek agar tetap berjalan tanpa integrasi *backend* dan menggunakan *local storage* untuk mengelola data sementara.
- Melakukan peninjauan terhadap komponen-komponen desain yang masih tersedia di Figma untuk mengidentifikasi elemen yang masih dapat dikembangkan, serta menjadikannya sebagai referensi dalam pengembangan komponen selanjutnya.