



MANUAL BOOK PENGEMBANGAN (FRONTEND)

PENGEMBANGAN DAN IMPLEMENTASI PEMBELAJARAN BAHASA ARAB PLATFORM BERBASIS WEB

“QOLAMI”

DOKUMENTASI MANUAL BOOK PENGEMBANGAN (FRONTEND)

Disusun oleh:

HERIKA HAYURANI, S.Kom., M. Kom.

ANGGA ADITYA PUTRA PRATAMA	140 2020 008
RIZAL ABIMANYU	140 2020 070
TARA THANIA ANANTA	140 2020 072

**PROGRAM SARJANA TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS YARSI
JAKARTA
JULI 2024**

KATA PENGANTAR

Puji syukur kami panjatkan ke hadirat Allah, karena atas berkat dan rahmat-Nya, kami dapat menyelesaikan penulisan *Manual Book Pengembangan (Frontend)*. Shalawat serta salam semoga senantiasa dilimpahkan kepada Nabi Muhammad, keluarganya, Para Sahabat, serta seluruh umatnya hingga hari akhir. *Manual Book Pengembangan (Frontend)* ini disusun sebagai panduan untuk memudahkan pengembang selanjutnya aplikasi Qolami Web pada bagian Front-End. Dengan adanya *Manual Book Pengembangan (Frontend)* ini, diharapkan pada tim pengembang selanjutnya dapat memahami fitur-fitur yang telah diterapkan dan memahami teknisi atau cara kerja sebagai pengembang front-end.

Aplikasi Qolami Web ini merupakan penerapan lanjutan dari aplikasi Qolami Mobile yang dikembangkan oleh Tim SOFUN dan Tim Indomitable. Aplikasi ini dibuat bertujuan agar “Qolami” tidak hanya memiliki akses pada satu (1) platform saja yaitu mobile dengan sistem operasi, tetapi memiliki beberapa platform seperti Web dan Mobile yang bisa digunakan di beberapa sistem operasi seperti Android dan Apple. Bapak Ali Ma’Sum, M.Ed sebagai klien dari Aplikasi Qolami Web berkenan memberikan kepercayaan kepada tim ART untuk membuat sebuah aplikasi Qolami berbasis platform web. Aplikasi ini juga dibuat dengan harapan dapat bermanfaat bagi yang ingin belajar Bahasa Arab yang bisa diakses dimana saja dan kapan saja.

Akhir kata, penulis mewakili segenap Tim ART. Berterima kasih kepada semua pihak yang berkontribusi terciptanya aplikasi terutama Klien kami yaitu Bapak Ali Ma’sum yang telah sangat keras menciptakan sebuah ide dan inovasi dalam membangun aplikasi Bahasa Arab demi kemajuan ilmu umat manusia. Semoga *Manual Book Pengembangan (Frontend)* ini dapat berguna baik dari sisi Pengembang selanjutnya dan Klien yang bertujuan dapat mengembangkan aplikasi ini secara berkala dari Fitur yang semakin maju dan bermanfaat. Semoga Allah SWT memberikan keberkahan yang berlimpah dari segala sisi baik.

Jakarta, 9 Juli 2024

Tim ART

Daftar Isi

BAB 1 PENDAHULUAN	4
1.1 LATAR BELAKANG	5
1.2 TUJUAN PENGGUNAAN	6
1.3 TECH-STACK	6
1.3.1 REACT.JS (JAVASCRIPT).....	6
1.3.2 TAILWIND (CSS)	6
1.3.3 BOOTSTRAP (CSS)	7
1.3.4 AXIOS	7
1.3.5 FIGMA	7
1.3.6 NODE.JS	7
1.3.7 GITHUB	7
1.3.8 VERCCEL	8
BAB 2 PENGEMBANGAN.....	9
2.1 PERSIAPAN	9
2.1.1 Tahap Pertama – Menuju Github Organisasi Qolami Web	9
2.1.2 Tahap Kedua – Menuju Repository Frontend Qolami Web	9
2.1.3 Tahap Ketiga – Menyalin Link Repository Frontend Qolami Web	10
2.1.4 Tahap Keempat – Menuju Commad Prompt dan melakukan Git Clone	10
2.1.5 Tahap Kelima – Menuju Text Editor (jika VS CODE)	10
2.1.6 Tahap Keenam – Menginstall Node Package Manager.....	11
2.1.7 Tahap Ketujuh – Menjalankan Aplikasi.....	13
2.2 DESKRIPSI FOLDER & FILE.....	14
2.2.1 /node_modules	14
2.2.2 /public	14
2.2.3 /src	14
2.1.4 tailwind.config.js	15
2.3 PENAMBAHAN FITUR	15
2.3.1 PELAJARAN	15
2.3.2 LATIHAN.....	19
2.3.3 UJIAN.....	26
BAB 3 PENUTUP	34
3.1 KESIMPULAN	34
3.2 SARAN	34

DAFTAR GAMBAR

Gambar 1 - Halaman Github Organization Repository “OQLAMI WEB”	9
Gambar 2 - Halaman Repository “qolami-web”	9
Gambar 3 - Halaman Repository "qolami-web"	10
Gambar 4- Halaman Command Prompt	10
Gambar 5 - Halaman Command Prompt	11
Gambar 6 - Halaman VS CODE	12
Gambar 7 - Halaman VS CODE	13
Gambar 8 - Halaman Qolami Web "Beranda"	14
Gambar 9 - Struktur Kode -KursusData.js-	16
Gambar 10 - Struktur Kode -KursusData.js-.....	16
Gambar 11 - Struktur Kode -ContentDetailKursus.jsx-.....	17
Gambar 12 - Struktur Kode -ContentDetailKursus.jsx-	17
Gambar 13 - Struktur Kode -ContentDetail2Kursus.jsx-	18
Gambar 14 - Struktur Kode -ContentDetail2Kursus.jsx-	18
Gambar 15 - Halaman hasil dari KursusData.js (Halaman Kursus).....	19
Gambar 16 - Struktur Kode -KursusData.js-.....	20
Gambar 17 - Struktur Kode -KursusData.js-.....	20
Gambar 18 - Struktur Kode -ContentDetailKursus.jsx-	21
Gambar 19 - Struktur Kode -ContentDetailKursus.jsx-	21
Gambar 20 - Struktur Kode -ContentDetailKursus.jsx-	22
Gambar 21 - Struktur Kode -ContentDetailKursus.jsx-	22
Gambar 22 - Struktur Kode -ContentDetailKursus.jsx-	23
Gambar 23 - Struktur Kode -ContentDetailKursus.jsx-	23
Gambar 24 - Halaman hasil dari KursusData.js (Halaman Kursus).....	24
Gambar 25 - Halaman hasil dari ContentDetailKursus.js (Halaman Detail Kursus)	24
Gambar 26 - Halaman hasil dari ContentDetail2Kursus.js (Halaman dalam Detail Kursus (Gambar))	25
Gambar 27 - Halaman hasil dari ContentDetail2Kursus.js (Halaman dalam Detail Kursus (Video))..	25
Gambar 28 - Halaman hasil dari ContentDetail2Kursus.js (Halaman dalam Detail Kursus (Audio)) .	26
Gambar 29 – Struktur Kode -KursusData.js-	26
Gambar 30 – Struktur Kode -KursusData.js-	27
Gambar 31 - Struktur Kode -ContentDetailKursus.jsx-	27
Gambar 32 - Struktur Kode -ContentDetailKursus.jsx-	27
Gambar 33 - Struktur Kode -ContentDetailKursus.jsx-	28
Gambar 34 - Struktur Kode -ContentDetailKursus.jsx-	28
Gambar 35 - Struktur Kode -ContentDetailKursus.jsx-	29
Gambar 36 - Struktur Kode -ContentDetailKursus.jsx-	29
Gambar 37 - Struktur Kode -ContentDetailKursus.jsx (useEffect)-.....	30
Gambar 38 - Struktur Kode -ContentDetailKursus.jsx- (exam-one).....	31
Gambar 39 - Struktur Kode -ContentDetailKursus.jsx- (exam-two).....	31
Gambar 40 - Struktur Kode -ContentDetailKursus.jsx- (exam-three).....	32
Gambar 41 – Struktur Kode -ContentDetailKursus.jsx-.....	32
Gambar 42 – Struktur Kode -ContentDetailKursus.jsx-.....	33
Gambar 43 - Halaman hasil dari KursusData.js (Halaman Kursus).....	33
Gambar 44 - Halaman hasil dari ContentDetailKursus.js (Halaman Ujian)	33

BAB 1

PENDAHULUAN

1.1 LATAR BELAKANG

Qolami Web merupakan website yang dirancang untuk menunjang pembelajaran bahasa Arab dengan metode Qolami. Dengan adanya website ini pengguna dapat dengan mudah dan nyaman mengakses berbagai materi pembelajaran bahasa Arab kapanpun dan dimanapun. Website ini merupakan evolusi dari android sebelumnya yang dikembangkan oleh tim Indomitable.

Salah satu fitur utama *Qolami Web* adalah mempelajari huruf Hijaiyah, hingga huruf berharakat. Pengguna dapat mempraktikkan pembelajarannya dalam format digital dengan tampilan visual yang jelas dan mudah dibaca. Selain itu, website ini juga menyediakan video pembelajaran tentang huruf Hijaiyah dengan berharakat sehingga membantu pengguna dalam memahami cara membaca dan mengenal bentuk bentuk huruf. Fitur ini sangat berguna bagi mereka yang sedang belajar dari awal atau ingin memahami huruf arab lebih dalam.

Selain itu, *Qolami Web* ini menawarkan fitur Latihan untuk membantu pengguna mempraktikkan pelajaran mereka sebelumnya. Menu Latihan ini mencakup tiga mode seperti; latihan gambar, video, dan audio. Pada mode ini terdapat 4 pertanyaan acak untuk menguji pemahaman pengguna terhadap proses pembelajaran sebelumnya. Soal-soal ini akan menunjukkan hasil sesi latihan dan dapat diulangi untuk mencapai skor yang diinginkan. Pada latihan gambar, pengguna akan memilih jawaban gambar yang sesuai dengan pertanyaan. Untuk video latihan, pengguna perlu memutar video yang tersedia dan kemudian memilih jawaban berdasarkan video tersebut. Selanjutnya bagian latihan audio, pengguna perlu menekan tombol audio untuk mendengar hasil jawaban yang diberikan. *Qolami Web* ini mempunyai fitur Ujian, dan pada mode ini terdapat 10 pertanyaan acak dari latihan gambar, video dan audio.

Aplikasi *Qolami Web* dikembangkan untuk mempermudah proses pembelajaran bahasa Arab baik bagi siswa maupun guru. Versi web aplikasi ini memungkinkan akses yang lebih luas dan fleksibilitas lebih. *Qolami Web* menjadi Solusi lengkap bagi siapa saja yang ingin meningkatkan pengetahuan pembelajaran Bahasa Arab.

1.2 TUJUAN PENGGUNAAN

Tujuan dibuatnya *Manual Book (Front-end)* ini adalah sebagai pedoman bagi pengembang Front-End dari Aplikasi *Qolami Web* versi selanjutnya. Front-End adalah sisi bagian depan pengguna yang dimana seorang pengguna bisa melihat suatu hal secara visual atau sekaligus bisa melakukan sebuah interaksi sesuai fitur yang telah disediakan seperti tombol dan lainnya. Front-End sangat berperan penting karena semakin bagus tampilan, akan semakin betah seorang pengguna menggunakan aplikasi kita, dan jika sebaliknya, meskipun pada aplikasi ini memiliki banyak fitur, jika tampilan yang dimilikinya sangat tidak cocok di mata, seorang pengguna akan berhenti menggunakan aplikasi tersebut atau seorang pengguna bisa beralih ke aplikasi lain yang memiliki fitur yang sama.

Seorang pengembang Front-End dalam *platform web* harus harus menerapkan responsif dalam aplikasi yang dikembangkannya sehingga mempertahankan kerapian tampilan aplikasi bila dibuka menggunakan platform mobile. Seorang Front-End juga harus memikirkan bagaimana struktur kode yang baik dan tidak terjadi pemborosan kode. Jika terjadi, Performa Aplikasi akan mengalami gangguan, lambatnya dalam merender sebuah konten, bahkan terjadi hal yang tidak diinginkan seperti crashnya sebuah aplikasi. Maka dari itu Pengembang harus memikirkan sebuah logika yang baik dan memilih sebuah *Library* dan *Framework* yang baik.

1.3 TECH-STACK

1.3.1 REACT.JS (JAVASCRIPT)¹

React.JS adalah Library Javascript yang dikembangkan oleh Facebook yang digunakan dalam pengembangan perangkat lunak seperti mobile dan desktop. *React.JS* memiliki fasilitas yang dimana yaitu pembuatan Komponen antarmuka yang interaktif, stateful sehingga dapat gunakan secara berulang kali.

1.3.2 TAILWIND (CSS)²

Tailwind CSS adalah *utility-first Framework CSS* yang didesain untuk mempermudah dan mempercepat aplikasi dalam membuat sebuah tampilan aplikasi yang lebih interaktif dan memanjakan mata. Tailwind CSS termasuk tipe utility yang lengkap dikarenakan meskipun setiap utility telah disediakan, kita bisa mengkustom utility sesuai dengan keperluan.

¹ <https://react.dev/>

² <https://tailwindcss.com/>

1.3.3 BOOTSTRAP (CSS)³

Bootstrap adalah Framework CSS yang berfokus pada bagian depan untuk membuat sebuah tampilan website yang modern, responsive dan mobile-friendly. Bootstrap sangat mudah digunakan bagi yang baru terjun ke Pengembangan dunia Web karena dia merupakan sistem Component yang Dimana Component yang dimaksud contoh seperti Header, *Boostrap* telah memiliki dan kita tinggal menyalin kode tersebut ke dalam berkas HTML atau lain sebagainya yang telah mendukung HTML.

1.3.4 AXIOS⁴

Axios adalah Klien HTTP berbasis Promise Node.Js untuk dan browser. Itu *Isomorfik* (dapat berjalan di browser dan nodejs dengan basis kode yang sama). Di sisi server menggunakan httpmodul node.js asli, sedangkan di sisi klien (browser) menggunakan XMLHttpRequests. *Axios* dapat mempermudah kita dalam berinteraksi dengan API dengan baris kode yang lebih sedikit disbanding tanpa menggunakannya.

1.3.5 FIGMA⁵

Figma merupakan tools yang membantu Pengembang dalam mendesain apapun, dalam segi Software yaitu mendesain aplikasi sebelum ke tahapan dalam SDLC yaitu Development. ([Link Figma Oolami Web](#))

1.3.6 NODE.JS⁶

Node.js adalah runtime environment untuk JavaScript yang bersifat open-source dan cross-platform. Dengan Node.js kita dapat menjalankan kode JavaScript di mana pun, tidak hanya terbatas pada lingkungan browser.

1.3.7 GITHUB⁷

Github adalah platform pengembangan Software berbasis cloud yang digunakan untuk menyimpan sebuah Proyek yang berisi kode, bahkan aplikasi ini bisa berkolaborasi yang Dimana satu (1) proyek atau repository bisa dikerjakan lebih dari satu (1) orang.

³ <https://getbootstrap.com/>

⁴ <https://axios-http.com/docs/intro>

⁵ <https://www.figma.com/>

⁶ <https://nodejs.org/id>

⁷ <https://github.com/>

1.3.8 VERCEL⁸

Vercel adalah platform cloud untuk situs statis dan fungsi tanpa server. Hal ini memungkinkan pengembang untuk membangun dan menyebarkan proyek web dengan mudah. Dengan Vercel, Anda dapat menyebarkan proyek Anda ke cloud dan mendapatkan CDN global, domain khusus, dan HTTPS otomatis.

1.3.9 GIT⁹¹⁰

Git adalah tools *version control system* (VCS) yang digunakan untuk mengelola perubahan kode selama pengembangan perangkat lunak. Tools ini digunakan para *programmer* untuk mengelola perubahan dan pelacakan pada *source code* secara lebih mudah.

⁸ <https://vercel.com/>

⁹ <https://www.dewaweb.com/blog/mengenal-git/>

¹⁰ <https://git-scm.com/downloads>

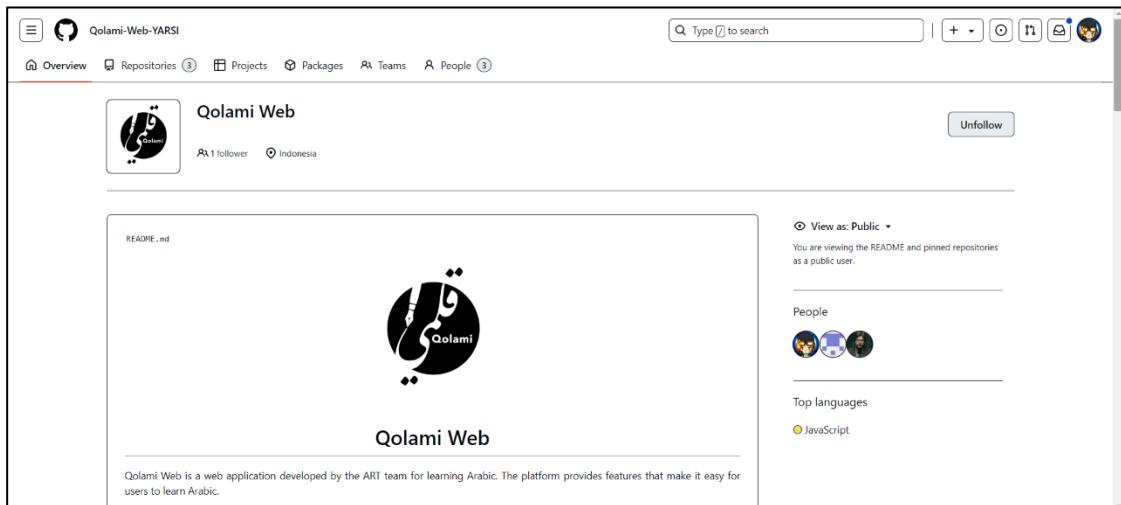
BAB 2

PENGEMBANGAN

2.1 PERSIAPAN

2.1.1 Tahap Pertama – Menuju Github Organisasi Qolami Web

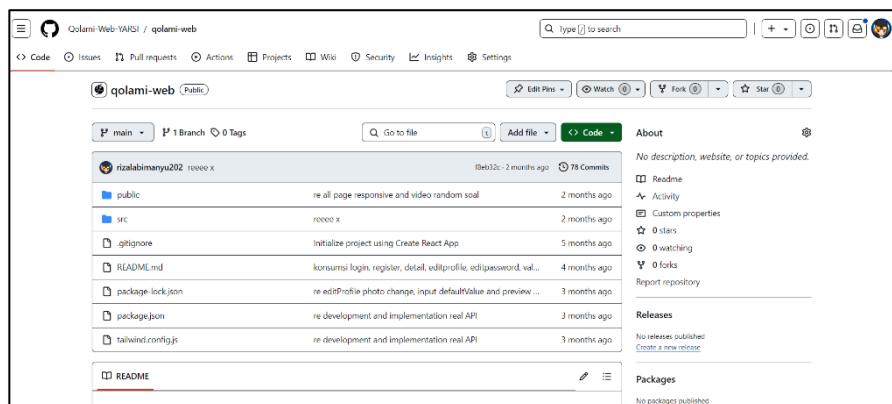
Untuk Pengembang Selanjutnya (opsional) tahap pertama yaitu menuju Github Organisasi [Qolami Web](#). Jika Sudah tampilan akan seperti ini.



Gambar 1 - Halaman Github Organization Repository “OQLAMI WEB”

2.1.2 Tahap Kedua – Menuju Repository Frontend Qolami Web

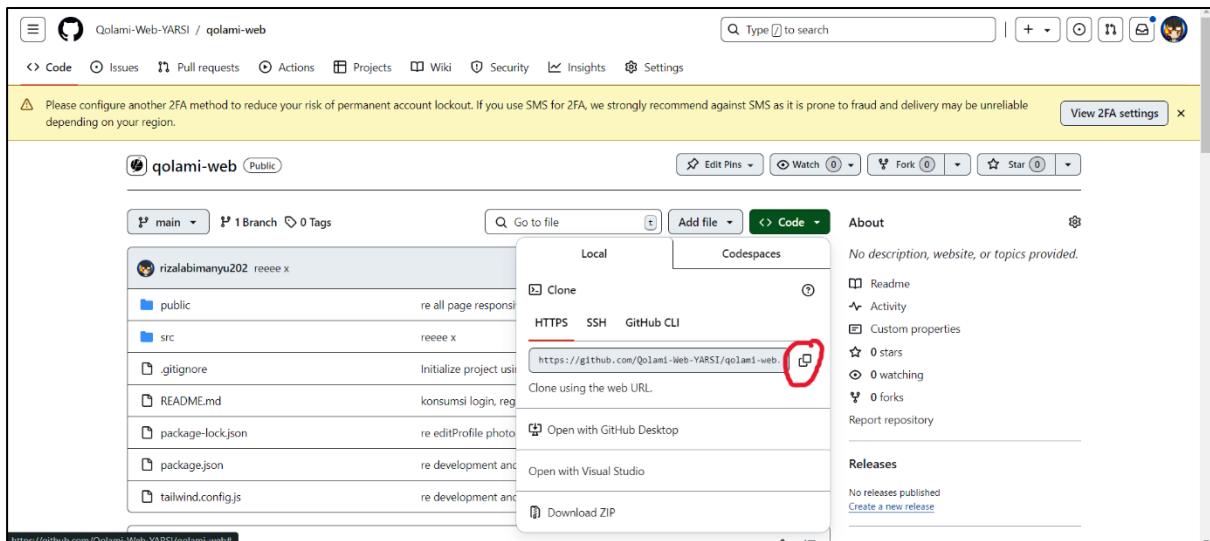
Selanjutnya yaitu menuju [Repository Frontend Qolami Web](#). Bisa diliat pada l pada gambar 2 jika sudah masuk.



Gambar 2 - Halaman Repository “qolami-web”

2.1.3 Tahap Ketiga – Menyalin Link Repository Frontend Qolami Web

Selanjutnya yaitu menyalin link repository Frontend Qolami Web. Untuk tahap pertama menuju tombol hijau bertulisan code lalu menekan tombol yang dilingkar merah (yaitu menyalin link repository).



Gambar 3 - Halaman Repository "qolami-web"

2.1.4 Tahap Keempat – Menuju Command Prompt dan melakukan Git Clone

Selanjutnya yaitu menuju *command prompt* (saat ini saya berada dihalaman C:\coba\haha) dengan mengetik git clone <link repository yang telah disalin>. Contohnya bisa diliat gambar dibawah ini.

```
C:\Windows\System32\cmd.exe Microsoft Windows [Version 10.0.22631.3880]
(c) Microsoft Corporation. All rights reserved.

C:\coba\haha>git clone https://github.com/Qolami-Web-YARSI/qolami-web.git
Cloning into 'qolami-web'...
remote: Enumerating objects: 1138, done.
remote: Counting objects: 100% (257/257), done.
remote: Compressing objects: 100% (148/148), done.
remote: Total 1138 (delta 187), reused 162 (delta 99), pack-reused 881
Receiving objects: 100% (1138/1138), 21.58 MiB | 1.40 MiB/s, done.
Resolving deltas: 100% (780/780), done.

C:\coba\haha>
```

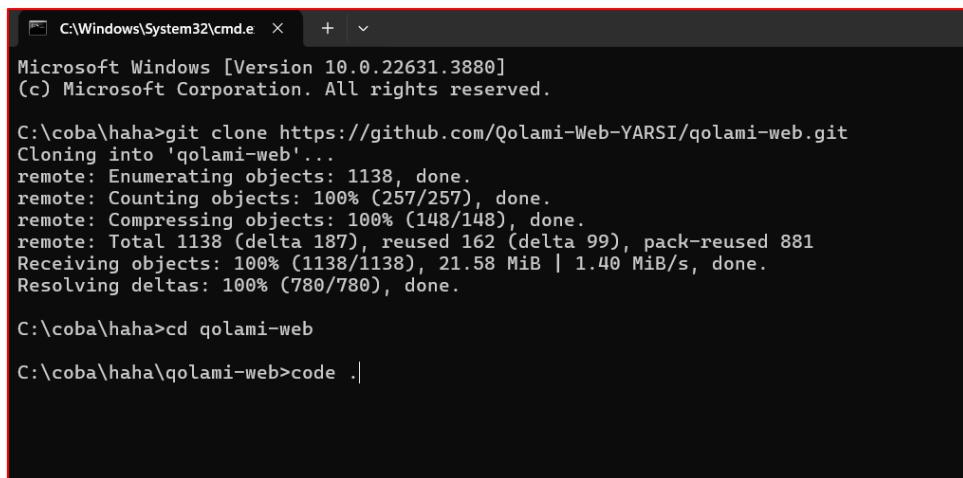
Gambar 4- Halaman Command Prompt

2.1.5 Tahap Kelima – Menuju Text Editor (jika VSCODE)

Selanjutnya yaitu menuju text editor dengan cara menuju folder yang telah kita clone dengan cara “cd <Nama Folder>”, biasanya jika kita mengclone repository ke

local, Nama foldernya sama dengan link repository paling terakhir, tapi tulisan .gitnya dihilangin. Bisa diliat contoh dibawah ini pada Text Highlight berwarna Hijau. Setelah berada difolder yang telah kita clone, kita mengetik “code .” untuk menuju ke Text Editor VS CODE.

Contoh: <https://github.com/Qolami-Web-YARSI/qolami-web.git>



```
C:\Windows\System32\cmd.e x + v
Microsoft Windows [Version 10.0.22631.3880]
(c) Microsoft Corporation. All rights reserved.

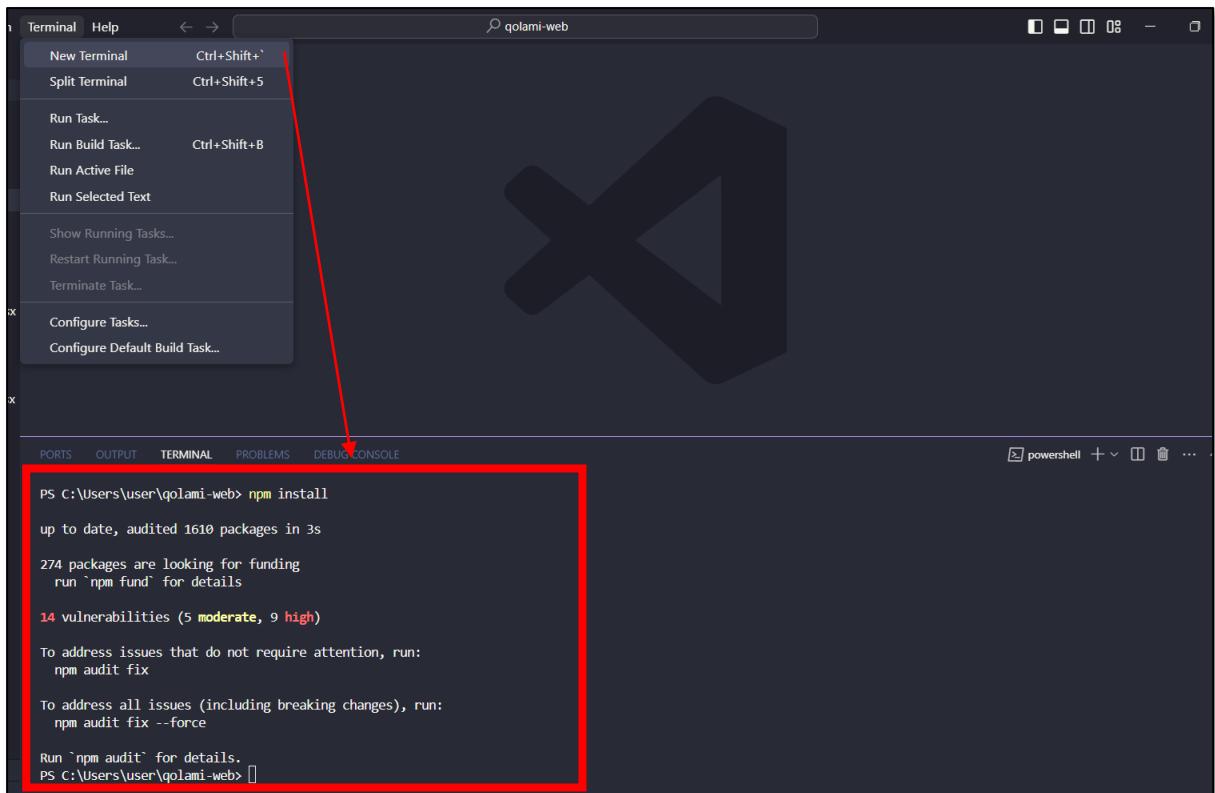
C:\coba\haha>git clone https://github.com/Qolami-Web-YARSI/qolami-web.git
Cloning into 'qolami-web'...
remote: Enumerating objects: 1138, done.
remote: Counting objects: 100% (257/257), done.
remote: Compressing objects: 100% (148/148), done.
remote: Total 1138 (delta 187), reused 162 (delta 99), pack-reused 881
Receiving objects: 100% (1138/1138), 21.58 MiB | 1.40 MiB/s, done.
Resolving deltas: 100% (780/780), done.

C:\coba\haha>cd qolami-web
C:\coba\haha\qolami-web>code .
```

Gambar 5 - Halaman Command Prompt

2.1.6 Tahap Keenam – Menginstall Node Package Manager

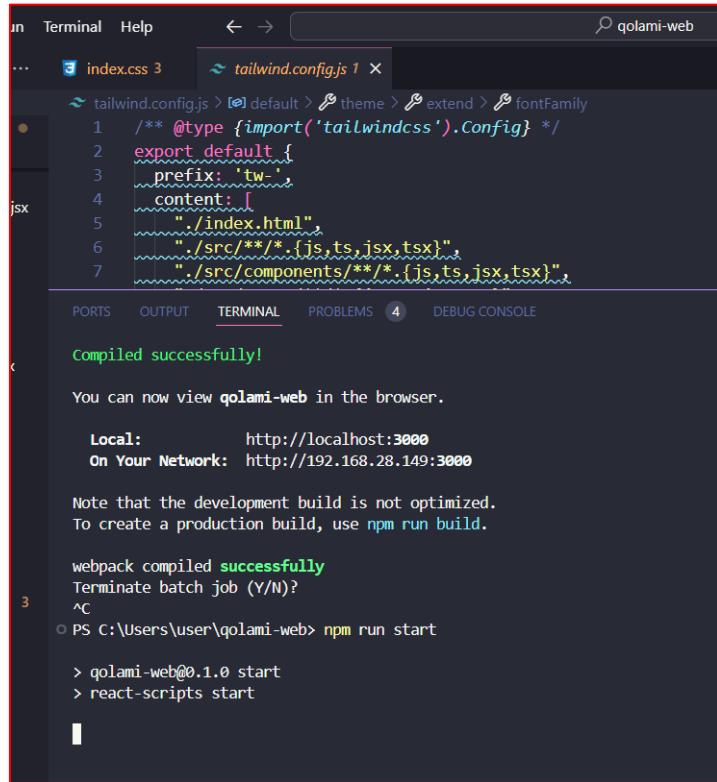
Setelah menuju VS Code, kita menuju terminal dengan cara tekan tombol “Terminal” dipaling atas lalu tekan “New Terminal”, lalu akan menuju kotak berwarna merah, lalu ketikkan “npm install” seperti pada gambar 6.



Gambar 6 - Halaman VS CODE

2.1.7 Tahap Ketujuh – Menjalankan Aplikasi

Setelah npm telah terinstall, kita jalankan aplikasinya dengan mengetik “npm run start” dan tunggu berapa detik, secara otomatis aplikasi akan terbuka di browser. Contohnya seperti pada gambar dibawah ini.



The screenshot shows the VS Code interface with the terminal tab active. The terminal window displays the following output:

```
tailwind.config.js
1  /**
2  * @type {import('tailwindcss').Config}
3  */
4  export default {
5   prefix: 'tw-',
6   content: [
7     './index.html',
8     './src/**/*.{js,ts,jsx,tsx}',
9     './src/components/**/*.{js,ts,jsx,tsx}'
```

Compiled successfully!

You can now view **qolami-web** in the browser.

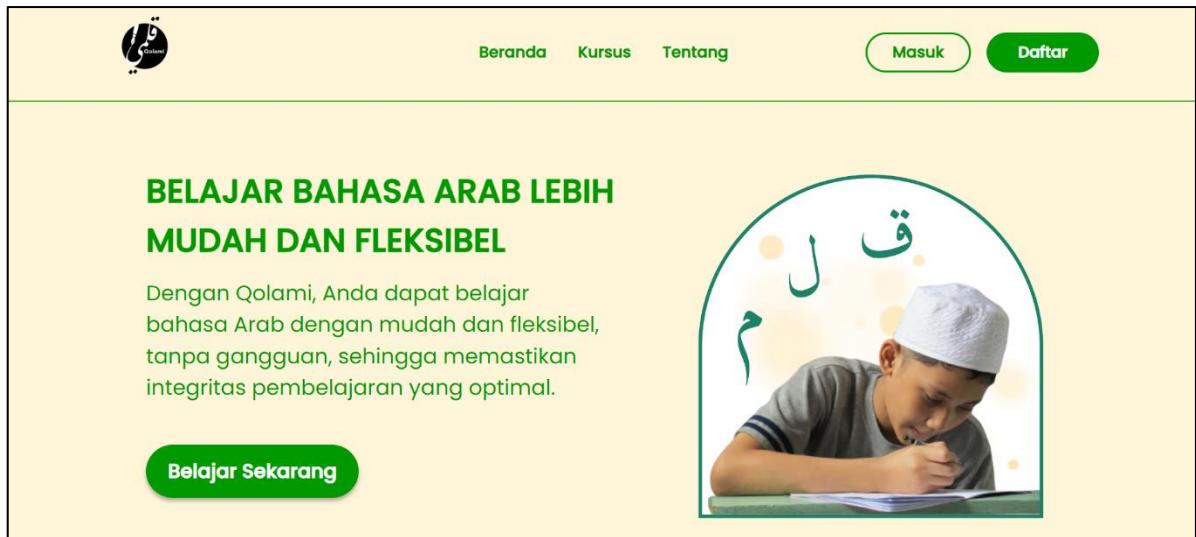
Local: http://localhost:3000
On Your Network: http://192.168.28.149:3000

Note that the development build is not optimized.
To create a production build, use `npm run build`.

webpack compiled successfully
Terminate batch job (Y/N)?
^C
PS C:\Users\user\qolami-web> npm run start

> qolami-web@0.1.0 start
> react-scripts start

Gambar 7 - Halaman VS CODE



Gambar 8 - Halaman Qolami Web "Beranda"

2.2 DESKRIPSI FOLDER & FILE

Pada source code yang sudah diunduh dari Github, terdapat beberapa folder dan file yang akan dijelaskan sebagai berikut.

2.2.1 /node_modules

Folder ini berisi sebuah import-an dan library yang telah diinstal untuk keperluan membangun proyek.

2.2.2 /public

Folder ini berisi file yang terdiri dari file HTML yang menjadi sebuah pacuan renderan, webmanifest untuk membuat aplikasi kita bisa diinstal didalam desktop pengguna dan berisi gambar logo dan gambar untuk keperluan konten aplikasi serta juga font dan folder font

2.2.2.1 /fonts

Folder ini berisi file untuk keperluan tampilin huruf pada aplikasi yang berformat .otf dan .ttf

2.2.3 /src

Folder ini berisi file yang sangat penting dalam membangun sebuah aplikasi yang mencakup file .jsx yang berisi file halaman dan file component yang digunakan untuk memanggil file ke halaman mana saja.

2.2.3.1 /components

Folder ini berisi file yang berperan sebagai komponen yang Dimana komponen ini akan digunakan pada file yang berperan halaman atau komponen digunakan didalam komponen

2.2.3.2 /data

Folder ini berisi file javascript yang berisi data dari setiap card kursus yang untuk sekarang memiliki 3 peranan yaitu Pelajaran, Latihan beserta data soal & jawaban dan Ujian besertadata soal & jawaban.

2.2.3.3 /pages

Folder ini berisi file javascript yang berperan sebagai satuan halaman yang berguna juga sebagai perpindahan halaman di sebuah aplikasi.

2.2.3.4 App.jsx

File ini berisi kode yang merupakan render lapisan dua (2) sebelum lapisan satu (1) yaitu file html yang berada di folder public dan juga merupakan interaksi aktifnya aplikasi dari pemanggilan router agar bisa berinteraksi ke halaman lain dari folder pages.

2.2.3.5 index.css

File ini berisi kode yang digunakan untuk menggunakan Framework Tailwind, memaksa background inputan edit berwarna #E9ECEF dan pemanggilan font.

2.1.4 tailwind.config.js

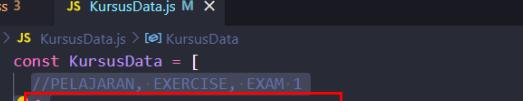
File ini berisi konfigurasi tailwind dari si pengembang dari menggunakan prefix dikarena mennggunakan lebih dari satu (1) framework css, membaca sebuah folder dimana yang menggunakan tailwind yang akan terbaca dengan file .js, .ts, .jsx, .tsx, membuat utility baru dari pemanggilan utility gambar, font, responsive halaman dan kecerahan.

2.3 PENAMBAHAN FITUR

2.3.1 PELAJARAN

1. Langkah Pertama kita menuju file KursusData.js lalu salin kode sesuai dengan garis kotak merah sesuai dengan **Gambar 9** lalu tempelkan kode pada array terakhir variable **KursusData** sesuai dengan garis

kotak merah sesuai dengan **Gambar 10** lalu sesuaikan dengan kebutuhan (*disini karena kita menambahkan Pelajaran 3, jadi sesuaikan dengan apa yang perlu diubah pada Pelajaran 3*).



```
const KursusData = [
  {
    id: "lessons-one",
    nama: "Pelajaran 1",
    gambar: "img1-1.svg",
    deskripsi: "Huruf Hijaiyah",
    detail: [],
    soalJawaban2: []
  }
]
```

Gambar 9 - Struktur Kode -KursusData.js-

```
9257 > { ...
12177 },
12178 [
12179   {
12180     id: "lessons-three",
12181     nama: "Pelajaran 3",
12182     gambar: "img1-1.svg",
12183     deskripsi: "Huruf Berharakat Dhammah",
12184     detail: [],
12185     soalJawaban2: []
12186   },
12187 ];
12188 export default KursusData;
12189
```

Gambar 10 - Struktur Kode -KursusData.js-

- Setelah selesai, Selanjutnya kita menuju ContentDetailKursus.jsx lalu salin kode sesuai dengan garis kotak merah sesuai dengan **Gambar 11** lalu tempelkan kode sesuai dengan garis kotak merah sesuai dengan **Gambar 12** lalu sesuaikan dengan kebutuhan (*disini karena kita menambahkan Pelajaran 3, jadi sesuaikan dengan apa yang diubah pada Pelajaran 3*).

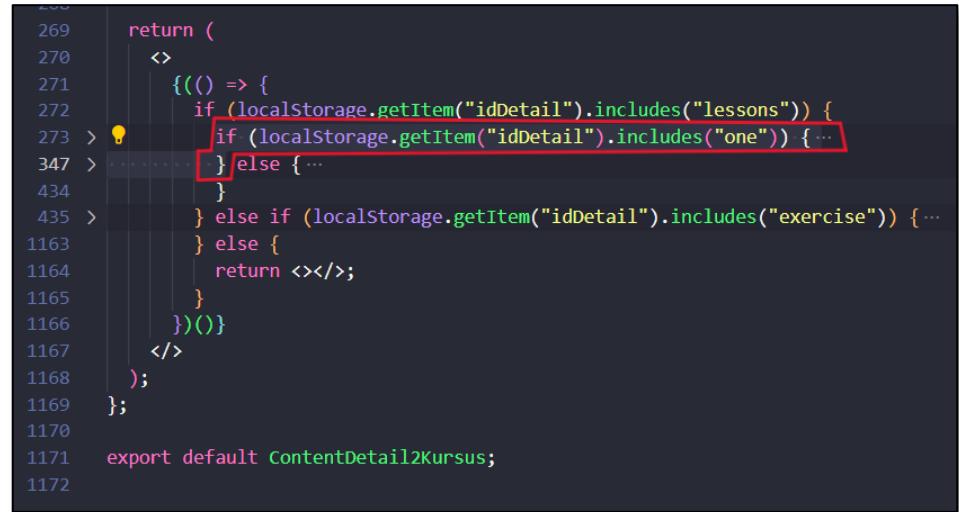
```
403
464     return (
465         <>
466         {(() => {
467             if (id.includes("lessons")) { ...
468         } else if (id.includes("exercise")) {
469             if (id.includes("one")) { ...
470         } else if (id.includes("two")) { ...
471         } else {
472             return <></>;
473         }
474         } else if (id.includes("exam")) { ...
475     } else {
476         return <></>;
477     }
478     })()
479     );
480 };
481
482 export default ContentDetailKursus;
```

Gambar 11 - Struktur Kode -ContentDetailKursus.jsx-

```
403
464     return (
465         <>
466         {(() => {
467             if (id.includes("lessons")) {
468                 if (id.includes("one")) { ...
469             } else if (id.includes("two")) { ...
470             } else if (id.includes("three")) { ...
471             } else {
472                 return <></>;
473             }
474             } else if (id.includes("exercise")) { ...
475         } else if (id.includes("exam")) { ...
476     } else {
477         return <></>;
478     }
479     })()
480     );
481
482 export default ContentDetailKursus;
```

Gambar 12 - Struktur Kode -ContentDetailKursus.jsx-

3. Setelah selesai, Selanjutnya kita menuju ContentDetail2Kursus.jsx lalu salin kode sesuai dengan garis kotak merah sesuai dengan **Gambar 13** lalu tempelkan kode sesuai dengan garis kotak merah sesuai dengan **Gambar 14** lalu sesuaikan dengan kebutuhan (*disini karena kita menambahkan Pelajaran 3, jadi sesuaikan dengan apa yang diubah pada Pelajaran 3*).



```

269   return (
270     <>
271       {(()) => {
272         if (localStorage.getItem("idDetail").includes("lessons")) {
273           if (localStorage.getItem("idDetail").includes("one")) { ...
274           } else { ...
275             }
276           } else if (localStorage.getItem("idDetail").includes("exercise")) { ...
277           } else {
278             return <></>;
279           }
280         })()
281       </>
282     );
283   };
284 
285   export default ContentDetail2Kursus;
286 
```

Gambar 13 - Struktur Kode -ContentDetail2Kursus.jsx-

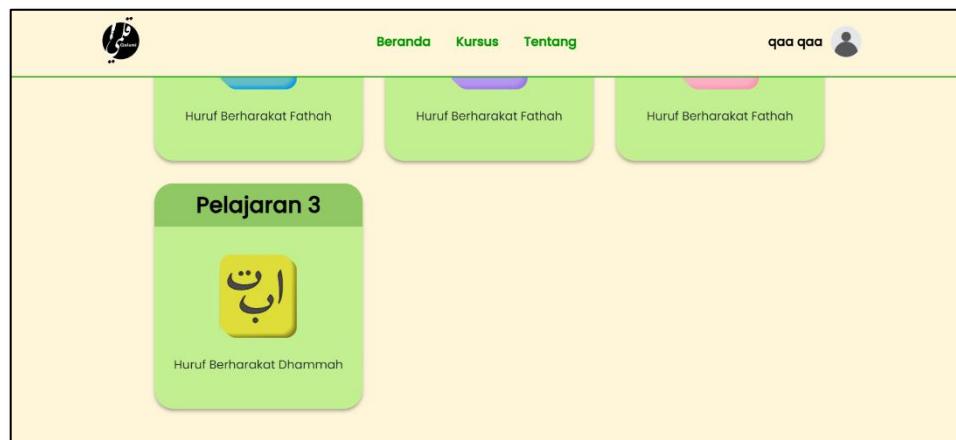


```

269   return (
270     <>
271       {(()) => {
272         if (localStorage.getItem("idDetail").includes("lessons")) {
273           if (localStorage.getItem("idDetail").includes("one")) { ...
274           } else if (localStorage.getItem("idDetail").includes("three")) { ...
275             }
276           } else if (localStorage.getItem("idDetail").includes("exercise")) { ...
277             } else {
278               return <></>;
279             }
280           })()
281         </>
282       );
283     };
284 
285   export default ContentDetail2Kursus;
286 
```

Gambar 14 - Struktur Kode -ContentDetail2Kursus.jsx-

- Hasil untuk **KursusData.js** seperti **Gambar 15**. Untuk **ContentDetailKursus.jsx** dan **ContentDetailKursus2** sepertinya perlu di diskusi dengan Back-End, karena pada Back-End Web ini hanya dibuatkan *lessons-one* dan *lessons-two*, sehingga *lessons-three* akan menghasilkan error dan tidak memunculkan content apapun, sehingga perlu dibuat oleh Back-End.



Gambar 15 - Halaman hasil dari KursusData.js (Halaman Kursus)

2.3.2 LATIHAN

- Langkah pertama kita menuju file KursusData.js lalu salin kode sesuai dengan garis kotak merah sesuai dengan **Gambar 16** lalu tempelkan kode pada array terakhir variable **KursusData** sesuai dengan garis kotak merah sesuai dengan **Gambar 17** lalu sesuaikan dengan kebutuhan (*disini karena kita menambahkan Latihan 3, jadi sesuaikan dengan apa yang perlu diubah pada Latihan 3*).

```
6298 > [ ...
6305 ...
6306 ...
6307 ...
6308 ...
6309 ...
6310 ...
6311 ...
6312 ...
6313 ...
6314 ...
6315 ...
6316 ...
6317 ...
6318 ...
6319 ...
7272 ...
7273 ...
7274 ...
7275 ...
7276 ...
7277 ...
7278 ...
7279 ...
7280 ...
7281 ...
8262 ...
8263 ...
8264 ...
8265 ...
8266 ...
8267 ...
8268 ...
8269 ...
8270 ...
8271 ...
9252 ...
9253 ...
9254 ...
9255 ...
9256 ... ]
```

Gambar 16 - Struktur Kode -KursusData.js-

```
1 const KursusData = [
12185 ...
12186 ...
12187 ...
12188 ...
12189 ...
12190 ...
12191 ...
12192 ...
12193 ...
12194 ...
12195 ...
12196 ...
12197 ...
12198 ...
13152 ...
13153 ...
13154 ...
13155 ...
13156 ...
13157 ...
13158 ...
13159 ...
13160 ...
13161 ...
14142 ...
14143 ...
14144 ...
14145 ...
14146 ...
14147 ...
14148 ...
14149 ...
14150 ...
14151 ...
15132 ...
15133 ...
15134 ...
15135 ...
15136 ...
15137 ... ];
```

Gambar 17 - Struktur Kode -KursusData.js-

- Setelah selesai, Selanjutnya kita menuju ContentDetailKursus.jsx lalu salin kode sesuai dengan garis kotak merah sesuai dengan **Gambar 18** lalu tempelkan kode sesuai dengan garis kotak merah sesuai dengan **Gambar 19** lalu sesuaikan dengan kebutuhan (*disini karena kita menambahkan Latihan 3, jadi sesuaikan dengan apa yang diubah pada Latihan 3*).

```

15
16 const ContentDetailKursus = ({ img, img2, img3 }) => {
17   const { id } = useParams();
18   const navigate = useNavigate();
19   const [pelajaran, setPelajaran] = useState(null);
20   const [tempIdCourse, setTempIdCourse] = useState([]);
21   const [tempExercise1, setTempExercise1] = useState([]);
22   const [tempExercise2, setTempExercise2] = useState([]); [tempExercise3, setTempExercise3] = useState([]);
23   const [tempExam1, setTempExam1] = useState(null);
24   const [tempExam2, setTempExam2] = useState(null);
25

```

Gambar 18 - Struktur Kode -ContentDetailKursus.jsx-

```

15
16 const ContentDetailKursus = ({ img, img2, img3 }) => {
17   const { id } = useParams();
18   const navigate = useNavigate();
19   const [pelajaran, setPelajaran] = useState(null);
20   const [tempIdCourse, setTempIdCourse] = useState([]);
21   const [tempExercise1, setTempExercise1] = useState([]);
22   const [tempExercise2, setTempExercise2] = useState([]); const [tempExercise3, setTempExercise3] = useState([]);
23   const [tempExam1, setTempExam1] = useState(null);
24   const [tempExam2, setTempExam2] = useState(null);
25

```

Gambar 19 - Struktur Kode -ContentDetailKursus.jsx-

- Setelah selesai, Kita tetap berada di ContentDetailKursus.jsx lalu salin kode sesuai dengan garis kotak merah sesuai dengan **Gambar 20** lalu tempelkan kode sesuai dengan garis kotak merah sesuai dengan **Gambar 21** lalu sesuaikan dengan kebutuhan (*disini karena kita menambahkan Latihan 3, jadi sesuaikan dengan apa yang diubah pada Latihan 3*).

```
82
83     const takeLatihanContent = () => {
84         KursusData.forEach((a) => {
85             a.detail.forEach((b) => {
86                 if (b.nama.toLowerCase().includes("1")) {
87                     tempExercise1.push(b);
88                 } else if (b.nama.toLowerCase().includes("2")) {
89                     tempExercise2.push(b);
90                 } else {
91                     console.log("Ops!!!");
92                 }
93             });
94         });
95     };
96 
```

Gambar 20 - Struktur Kode -ContentDetailKursus.jsx-

```
82
83     const takeLatihanContent = () => {
84         KursusData.forEach((a) => {
85             a.detail.forEach((b) => {
86                 if (b.nama.toLowerCase().includes("1")) {
87                     tempExercise1.push(b);
88                 } else if (b.nama.toLowerCase().includes("2")) {
89                     tempExercise2.push(b);
90                 } else if (b.nama.toLowerCase().includes("3")) {
91                     tempExercise3.push(b);
92                 } else {
93                     console.log("Ops!!!");
94                 }
95             });
96         });
97     };
98 
```

Gambar 21 - Struktur Kode -ContentDetailKursus.jsx-

4. Setelah selesai, Kita tetap berada di ContentDetailKursus.jsx lalu salin kode sesuai dengan digaris kotak merah sesuai dengan **Gambar 22** lalu tempelkan kode sesuai dengan digaris kotak merah sesuai dengan **Gambar 23** lalu sesuaikan dengan kebutuhan (*disini karena kita menambahkan Latihan 3, jadi sesuaikan dengan apa yang diubah pada Latihan 3*).

```
463     return (
464       <>
465         {(() => {
466           >     if (id.includes("lessons")) { ...
932           } else if (id.includes("exercise")) {
933             >       if (id.includes("one")) { ...
1004             >         } else if (id.includes("two")) { ...
1075             >           } else {
1076               >             return <></>;
1077             }
1078           } else if (id.includes("exam")) { ...
1908           } else {
1909             >             return <></>;
1910           }
1911         })()
1912       </>
1913     );
1914   };
1915 
```

Gambar 22 - Struktur Kode -ContentDetailKursus.jsx-

```
463     return (
464       <>
465         {(() => {
466           >     if (id.includes("lessons")) { ...
932           } else if (id.includes("exercise")) {
933             >       if (id.includes("one")) { ...
1004             >         } else if (id.includes("two")) { ...
1075             >           } else if (id.includes("three")) { ...
1146             >             } else {
1147               >                 return <></>;
1148             }
1149           } else if (id.includes("exam")) { ...
1979           } else {
1980             >             return <></>;
1981           }
1982         })()
1983       </>
1984     );
1985   };
1986 
```

Gambar 23 - Struktur Kode -ContentDetailKursus.jsx-

5. Hasil untuk **KursusData.js** seperti **Gambar 24**. Untuk **ContentDetailKursus.jsx** seperti **Gambar 25** (*Untuk Soal Jawaban masih menggunakan pacuan dari exercise-two*). Dan untuk **ContentDetailKursus2** seperti **Gambar 26**, **Gambar 27**, dan **Gambar 28** (*Untuk Soal Jawaban masih menggunakan pacuan dari exercise-two*).

The screenshot displays a web application interface for a course titled "Kursus Data". At the top, there is a navigation bar with links for "Beranda", "Kursus", and "Tentang", along with a user profile icon labeled "qaa qaa". Below the navigation bar, there are three green rounded rectangular boxes, each containing the text "Huruf Berharakat Fathah" and a small icon. The first box is associated with "Pelajaran 3" and contains a yellow square icon with a black Arabic character. The second box is associated with "Latihan 3" and contains a purple square icon with a black Arabic character.

Gambar 24 - Halaman hasil dari KursusData.js (Halaman Kursus)

The screenshot displays a web application interface for a course titled "Latihan 3" under the heading "Huruf Berharakat Dhammah". At the top, there is a navigation bar with links for "Beranda", "Kursus", and "Tentang", along with a user profile icon labeled "profile". Below the navigation bar, there are three icons with labels: "Gambar" (image), "Video" (video camera), and "Audio" (headphones). Each icon has a corresponding colored square icon above it.

Gambar 25 - Halaman hasil dari ContentDetailKursus.js (Halaman Detail Kursus)

Beranda Kursus Tentang qaa qaa

Latihan 3 - Gambar

Huruf Berharakat Dhammah

1. Yang manakah huruf "A"?

A B C D

2. Yang manakah huruf "Qa"?

A B C D

Gambar 26 - Halaman hasil dari ContentDetail2Kursus.js (Halaman dalam Detail Kursus (Gambar))

Beranda Kursus Tentang qaa qaa

Latihan 3 - Video

Huruf Berharakat Dhammah

1. Huruf apa yang diucapkan dari video berikut ini?

Gambar 27 - Halaman hasil dari ContentDetail2Kursus.js (Halaman dalam Detail Kursus (Video))

The screenshot shows a web page with a yellow header bar. On the left is a logo of a person with a tilde over their head. In the center are navigation links: 'Beranda' (Home), 'Kursus' (Courses), and 'Tentang' (About). On the right are user icons: 'qaa qaa' and a profile picture. Below the header, the title 'Latihan 3 - Audio' is displayed in green, followed by the subtitle 'Huruf Berharakat Dhammaha'. A question '1. Huruf apa yang terdengar?' is shown, accompanied by a large green speaker icon. Below the speaker are four options labeled A, B, C, and D, each with a small teal diagonal line icon.

Gambar 28 - Halaman hasil dari ContentDetail2Kursus.js (Halaman dalam Detail Kursus (Audio))

2.3.3 UJIAN

1. Langkah pertama kita menuju file KursusData.js lalu salin kode sesuai dengan garis kotak merah sesuai dengan **Gambar 29** lalu tempelkan kode pada array terakhir variable **KursusData** sesuai dengan garis kotak merah sesuai dengan **Gambar 30** lalu sesuaikan dengan kebutuhan (*disini karena kita menambahkan Ujian 3, jadi sesuaikan dengan apa yang perlu diubah pada Ujian 3*).

A screenshot of a code editor showing a portion of the KursusData.js file. The code is a JSON object with several properties. A red rectangular box highlights the 'soalJawaban2' property, which contains an array. The code is as follows:

```
9250 9251
9252 9253
9254 9255
9256 9257
9257 9258
9258 9259
9259 9260
9260 9261
9261 9262
9262 9263
9263 > 9264
12176 9265
12177 9266
12177 9267
12178 >
```

```
  id: "exam-two",
  nama: "Ujian 2",
  gambar: "img2-3.svg",
  deskripsi: "Huruf Berharakat Fathah",
  detail: [],
  soalJawaban2: [ ... ]
```

Gambar 29 – Struktur Kode -KursusData.js-

```

15136 },
15137 {
15138   id: "exam-three",
15139   nama: "Ujian 3",
15140   gambar: "img2-3.svg",
15141   deskripsi: "Huruf Berharakat Dhammah",
15142   detail: [],
15143   soalJawaban2: [ ... ],
15144   ],
15145 },
18056 ],
18057 ];
18058
18059

```

Gambar 30 – Struktur Kode -KursusData.js-

- Setelah selesai, Selanjutnya kita menuju ContentDetailKursus.jsx lalu salin kode sesuai dengan digaris kotak merah sesuai dengan **Gambar 31** lalu tempelkan kode sesuai dengan digaris kotak merah sesuai dengan **Gambar 32** lalu sesuaikan dengan kebutuhan (*disini karena kita menambahkan Ujian 3, jadi sesuaikan dengan apa yang diubah pada Ujian 3*).

```

16 const ContentDetailKursus = ({ img, img2, img3 }) => {
17   const { id } = useParams();
18   const navigate = useNavigate();
19   const [pelajaran, setPelajaran] = useState(null);
20   const [tempIdCourse, setTempIdCourse] = useState([]);
21   const [tempExercise1, setTempExercise1] = useState([]);
22   const [tempExercise2, setTempExercise2] = useState([]);
23   const [tempExercise3, setTempExercise3] = useState([]);
24   const [tempExam1, setTempExam1] = useState(null);
25   const [tempExam2, setTempExam2] = useState(null);
26
27   //USESTATE UJIAN
28   const [score, setscore] = useState(
29     parseInt(localStorage.getItem("score")) || 0
30   );

```

Gambar 31 - Struktur Kode -ContentDetailKursus.jsx-

```

15
16 const ContentDetailKursus = ({ img, img2, img3 }) => {
17   const { id } = useParams();
18   const navigate = useNavigate();
19   const [pelajaran, setPelajaran] = useState(null);
20   const [tempIdCourse, setTempIdCourse] = useState([]);
21   const [tempExercise1, setTempExercise1] = useState([]);
22   const [tempExercise2, setTempExercise2] = useState([]);
23   const [tempExercise3, setTempExercise3] = useState([]);
24   const [tempExam1, setTempExam1] = useState(null);
25   const [tempExam2, setTempExam2] = useState(null);
26   const [tempExam3, setTempExam3] = useState(null);
27
28   //USESTATE UJIAN
29   const [score, setScore] = useState(
30     parseInt(localStorage.getItem("score")) || 0
31   );

```

Gambar 32 - Struktur Kode -ContentDetailKursus.jsx-

- Setelah selesai, Selanjutnya kita menuju ContentDetailKursus.jsx lalu salin kode sesuai dengan garis kotak merah sesuai dengan **Gambar 33** lalu tempelkan kode sesuai dengan garis kotak merah sesuai dengan **Gambar 34** lalu sesuaikan dengan kebutuhan (*disini karena kita menambahkan Ujian 3, jadi sesuaikan dengan apa yang diubah pada Ujian 3*).

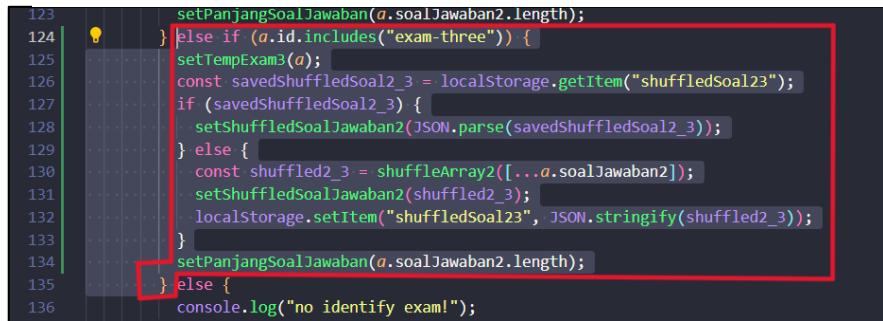


```

1.11
112     localStorage.setItem("shuffledSoal2_1", JSON.stringify(shuffledSoal2_1));
113 } else if (a.id.includes("exam-two")) {
114     setTempExam2(a);
115     const savedShuffledSoal2_2 = localStorage.getItem("shuffledSoal2");
116     if (savedShuffledSoal2_2) {
117         setShuffledSoalJawaban2(JSON.parse(savedShuffledSoal2_2));
118     } else {
119         const shuffled2_2 = shuffleArray2([...a.soalJawaban2]);
120         setShuffledSoalJawaban2(shuffled2_2);
121         localStorage.setItem("shuffledSoal2", JSON.stringify(shuffled2_2));
122     }
123     setPanjangSoalJawaban(a.soalJawaban2.length);
124 } else {
125     console.log("no identify exam!");
126 }
127

```

Gambar 33 - Struktur Kode -ContentDetailKursus.jsx-



```

123     setPanjangSoalJawaban(a.soalJawaban2.length);
124 } else if (a.id.includes("exam-three")) {
125     setTempExam3(a);
126     const savedShuffledSoal2_3 = localStorage.getItem("shuffledSoal2");
127     if (savedShuffledSoal2_3) {
128         setShuffledSoalJawaban2(JSON.parse(savedShuffledSoal2_3));
129     } else {
130         const shuffled2_3 = shuffleArray2([...a.soalJawaban2]);
131         setShuffledSoalJawaban2(shuffled2_3);
132         localStorage.setItem("shuffledSoal2", JSON.stringify(shuffled2_3));
133     }
134     setPanjangSoalJawaban(a.soalJawaban2.length);
135 } else {
136     console.log("no identify exam!");

```

Gambar 34 - Struktur Kode -ContentDetailKursus.jsx-

- Setelah selesai, Selanjutnya kita menuju ContentDetailKursus.jsx lalu salin kode sesuai dengan garis kotak merah sesuai dengan **Gambar 35** lalu tempelkan kode sesuai dengan garis kotak merah sesuai dengan **Gambar 36** lalu sesuaikan dengan kebutuhan (*disini karena kita menambahkan Ujian 3, jadi sesuaikan dengan apa yang diubah pada Ujian 3*).

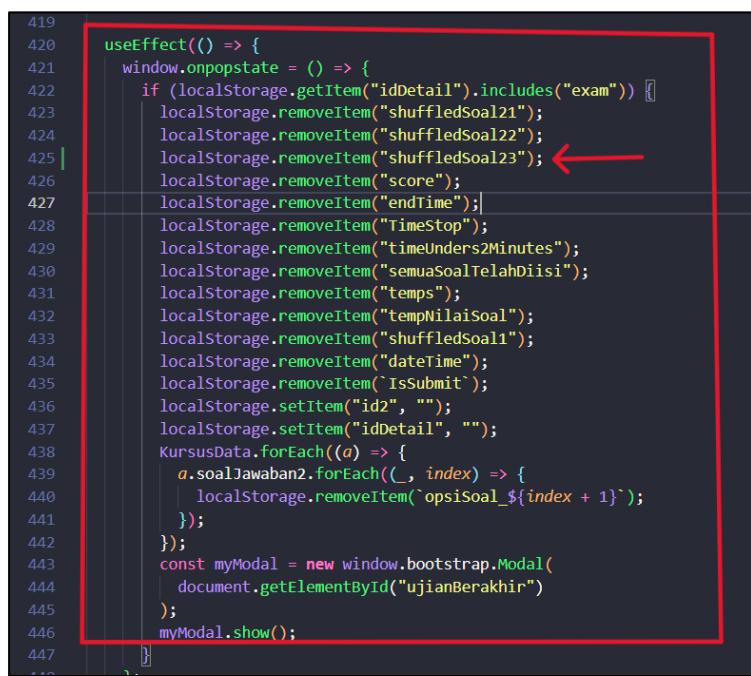
```
400
467     {(() => {
468 >       if (id.includes("lessons")) { ...
934 >       } else if (id.includes("exercise")) { ...
1151 >       } else if (id.includes("exam")) {
1152 >         if (id.includes("one")) { ...
1429 >           } else if (id.includes("two")) { ...
1705 >             } else {
1706             return <></>;
1707           }
1708 >         } else { ...
1710           }
1711         })()
1712       </>
1713     );
1714   };
1715
```

Gambar 35 - Struktur Kode -ContentDetailKursus.jsx-

```
465   return (
466     <>
467     {(() => {
468 >       if (id.includes("lessons")) { ...
934 >       } else if (id.includes("exercise")) { ...
1151 >       } else if (id.includes("exam")) {
1152 >         if (id.includes("one")) { ...
1429 >           } else if (id.includes("two")) { ...
1705 >             } else if (id.includes("three")) { ...
1981 >               } else {
1982               return <></>;
1983             }
1984 >           } else { ...
1986             }
1987           })()
1988         </>
1989       );
1990     };
1991
```

Gambar 36 - Struktur Kode -ContentDetailKursus.jsx-

5. Setelah selesai, Jangan lupa pada *useEffect* seperti pada **Gambar 37** dan setiap percabangan exam menambahkan kode “*localStorage.removeItem("shuffledSoal23");*” pada bagian yang digaris kotak seperti pada **Gambar 38**, **Gambar 39** dan **Gambar 40**. Sedikit penjelasan, Pada kata bertuliskan *savedShuffledSoal2_3*, *shuffledSoal23*, dan *shuffled2_3* yang dimana angka 2 mendefinisikan soal acak bertipe ujian (angka 1 mendefinisikan soal acak bertipe latihan). Sedangkan angka 3 mendefinisikan soal acak ujian ke berapa.



```

419
420
421
422
423
424
425 | useEffect(() => {
426   window.onpopstate = () => {
427     if (localStorage.getItem("idDetail").includes("exam")) {
428       localStorage.removeItem("shuffledSoal21");
429       localStorage.removeItem("shuffledSoal22");
430       localStorage.removeItem("shuffledSoal23"); ←
431       localStorage.removeItem("score");
432       localStorage.removeItem("endTime");
433       localStorage.removeItem("timeStop");
434       localStorage.removeItem("timeUnder2Minutes");
435       localStorage.removeItem("semuaSoalTelahDiisi");
436       localStorage.removeItem("temp");
437       localStorage.removeItem("tempNilaiSoal");
438       localStorage.removeItem("shuffledSoal1");
439       localStorage.removeItem("dateTime");
440       localStorage.removeItem("isSubmit");
441       localStorage.setItem("id2", "");
442       localStorage.setItem("idDetail", "");
443       KursusData.forEach((a) => {
444         a.soalJawaban2.forEach((_, index) => {
445           localStorage.removeItem(`opsiSoal_${index + 1}`);
446         });
447       });
448       constmyModal = new window.bootstrap.Modal(
449         document.getElementById("ujianBerakhir")
450       );
451      myModal.show();
452     }
453   }
454 }

```

*Gambar 37 - Struktur Kode -ContentDetailKursus.jsx
(useEffect)-*

```
1375
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397 | <> <button onClick={() => {
1398   navigate(`/kursus`, { replace: true });
1399   tempExam1.soalJawaban2.forEach((_, index) => {
1400     localStorage.removeItem(
1401       `opsiSoal_${index + 1}`
1402     );
1403   });
1404   localStorage.removeItem(`tempNilaiSoal`);
1405   localStorage.removeItem(`shuffledSoal21`);
1406   localStorage.removeItem(`shuffledSoal22`);
1407   localStorage.removeItem(`shuffledSoal23`); ←
1408   localStorage.removeItem(`dateTime`);
1409   localStorage.setItem("id2", "");
1410   localStorage.setItem("idDetail", "");
1411   window.location.reload();
1412 }
1413 >
1414 <div className="tw-bg-[#009900] tw-hover:tw-bg-[#007100] tw-w-[45%] xs:tw-w-[
```

Gambar 38 - Struktur Kode -ContentDetailKursus.jsx- (exam-one)

```
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673 | <> <button onClick={() => {
1674   navigate(`/kursus`, { replace: true });
1675   tempExam1.soalJawaban2.forEach((_, index) => {
1676     localStorage.removeItem(
1677       `opsiSoal_${index + 1}`
1678     );
1679   });
1680   localStorage.removeItem(`tempNilaiSoal`);
1681   localStorage.removeItem(`shuffledSoal21`);
1682   localStorage.removeItem(`shuffledSoal22`);
1683   localStorage.removeItem(`shuffledSoal23`); ←
1684   localStorage.removeItem(`dateTime`);
1685   localStorage.setItem("id2", "");
1686   localStorage.setItem("idDetail", "");
1687   window.location.reload();
1688 }
1689 >
1700 <div className="tw-bg-[#009900] tw-hover:tw-bg-[#007100] tw-w-[45%] xs:
```

Gambar 39 - Struktur Kode -ContentDetailKursus.jsx- (exam-two)

```

1928   <>
1929   {JSON.parse(localStorage.getItem(`IsSubmit`)) ===
1930     true ? (
1931       <button
1932         onClick={() => {
1933           navigate(`/kursus`, { replace: true });
1934           tempExam1.soalJawaban2.forEach(_, index) => {
1935             localStorage.removeItem(
1936               `opsisoal_${index + 1}`
1937             );
1938           });
1939           localStorage.removeItem(`temp`);
1940           localStorage.removeItem(`semuaSoalTelahDiisi`);
1941           localStorage.removeItem(`IsSubmit`);
1942           localStorage.removeItem(`endTime`);
1943           localStorage.removeItem(`timeUnder2Minutes`);
1944           localStorage.removeItem(`TimeStop`);
1945           localStorage.removeItem(`score`);
1946           localStorage.removeItem(`tempNilaiSoal`);
1947           localStorage.removeItem(`shuffledSoal21`);
1948           localStorage.removeItem(`shuffledSoal22`);
1949           localStorage.removeItem(`shuffledSoal23`); ←
1950           localStorage.removeItem(`dateTime`);
1951           localStorage.setItem(`id2`, ``);
1952           localStorage.setItem(`idDetail`, ``);
1953           window.location.reload();
1954         }
1955         className="tw-bg-[#009900] tw-hover:bg-[#007100] tw-w-
1956       >
1957         OK
1958       </button>
1959     ) : (
1960       <button

```

Gambar 40 - Struktur Kode -ContentDetailKursus.jsx- (exam-three)

6. Jangan lupa juga untuk kebaris yang digaris kotak seperti pada **Gambar 41** dan tambahkan kode sesuai yang digaris kotak seperti pada **Gambar 42**.

```

313   }, [ta]);
314
315   useEffect(() => {
316     takeLatihanContent();
317   }, [id, tempExercise1, tempExercise2]);
318
319 >   useEffect(() => {
320     ...
321   }, [id, tempExam1, tempExam2]);
322

```

Gambar 41 – Struktur Kode -ContentDetailKursus.jsx-

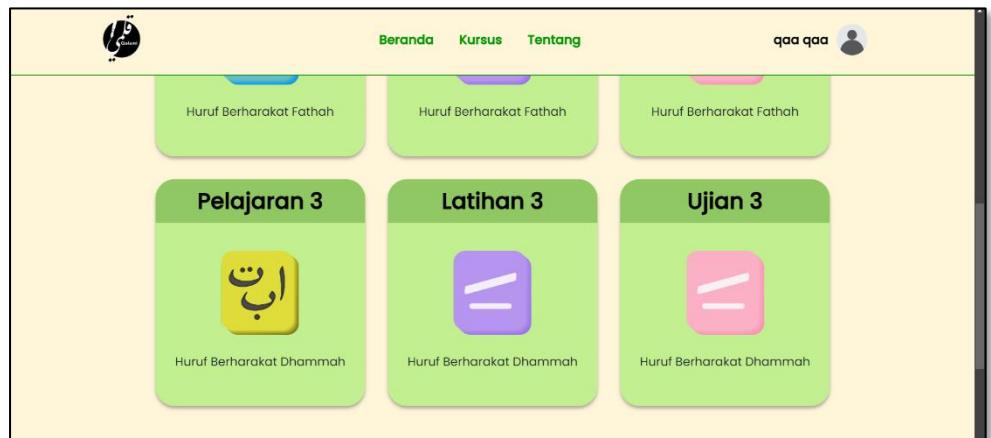
```

314
315
316
317 >   useEffect(() => {
318     takeLatihanContent();
319 }, [id, tempExercise1, tempExercise2, tempExercise3]);
320 >   useEffect(() => {
321     [id, tempExam1, tempExam2, tempExam3]);
322 }, [time]);
323 >   useEffect(() => {
324     [id, tempExam1, tempExam2, tempExam3]);
325 }, [time]);

```

Gambar 42 – Struktur Kode -ContentDetailKursus.jsx-

7. Hasil untuk **KursusData.js** seperti **Gambar 43**. Untuk **ContentDetailKursus.jsx** seperti **Gambar 44** (*Untuk Soal Jawaban masih menggunakan pacuan dari exercise-two*).



Gambar 43 - Halaman hasil dari KursusData.js (Halaman Kursus)

Gambar 44 - Halaman hasil dari ContentDetailKursus.js (Halaman Ujian)

BAB 3

PENUTUP

3.1 KESIMPULAN

Alhamdulillah, dengan rahmat dan berkah dari Allah SWT, proyek akhir kami berhasil diselesaikan. *Manual Book Pengembangan (Front-End)* ini telah dirancang untuk memberikan panduan yang jelas dan praktis bagi pengembang selanjutnya dalam mengembangkan aplikasi *Qolami Web*. Dengan adanya panduan ini, diharapkan tim pengembang berikutnya dapat memahami dan melanjutkan pengembangan aplikasi dengan lebih efektif dan efisien.

Aplikasi Qolami ini, yang sebelumnya hanya tersedia pada platform mobile, kini telah diperluas ke platform desktop, sehingga dapat diakses lebih luas oleh pengguna. Kami berharap aplikasi ini dapat memberikan manfaat yang besar bagi siapa saja yang ingin belajar Bahasa Arab dengan mudah dan praktis.

3.2 SARAN

Kami menyarankan agar tim pengembang selanjutnya terus melakukan inovasi dan pengembangan fitur-fitur aplikasi Qolami Web, seperti Latihan menulis arab menggunakan Pen Pad/Mouse/Mousepad dan hasil tulis bisa diperiksa secara otomatis dengan menerapkan AI atau Machine Learning.

Atau juga sang pengembang bisa membuat 2 role, yaitu Guru dan Murid, yang dimana hasil tulis dari Role Murid hanya bisa diperiksa oleh user yang memiliki Role Guru. Jika diterapkannya 2 Role yaitu Guru dan Murid, Untuk menjadi Guru, si user harus melalui ujian bertahap agar Role si user menjadi Guru.

Sempat disinggung juga, Sepertinya diperlukan diskusi dengan Back-End, karena pada Back-End Web ini hanya dibuatkan *lessons-one* dan *lessons-two*, sehingga *lessons-three* akan menghasilkan error dan tidak memunculkan content apapun, sehingga perlu dibuat oleh Back-End. Terima kasih kepada semua pihak yang telah berkontribusi dalam proyek ini. Semoga Allah SWT selalu memberikan keberkahan dan kemudahan dalam setiap langkah kita.