

# Pembuatan Sistem Aplikasi Berbasis Website Konsultasi Orang Tua dengan Psikolog untuk Kesehatan Mental Anak

Ilman Nawali<sup>#1</sup>, Bernard Renaldy Suteja<sup>#2</sup>

*#Program Studi SI Teknik Informatika, Fakultas Teknologi Informasi, Universitas Kristen Maranatha  
Jln. Prof. Drg. Surya Sumantri No. 65, Sukawarna, Bandung, Indonesia*

<sup>1</sup>1972003@maranatha.ac.id

<sup>2</sup>bernard.rs@it.maranatha.edu

**Abstract** — Treason for writing a report entitled "DEVELOPMENT OF PARENTAL CONSULTATION WEBSITE-BASED APPLICATION SYSTEM WITH PSYCHOLOGIST FOR CHILDREN'S MENTAL HEALTH." Because currently is participating in the Certified MBKM program created by the government so that later can give birth to or produce competent human resources in their fields. And on this occasion the MBKM program that was participated in was Front-End and Back-End Developers at Dicoding. When participating in this program, will be taught how to create a website that suits industry needs. As learning materials while participating in this program there are several materials in outline to learn about HTML, CSS, Javascript, Node Js, and Hapi, and in each of these materials there is a final assignment to proceed to the next material and if the final assignment does not meet the requirements then the final assignment will be submitted. sent will be rejected and vice versa. And also not only learning hard skills, they are also given learning about soft skills such as evaluating the way oneself manages self-productivity, designing time management, applying moral principles that are in accordance with applicable standards of behavior and regulations, and knowledge about what to prepare when will be interviewed. And with the MBKM Independent Study of Front-End and Back-End Developers it is very helpful to be able to produce websites that are good and correct and in accordance with industry needs. Later at the end of the program, will work on final project or project capstone, namely creating a website for parents so they can discuss with a psychologist about children's mental health

**Keywords**— MBKM, Front-End, Back-End.

## I. PENDAHULUAN

Saat ini kemajuan dalam berbagai bidang berkembang pesat, salah satunya bidang digital perkembangan website. Dengan pesatnya bidang digital perkembangan website maka dari itu Sumber Daya Manusia harus dituntut untuk berkembang dan berkompeten. MBKM hadir untuk membantu menghasilkan Sumber Daya Manusia yang berkompeten nantinya.

MBKM merupakan bagian dari kebijakan merdeka belajar oleh kementerian pendidikan kebudayaan, riset, dan teknologi republik indonesia yang memberikan kesempatan kepada seluruh mahasiswa/i diseluruh indonesia untuk mengasah kemampuan sesuai minat dan bakat dengan terjun langsung ke dalam dunia kerja sebagai persiapan untuk masa depan yang akan datang. banyak sekali program kampus merdeka atau MBKM yang dapat diikuti oleh mahasiswa guna mengasah skill untuk mempersiapkan karier di masa depan, salah satunya adalah Studi Independen Bersertifikat, Studi Independen Bersertifikat adalah bagian dari program Kampus Merdeka yang bertujuan untuk memberikan kesempatan kepada mahasiswa untuk belajar dan mengembangkan diri melalui aktivitas di luar kelas perkuliahan, namun tetap diakui sebagai bagian dari perkuliahan.

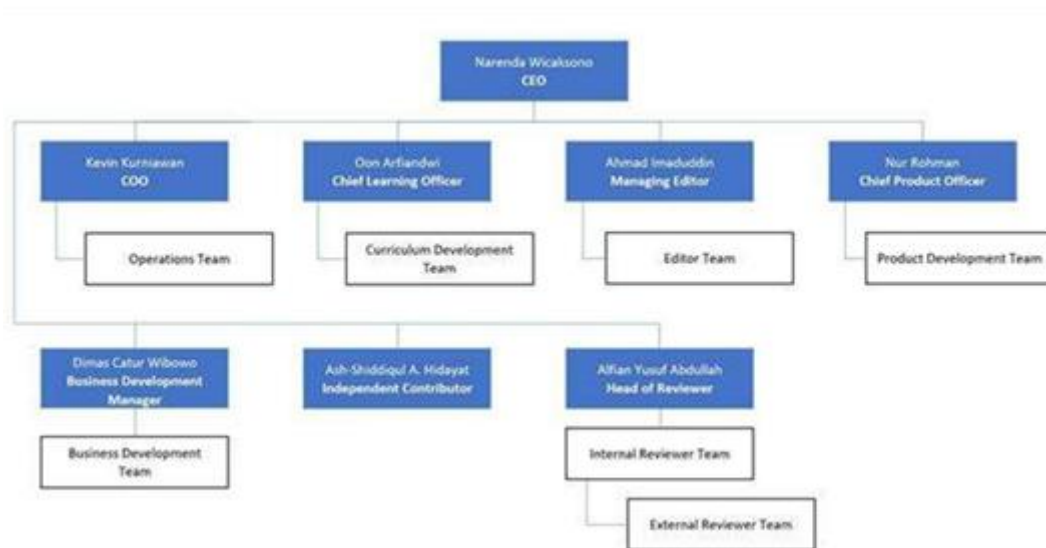
Pada kegiatan Studi Independen tersebut mahasiswa harus mengikuti kegiatan yang diberi oleh mitra, Salah satunya adalah Dicoding Indonesia, Dicoding dikenal sebagai tempat platform edukasi teknologi terdepan yang mendorong akses literasi digital yang lebih luas untuk semua yang memiliki beberapa program pembelajaran antara lain Pengembang Aplikasi Android, Pengembang Front-End dan React, Pengembangan Machine Learning dan Front-End Web, Pengembang Front-End dan Back-End, dan masih ada lagi yang lainnya. Program ini dilaksanakan secara online dengan cara diberikan akses token belajar untuk nantinya belajar melalui website Dicoding, dan di setiap modul pembelajaran nantinya akan disediakan tugas akhir untuk melanjutkan ke modul selanjutnya.

Pada saat mengikuti kegiatan Studi Independen ini di Dicoding, program pembelajaran yang diambil adalah Pengembang Front-End dan Back-End. karena pada saat ini kegunaan website tidak hanya menjadi media informasi saja melainkan bisa menjadi tempat jual beli online, booking tiket, dan lain sebagainya. Website juga bisa menjadikan peluang kerja yang luas nantinya.

## II. PROFIL PERUSAHAAN

Dicoding merupakan sebuah platform bagi developer Indonesia untuk belajar, membuat karya digital, meraih validasi industri atas karya tersebut, serta mencari pekerjaan di perusahaan mitra. Dicoding menawarkan developer journey yang utuh, dimulai dari Academy.

Visi Dicoding adalah menjadi platform edukasi teknologi terdepan yang mendorong akses literasi digital yang lebih luas untuk semua. Dicoding memiliki misi untuk mengakselerasi transisi Indonesia menuju dunia digital melalui pendidikan teknologi yang mentransformasi kehidupan.



Dicoding juga mempunyai struktur organisasi perusahaan yaitu beranggotakan seperti berikut ini:

- Narendra Wicaksono sebagai CEO Dicoding Lulusan Teknik Informatika ITB. Mendirikan dan menjadi CEO Dicoding sejak 2015. Sebelumnya bekerja di Microsoft dan Nokia. Mengembangkan developer serta jejaringnya di Indonesia hingga Asia Tenggara.
- Kevin Kurniawan sebagai COO Bergabung sejak awal Dicoding berdiri tahun 2015. Pernah bekerja di Nokia untuk mengembangkan developer di Indonesia. Kini fokus memberikan experience terbaik pada user dan partner Dicoding.
- Oon Arfiandi sebagai Chief Learning Officer Lulusan Teknik Informatika ITB. Berpengalaman dalam membangun berbagai sistem dengan perhatian khusus di edukasi. Saat ini bertanggung jawab mengembangkan pengalaman belajar untuk siswa di Dicoding.
- Ahmad Imaduddin sebagai Managing Editor Memulai karir sebagai Curriculum Developer di tahun 2016. Setelah berhasil mengembangkan banyak kelas berkualitas di Academy, saat ini bertanggung jawab menyunting materi kelas agar tetap terjaga kualitasnya.
- Nur Rohman sebagai Chief Product Officer Meraih gelar LINE API Expert. Pernah bekerja di Gojek sebagai Product Engineer. Setelah berhasil membangun tim Reviewer, kini fokus untuk memastikan dicoding.com bisa menyelesaikan masalah para pengguna.
- Dimas Catur Wibowo sebagai Business Development Manager Lulusan Manajemen Bisnis Telekomunikasi & Informatika Telkom University. Saat ini bertanggung jawab dalam mengembangkan strategi dan menciptakan peluang bisnis untuk perusahaan.
- Ash-Shiddiqul A. Hidayat sebagai Independent Contributor Melakukan commit code pertama untuk dicoding.com. Fokus membangun fondasi dan platform yang stabil, tangguh dan aman. Tujuannya memudahkan engineer dalam memberi dampak yang maksimal.

● Alfian Yusuf Abdullah sebagai Head of Reviewer Peraih gelar Google Certified Associate Android Developer sejak 2019. Bergabung dengan Dicoding untuk membantu developer Indonesia menuliskan kode dengan baik lewat Code Review.

### III. LANDASAN TEORI

#### A. HTML

Hypertext Markup Language atau HTML adalah bahasa markup standar yang digunakan untuk membuat halaman website dan aplikasi web. penggunaan umum HTML adalah untuk menyusun bagian paragraf, heading, maupun link pada halaman web. Tapi, meskipun susunannya seperti coding, perlu diketahui bahwa HTML bukanlah bahasa pemrograman. HTML tidak dianggap sebagai bahasa pemrograman karena tidak bisa memberikan fungsi yang dinamis. Sekarang, bahasa markup ini dianggap sebagai standar web resmi, dikelola oleh World Wide Web Consortium (W3C), yang juga bertugas merilis pembaruan rutin HTML.

#### B. CSS

Cascading Style Sheet atau biasa disingkat CSS merupakan W3C standar yang digunakan untuk mengatur visualisasi berkas yang ditulis pada HTML. CSS syntax berbeda dengan HTML, begitu pula dengan JavaScript. CSS bukanlah sebuah bahasa pemrograman karena di dalamnya tidak terdapat logika, tidak dapat sintaks pengondisian, tidak adanya proses iterasi, dsb. CSS hanya sebuah declarative language yang digunakan untuk mendeklarasikan suatu nilai yang nantinya digunakan untuk mengatur seperti apa sebuah elemen HTML ditampilkan pada browser.

#### C. JavaScript

JavaScript adalah bahasa pemrograman tingkat tinggi yang pada awalnya dikembangkan untuk membuat website menjadi lebih “hidup”. Bersama dengan HTML dan CSS, JavaScript menjadi bahasa pemrograman paling populer untuk mengembangkan aplikasi berbasis web. Bahasa ini mampu memberikan logic ke dalam website, sehingga website tersebut memiliki fungsionalitas tambahan dan lebih interaktif. JavaScript termasuk ke dalam kategori scripting language. Apa maksudnya? Salah satu ciri-ciri utama dari bahasa scripting adalah kode tidak perlu dikompilasi agar bisa dijalankan. Scripting language menggunakan interpreter untuk menerjemahkan kode atau perintah yang ditulis supaya dimengerti oleh mesin. Itulah kenapa bahasa scripting tidak membutuhkan banyak kode yang perlu ditulis agar sebuah program bisa dijalankan.

#### D. Web Storage

Web storage adalah salah satu Web API yang dapat menyimpan data secara lokal pada sisi client. Berbeda dengan objek atau array, data yang disimpan pada objek atau array JavaScript bersifat sementara, dan akan hilang jika terjadi reload atau pergantian URL pada browser. Sedangkan data yang disimpan pada Web Storage akan bertahan lebih lama karena data akan disimpan pada storage browser. Sebelum HTML5, untuk melakukan hal ini dengan menggunakan cookies, namun penyimpanan cookies terbatas. Dengan hadirnya Web Storage dapat menampung data lebih besar dan tentunya lebih aman.

Data yang disimpan web storage tersedia berdasarkan domain. Artinya, data pada suatu domain web hanya dapat diakses oleh domainnya itu sendiri. Web storage dapat menampung sebesar 10MB untuk satu domain.

#### E. PWA (Progressive Web Apps)

Istilah Progressive Web Apps pertama diperkenalkan oleh Alex Russel, salah seorang Google Chrome Engineer, dan Frances Berriman, designer yang bekerja untuk Google pada tahun 2015. Google mendefinisikan PWA sebagai berikut:

“Progressive Web Apps use modern web capabilities to deliver an app-like user experience. They evolve from pages in browser tabs to immersive, top-level apps, maintaining the web's low friction at every moment.”

Tujuan hadirnya PWA adalah memungkinkan web developer mengubah web yang sudah ada agar bisa berperilaku layaknya aplikasi mobile native tanpa banyak perubahan atau menambah programmer khusus. Tujuan ini bisa dicapai berkat kumpulan teknologi web. Ia bisa memanfaatkan kelebihan-kelebihan web dan aplikasi mobile native sekaligus. Kumpulan teknologi ini memungkinkan aplikasi web tradisional mampu diakses tanpa koneksi internet (offline), bisa dipasang di homescreen seperti aplikasi native, bisa melakukan sinkronisasi data ke server, bisa mengirim push notification dan lain-lain.

#### F. Node JS

Node.js adalah platform buatan Ryan Dahl untuk menjalankan aplikasi web berbasis JavaScript yang dikenalkan pada tahun 2009. Dengan platform ini, Anda dapat menjalankan JavaScript dari sisi server. Untuk mendukung kemampuan tersebut, Node.js dibangun dengan engine Javascript V8 milik Google. Di samping itu, Node.js juga memiliki pustaka server sendiri sehingga Anda tidak perlu menggunakan program server web seperti Nginx dan Apache.

### G. Hapi

Hapi.js atau disingkat Hapi merupakan Node.js framework yang digunakan untuk membangun web server yang kompleks karena Hapi cocok untuk itu sesuai dengan tagline Hapi “Build powerful, scalable applications, with minimal overhead and full out-of-the-box functionality”. Hapi sangat cocok untuk membangun web server yang kompleks. Hapi awalnya dikembangkan oleh e-commerce Walmart meskipun demikian Hapi juga digunakan oleh banyak perusahaan lainnya.

### H. Software Development

Software development adalah proses di mana perangkat lunak mandiri atau terpisah dibuat menggunakan bahasa pemrograman tertentu. Ini melibatkan penulisan serangkaian kode pemrograman yang saling terkait, yang menyediakan fungsionalitas perangkat lunak yang dikembangkan.

Software Management Project adalah perangkat lunak yang dibuat untuk mengatur manajemen tugas, waktu pengerjaan, dan kerjasama dalam sebuah tim.

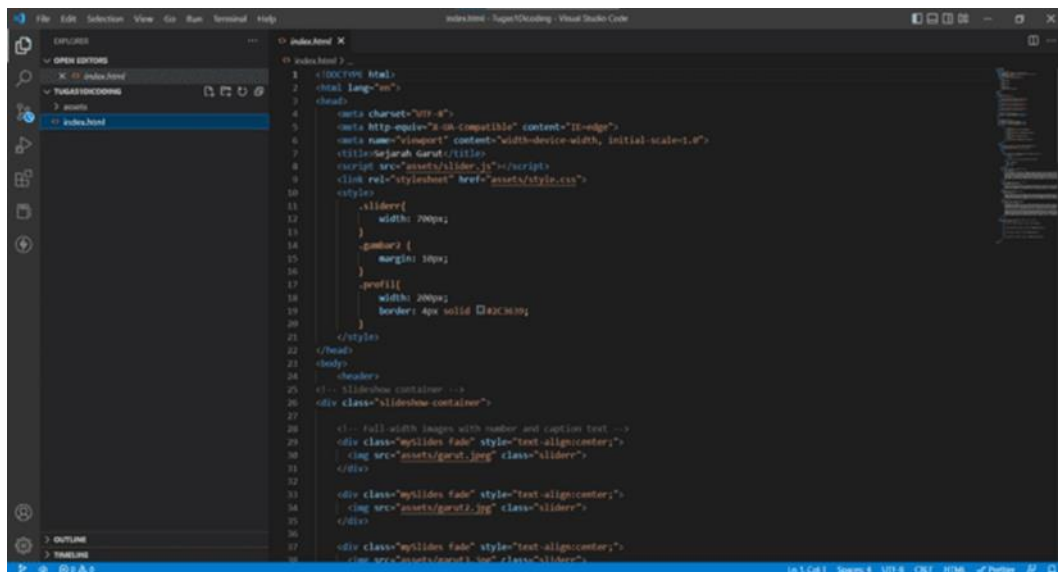
- 1) *Front-End Developer: Bagian frontend dari sebuah website adalah bagian yang langsung dilihat oleh user. User juga bisa langsung berinteraksi pada bagian ini. Bagian ini dibangun menggunakan HTML, CSS, JavaScript. HTML (HyperText Markup Language) adalah fondasi dari web.*
- 2) *Back-End Developer: Backend adalah bagian belakang layar dari sebuah website. Bahasa pemrograman untuk backend development diantaranya adalah PHP, Ruby, Python, dan banyak lainnya. Satu hal yang akan kita temui: tidak akan banyak perusahaan yang mem-posting lowongan kerja dengan kalimat “backend developer”. Yang akan lebih sering kita temui adalah lowongan pekerjaan untuk “Ruby developer”, “PHP developer”, dan lain sebagainya. Ada banyak hal yang tidak bisa dilakukan oleh JavaScript, namun dapat dilakukan oleh bahasa pemrograman backend.*

## IV. HASIL PEKERJAAN

### A. Membuat Website HTML, CSS dan Javascript.

Pada submission ini peserta diberikan tugas yaitu membuat website yang memiliki kriteria seperti:

- Dasar HTML.
- Penerapan elemen semantic.
- Menerapkan styling menggunakan CSS.
- Layouting menggunakan float ataupun flexbox.



Gambar 1. Potongan Code Program.

Gambar 1 adalah contoh potongan code program untuk membuat tampilan website, terdapat tag <script> dan <style> itu merupakan tag untuk mengimport atau memasukan file css dan javascriptnya yang dibuat.



Gambar 2. Tampilan Website 1



Gambar 3. Tampilan Website 2

Gambar 2 dan Gambar 3 adalah contoh tampilan jika file HTML di buka. Terdapat suatu gambar daerah di Garut dan juga ada latar belakang Kota Garut dan juga ada Profil pembuat.

B. Membuat Website Pengelolaan Data Menggunakan DOM dan Web Storage.

Pada submission ini peserta harus mengerjakan tugas yang memiliki kriteria seperti berikut ini:

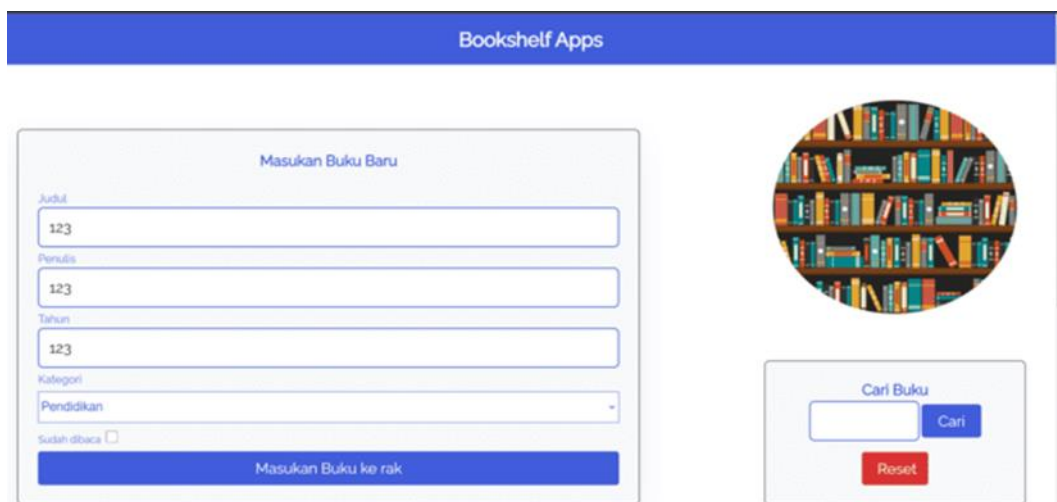
- Dasar-dasar JavaScript mulai dari statement, expressions, pembuatan variabel, tipe data, operator, percabangan if dan switch, data structure dengan array dan object, hingga pembuatan function.
- Mengetahui Browser Object Model (BOM), Document Object Model (DOM), dan Event di dalam website. Serta Anda mampu mengubah konten website melalui teknik manipulasi DOM dan memberikan Event.
- Mengenal localStorage dan sessionStorage sebagai Web Storage. Serta menggunakan keduanya.

```

11 <body>
12 <header>
13 <h1>Bookshelf Apps</h1>
14 </header>
15 <div class="wrapper">
16 <table>
17 <tr>
18 <td>
19 <div
20 class="container bg white shadow"
21 id="add-book">
22 <div class="text-center">Masukan Buku Baru</div>
23 <form class="form action="s" id="form" style="width: auto">
24 <div class="form-group form-title">
25 <label for="title">Judul</label>
26 <input type="text" id="title" name="title" required />
27 </div>
28 <div class="form-group form-title">
29 <label for="name">Penulis</label>
30 <input type="text" id="name" name="name" required />
31 </div>
32 <div class="form-group form-title">
33 <label for="year">Tahun</label>
34 <input type="text" id="year" name="year" required />
35 </div>
36 <div class="form-title form-group">
37 <label for="category">Kategori</label>
38 <select
39 id="category"
40 name="category"
41 style="
42 height: 50px;
43 font-size: 20px;
44 background-color: # white;
45 margin-top: 5px;
46 border: 1px solid #ccc;">
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
    
```

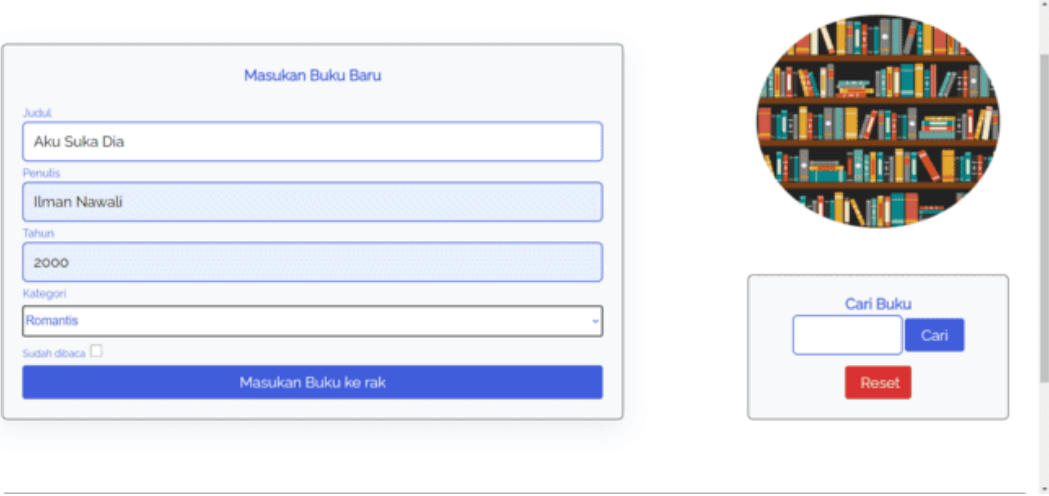
Gambar 4. Potongan Code Program HTML

Gambar 4 adalah merupakan contoh potongan code yang sudah dibuat untuk tampilan yang nanti akan ditampilkan didalam website.



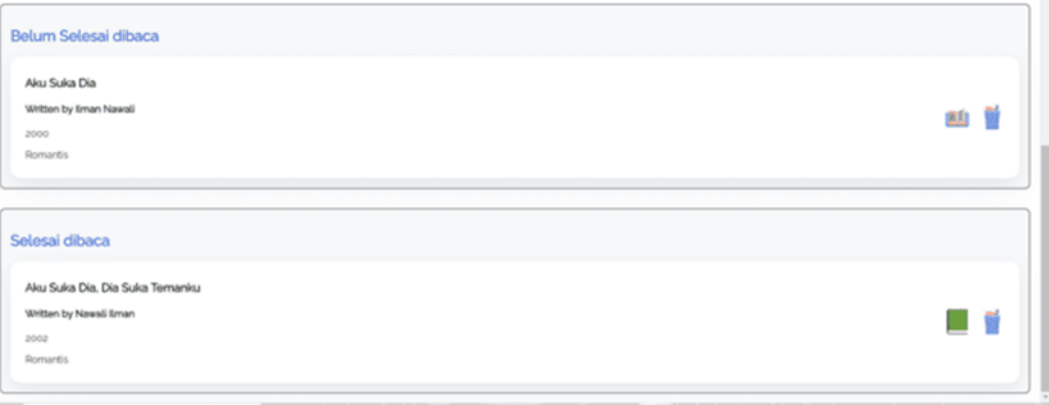
Gambar 5. Tampilan Awal Website

Pada gambar 5 adalah contoh tampilan awal website yang sudah dibuat dengan code HTML dan CSS.



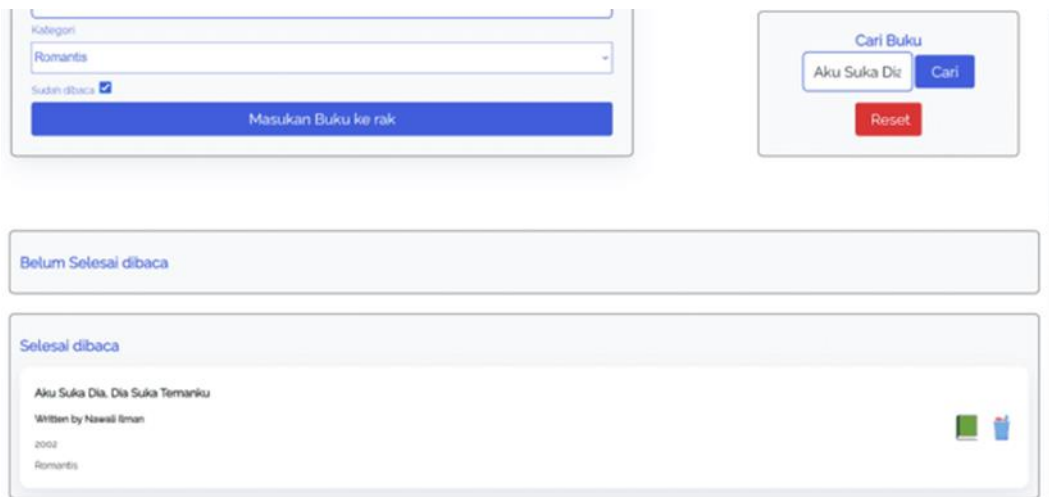
Gambar 6. Tampilan Mengisi data

Gambar 6 adalah contoh tampilan jika mengisi data judul buku, penulis buku, tahun terbit buku, katategori buku, dan buku itu sudah dibaca atau belum dan pada gambar 6 memberikan contoh jika memasukan buku yang belum pernah dibaca.



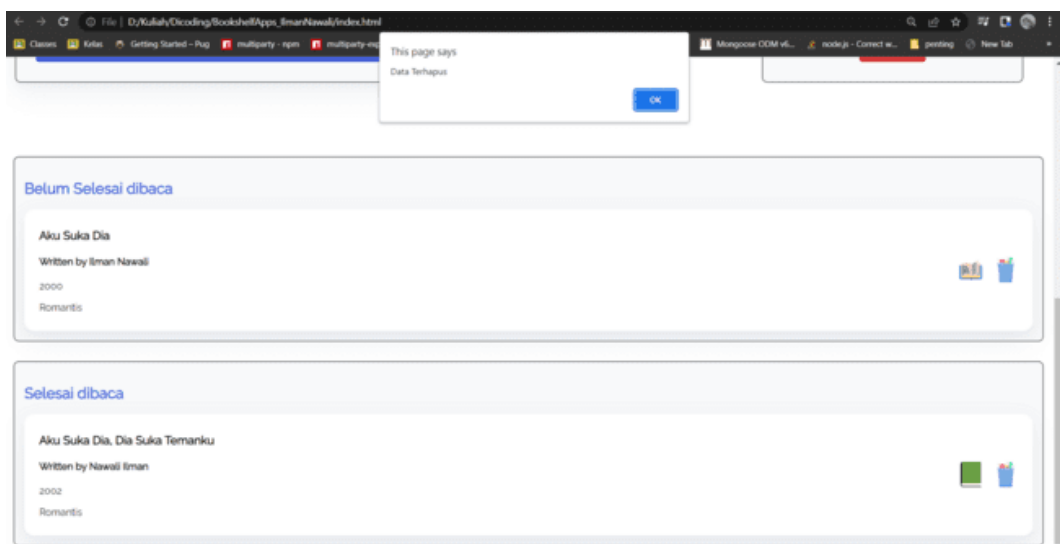
Gambar 7. Tampilan untuk Menampilkan Data

Pada gambar 7 adalah contoh tampilan menampilkan data, data yang sudah dimasukan pada gambar 6 akan ditampilkan ada menu buku yang belum selesai dibaca, dan contoh lain adalah data buku yang sudah dibaca, dan terlihat pada gambar 7 ada gambar tong sampah itu berfungsi untuk menghapus data buku dan gambar buku terbuka adalah menandakan buku itu sudah dibaca dan buku tertutup menandakan buku itu belum selesai dibaca.



Gambar 8. Tampilan Jika Mencari Buku

Gambar 8 adalah contoh tampilan jika mencari buku, dan nantinya hanya akan menampilkan buku yang sesuai yang dengan yang dicari, pada gambar 8 akan mencari buku tentang “Aku Suka Dia, Dia Suka Temanku” dan yang ditampilkan hanyalah buku yang sesuai dengan nama yang dicari, dan tombol reset berfungsi untuk mereset daftar buku tersebut agar ditampilkan semua buku lagi.



Gambar 9 Alert Jika Menghapus Data

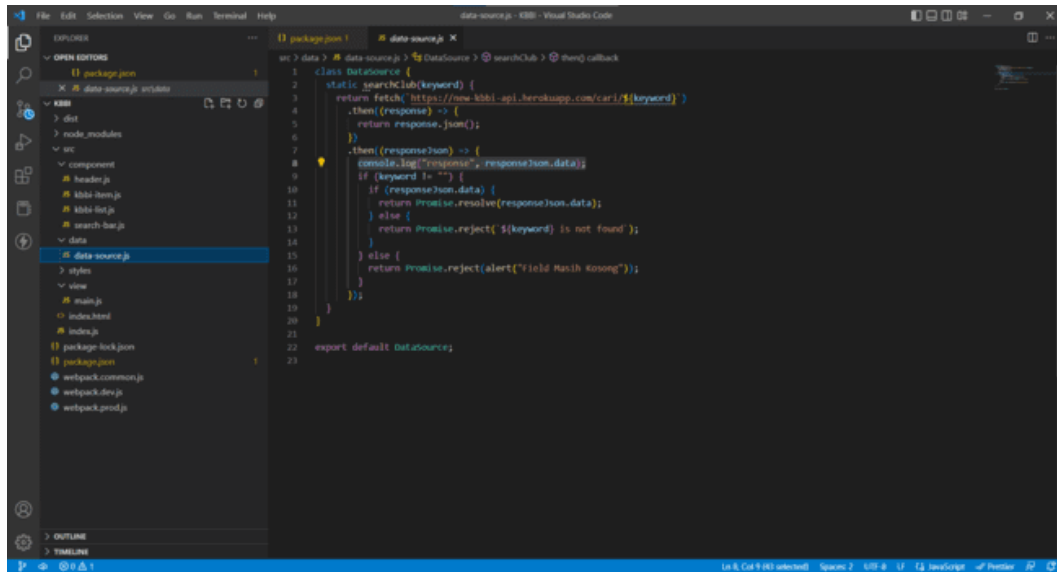
Pada gambar 9 adalah contoh jika menghapus data maka akan muncul tampilan alert jika data bukunya sudah terhapus.

### C. Membuat Website dengan ES6, Custom Element, NPM, Webpack, dan AJAX

Pada submission ini peserta harus mengerjakan website yang memiliki kriteria seperti berikut:

- Sintaks ES6.
- Web Components.
- Node Package Manager.
- Webpack.
- Asynchronous JavaScript Request menggunakan XHR dan Fetch.

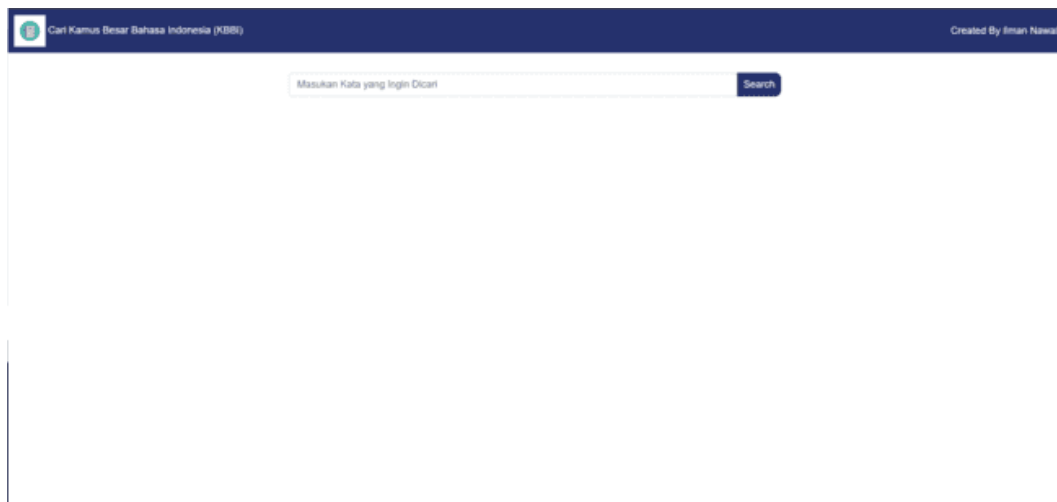




```
1 class DataSource {
2   static searchClub(keyword) {
3     return fetch('https://www.kbbi-api.herokuapp.com/cari/' + keyword)
4       .then(response => {
5         return response.json();
6       })
7       .then(responseJson => {
8         console.log('response', responseJson.data);
9         if (keyword != '') {
10          if (responseJson.data) {
11            return Promise.resolve(responseJson.data);
12          } else {
13            return Promise.reject(`${keyword} is not found`);
14          }
15        } else {
16          return Promise.reject(alert('Tolong Masuk Kosong'));
17        }
18      });
19   }
20 }
21
22 export default DataSource;
```

Gambar 10 Code Program Untuk Menggunakan API

Pada gambar 10 adalah contoh code untuk memanggil atau menggunakan *API*, karena pada tugas ini dibebaskan untuk menggunakan *API* apa saja, dan disini *API* yang dipakai adalah *API* untuk mencari kata kbbi.



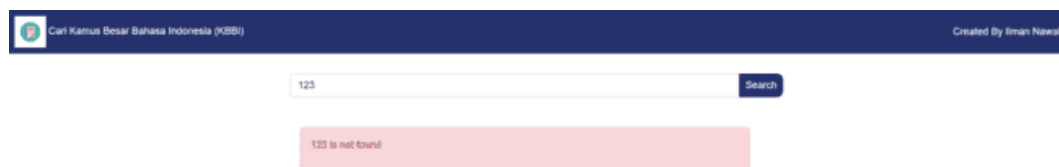
Gambar 11 Tampilan Awal Website

Gambar 4.38 menampilkan tampilan awal website jika dijalankan masih kosong karena disini belum mencari kata apapun.



Gambar 12 Tampilan Jika Mencari Kata "Sepak Bola" dan Kata yang Dicari Ditemukan

Gambar 12 adalah contoh jika mencari kata yang dicari ditemukan, pada gambar 12 mencari kata "sepak bola" dan ditemukan dan menampilkan tentang seluruh kata sepak bola yang berada dalam API.



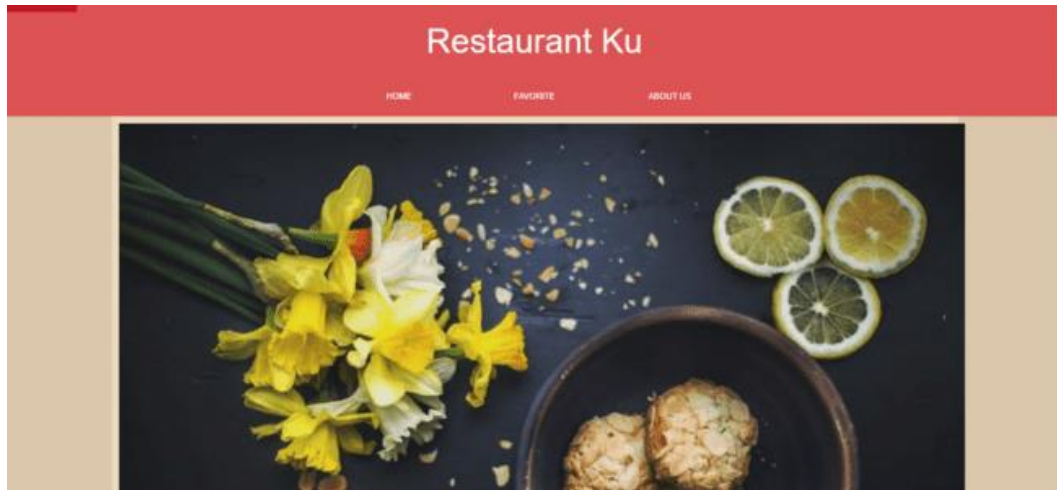
Gambar 13 Tampilan Jika Kata yang Dicari Tidak Ditemukan

Gambar 13 adalah contoh tampilan website jika memasukan kata yang dicari hasilnya tidak ditemukan.

#### D. Membuat Website Restoran PWA + Testing and Optimized

Pada submission ini peserta harus mengerjakan website yang memiliki kriteria seperti berikut:

- Penerapan PWA.
- Responsibilitas tampilan.
- Integration Test.
- End 2 End Test.
- Image Optimization.



Gambar 14 Header Website

Gambar 14 adalah contoh tampilan header website yang sudah dibuat.



Gambar 15 Tampilan List Restaurant

Gambar 15 adalah contoh potongan tampilan website daftar restaurant yang menampilkan gambar restaurant, kota, nama, rating, dan deskripsi.



Gambar 16 Tampilan Halaman Detail Restaurant 1



Gambar 17 Tampilan Halaman Detail Restaurant 2

Di dalam gambar 16 dan 17 adalah contoh dari tampilan halaman detail restaurant yang menampilkan gambar, nama, kota, rating, deskripsi, daftar menu, dan review dari customer.

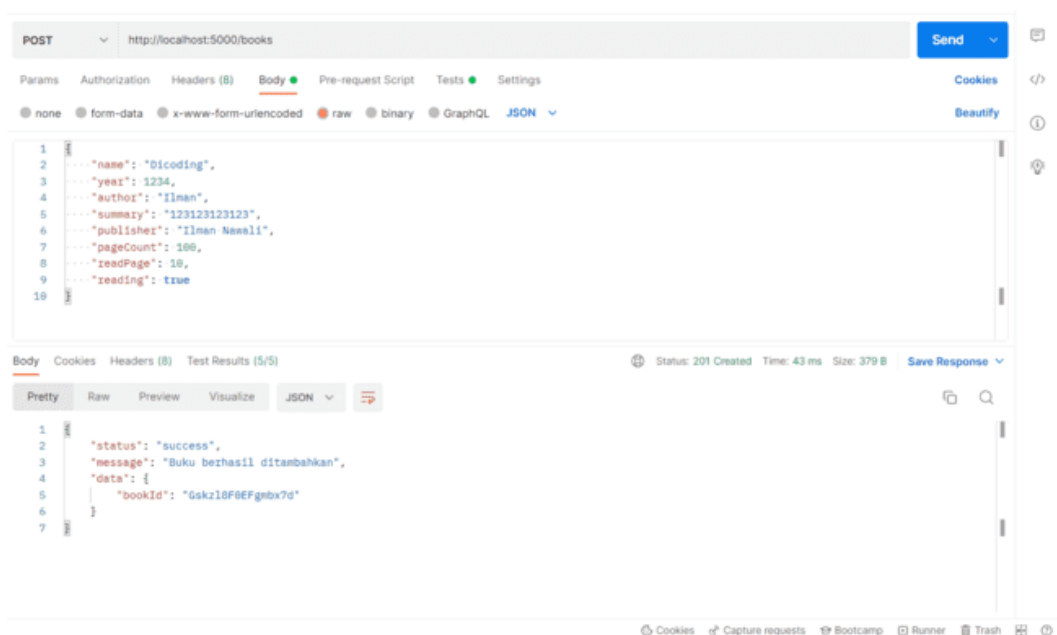


Gambar 18 Tampilan Jika Halaman Favorit Restaurant sudah mempunyai data

Gambar 18 adalah contoh jika halaman favorit sudah mempunyai data, dan tampilan ini sama seperti tampilan utama yaitu menampilkan nama, kota, gambar, rating dan deskripsi, yang membedakan pada halaman ini hanya menampilkan restaurant yang sudah disukai.

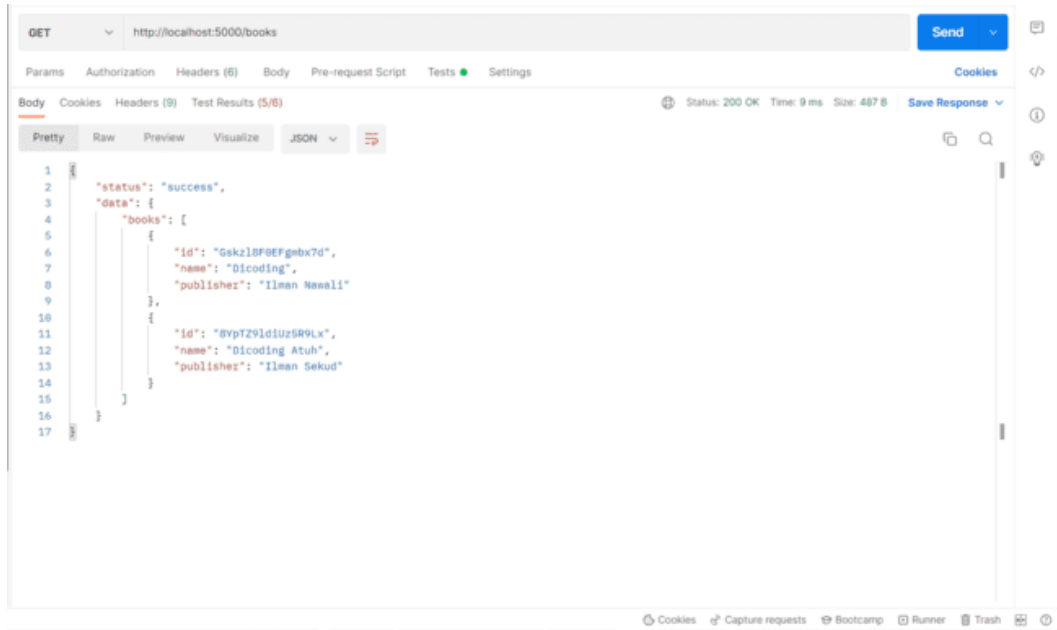
#### E. Membuat Bookshelf API

Pada *submission* ini peserta ditugaskan untuk membuat *back-end* tentang *Bookshelf Api* yang nantinya data tersebut akan disimpan pada *array*.



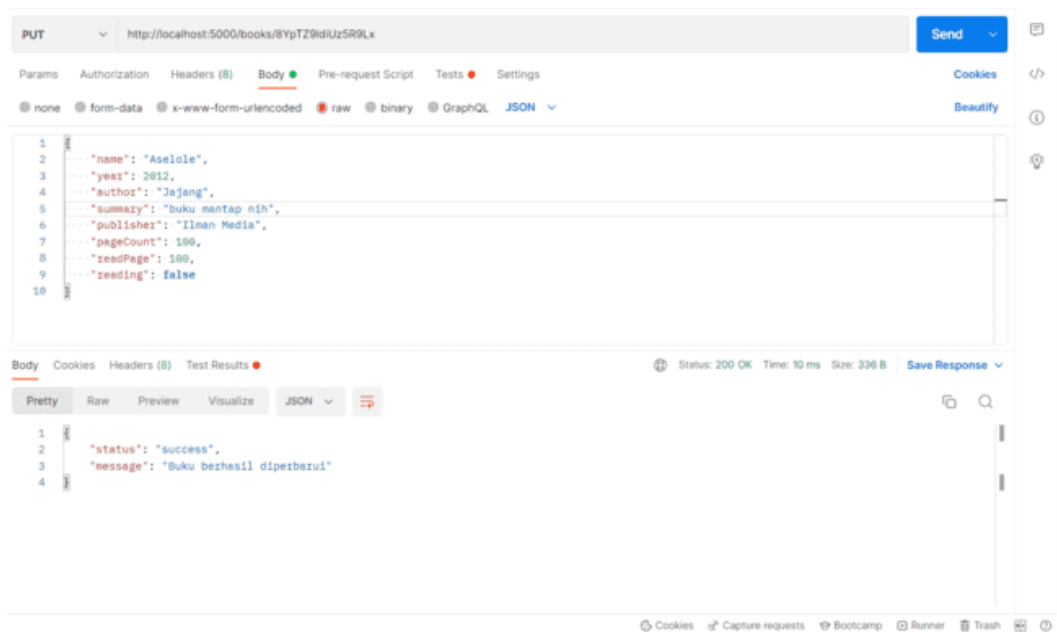
Gambar 19 Cek API di Postman 1

Gambar 19 contoh jika mencoba mengirimkan data dengan data yang lengkap yaitu ada data antara lain name, year, author, summary, publisher, pageCount, readPage, reading dan jika mengirimkan data tersebut maka akan muncul pesan "Buku berhasil ditambahkan".



Gambar 20 Cek API di Postman 2

Gambar 20 adalah contoh jika mencoba mengambil atau menampilkan data, disini hanya disuruh untuk hanya menampilkan data id, nama dan publisher saja.



Gambar 21 Cek API di Postman 3

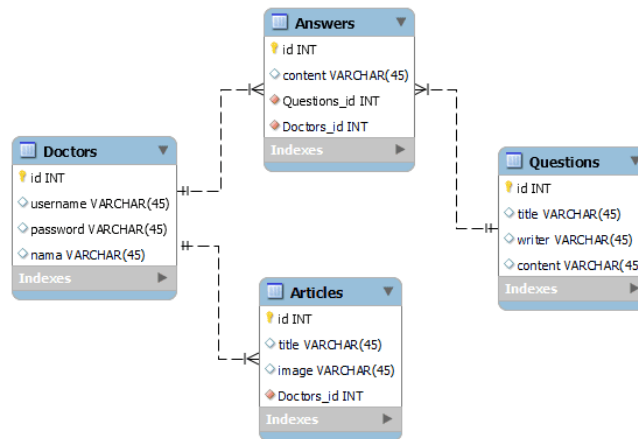
Gambar 4.85 contoh jika akan mencoba mengupdate data, disini mengupdate data sama halnya seperti akan memasukan data yaitu data harus lengkap dan data nama tidak boleh kosong juga dan tentunya readPage tidak boleh lebih besar dari pageCount.

#### F. Parent-Care

Pada tugas ini adalah membuat website Parent-Care yaitu ini adalah tugas capstone atau proyek akhir dari program Studi Independen ini. Parent-Care adalah website untuk melakukan konsultasi mengenai kesehatan mental anak yaitu orang tua akan berkonsultasi dengan seorang psikolog nantinya.

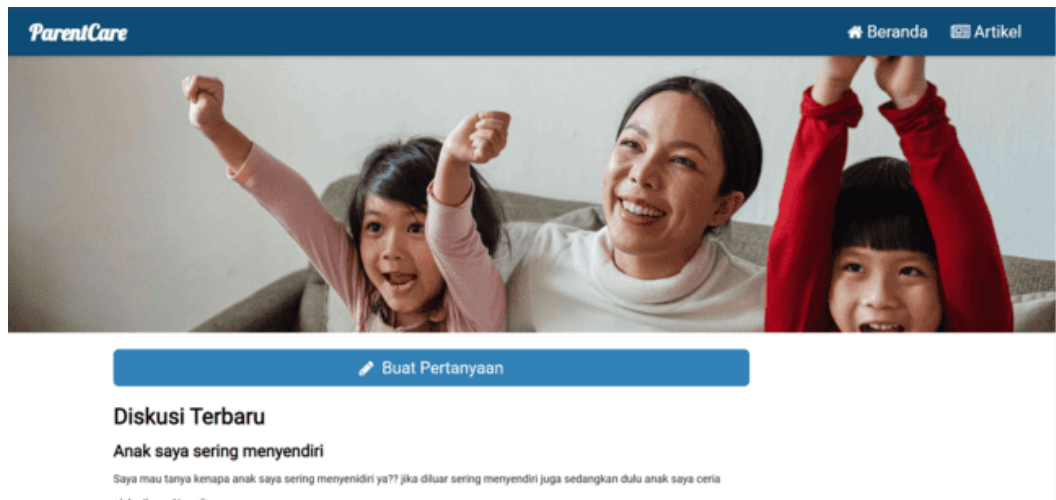
Tahapan-tahapan pengerjaannya yaitu:

1. Memikirkan data apa saja yang akan dipakai.
2. Membuat ERD.
3. Membuat Back-End/Membuat API.
4. Membuat Front-End tampilan website.
5. Mengintegrasikan dengan tampilan yang sudah dibuat dengan API.



Gambar 22 ERD

Pada gambar 22 adalah contoh ERD yang sudah dibuat yaitu didalamnya ada data yang sudah dipikirkan yaitu Answers ada data id, content, question\_id, doctors\_id, dan lain-lain.



Gambar 23 Tampilan Awal Untuk Orang Tua

Gambar 23 adalah tampilan awal website untuk tampilan orant tua, untuk orang tua tidak perlu login nantinya akan langsung masuk kedalam halaman 23 nantinya jika ingin membuat pertanyaan atau diskusi klik saja tombol Buat Pertanyaan.

## Diskusi Terbaru

### Anak saya sering menyendiri

Saya mau tanya kenapa anak saya sering menyendiri ya?? jika diluar sering menyendiri juga sedangkan dulu anak saya ceria

oleh : Ilman Nawali

### Anak Saya Sering Menyendiri

saya mau tanya bagaimana cara tau kenapa anak saya sering menyendiri yaa??

oleh : Agus Samsudin

### Anak bandel sekali

dok mau tanya anak saya bandel sekalii kenapa yaa??

oleh : Ilman Nawali

Gambar 24 Tampilan List Diskusi atau Konsultasi

Gambar 24 adalah untuk tampilan seluruh list data diskusi atau konsultasi yang ada didalam API.



Gambar 25 Tampilan Konsultasi

Gambar 25 adalah tampilan detail tentang konsultasi, disini ada pertanyaan yang sudah dibuat dan jawaban yang sudah dijawab oleh psikolog atau dokter.





Gambar 26 Tampilan Form Pertanyaan

Gambar 26 adalah tampilan jika ingin membuat pertanyaan, untuk form pertanyaan harus mengisi nama, judul pertanyaan, dan pertanyaa.



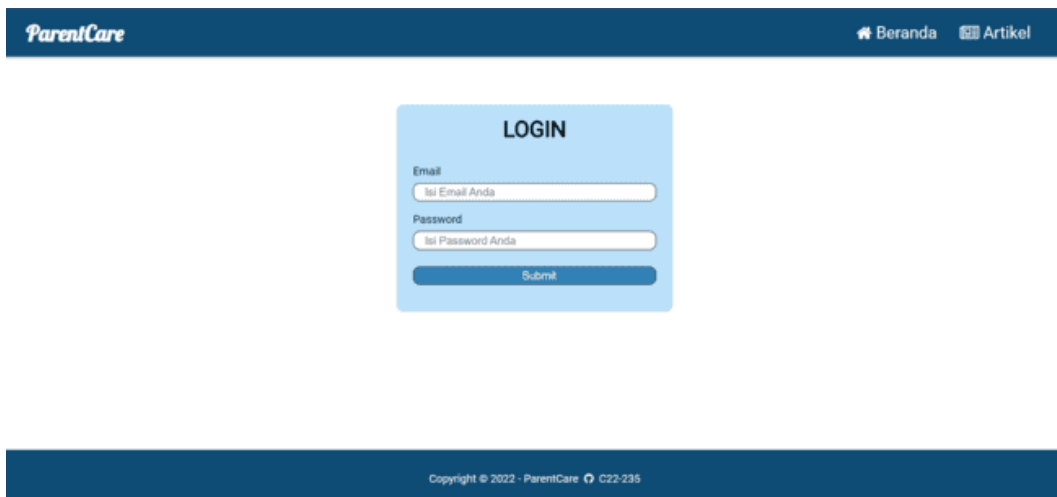
Gambar 27 Tampilan Artikel List

Gambar 27 adalah contoh tampilan seluruh list artikel yang sudah dibuat oleh psikolog atau dokter.



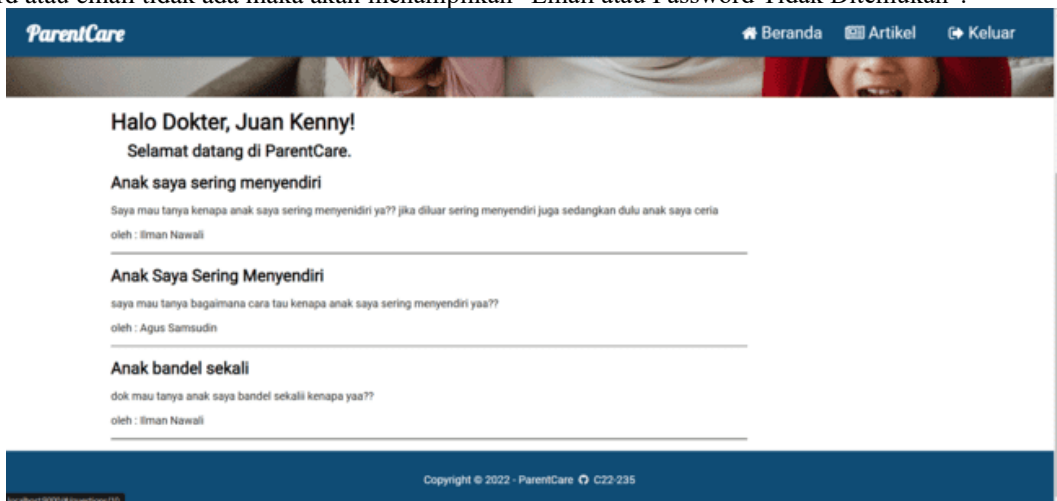
Gambar 28 Tampilan Artikel

Gambar 28 adalah tampilan untuk membaca artikel yang sudah dibuat oleh psikolog dan dokter.



Gambar 29 Tampilan Login Doctor/Psikolog

Gambar 29 adalah tampilan login untuk dokter atau psikolog, untuk dokter perlu login agar bisa masuk kedalam website, jika password atau email tidak ada maka akan menampilkan “Email atau Password Tidak Ditemukan”.



Gambar 30 Tampilan Awal Halaman Dokter/Psikolog

Gambar 30 adalah contoh tampilan awal halaman untuk dokter atau psikolog yang membedakan disini adalah untuk tampilan dokter ada tulisan Hallo Dokter, nama dokter, dan jika ada jawaban yang belum dijawab maka akan ada tombol “Balas Diskusi ini”.



Gambar 31 Form Artikel

Gambar 31 adalah tampilan form artikel untuk menambahkan artikel, untuk membuat data artikel membutuhkan data judul, gambar, isi.

## V. SIMPULAN DAN SARAN

Dengan adanya MBKM ini akan membantu menghasilkan Sumber Daya Manusia yang berkompeten, dan juga dengan adanya MBKM studi independen di Dicoding sebagai pengembang front-end dan back-end memberikan banyak manfaat, yaitu penulis menjadi tau seperti apa pembuatan website yang baik, benar dan sesuai kebutuhan industri dan juga tentunya penulis mendapatkan banyak sekali ilmu tidak hanya hard skill saja yang didapatkan, penulis juga mendapatkan ilmu tentang soft skill seperti mengevaluasi cara diri selama mengelola produktivitas diri, merancang manajemen waktu, menerapkan prinsip-prinsip moral yang sesuai dengan standar perilaku dan peraturan yang berlaku, dan ilmu tentang apa saja yang harus dipersiapkan saat akan diinterview, itu adalah ilmu soft skill yang didapat selama mengikuti studi independen di Dicoding.

Sebelum mengikuti program studi independen di dicoding, harus menyiapkan surat atau data data seperti surat rekomendasi, surat SPTJM dan sebelum mendapatkan surat rekomendasi disini diharuskan mengisi matakuliah apa saja yang akan di convert SKSnya, dan setelah itu jika mendaftar akan mendapatkan tes untuk masuk ke program yang diinginkan jika sudah lolos akan melanjutkan menjalankan program studi independen, dan pada saat penulis mengikuti program studi independen penulis juga mendapatkan sedikit hambatan tentang waktu, pengetahuan yang dirasa kurang untuk mengikuti studi independen ini. Oleh karena itu, alangkah baiknya untuk bisa mengatur waktu selama mengikuti studi independen ini, dan juga harus mempersiapkan diri terlebih dahulu dengan membaca materi yang sesuai dengan studi yang nanti akan diikuti agar tidak tertinggal dan bisa menjalankan program studi independen dengan baik.

## DAFTAR PUSTAKA

- [1] Dicoding Academy, "Atribut HTML" [Online]. Availabel: <https://www.dicoding.com/academies/123/tutorials/4119>.
- [2] Dicoding Academy, "Pengenalan HTML" [Online]. Availabel: <https://www.dicoding.com/academies/123/tutorials/4113>.
- [3] Dicoding Academy, "Struktur Dasar HTML" [Online]. Availabel: <https://www.dicoding.com/academies/123/tutorials/4115>.
- [4] gilacoding, "Tag-tag Pada HTML beserta Fungsinya" [Online]. Available: <https://www.gilacoding.com/read/tag-tag-pada-html-beserta-fungsinya>
- [5] Dicoding Academy, "Pengenalan ke Styling" [Online]. Availabel: <https://www.dicoding.com/academies/123/tutorials/4180>.
- [6] Dicoding Academy, "Perkenalan ke Web Storage" [Online]. Availabel: <https://www.dicoding.com/academies/123/tutorials/5888>.
- [7] Dicoding Academy, "Pendahuluan Dasar JavaScript" [Online]. Availabel: <https://www.dicoding.com/academies/315/tutorials/16629>.
- [8] Dicoding Academy, "Tipe Data" [Online]. Availabel: <https://www.dicoding.com/academies/315/tutorials/16655>.
- [9] Dicoding Academy, "Dasar-Dasar Node.js untuk Back-End" [Online]. Availabel: <https://www.dicoding.com/academies/261/tutorials/14187>. [Diakses 25 November 2022]
- [10] Arasopraza, " Membuat Web Server Dengan Hapi.js" [Online]. Availabel: <https://arasopraza.medium.com/membuat-web-server-dengan-hapi-js-ee8d785b53e1>.
- [11] BADR, "Software Developer: Fullstack, Frontend, Backend. Apa Bedanya?" [Online]. Available: <https://badr.co.id/software-developer-fullstack-frontend-backend-apa-bedanya/>.

- [12] Glints, "Mengetahui Proses dan Peran Penting Software Development" [Online]. Available: <https://glints.com/id/lowongan/proses-peran-software-development/#.Y7wuLXZBzIU>.