

Rancangan Sistem Informasi Katalog Perpustakaan RPTRA Villa Sawo

Laras Dewi Adistia, Rozi dan Timoti Bagus Setiawan

Sistem Informasi, STMIK Jakarta STI&K
Jl. BRI No.17, Kebayoran Baru, Jakarta Selatan
ldadistia@gmail.com, roziborang72@gmail.com dan timotibagus@outlook.com

Abstrak

RPTRA Villa Sawo merupakan Ruang Publik Terpadu Ramah Anak atau sebuah taman yang didalamnya terdapat sebuah perpustakaan kecil untuk anak-anak membaca buku-buku yang sudah disediakan. Perpustakaan ini masih menggunakan pencatatan manual untuk setiap bukunya. Dalam penulisan ini dibahas bagaimana membuat sistem informasi katalog elektronik untuk digunakan pada perpustakaan kecil di RPTRA Villa sawo yang sebelumnya masih menggunakan pencatatan manual pada setiap buku yang disimpan dengan adanya katalog elektronik ini akan memudahkan petugas mencatat, mencari, menghapus data buku. Sehingga proses penyimpanan buku menjadi lebih efisien serta mudah digunakan. Bahasa pemrograman yang akan digunakan yaitu Javascript. Terdapat 2 proses pembuatan program yaitu FrontEnd yang berhubungan dengan tampilan antar muka serta Backend yang berhubungan dengan server atau database. Baik Frontend/Backend sama-sama menggunakan Javascript. Pada Frontend akan menggunakan library React.js. Pada Backend akan menggunakan framework Express.js serta database MongoDB. Backend akan bertugas menyediakan API (Application Programming Interface) agar data bisa ditampilkan Frontend.

Kata kunci: rancangan, sistem informasi, perpustakaan, RPTRA villa sawo, javascript

Pendahuluan

Teknologi informasi dan komputer pada masa ini telah menunjukkan perkembangan yang signifikan. Teknologi ini memiliki beberapa prinsipnya, yaitu untuk melayani kebutuhan informasi secara tepat guna, tepat waktu, dan tepat sasaran. Kebutuhan tepat guna atau akurat dapat terpenuhi jika pengambilan keputusan bersumber dari kebenaran informasi yang dihasilkan. Kebutuhan tepat guna akan terpenuhi jika informasi yang dihasilkan benar sehingga mendukung pengambilan keputusan yang benar. Informasi dapat dikatakan tepat waktu jika informasi yang kita butuhkan tersedia pada saat dibutuhkan, sehingga kecepatan proses menjadi peranan yang penting. Sementara penggunaan informasi tepat sasaran yaitu jika informasi yang dihasilkan dapat digunakan pada orang yang membutuhkan atau memerlukannya sehingga informasi menjadi relevan dengan penggunaannya dan Internet merupakan salah satu media dan sarana untuk mendapatkan informasi dengan tepat guna, tepat waktu / cepat, serta tepat sasaran. Dengan adanya internet perkembangan digital menjadi begitu modern sehingga memunculkan suatu era yang disebut era komputerisasi. Komputerisasi ialah dimana pengelolaan data yang tadinya manual diganti oleh alat bantu komputer. [1] Dengan kemudahan yang ditawarkan komputerisasi memberikan

kemudahan untuk pekerjaan kita bahkan menjadi lebih akurat, efisien dan efektif.

Dikarenakan pada era ini komputerisasi telah berkembang, begitupun pada perpustakaan yang akan menjadi lebih modern jika diterapkan sistem informasi digital. Maka dalam hal ini penulis membuat penelitian dengan judul “Rancangan Sistem Informasi Katalog Perpustakaan RPTRA Villa Sawo” untuk dapat mempermudah pencarian buku pada perpustakaan, dibutuhkan sebuah sistem informasi yang menyediakan daftar buku yang ada secara komputerisasi maka, dibuatkan katalog elektronik. Pada pembuatan ini penulis menggunakan bahasa pemrograman HTML, [3] CSS, Javascript, dan database menggunakan MongoDB. Aplikasi antar muka yang dibuat menggunakan pustaka / library dari Javascript yaitu React. Untuk aplikasi pada bagian server menggunakan Express.js framework dari Node.js. [4] .

Batasan Masalah

Hanya membahas: 1. Cara penggunaan website secara offline, 2. Aplikasi yang dibuat menggunakan library Reac, 3. Sistem katalog elektronik dalam penulisan ini dibuat dengan bahasa pemrograman HTML(Hypertext Markup Language), CSS(Cascading Style Sheets), Javascript.4. Pada DBMS (Database Management System) menggu-

nakan MongoDB.

Tujuan

Adapun tujuan penulisan ini dibuat ialah : Melakukan komputerisasi pada perpustakaan yang masih menggunakan sistem manual. Mengembangkan website guna mempermudah dalam penginputan serta pencarian ketersediaan buku.

Metode Penelitian

Dalam mendapatkan informasi untuk pengembangan proses pembuatan aplikasi, dilakukan beberapa metode penelitian untuk mendapatkan keterangan dan penjelasan mengenai masalah yang dihadapi. Adapun metode yang digunakan, yaitu: Studi Pustaka melalui metode ini, dapat memperoleh data dan informasi yang dibutuhkan dengan membaca buku, buku digital, makalah, internet dan sumber lainnya untuk menunjang penulisan ini. Studi lapangan untuk mendapatkan informasi serta data – data yang dibutuhkan, maka penelitian ini dilakukan dengan cara observasi pada lingkungan RPTRA Taman Villa Sawo.

Analisa Masalah

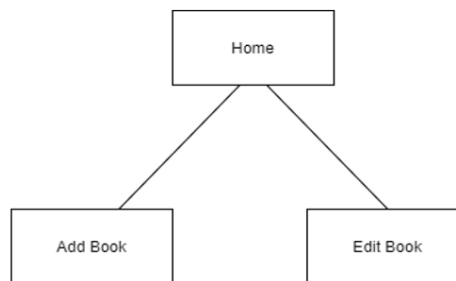
Pada perpustakaan RPTRA Villa Sawo belum pada sistem dan teknologi untuk mengelola buku-buku yang ada secara digital. Untuk saat ini pencatatan buku-buku masih menggunakan cara manual atau tulis tangan yang dilakukan petugas perpustakaan. Cara ini tentu tidak efisien melihat pencatatan masih menggunakan cara manual, ini tentunya menimbulkan beberapa resiko yaitu data buku bisa hilang sewaktu-waktu, pencarian data yang lama. Dengan adanya beberapa resiko, maka diperlukan sebuah sistem komputerisasi untuk mengelola data-data buku sebagai sebuah sistem informasi yang baru.

Pemecahan Masalah

Dari permasalahan yang dihadapi, menggunakan sistem informasi membantu proses pencatatan buku-buku menjadi lebih cepat dan data tersimpan dengan aman sehingga membantu proses kerja dari penjaga perpustakaan. Maka dari itu akan dibuat sebuah sistem informasi katalog berbasis website yang berisikan proses penginputan data, perubahan data, serta mengambil data-data buku. Petugas perpustakaan hanya membutuhkan satu perangkat keras baik itu komputer ataupun laptop untuk menaruh data di database dan mengasesnya melalui browser. Koneksi internet tidak diperlukan karena data akan tetap berjalan secara local atau tanpa internet.

Struktur Navigasi

Pada pembuatan aplikasi ini akan menggunakan struktur navigasi hirarki yang selanjutnya dibuatkan peta navigasinya. Peta navigasi yang akan mengarahkan alur pada halaman website. Terdapat 3 halaman website, yaitu Home, Add Book, dan Edit Book.



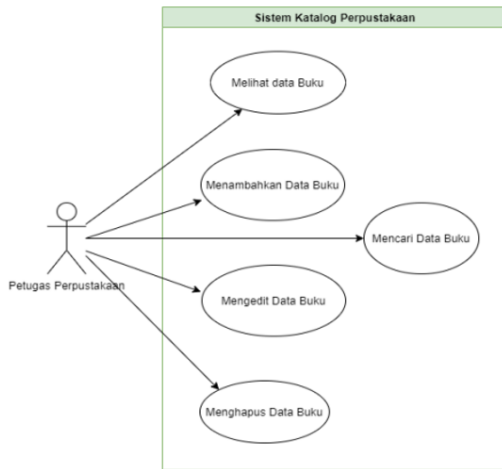
Gambar 1: Peta Struktur Navigasi Hirarki

Perancangan Sistem

Untuk membantu dalam perancangan sistem perangkat lunak, menggunakan UML (Unified Modelling Language) yang merupakan metode pemodelan berorientasi objek. Dengan adanya UML bisa memberikan permodelan visual dari berbagai macam bahasa pemrograman serta sebagai blue print dari koding pada suatu program. Dalam penulisan ini diagram UML yang akan digunakan Use Case Diagram, Sequence Class, Diagram Activity.

Use Case Diagram

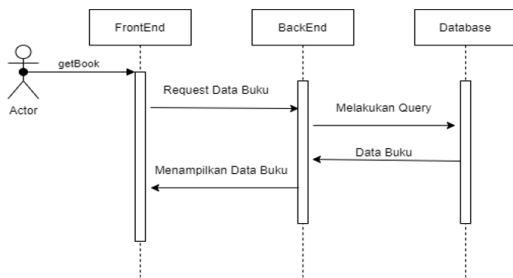
Use Case diagram menggambarkan grafis dari beberapa actor dan interaksi dengan beberapa aktivitas untuk memperkenalkan suatu sistem. Pada gambar 2 hanya terdiri dari 1 aktor saja, yaitu pengguna (petugas perpustakaan). Aktor dapat melakukan penambahan data, menghapus data, merubah isi data serta melihat data keseluruhan. Dalam menambah dan merubah isi data pengguna harus mengisi form terlebih dahulu, yaitu judul, pengarang, tahun terbit.



Gambar 2: Use Case Diagram Sistem Perpustakaan

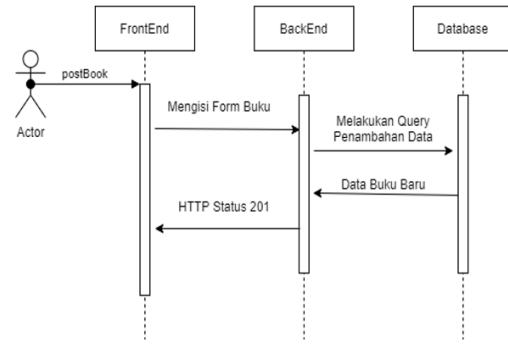
Sequence Diagram

Sequence diagram merupakan diagram yang dipakai untuk menggambarkan kolaborasi dinamis antara objek satu dengan lainnya. Ada 4 komponen utama dari sequence diagram objek, activation boxes, aktor, lifeline. Terdapat 4 HTTP Method pada aplikasi ini yaitu GET, POST, PATCH, DELETE.



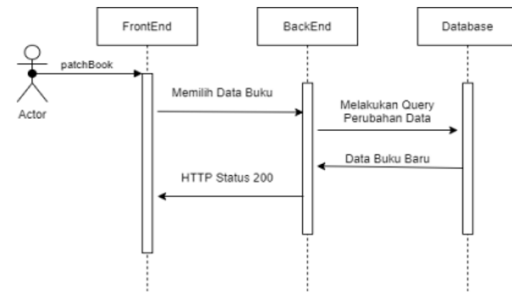
Gambar 3: Sequence Diagram getBook

Gambar 3 menjelaskan tentang proses penampilan data buku yang ditampilkan melalui proses yang terjadi di FrontEnd(React.js), Back-End(Express.js), dan Database. FrontEnd melakukan method GET, lalu direspon oleh Backend untuk melakukan query pada database. Database memberikan response berbentuk dan diterima oleh Backend dan dikirim ke Frontend berupa JSON (JavaScript Object Notation) sehingga data itu diolah dengan menampilkan table data.



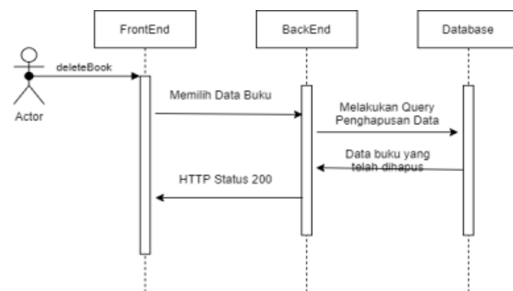
Gambar 4: Sequence Diagram postBook

Gambar 4 menjelaskan proses HTTP Method POST untuk menambahkan data buku yang baru. Dengan cara mengisi form buku yang terdiri dari judul, pengarang, dan tahun terbit lalu data dikirim ke backend dan jika berhasil maka akan memberikan response HTTP status 201. Status 201 bisa diartikan data berhasil dibuat.



Gambar 5: Sequence Diagram patchBook

Gambar 5 menjelaskan proses PATCH method yang dilakukan oleh petugas perpustakaan dengan cara, memilih buku yang ingin datanya diubah. Lalu user akan dimintai untuk mengubah data form judul, pengarang, tahun. Jika data sudah diubah lalu click submit, jika berhasil maka akan langsung menuju home.



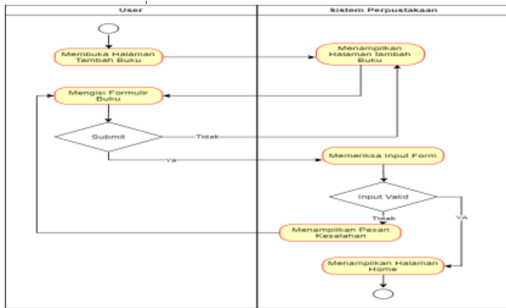
Gambar 6: Sequence Diagram deleteBook

Gambar 6 menjelaskan proses penghapusan data buku. Dengan memilih data buku mana yang akan dihapus. Maka data akan langsung dihapus dari database dan pada daftar buku di frontend.

Diagram Activity

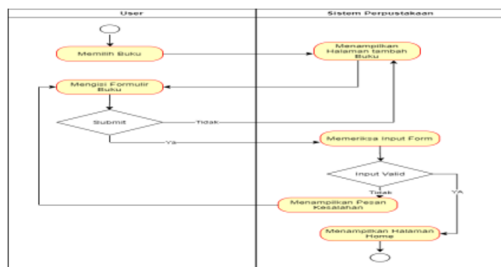
Diagram Activity, menjelaskan mengenai aktivitas yang terjadi antara pengguna, sistem atau sistem dengan sistem. Fungsi diagram activity yaitu untuk memperlihatkan urutan dari aktivitas yang dilakukan oleh user yang selanjutnya akan direspon oleh sistem juga membantu menggambarkan proses bisnis lebih detail. Terdapat 3 diagram activity pada sistem ini yaitu, menambahkan buku, mengubah data buku, dan menghapus buku.

Menambahkan Buku



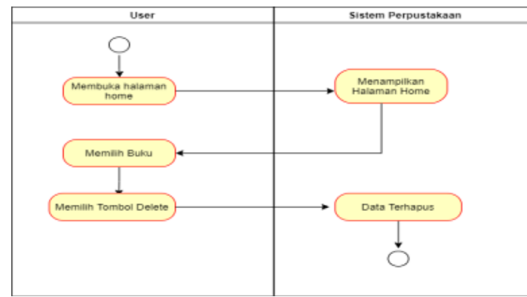
Gambar 7: Diagram Activity Menambahkan Buku

Mengubah Data Buku



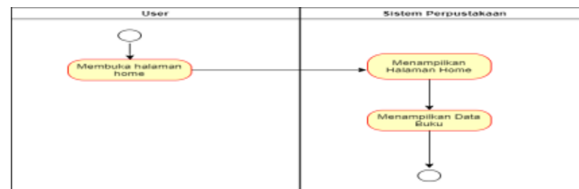
Gambar 8: Diagram Activity Mengubah Buku

Menghapus Buku



Gambar 9: Diagram Activity Menghapus Buku

Menampilkan Buku



Gambar 10: Diagram Activity Menampilkan Buku

Perancangan Antarmuka

Pembuatan desain antarmuka merupakan langkah awal untuk menentukan elemen-elemen apa saja yang nanti akan dipakai pada saat proses coding sebuah website. Desain antarmuka juga bertujuan agar aplikasi yang dibuat mudah digunakan dan tidak rumit. Selain itu desain antarmuka mengatur penyusunan tata letak sebuah elemen, ilustrasi, serta bentuk tombol.

Rancangan Antarmuka Halaman Utama

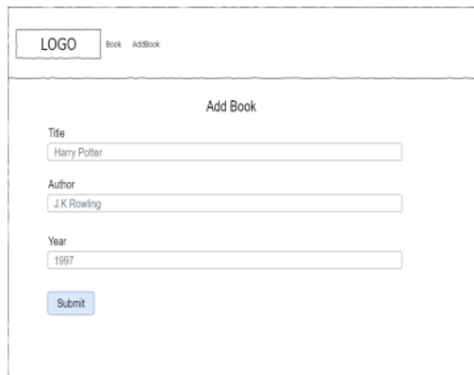
Perancangan tampilan antarmuka halaman utama akan menampilkan keseluruhan data buku-buku yang berupa tabel, setiap baris memiliki tombol untuk menghapus dan mengubah data, dan juga terdapat fitur pencarian buku.



Gambar 11: Rancangan Antarmuka Halaman Utama

Rancangan Antarmuka Halaman Tambah Buku

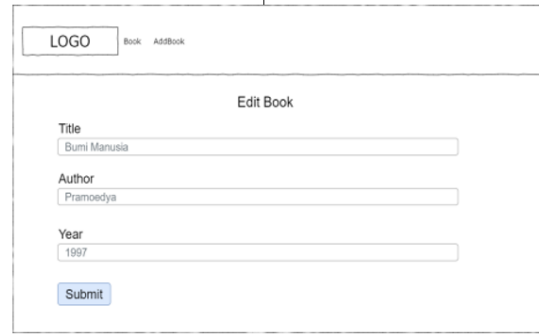
Rancangan antarmuka pada halaman tambah buku akan menampilkan form input seperti nama judul, pengarang, serta tahun. Pada halaman ini form input judul dan pengarang berupa huruf, serta tahun berupa angka. Terdapat tombol submit untuk menambahkan data buku sesuai form.



Gambar 12: Rancangan Antarmuka Halaman Tambah Buku

Rancangan Antarmuka Halaman Ubah Buku

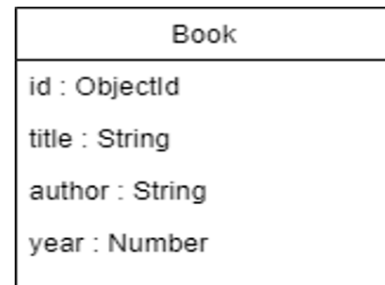
Rancangan antarmuka halaman ubah buku berisikan form informasi dari data yang ingin diubah, beserta tombol submit untuk mengubah data tersebut.



Gambar 13: Rancangan Antarmuka Halaman Ubah Buku

Perancangan Database

Perancangan sistem basis data pada aplikasi ini menggunakan NoSQL yaitu MongoDB. Pada sistem database NoSQL merupakan database yang tidak memerlukan skema dan relasi untuk setiap tabelnya. Sebuah dokumen yang disimpan pada NoSQL yaitu berupa JSON (Javascript Object Notation). Berikut merupakan struktur pada database, yaitu hanya tabel Book.



Gambar 14: Struktur Tabel Book

Implementasi Database Aplikasi

Setelah proses perancangan aplikasi tahapan selanjutnya ialah implementasi aplikasi. Pada tahapan implementasi ini, elemen-elemen pada proses perancangan akan realisasikan kedalam suatu bahasa program. Pengaksesan aplikasi katalog buku dapat dibuka melalui aplikasi browser (Google Chrome, Mozilla Firefox, dll) yang tersedia pada perangkat komputer admin. Dengan url <http://localhost:3000> admin langsung bisa mengakses aplikasi

Membuat Database di MongoDB

Pembuatan database dapat dilakukan secara otomatis melalui bahasa pemrograman Javascript dengan Framework Express.js. Berikut langkah-langkah untuk membuat database:

1. Buka Visual Studio Code. Dengan Visual Studio Code kita memulai proses pengkodean.
2. Install dependencies “mongoose” sebagai driver untuk MongoDB menggunakan CMD. Ketik “npm install mongoose”
3. Jika proses instalasi berhasil, Langkah selanjutnya ialah membuat file server.js yang berisi kode seperti dibawah ini.

```
const mongoose = require('mongoose');
const bookSchema = new mongoose.Schema({
  title: {
    type: String,
    required: true
  },
  author: {
    type: String,
    required: true
  },
  year: {
    type: Number,
    required: true
  }
});
module.exports = mongoose.model('Buku', bookSchema);
```

Gambar 16: Koding Collection

```
const mongoose = require('mongoose')

mongoose.connect('mongodb://localhost:27017/perpustakaan', {
  useNewUrlParser: true,
  useUnifiedTopology: true
});
const db = mongoose.connection;
db.on('error', (err) => console.log(err));
db.once('open', () => console.log('Connected to database!'));
```

Gambar 15: Koding Database

Pada tahap ini kita akan menentukan struktur pada Collection yang kita buat. Title, author, dan year merupakan schema pada collection yang nanti akan menjadi struktur data serta tipe data. Untuk nama collection yaitu, “Buku” yang tertera pada baris kode terakhir.

2. Jika langkah pembuatan database dan collection sudah, kita akan mencoba kodingan dengan cara “node server.js”, seperti pada gambar dibawah.

Pada koding diatas kita akan membuat sebuah database dengan nama “perpustakaan” yang diketik pada URL database “mongodb://localhost:27017/perpustakaan?”. Maka pada MongoDB database kita akan dibuat secara otomatis.

```
PS D:\Koding\backend> node server
(node:11376) warning: Accessing non-existent property 'MongoError' of module exports inside circular dependency
(Use 'node --trace-warnings ...' to show where the warning was created)
Server is running on port 5000
Connected to database
```

Gambar 17: Command Line Server

Membuat Collection

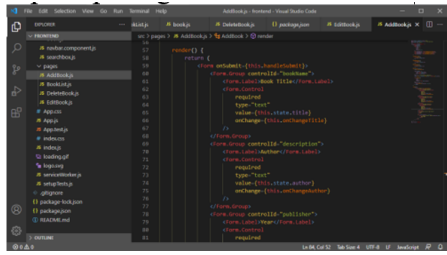
Jika pembuatan database sudah, selanjutnya ialah pembuatan Collection atau pada database SQL disebut dengan tabel. Collection ini nantinya akan menyimpan data-data buku. Adapun langkah untuk membuat sebuah collection dengan pengkodean yaitu :

1. Pengkoding untuk membuat collection ialah seperti pada gambar dibawah ini. Dengan membuat file baru yaitu book.model.js

Jika pesan “Connected to database” berarti kita sudah terkoneksi dengan database. Serta Database dan Collection yang kita buat berhasil.

Implementasi Tampilan Aplikasi

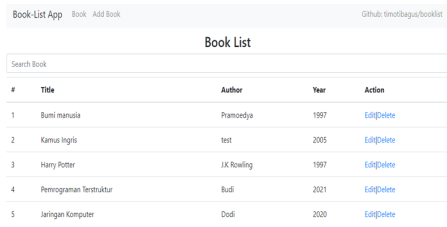
Pengimplementasi tampilan antar muka pada website terbentuk dari gabungan bahasa pemrograman HTML serta Javascript. Pengkodean akan dilakukan dengan menggunakan aplikasi editor Visual Studio Code, seperti pada gambar dibawah ini.



Gambar 18: Tampilan Visual Studio Code

Tampilan Halaman Utama

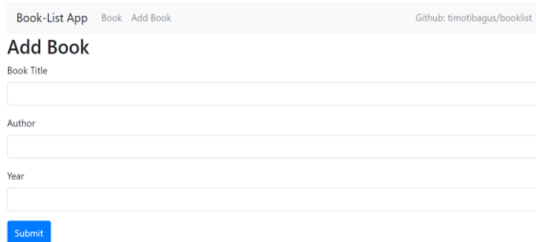
Halaman utama, merupakan halaman pertama yang akan ditampilkan ketika membuka url website. Halaman ini sekaligus menampilkan daftar buku-buku yang ada pada database, serta terdapat fungsi pencarian buku. Halaman utama dapat dilihat seperti pada gambar dibawah ini. Serta terdapat menu navigasi untuk berpindah halaman pada bagian atas.



Gambar 19: Halaman Utama atau Home

Tampilan Halaman Tambah Buku

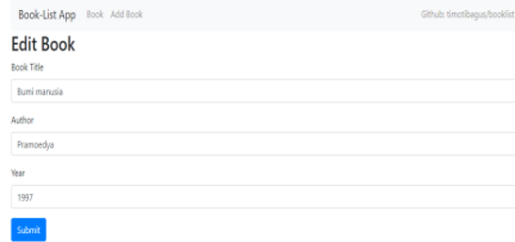
Halaman tambah buku merupakan halaman untuk menambahkan data buku melalui form. Dengan mengisi judul buku, nama pengarang serta tahun buku data buku baru bisa ditambahkan dengan tombol submit. Halaman Tambah Buku dapat dilihat seperti pada gambar dibawah ini.



Gambar 20: Halaman tambah buku

Halaman Edit Buku

Halaman Edit Buku merupakan halaman untuk mengubah data buku yang sudah ada. Dengan mengubah data yang sudah ada pada form judul buku, nama pengarang dan tahun buku. Halaman edit buku dapat dilihat pada gambar dibawah ini.



Gambar 21: Halaman Edit Buku

Pengujian

Proses perancangan, pengembangan serta pengujian aplikasi ini dilakukan menggunakan komputer dengan spesifikasi sebagai berikut :

Perangkat Keras (Hardware)

1. Processor : Intel Pentium G3240
2. RAM : 8 GB DDR3
3. Penyimpanan : 120 GB SSD
4. VGA Card : Nvidia GTX 750 1 GB

Perangkat Lunak (Software)

1. Windows 10
Sistem operasi yang digunakan pada perangkat komputer
2. Visual Studio Code
Sebagai kode editor, proses pengkodean program pada aplikasi berbasis web yang dibuat.
3. Google Chrome
Sebagai browser yang dipakai untuk menjalankan website.
4. MERN Stack
Merupakan kumpulan service-service bahasa pemrograman yang akan digabungkan menjadi satu aplikasi:
 - (a) MongoDB Digunakan sebagai Database berbasis NoSQL.
 - (b) Express Framework web server dari Node.js, berfungsi sebagai backend.
 - (c) React Framework client-side website, berfungsi sebagai frontend.

- (d) Node Js Framework Javascript yang memungkinkan Javascript bisa berjalan diluar website.
- 5. Postman
Sebagai API testing untuk melakukan HTTP method (POST, GET, Patch, Delete) dari URL yang tersedia pada backend.

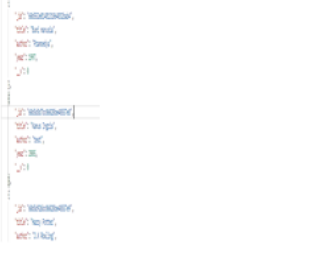

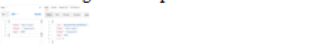


Proses pengujian akan dibagi menjadi 2, yaitu pengujian untuk backend dan pengujian frontend. Pada pengujian backend akan mengunakan aplikasi Postman pada setiap alamat Routing . Pengujian frontend akan dilakukan di Google Chrome, untuk melakukan pengecekan pada alamat website, tombol, dan fungsi lainnya.

Skenario Pengujian

Skenario pengujian sebagai tahap-tahap yang dilakukan untuk menguji aplikasi yang telah dibuat.



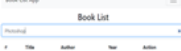


Pengujian Backend

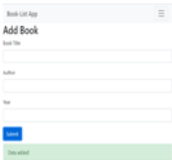

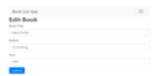

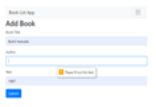
Tabel 1: *Pengujian Backend*

No	Skenario Pengujian	Hasil yang diharapkan	Kesimpulan
1	Jika membuka link <code>http://localhost:5000/books</code> dengan method GET.	Maka server akan mengirimkan respons berupa semua list buku-buku pada database berupa JSON. 	Valid
2	Jika membuka link <code>http://localhost:5000/books/id</code> dengan method GET dan penambahan id buku. Contoh id <code>60d552e8148131064832bab4</code> .	Maka server akan merespon buku sesuai dengan id yang diberikan pada url, berupa JSON. 	Valid
3	Jika membuka link <code>http://localhost:5000/books</code> dengan method POST. Serta mengisi form berupa JSON yang berisi title, author, year.	Maka server akan memberikan data buku beserta dengan id baru pada data tersebut. 	Valid
4	Jika membuka link <code>http://localhost:5000/books/id</code> dengan method PATCH beserta id buku, yang ingin datanya diubah.	Maka server akan mengubah data buku dengan id yang diberikan pada url serta akan memberikan respons "updated". 	Valid
5	Jika membuka link <code>http://localhost:5000/books/id</code> dengan method DELETE beserta id buku.	Maka server akan menghapus data buku sesuai dengan id yang tertera pada url, dan memberikan respon "deleted". 	Valid

Pengujian FrontEnd

Tabel 2: *Pengujian Backend*

No	Skenario Pengujian	Hasil yang diharapkan	Kesimpulan
1	Mebuka halaman menu utama	Maka akan menampilkan daftar buku-buku beserta fitur pencarian 	Valid
2	Mencari buku yang ada pada database	Maka daftar buku akan terfilter sesuai judul buku yang dicari 	Valid
3	Mencari buku yang tidak terdapat pada database	Maka daftar buku tidak menampilkan apa-apa 	Valid
4	Menekan tombol hapus pada buku yang ingin dihapus	Menampilkan pesan "Data deleted" 	Valid
5	Menekan tombol "Add Book" pada bagian navigasi	Menampilkan halaman tambah buku 	Valid
6	Ketika mengisi semua form, lalu menekan tombol submit	Muncul pesan "Data Added" dibawah tombol submit	Valid

				
7	Ketika tidak mengisi salah satu form pada halaman tambah buku, lalu menekan submit	Menampilkan peringatan untuk mengisi form yang kosong dengan pesan "Please fill out this field"		Valid
8	Ketika menekan edit di halaman home pada buku yang ingin datanya dirubah	Menampilkan halaman edit book dengan form yang sudah terisi sesuai dengan data buku		Valid
9	Ketika mengganti data pada halaman edit buku dengan semua form terisi, lalu menekan tombol submit	Maka akan menampilkan pesan "Data Updated" dan akan langsung mengarahkan kehalaman home.		Valid
10	Ketika tidak mengisi salah satu form pada halaman edit book, lalu menekan tombol submit	Menampilkan peringatan untuk mengisi form yang kosong dengan pesan "Please fill out this field"		Valid

Penutup

Simpulan

Berdasarkan hasil penelitian ini telah dihasilkan aplikasi berbasis website yang digunakan sebagai katalog penyimpanan buku-buku secara digital. Pada hasil uji coba di bab sebelumnya, aplikasi berbasis website berjalan sesuai dengan fungsinya. Baik frontend, backend, serta database MongoDB berbasis NoSQL yang terkoneksi dengan baik. Dibandingkan dengan cara sebelumnya yang masih manual, dengan aplikasi berbasis website ini pengguna bisa lebih cepat dan efisien menyimpan serta pencarian data-data buku. Dengan memakai database MongoDB berbasis NoSQL, database memiliki performa yang lebih unggul jika dibandingkan dengan database berbasis Sehingga walaupun data yang disimpan banyak tetapi performansi dari database akan

tetap stabil. Penggunaan Javascript pada pembuatan aplikasi ini sangat memudahkan dalam proses pengembangan, karena hanya membutuhkan satu bahasa pemrograman untuk membuat aplikasi frontend dan backend.

Daftar Pustaka

- [1] William H. Leffingwell, Edwin M. Robinson *Textbook of Management*, New York, McGraw-Hill Book Company, 1998
- [2] Hartono, *Manajemen Perpustakaan Sekolah*. Yogyakarta: Ar-Ruzz Media, 2016
- [3] *Sistem Informasi Manajemen - Perspektif Organisasi* Yogyakarta Gudang Penerbit, 2020

- [4] Irfan Dwi Jaya, Dasar Pemrograman WEB Menggunakan PHP dan MYSQL Surabaya: Prenada Media Group,2020
- [5]]Greg Lim , Beginning Node.js, Express & MongoDB Development Singapura, 2020