

LAPORAN KERJA PRAKTIK

REKAYASA PERANGKAT LUNAK APLIKASI LIBRARY APP
MENGGUNAKAN JSON UNTUK MEREKAM DATA BUKU
DI PT RUANG RAYA INDONESIA

Diajukan untuk memenuhi persyaratan kelulusan

Matakuliah FTI335 – Kerja Praktik

Oleh :

TEJA KUSUMAH / 301200033



PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS BALE BANDUNG
2024

LEMBAR PENGESAHAN
PROGRAM STUDI TEKNIK INFORMATIKA
REKAYASA PERANGKAT LUNAK APLIKASI LIBRARY APP
MENGGUNAKAN JSON UNTUK MEREKAM DATA BUKU
DI PT RUANG RAYA INDONESIA

Oleh :

TEJA KUSUMAH / 301200033

Disetujui dan disahkan sebagai
LAPORAN KERJA PRAKTIK

Bandung, 11 Januari 2024

Koordinator Kerja Praktik

Yusuf Muharam, S.Kom., M.Kom.

NIK. 04104820003

LEMBAR PENGESAHAN
PEMBIMBING KERJA PRAKTIK
REKAYASA PERANGKAT LUNAK APLIKASI LIBRARY APP
MENGGUNAKAN JSON UNTUK MEREKAM DATA BUKU
DI PT RUANG RAYA INDONESIA

Oleh :
TEJA KUSUMAH / 301200033

Disetujui dan disahkan sebagai
LAPORAN KERJA PRAKTIK

Bandung, 11 Januari 2024
Pembimbing Kerja Praktik

Yudi Herdiana S.T., M.T.
NIK. 04104808008

LEMBAR PENGESAHAN
PT RUANG RAYA INDONESIA

REKAYASA PERANGKAT LUNAK APLIKASI LIBRARY APP
MENGGUNAKAN JSON UNTUK MEREKAM DATA BUKU
DI PT RUANG RAYA INDONESIA

Oleh :

TEJA KUSUMAH / 301200033

Disetujui dan disahkan sebagai
LAPORAN KERJA PRAKTIK

Jakarta, 11 Januari 2024

Pembimbing Perusahaan



Pratama Agung Sumirat. S.T.

ABSTRAKSI

Kerja Praktik (KP) yang dilaksanakan di PT Ruang Raya Indonesia pada Kementerian Pendidikan dan Kebudayaan Republik Indonesia melalui program Merdeka Belajar Kampus Merdeka (MBKM) pada program Magang dan Studi Independen Bersertifikat (MSIB). Saya mengikuti Program Studi Independen dan membahas bidang General Competence, Software Engineering, Pemograman Frontend Dasar With JavaScript, Pemograman Frontend Lanjutan With JavaScript, JSON pada Basic Frontend Web Development, Dasar Pemograman Frontend dengan React, dan Dasar Pemograman Frontend dengan React Lanjutan. Kerja praktik yang saya lakukan adalah rekayasa perangkat lunak aplikasi Library App menggunakan JSON JavaScript (JS) untuk merekam data buku. Latar belakang pengembangan aplikasi ini muncul dari kebutuhan akan platform yang efisien dan interaktif untuk mengelola dan merekam data buku di lingkungan PT Ruang Raya Indonesia. Permasalahan yang dihadapi melibatkan ketidakefisienan dalam proses manajemen data buku, ketidakmudahan pengguna dalam mengakses dan menyimpan informasi, serta kurangnya keterlibatan pengguna akhir dalam proses pencatatan data buku. Dalam rangka mengatasi permasalahan tersebut, penerapan JSON sebagai teknologi utama pengembangan aplikasi dianggap sebagai solusi yang tepat. Solusi yang diusulkan melibatkan desain antarmuka yang intuitif, kemampuan respon yang cepat, dan kemudahan penggunaan dalam mengelola dan merekam data buku. Melalui implementasi JSON, aplikasi Library App dapat memberikan pengalaman pengguna yang lebih dinamis dan efektif, meningkatkan efisiensi dalam manajemen data buku, serta memberikan keterlibatan yang lebih besar kepada pengguna. Kesimpulan dari kerja praktik ini menunjukkan bahwa penggunaan JSON dalam pengembangan Library App memberikan hasil yang positif dalam meningkatkan kinerja dan fungsionalitas aplikasi. Dengan demikian, solusi ini dapat dijadikan model untuk pengembangan aplikasi serupa dalam lingkungan pendidikan.

Kata Kunci: Antarmuka Pengguna, Efisiensi Sistem, JSON, Library App, Manajemen Data Buku

KATA PENGANTAR

Puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa atas berkat dan rahmat serta karunia-Nya, penulis dapat mengikuti kegiatan Studi Independen Bersertifikat Program Frontend Engineering Bootcamp di PT Ruang Raya Indonesia (Ruang Guru). Laporan Kerja Praktik (KP) diajukan sebagai salah satu persyaratan menyelesaikan kurikulum Program Studi Teknik Informatika Fakultas Teknologi Informasi Universitas Bale Bandung (FTI UNIBBA).

Penulis menyampaikan terima kasih sebesar-besarnya kepada pihak-pihak yang telah ikut serta membantu dalam mengerjakan KP:

1. Rektor Universitas Bale Bandung Dr. Ir. H. Ibrahim Danuwikarsa, M.S.
2. Warek I Universitas Bale Bandung Dr. Diana Silaswati, Dra., M.Pd.
3. Dekan Fakultas Teknologi Informasi dan Dosen Pembimbing Kerja Praktik Yudi Herdiana, S.T., M.T.
4. Ketua Program Studi Teknik Informatika Yusuf Muharam, S.Kom., M.Kom.
5. Pembimbing pada PT Ruang Raya Indonesia Pratama Agung Sumirat S.T. yang telah memberikan waktu, ilmu, serta arahannya mengenai materi pada saat kegiatan KP berlangsung.
6. Rekan – rekan seperjuangan baik di PT Ruang Raya Indonesia, maupun di Program Studi Teknik Informatika FTI UNIBBA
7. Orang tua yang sudah memberikan semangat serta dukungannya kepada penulis.

Bandung, 11 Januari 2024

Teja Kusumah

301200033

DAFTAR ISI

BAB I PENDAHULUAN	1
I.1 Latar belakang	1
I.2 Lingkup	2
I.3 Tujuan.....	3
BAB II LINGKUNGAN KERJA PRAKTIK	4
II.1 Struktur Organisasi.....	4
II.2 Lingkup Pekerjaan Organisasi.....	6
II.3 Deskripsi Pekerjaan	7
II.4 Jadwal Kerja	8
BAB III TEORI PENUNJANG KERJA PRAKTIK	9
III.1 Teori Penunjang	9
III.2 Peralatan Pengembangan Aplikasi	17
BAB IV PELAKSANAAN KERJA PRAKTIK	36
IV.1 Input.....	36
IV.2 Proses.....	38
IV.3 Pencapaian Hasil	48
BAB V PENUTUP.....	56
V.1 Kesimpulan dan Saran Mengenai Pelaksanaan	56
V.1.1 Kesimpulan Pelaksanaan Kerja Praktik	56
V.1.2 Saran Pelaksanaan Kerja Praktik.....	57
V.2 Kesimpulan dan Saran Mengenai Substansi	58
V.2.1 Kesimpulan	58
V.2.2 Saran.....	59

DAFTAR TABEL

Tabel III 1 Use Case Diagram.....	12
Tabel III 2 Activity Diagram.....	13
Tabel III 3 Class Diagram	14
Tabel III 4 Sequence Diagram	16
Tabel IV. 1 Kebutuhan Perangkat Keras.....	37
Tabel IV. 2 Minimum kebutuhan perangkat keras.....	38
Tabel IV. 3 Kebutuhan perangkat lunak	38
Tabel IV. 4 Database di JSON	48

DAFTAR GAMBAR

Gambar II. 1 Struktur Organisasi PT. Ruang Raya Indonesia (Ruang Guru).....	4
Gambar IV 1 Use Case Diagram.....	42
Gambar IV 2 Activity Diagram Library App.....	44
Gambar IV 3 Activity Diagram Beranda	46
Gambar IV 4 Class Diagram	47
Gambar IV 5 Tampilan Beranda Library App sebelum ada data buku.....	49
Gambar IV 6 Tampilan pada fitur tambah buku sebelum diisi data	50
Gambar IV 7 Tampilan pada fitur tambah buku sesudah diisi data.....	51
Gambar IV 8 Tampilan pada beranda Library App sesudah diisi data buku	52
Gambar IV 9 Tampilan pada fitur edit data buku	53
Gambar IV 10 Tampilan fitur hapus data buku sebelum data buku dihapus	54
Gambar IV 11 Tampilan fitur hapus data buku sesudah data buku dihapus.....	55

BAB I

PENDAHULUAN

I.1 Latar belakang

Perpustakaan digital adalah aplikasi yang digunakan untuk mengelola data buku di perpustakaan. Aplikasi ini biasanya menyediakan berbagai fitur, seperti pencarian buku, peminjaman buku, pengembalian buku, daftar koleksi buku, dan pemesanan buku. Perpustakaan digital memiliki banyak manfaat, antara lain meningkatkan efisiensi pengelolaan data buku, meningkatkan kemudahan akses data buku bagi pengguna, dan meningkatkan kenyamanan pengguna dalam mengelola data buku.

Menurut penelitian yang dilakukan oleh Nielsen Norman Group, 79% pengguna internet akan meninggalkan situs web jika mereka tidak menemukan apa yang mereka cari dalam 10 detik pertama. Hal ini menunjukkan bahwa pengalaman pengguna adalah faktor yang sangat penting dalam keberhasilan sebuah situs web. Dalam hal perpustakaan digital, pengalaman pengguna yang baik dapat meningkatkan kepuasan pengguna dan mendorong mereka untuk menggunakan perpustakaan tersebut secara lebih sering. Perpustakaan digital yang berkualitas tinggi akan memudahkan pengguna untuk menemukan informasi yang mereka butuhkan, menggunakan fitur-fitur perpustakaan, dan memberikan umpan balik.

Saat ini, data buku di PT Ruang Raya Indonesia dikelola menggunakan sistem yang bersifat manual. Sistem ini memiliki beberapa kekurangan, antara lain sulit untuk diakses oleh pengguna dari jarak jauh, sulit untuk diintegrasikan dengan sistem lain, dan sulit untuk dianalisis. Kekurangan-kekurangan tersebut dapat menyebabkan beberapa permasalahan, antara lain pengguna kesulitan untuk mengakses data buku dari jarak jauh, data buku sulit untuk diintegrasikan dengan sistem lain, dan data buku sulit untuk dianalisis untuk keperluan pengambilan keputusan.

Untuk mengatasi permasalahan-permasalahan tersebut, PT Ruang Raya Indonesia berencana untuk mengembangkan aplikasi perpustakaan digital baru yang menggunakan JSON untuk merekam data buku. JSON (JavaScript Object Notation) adalah sebuah format data yang bersifat ringan dan mudah dibaca oleh manusia maupun mesin. Format ini sering digunakan untuk pertukaran data antar aplikasi.

Pengembangan aplikasi perpustakaan digital baru ini diharapkan dapat meningkatkan efisiensi dan efektivitas pengelolaan data buku di PT Ruang Raya Indonesia.

I.2 Lingkup

Pada PT Ruang Raya Indonesia, saya ditugaskan untuk membuat perpustakaan digital dan diposisikan pada Front End Engineering. Front End Engineering memiliki peran menciptakan situs web perpustakaan digital yang berkualitas tinggi dan aplikasi yang dinamis untuk memberikan pengalaman pengguna terbaik. Pengalaman pengguna yang baik sangat penting untuk meningkatkan daya saing sebuah produk atau layanan, khususnya dalam hal manajemen pustaka. Front End Engineering berperan penting dalam menciptakan pengalaman pengguna yang baik untuk perpustakaan digital. Front End Engineer bertanggung jawab untuk mendesain dan mengembangkan situs web dan aplikasi perpustakaan yang responsif, mudah digunakan, dan menarik secara visual.. Berikut ini ruang lingkup dalam membangun aplikasi library app diantaranya:

1. General Competence
2. Software Engineering
3. Pemograman Frontend Dasar With JavaScript
4. Pemograman Frontend Lanjutan With JavaScript
5. JSON pada Basic Frontend Web Development

Namun, pada laporan ini penulis hanya akan menjelaskan salah satu dari final project diatas yaitu fokus JSON pada Dasar Pemograman Frontend. Dalam pengembangan aplikasi perpustakaan digital baru, JSON digunakan untuk

menyimpan data buku. Hal ini karena JSON memiliki beberapa kelebihan, antara lain: 1) Mudah untuk diakses oleh pengguna dari jarak jauh. Data buku yang direkam dalam format JSON dapat diakses dengan mudah menggunakan berbagai perangkat, baik komputer, laptop, maupun smartphone, 2) Mudah untuk diintegrasikan dengan sistem lain. Format JSON adalah format data yang umum digunakan, sehingga dapat diintegrasikan dengan mudah dengan sistem lain, seperti sistem pembayaran dan sistem manajemen pelanggan. 3) Mudah untuk dianalisis. Data buku yang direkam dalam format JSON dapat dianalisis dengan mudah menggunakan berbagai perangkat lunak analisis data.

I.3 Tujuan

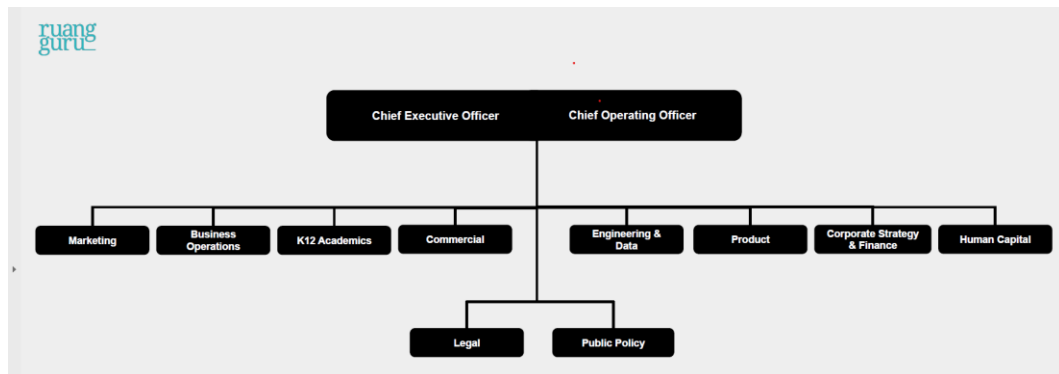
Tujuan kerja praktik diharapkan peserta dapat membangun aplikasi website perpustakaan digital yang interaktif dengan menggunakan teknologi terbaru yang banyak digunakan di pengembangan web, dan pengembangan perangkat lunak sebagai Front-End Engineer.

1. Memudahkan mengakses data buku dari jarak jauh
2. Memudahkan mengintegrasikan dengan sistem lain
3. Memudahkan analisis data buku untuk keperluan pengambilan keputusan.

BAB II

LINGKUNGAN KERJA PRAKTIK

II.1 Struktur Organisasi



Gambar II. 1 Struktur Organisasi PT. Ruang Raya Indonesia (Ruang Guru)

PT RUANG RAYA INDONESIA (Ruang Guru) adalah perusahaan terbatas yang bergerak di bidang pendidikan nonformal yang didirikan berdasarkan hukum yang berlaku di Indonesia serta telah memperoleh Izin Pendirian Satuan Pendidikan Nonformal dan Izin Operasional Lembaga Kursus Pelatihan dengan Nomor 3/A.5a/31.74.01/-1.851.332/2018.

Ruang Guru merupakan perusahaan teknologi terbesar di Indonesia yang berfokus pada layanan berbasis pendidikan. Ruang Guru telah memiliki lebih dari 22.000.000 pengguna serta mengelola 300.000 guru yang menawarkan jasa di lebih dari 100 bidang pelajaran. Ruang Guru mengembangkan berbagai layanan belajar berbasis teknologi, termasuk layanan kelas virtual, platform ujian online, video belajar berlangganan, marketplace les privat, serta konten-konten pendidikan lainnya yang bisa diakses melalui web dan aplikasi Ruang Guru. Ruang Guru juga telah dipercaya untuk bermitra dengan 32 (dari 34) Pemerintah Provinsi dan 326 Pemerintah Kota dan Kabupaten di Indonesia.

Ruang Guru juga telah memenangkan sejumlah penghargaan di dalam dan luar negeri, termasuk Solver of MIT, Atlassian Prize, UNICEF Innovation to Watch, Google Launchpad Accelerator, dan ITU Global Industry Award.

Perusahaan ini didirikan sejak tahun 2014 oleh Belva Devara dan Iman Usman, yang keduanya berhasil masuk dalam jajaran pengusaha sukses di bawah 30 tahun melalui Forbes 30 under 30 untuk sektor teknologi konsumen di Asia. Di tahun 2019, Peserta mendapat penghargaan sebagai Emerging Entrepreneur dari Ernst & Young.

Ruang Guru memiliki misi untuk menyediakan dan memperluas akses terhadap pendidikan berkualitas melalui teknologi untuk semua siswa, kapan saja dan di mana saja. Ruang Guru percaya bahwa pendidikan adalah hak setiap manusia. Ruang Guru juga meyakini bahwa pendidikan adalah tiket untuk masa depan yang lebih baik. Maka dari itu, Ruang Guru bertujuan untuk menyediakan layanan pendidikan dan materi pembelajaran dari guru-guru terbaik Indonesia, yang bisa diakses oleh seluruh siswa di mana saja Peserta berada dengan biaya yang terjangkau.

Ruang Guru juga bertujuan untuk meningkatkan kualitas guru dengan menciptakan lapangan pekerjaan dan tambahan penghasilan bagi guru di Indonesia. Ruang Guru percaya, dengan meningkatnya kualitas guru, maka mutu pendidikan di Indonesia juga akan menjadi lebih baik. Ruang Guru juga percaya bahwa teknologi memudahkan siswa untuk mengakses materi pembelajaran yang berkualitas. Di mana pun Peserta berada, Ruang Guru dapat membantu proses belajar siswa tanpa batasan ruang dan waktu. Ruang Guru juga meyakini bahwa teknologi dapat membantu siswa, guru, dan orang tua untuk menjalankan aktivitasnya menjadi lebih efektif dan efisien.

Ruang Guru CAMP merupakan program studi independent kampus merdeka yang dikembakan oleh Ruang Guru. CAMP memiliki Learning Management Sistem (LMS) berbasis mobile app dan desktop untuk kemudahan belajar bagi mahasiswa. Adapun struktur organisasi dari PT. Ruang Raya Indonesia (Ruang Guru) mencerminkan kerangka yang mencakup berbagai tingkatan dan fungsi dalam perusahaan serta program studi independen berada di bawah divisi Business Operations dari Ruang Guru.

II.2 Lingkup Pekerjaan Organisasi

Dalam program Studi Independen Bersertifikat di PT Ruang Raya Indonesia (Ruang Guru) saya sebagai peserta (mentee) melakukan pembelajaran synchronus dan asynchronus didampingi oleh seorang mentor untuk membimbing kami selama kegiatan berlangsung. Kegiatan Studi Independen ini berlangsung pada tanggal 16 Februari 2023 sampai dengan 30 Juni 2023 selama 4 bulan. Tanggung jawab saya dalam pekerjaan ini adalah mempelajari dan menyelesaikan tugas dari materi yang diajarkan sesuai dengan lingkup dari divisi Business Operations antara lain:

1. General Competencies, mempelajari Communication and Presentations Skill, Analytical & Strategic Thinking Skill.
2. Software Engineering, mempelajari penerapan proses pengembangan software dengan menggunakan VSCode, Terminal, dan Version Control. Serta memahami metode dan tools Project Management;
3. Pemrograman Frontend dasar with Javascript, mempelajari konsep konsep dasar pemrograman pada 6ontro pemrograman Javascript, mulai dari nodeJS environment, pengenalan 6ontrol6, struktur 6ontrol, function, struktur data array, object, array of object dan modules;
4. Pemrograman Frontend Lanjutan with Javascript, mempelajari konsep lanjutan yang diperlukan oleh Frontend Developer, mulai dari OOP, asynchronous JS hingga REST API;
5. Basic Frontend Web Development, mempelajari cara penulisan HTML, penggunaan CSS, memanipulasi Document Object Model dalam Web Development dasar serta memanfaatkan object-object pada browser (Browser API) khususnya Web Storage API;
6. Dasar pemrograman Frontend dengan React, mempelajari fundamental library ReactJS (JSX, Component, Props, State, LifeCycle, Handling Event) dan menggunakan ReactJS untuk membuat Single Page Application;

7. Dasar pemrograman Frontend dengan React lanjutan, mempelajari cara menggunakan State Management dan Chakra UI sebagai komponen untuk styling dan melakukan deployment aplikasi ReactJS yang sudah dibuat sehingga bisa diakses online dengan menggunakan layanan Netlify and Firebase.

II.3 Deskripsi Pekerjaan

Dalam rangka kegiatan kerja praktik di PT Ruang Raya Indonesia, saya diberikan tanggung jawab untuk mengembangkan aplikasi bernama Library App menggunakan JSON untuk merekam data buku. Fokus proyek ini adalah merancang sebuah website perpustakaan digital yang bertujuan meningkatkan efisiensi pengelolaan informasi buku. Data buku yang dibutuhkan pada library app ialah judul buku, informasi penerbit, tahun terbit, dan jumlah buku yang dimiliki. Saya merancang sistem dengan penekanan pada penggunaan JSON untuk penyimpanan data buku, dan melanjutkan dengan tahap pengembangan aplikasi.

Pada tahap implementasi teknis, sebuah skeleton code client telah disediakan dengan beberapa function yang perlu diimplementasikan, termasuk operasi CRUD seperti menambahkan, mengubah, dan menghapus data buku. Dalam proses pengembangan, client akan berjalan pada `http://localhost:3000`, dan aplikasi ini memiliki beberapa persyaratan umum seperti menampilkan data buku dari database ke dalam tabel saat halaman web di-render pertama kali. Tombol "Add Book" akan memicu perubahan tampilan menjadi formulir untuk menambahkan data buku, dan data yang ditambahkan akan langsung ditampilkan secara real-time tanpa perlu melakukan refresh halaman. Sistem juga mendukung operasi pengeditan dan penghapusan data buku, dengan kemampuan tampilan dan penyimpanan data yang diperbarui secara otomatis. Akhirnya, aplikasi ini harus di-deploy ke Netlify, dengan link deploy dijadikan variabel `NetlifyDeployUrl` pada file `deployData.js`, disertai dengan mencantumkan nama dan CAMPID pada file tersebut seperti contoh yang telah diberikan.

II.4 Jadwal Kerja

PT Ruang Raya Indonesia (Ruang Guru) menerapkan pendekatan pembelajaran asinkron dalam menjalankan program MSIB. Peserta memiliki fleksibilitas membaca materi di Learning Management Sistem (LMS) sebelum mengerjakan latihan atau proyek, dengan pre-test sebelum mempelajari materi dan post-test sebagai evaluasi kemampuan. Dalam setiap kelompok "house," peserta dibimbing oleh seorang mentor melalui sesi mentoring mingguan, yang mencakup pembahasan materi dan bantuan implementasi. Sesi Live QnA juga diadakan untuk mendukung peserta dalam mengerjakan latihan, tugas, dan proyek.

Program Frontend Engineering di Ruang Guru terdiri dari tujuh materi yang melibatkan berbagai kompetensi dalam pengembangan frontend. Materi tersebut mencakup General Competence, Software Engineering, Pemrograman Frontend Dasar hingga Pemrograman Frontend Lanjutan dengan Javascript dan React. Setiap materi melibatkan tugas-tugas seperti exercise, assignment, dan final project yang dikerjakan secara individu dengan batas waktu pengumpulan. Peserta juga mendapatkan akses ke Neurons, platform pembelajaran milik Ruang Guru, yang menyediakan materi dalam berbagai format. Program ini bertujuan untuk mempersiapkan peserta menjadi frontend engineer yang kompeten melalui materi yang disampaikan, tugas-tugas praktis, dan dukungan pembelajaran dari mentor dan sesi Live QnA.

BAB III

TEORI PENUNJANG KERJA PRAKTIK

III.1 Teori Penunjang

Selama pelaksanaan kerja praktik di PT Ruang Raya Indonesia saya Menggunakan pengetahuan yang saya peroleh dari kampus selama masa perkuliahan sebagai landasan teori frontend engineering di PT Ruang Raya Indonesia. Pengetahuan dan teori yang digunakan antara lain sebagai berikut:

1. Teori tentang pengenalan pemrograman

Pada mata kuliah TIF301 Algoritma dan Pemrograman memberikan pemahaman dasar tentang langkah-langkah pembuatan program, pedoman desain aplikasi, dan pentingnya algoritma pemrograman. Mahasiswa mempelajari prinsip-prinsip kunci untuk merancang aplikasi yang efektif, dengan penekanan pada logika dan implementasi algoritma.

2. Teori Struktur Data

Mata kuliah TIF304 Struktur Data membahas konsep dan implementasi struktur data dalam pemrograman. Mahasiswa belajar tentang berbagai jenis struktur data seperti array, linked list, stack, dan queue. Penekanan diberikan pada pemahaman efisiensi dan kegunaan setiap struktur data dalam memecahkan masalah pemrograman. Materi ini memberikan dasar yang kuat bagi mahasiswa untuk mengembangkan pemahaman tentang bagaimana menyusun dan mengelola data secara efektif dalam konteks pengembangan perangkat lunak.

3. Teori Interaksi Manusia dan Komputer

Teori Interaksi Manusia dan Komputer mempelajari bagaimana pengguna berinteraksi dengan sistem komputer. Teori ini membantu dalam merancang antarmuka pengguna yang intuitif, mudah digunakan, dan responsif. Dengan memahami prinsip-prinsip desain antarmuka yang baik, Anda dapat meningkatkan pengalaman pengguna dan efisiensi dalam penggunaan aplikasi. diperoleh pada mata kuliah FTI307 interaksi manusia dan komputer

4. Teori Basis Data

Mata kuliah TIF310 Basis Data memperkenalkan mahasiswa pada konsep basis data dan sistem manajemen basis data (DBMS). Mahasiswa memahami desain struktur data yang optimal untuk menyimpan dan mengelola informasi dalam database. Selain itu, mereka mempelajari bahasa SQL untuk melakukan manipulasi data dan query. Pemahaman ini penting untuk merancang dan mengelola basis data dengan efisien dalam proyek pengembangan perangkat lunak.

5. Teori Rekayasa Perangkat Lunak

Mata kuliah TIF316 Rekayasa Perangkat Lunak membahas proses pengembangan perangkat lunak dari perencanaan hingga implementasi. Mahasiswa mempelajari metodologi pengembangan perangkat lunak, analisis kebutuhan, perancangan sistem, dan uji coba. Teori ini memberikan landasan bagi mahasiswa untuk memahami siklus hidup pengembangan perangkat lunak dan menerapkannya dalam proyek-proyek pengembangan perangkat lunak yang kompleks.

6. Teori Pemrograman Internet

Mata kuliah TIF319 Pemrograman Internet membahas konsep dan teknologi terkini dalam pengembangan aplikasi web dan layanan online. Mahasiswa memahami dasar-dasar pemrograman web, protokol internet, dan integrasi database. Pemahaman ini penting dalam mengembangkan aplikasi web yang responsif dan efisien, serta berkontribusi pada kemampuan mahasiswa dalam merancang dan mengimplementasikan solusi di ranah internet.

7. Teori Manajemen Proyek

Teori Manajemen Proyek mencakup prinsip-prinsip, metodologi, dan alat-alat untuk mengelola proyek secara efektif. Dalam kerja praktik pembuatan website yang responsif, teori ini membantu dalam perencanaan, pengorganisasian, dan pengendalian proyek pengembangan website. Memahami tahapan pengembangan, alokasi sumber daya, dan manajemen risiko akan membantu menjaga proyek tetap terjadwal dan sesuai dengan tujuan yang ditetapkan. diperoleh pada mata kuliah FTI318 Manajemen proyek perangkat lunak

8. Teori Pendukung

a. Unified Modeling Language (UML)

Unified Modeling Language (UML) adalah sebuah bahasa grafis yang memiliki tujuan untuk menggambarkan, merancang, dan mendokumentasikan sistem perangkat lunak. Dalam pengembangan perangkat lunak, UML memberikan kumpulan notasi dan teknik yang dapat digunakan untuk mengilustrasikan berbagai aspek dari sistem yang sedang dibangun. Aspek-aspek tersebut mencakup struktur, fungsi, dan interaksi antar komponen sistem.

UML menjadi sangat berguna pada tahap perencanaan dan pengembangan sistem perangkat lunak. Para pengembang perangkat lunak dapat memanfaatkan UML untuk membuat diagram yang memberikan gambaran visual tentang struktur kelas, relasi antar kelas, alur proses, dan tingkah laku sistem secara keseluruhan. Dengan demikian, UML membantu para pengembang dalam mengkomunikasikan ide, merancang sistem, dan memahami kompleksitas dari suatu proyek perangkat lunak.

Penggunaan UML tidak hanya terbatas pada aspek fungsional sistem. UML juga memberikan dukungan untuk pemodelan aspek non-fungsional, seperti keamanan dan skalabilitas. Hal ini memungkinkan para pengembang untuk memperhitungkan dan merancang tidak hanya bagaimana sistem berfungsi, tetapi juga bagaimana sistem dapat mencapai tingkat keamanan yang diinginkan dan dapat berkembang secara efisien seiring waktu.

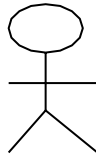



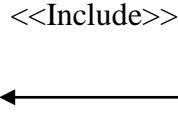
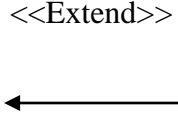
1. Use Case Diagram

Use Case Diagram merupakan jenis diagram dalam pemodelan perangkat lunak yang dirancang untuk menggambarkan fungsionalitas yang diinginkan dari suatu sistem. Fungsionalitas ini dilihat dari sudut pandang pengguna atau aktor yang berinteraksi dengan sistem tersebut. Diagram ini memberikan gambaran visual tentang cara pengguna atau aktor eksternal berinteraksi dengan sistem untuk mencapai tujuan tertentu.

Dalam Use Case Diagram, elemen-elemen utama yang digambarkan meliputi aktor, use case, dan hubungan antara keduanya.

Aktor adalah entitas luar yang berinteraksi dengan sistem, seperti pengguna, sistem eksternal, atau entitas lain yang dapat memengaruhi atau menerima informasi dari sistem. Use case, di sisi lain, merepresentasikan fungsi atau tindakan yang dapat dilakukan oleh aktor dalam interaksi dengan sistem.

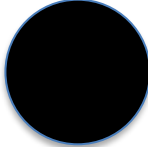

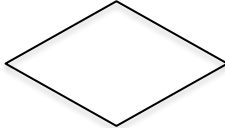

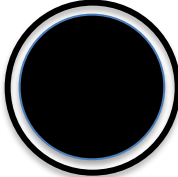
Tabel III 1 Use Case Diagram

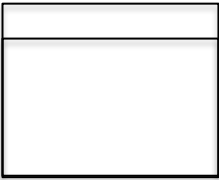
No	Simbol	Keterangan
1.		Aktor : Mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan use case
2.		Use Case : Abstraksi dan interaksi antara sistem dan actor
3.		Association : Abstraksi dari penghubung antara aktor dan use case
4.		Generalisasi :Menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan use Case
5.		Menunjukkan bahwa suatu use case seluruhnya merupakan fungsionalitas dari use case lainnya.
6.		Menunjukkan bahwa suatu use case merupakan tambahan fungsional dari use case lainnya jika suatu kondisi terpenuhi.

2. Activity Diagram

Diagram aktivitas merupakan bentuk visualisasi yang digunakan dalam pemodelan perangkat lunak untuk menggambarkan alur atau urutan aktivitas dalam suatu proses atau sistem.

Tabel III 2 Activity Diagram


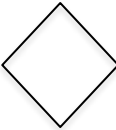
No	Simbol	Nama	Keterangan
1.		Status awal	Sebuah diagram aktivitas memiliki kondisi awal.
2.		Aktivitas	Aktivitas yang dilakukan sistem biasanya diawali dengan kata kerja.
3.		Percabangan/ Descisiom	Percabangan dimana ada pilihan aktivitas lebih dari satu.
4.		Penggabungan/ Join	Penggabungan dimana yang lebih dari satu aktivitas lalu digabungkan jadi satu.
5.		Status Akhir	Status akhir yang dilakukan sistem, sebuah diagram aktivitas.

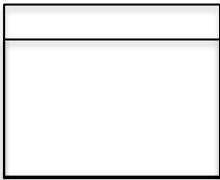




6.		Swimlane	Swimlane memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.
----	---	----------	---

3. Class Diagram

Diagram ini memberikan gambaran struktur sistem, menggambarkan kelas-kelas yang ada, interaksi di antara mereka, atribut-atribut (data) yang dimiliki oleh kelas, dan metode-metode (fungsi) yang terkait. Dengan menyajikan informasi ini secara visual, diagram kelas membantu pemahaman mendalam tentang organisasi sistem dan bagaimana kelas-kelas berkomunikasi satu sama lain. Atribut mencerminkan data yang dimiliki oleh kelas, sementara metode mencakup fungsi atau operasi yang dapat dilakukan oleh kelas tersebut. Diagram kelas sering digunakan dalam pemodelan sistem berbasis objek untuk merancang dan menggambarkan struktur yang kompleks dengan cara yang lebih mudah dipahami.

Tabel III 3 Class Diagram

No	Simbol	Nama	Keterangan
1.		Generalization	Hubungan dimana objek anak (descendent) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (ancestor)
2.		Navy Association	Upaya untuk menghindari

			asosiasi dengan lebih dari 2 objek
3.		Class	Himpunan dari objek-objek yang Berbagi atribut serta operasi yang sama.
4.		Collaboration	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu Aktor
5.		Realization	Operasi yang benar-benar dilakukan oleh suatu objek
6.		Dependency	Hubungan antara dua kelas di mana satu kelas tergantung pada kelas lain dalam konteks tertentu
7.		Association	Apa yang menghubungkan antara objek satu dengan objek lainnya

4. Sequence Diagram

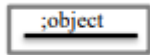
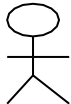



Sequence Diagram digunakan sebagai alat pemodelan perangkat lunak untuk merinci urutan interaksi antara objek-objek dalam suatu skenario atau proses. Diagram ini menyajikan gambaran visual tentang



bagaimana objek-objek berkomunikasi dan berinteraksi sepanjang waktu dalam konteks tertentu.

Dalam sebuah Sequence Diagram, setiap objek direpresentasikan sebagai garis vertikal, dan pesan-pesan yang dikirim antar objek diilustrasikan dengan panah dan urutan waktu. Ini memungkinkan pemahaman yang jelas tentang hubungan sebab-akibat antar elemen sistem, membantu pengembang untuk memahami logika operasional sistem secara rinci.

Sequence Diagram tidak hanya memetakan interaksi objek-objek, tetapi juga membantu dalam mengidentifikasi titik-titik penting seperti penciptaan dan penghancuran objek, serta urutan pesan yang dikirim antar mereka. Oleh karena itu, Sequence Diagram menjadi instrumen yang sangat efektif dalam merencanakan, memahami, dan mengkomunikasikan logika alur kerja dan interaksi objek dalam suatu perangkat lunak.

Tabel III 4 Sequence Diagram

No	Simbol	Nama	Keterangan
1.		Object	Komponen utama Sequence Diagram
2.		Actor	Menggambarkan orang yang sedang berinteraksi dengan sistem
3.		Entity Class	Menggambarkan hubungan kegiatan yang dilakukan
4.		Boundary Class	Menggambarkan sebuah penggambaran dari form
5.		Control Class	Menggambarkan penghubung antara boundary dengan tabel

6.		Life Line	Menggambarkan tempat mulai dan berakhirnya sebuah message
7.		Message	Menggambarkan pengiriman pesan

III.2 Peralatan Pengembangan Aplikasi

Peralatan atau tools yang digunakan dalam pembuatan aplikasi *Library App* ini diantaranya:

A. Software

1. Visual Studio Code

Visual Studio Code (VS Code) adalah editor source code buatan Microsoft yang beroperasi pada komputer desktop dan kompatibel dengan sistem operasi Windows, macOS, dan Linux. VS Code cocok untuk JavaScript, TypeScript, dan Node.js. Selain itu, editor ini memiliki ekosistem extension yang luas yang dapat mendukung berbagai bahasa pemrograman dan runtime lainnya, termasuk C++, C#, Java, Python, PHP, Go, dan .NET. Karena serbaguna dan memiliki extension yang beragam, Visual Studio Code menjadi pilihan populer di kalangan developer untuk beragam tugas pemrograman.

Fitur penting VS Code di antaranya adalah:

a. IntelliSense

IntelliSense adalah fitur cerdas yang menyediakan saran kode dengan mempertimbangkan konteks saat mengetik. Hal ini memungkinkan pengembang untuk menulis kode dengan lebih cepat dan mengurangi kemungkinan kesalahan karena saran kode yang relevan dan otomatis muncul.

b. Run and Debug

VS Code dilengkapi dengan fitur bawaan untuk menjalankan dan melakukan debugging kode untuk berbagai bahasa pemrograman.

Pengembang dapat melakukan setting breakpoint, memeriksa variabel, dan menelusuri kode, membantu mereka mengidentifikasi serta memperbaiki masalah dengan efektif.

c. Built-in Git

Visual Studio Code memiliki integrasi dengan Git sehingga developer dapat mengelola version control secara langsung di dalam editor. Dengan kata lain, fitur ini memungkinkan untuk mengatur dan membuat perubahan ke repositori tanpa beralih ke Git client yang terpisah.

d. Extensions

Visual Studio Code memiliki library extension yang sangat banyak tersedia di marketplace-nya. Extension ini dapat menambahkan dukungan untuk berbagai bahasa pemrograman, framework, dan alat lainnya sehingga developer dapat menyesuaikan editor dengan kebutuhan spesifiknya.

e. Tema yang dapat disesuaikan

Dengan beragam tema yang tersedia, pengguna dapat mengubah tampilan dan skema warna editor agar sesuai dengan preferensi mereka. Ini tidak hanya memberikan kebebasan estetika tetapi juga membantu mengurangi ketegangan mata selama sesi coding yang berkepanjangan. Dengan demikian, fitur-fitur ini menjadikan Visual Studio Code sebagai pilihan yang powerful dan fleksibel untuk kebutuhan pengembangan perangkat lunak.

Berikut merupakan kelebihan Visual Studio Code:

a. Gratis dan open-source

Visual Studio Code menonjol sebagai editor yang tidak memerlukan biaya lisensi dan bersifat open-source. Ketersediaan ini memungkinkan akses tanpa hambatan biaya bagi para pengembang, memberikan keleluasaan untuk digunakan dan dimodifikasi sesuai kebutuhan.

b. Ringan dan cepat

Desain ringan dari VS Code memastikan pengalaman pengguna yang responsif. Kecepatan loading yang tinggi dan performa yang optimal membuatnya dapat berjalan lancar bahkan pada perangkat dengan sumber daya terbatas, meningkatkan efisiensi pengembangan.

c. Dukungan bahasa yang luas

Kelebihan ini terletak pada kemampuan VS Code untuk menyediakan dukungan yang luas terhadap berbagai bahasa pemrograman. Ini memungkinkan pengembang bekerja dengan teknologi berbeda tanpa harus berganti editor.

d. Otomatisasi tugas

VS Code dilengkapi dengan task runner bawaan yang memungkinkan otomatisasi tugas berulang. Kemampuan untuk mengotomatiskan proses seperti pembangunan, pengujian, dan implementasi proyek meningkatkan produktivitas pengembang secara keseluruhan.

e. Mampu lintas platform

Visual Studio Code dirancang untuk mendukung lintas platform, tersedia di Windows, macOS, dan Linux. Konsistensi antar platform memastikan pengalaman yang seragam bagi pengembang, tidak tergantung pada sistem operasi yang digunakan. Ini memberikan fleksibilitas bagi tim pengembang yang mungkin menggunakan berbagai lingkungan kerja.

Cara menggunakan Visual Studio Code (VS Code) dimulai dengan langkah-langkah berikut:

a. Download dan Install VS Code:

Unduh dan instal VS Code dari situs web resmi, yaitu <https://code.visualstudio.com/>](<https://code.visualstudio.com/>) sesuai dengan sistem operasi yang Anda gunakan.

b. Mulai Coding:

Buka VS Code dan pilih folder atau file yang relevan untuk memulai proses coding Anda.

c. Area Editor:

Tulis atau edit kode Anda di area editor, dengan dukungan fitur seperti IntelliSense untuk memudahkan penulisan kode.

d. Enhance dengan Extension:

Tingkatkan fungsionalitas VS Code dengan menginstal extension yang sesuai dengan kebutuhan Anda.

e. Gunakan Terminal Terintegrasi:

Manfaatkan terminal terintegrasi untuk menjalankan dan men-debug kode Anda langsung dari VS Code.

f. Kelola Version Control:

Mengelola version control menjadi lebih mudah dengan fitur Git bawaan di dalam VS Code.

g. Personalisasi:

Sesuaikan VS Code dengan preferensi Anda, termasuk pilihan tema dan pengaturan lainnya.

h. Kolaborasi dengan Live Share:

Jika diperlukan, Anda dapat melakukan kolaborasi secara real-time dengan orang lain menggunakan fitur Live Share.

i. Simpan Perubahan dan Tutup VS Code:

Pastikan untuk menyimpan perubahan yang telah Anda buat, dan tutup VS Code ketika Anda telah selesai dengan sesi coding Anda.

2. CSS (*Cascading Style Sheets*)

CSS adalah bahasa Cascading Style Sheet dan biasanya digunakan untuk mengatur tampilan elemen yang tertulis dalam bahasa markup, seperti HTML (Hypertext Markup Language) . CSS berfungsi untuk memisahkan konten dari tampilan visualnya di situs. CSS dibuat dan dikembangkan oleh W3C (World Wide Web Consortium) pada tahun

1996 untuk alasan yang sederhana. Dulu HTML tidak dilengkapi dengan tags yang berfungsi untuk memformat halaman. Anda hanya perlu menulis markup untuk situs.

Hubungan antara CSS dengan HTML yaitu HTML (Hypertext Markup Language) adalah bahasa pemrograman yang mendominasi dalam pembuatan situs web. Sebagai elemen dasar di balik layar tampilan halaman web, HTML memainkan peran krusial dalam menyajikan konten yang dapat diakses oleh pengguna sehari-hari. Tanpa HTML, struktur dan representasi visual dari sebuah situs web akan menjadi kacau. Selanjutnya, hubungan antara HTML dan CSS menjadi krusial dalam pengembangan web. CSS, atau Cascading Style Sheets, merupakan kumpulan kode yang digunakan untuk mendesain halaman HTML. Untuk memberikan analogi sederhana, HTML bisa diibaratkan sebagai handphone, sedangkan CSS berperan sebagai casing yang menambahkan estetika pada handphone tersebut.

CSS memiliki peran signifikan dalam memberdayakan para desainer web untuk mengubah dan memperindah elemen-elemen dalam halaman HTML. Misalnya, CSS memungkinkan desainer untuk mengatur teks, gambar, latar belakang, dan elemen-elemen lainnya. Sebuah contoh praktis adalah penggunaan CSS untuk mengubah warna teks. Dengan memberikan perintah pada tag HTML, seperti `` dengan atribut `class="nama-warna"`, desainer dapat memberikan warna spesifik pada teks dan menghasilkan tampilan yang menarik secara visual.

Berikut beberapa fungsi CSS lainnya yang baik untuk website, antara lain:

a. Proses loading halaman web lebih cepat

Satu rangkaian kode CSS dapat diaplikasikan untuk beberapa halaman website sekaligus sehingga jumlah kode dapat diminimalkan. Hal ini tentunya akan membuat beban loading

website menjadi lebih kecil. Alhasil, kecepatan saat loading website pun akan menjadi lebih cepat.

b. Beragam variasi

Seperti yang kita ketahui, HTML memang dapat digunakan untuk mengatur tampilan website. Sayangnya, terbatas. Namun hal ini tidak berlaku bagi CSS yang menawarkan lebih banyak style tampilan pada halaman website.

c. Tampilan website akan lebih rapi

Dengan menggunakan CSS, tampilan halaman website pun akan rapi dan optimal di berbagai ukuran layar pengguna, baik itu desktop maupun smartphone. Sehingga pemilik website tidak perlu repot mengatur dan pusing memikirkan website tidak maksimal dibuka dari perangkat-perangkat pengunjung.

CSS dibagi menjadi beberapa macam tergantung fungsi penggunaannya, namun ada tiga macam jenis yang paling sering digunakan oleh web designer.

- a. Inline Style CSS merupakan kode yang dituliskan di dalam file HTML dan menggunakan elemen spesifik yang memuat tag `<style>`. Jenis CSS ini hanya mempengaruhi satu baris kode HTML dan harus menempel pada elemen tulisan tersebut.
- b. Internal CSS merupakan kode yang dituliskan pada bagian header file HTML. Sayangnya, jenis ini terkadang membuat loading menjadi semakin lama.
- c. External CSS merupakan kode CSS yang diletakkan di luar dokumen HTML sebagai file `.css`. External CSS memiliki fungsi untuk mengkustomisasi semua tampilan halaman website yang sudah ditentukan atau beberapa halaman sekaligus.

3. JavaScript

JavaScript adalah Bahasa pemrograman yang memungkinkan Web Developer untuk menerapkan fitur yang kompleks pada halaman web.

Contoh fiturnya ialah animasi 2D/3D, Photo Sideshow, Autocomplete text suggestion, dan masih banyak lagi. JavaScript merupakan layer ke tiga dari Standar Website, dua diantaranya ialah HTML dan CSS. JavaScript ditemukan oleh Brendan Eich pada tahun 1995. Pertama kali dikembangkan untuk Netscape 2, dan menjadi standar ECMA-262 pada tahun 1997. Setelah Netscape menyerahkan JavaScript ke ECMA, Perusahaan Raksasa Mozilla terus mengembangkan JavaScript untuk browser Firefox.

Berikut merupakan kelebihan dan kekurangan Bahasa pemrograman JavaScript

a. Kelebihan :

- 1) Ringan: JavaScript bersifat Client – Side yaitu seluruh proses halaman akan di proses oleh client . jadi server hanya perlu memproses HTTP request saja. Kebanyakan Bahasa pemrograman yang lain bersifat Sisi-Server sehingga menjadikan Bahasa pemrograman tersebut lebih berat daripada JavaScript.
- 2) Mudah Dipelajari: JavaScript merupakan salah satu Bahasa pemrograman yang paling banyak digunakan oleh web developer saat ini , dikarenakan JavaScript mudah untuk dipelajari dari pada Bahasa pemrograman yang lain.
- 3) MultiPlatform Sistem Operasi: JavaScript bisa dijalankan oleh banyak platform seperti Windows , Linux , IOS , Android , dan masih banyak lagi Sistem Operasi lainnya.
- 4) Dinamis: Dinamis merupakan salah satu keunggulan dari JavaScript. Dinamis yang dimaksud disini ialah fleksibilitas dari pengembangan aplikasi atau web yang menggunakan JavaScript.

b. Kekurangan :

- 1) Script yang digunakan tidak terenkripsi: Web atau Aplikasi yang menggunakan JavaScript memiliki tingkat enkripsi yang

rendah atau bahkan tidak terenkripsi, hal tersebut akan berakibat web atau aplikasi mudah diretas atau dibajak.

- 2) Tidak Konsisten: Tidak konsisten yang dimaksud disini ialah sering kali data dari masing-masing perangkat dirender berbeda satu sama lain.

4. Node JS

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. Node.js menjalankan V8 JavaScript engine (yang juga merupakan inti dari Google Chrome) di luar browser. Ini memungkinkan Node.js memiliki performa yang tinggi. Node.js juga menyediakan banyak library/module JavaScript yang membantu menyederhanakan pengembangan aplikasi web. Berikut ini adalah beberapa fitur penting dari Node.js yang menjadikannya pilihan utama dalam pengembangan aplikasi:

- 1) *Asynchronous & Event-driven*. Semua API dari Node.js bersifat asynchronous, artinya tidak memblokir proses lain sembari menunggu satu proses selesai. Server Node.js akan melanjutkan ke ke pemanggilan API berikutnya lalu memanfaatkan mekanisme event notification untuk mendapatkan respon dari panggilan API sebelumnya.
- 2) *Very Fast*. Eksekusi kode dengan Node.js sangat cepat karena berjalan pada V8 JavaScript Engine dari Google Chrome.
- 3) *Single Threaded but Highly Scalable*. Node.js menggunakan model single thread dengan event looping. Mekanisme ini membantu server untuk merespon secara asynchronous dan menjadikan server lebih scalable dibandingkan server tradisional yang menggunakan banyak thread untuk menangani permintaan.

Node.js dirancang untuk aplikasi dengan proses I/O yang intensif seperti network server atau backend API. Pemrograman dengan multithreading relatif lebih berat dan sulit untuk dilakukan. Jika kita ingin membuat web server yang bisa menangani ratusan request bersamaan, menggunakan ratusan thread akan membutuhkan memori yang besar. Oleh karena itu, karakteristik Node yang asynchronous dan single thread dirancang untuk memungkinkan implementasi server yang dapat menangani banyak request pada waktu yang sama.

Setelah berhasil menjadi JavaScript Runtime yang dapat mengeksekusi kode JavaScript di luar browser, Node.js seolah-olah menjadi gerbang bagi para JavaScript Developer untuk mengembangkan sistem di luar dari browser. Dengan adanya Node.js, JavaScript menjadi bahasa multiplatform yang banyak menggiring developer untuk menggunakannya. Popularitas JavaScript pun meroket! JavaScript menjadi salah satu pilihan tepat dalam membangun web server, terlebih bila Anda adalah seorang Front-End Web Developer. Anda tentu tidak perlu menggunakan bahasa yang berbeda dalam membangun Back-End. Anda bisa menjadi Full-Stack Developer dengan mempelajari satu bahasa pemrograman saja.

5. MongoDB

MongoDB adalah salah satu jenis database NoSQL (Not Only SQL) yang dirancang untuk menyimpan data dalam bentuk dokumen JavaScript object notation (JSON). Database ini terbilang cukup populer digunakan dalam pengembangan website. Tidak heran, banyak perusahaan besar yang mengandalkan MongoDB, seperti Google, Adobe, dan eBay. MongoDB juga dirancang untuk mengatasi beberapa masalah yang dihadapi oleh database relasional tradisional, seperti skalabilitas horizontal, fleksibilitas schema, dan kinerja yang lebih baik untuk aplikasi dengan volume data yang besar dan kompleks.

MongoDB menyimpan data dalam dokumen yang dapat berisi berbagai jenis data seperti string, angka, daftar, bahkan dokumen lain. Dokumen tersebut kemudian dikelompokkan ke dalam koleksi, yang setara dengan tabel dalam database relasional tradisional.

Salah satu kelebihan MongoDB adalah kemampuannya untuk skalabilitas horizontal, yang berarti dapat dengan mudah menambahkan lebih banyak server untuk meningkatkan kinerja dan kapasitas penyimpanan. Selain itu, MongoDB juga menyediakan berbagai fitur untuk manajemen data seperti index, replikasi, dan sharding.

Ada beberapa komponen penting dalam sistem database MongoDB, antara lain:

- 1) Database yang merupakan wadah dengan struktur penyimpanan yang disebut collection.
- 2) Collection yang merupakan kumpulan informasi data yang berbentuk dokumen.
- 3) Document merupakan satuan unit terkecil dalam MongoDB.

Berikut merupakan contoh data pada MongoDB:

```
{
  {
    "title": "Laskar Pelangi",
    "author": "Andrea Hirata",
    "year": "2005",
    "quantity": "3",
    "id": 1
  }
}
```

Dalam contoh di atas, dokumen ini merincikan informasi mengenai suatu buku dalam format yang sesuai untuk penyimpanan dalam database MongoDB. Judul buku adalah "Laskar Pelangi," ditulis oleh Andrea Hirata, dan diterbitkan pada tahun 2005. Terdapat tiga salinan

dari buku tersebut, yang dapat diidentifikasi melalui atribut "quantity." Setiap buku dalam koleksi memiliki identifikasi unik, dan dalam contoh ini, ID buku adalah 1. Struktur data ini mencerminkan cara MongoDB menyimpan informasi, dengan setiap dokumen di dalam koleksi mewakili entitas tunggal, dalam hal ini, buku "Laskar Pelangi."

Dengan contoh format dokumen tersebut membuktikan bahwa MongoDB mampu menampung data yang lebih bervariasi dan kompleks. Tidak hanya itu, pengguna juga dapat melakukan scale-out database untuk meningkatkan kapasitas data tanpa mengganggu performa server.

Beberapa kelebihan yang akan dijelaskan seperti di bawah:

1) Performa cepat

MongoDB memiliki performa yang sangat cepat dalam memproses data. Hal ini terjadi karena MongoDB mendukung indexing dan query yang efisien. MongoDB dapat mengindex data dengan banyak cara, termasuk indexing single field, indexing composite field, indexing text, dan indexing geo-spatial data. Selain itu, MongoDB juga mendukung query yang efisien untuk mengambil data dari database.

2) Fleksibel dan scalable

MongoDB dirancang untuk menjadi sistem database yang sangat fleksibel dan scalable. Struktur datanya yang berbasis dokumen memungkinkan pengguna untuk menyimpan data dengan cara yang sangat fleksibel, bahkan jika skema datanya berubah dari waktu ke waktu. Sebagai contoh, pengguna dapat menyimpan dokumen dengan field yang berbeda untuk setiap dokumen, atau bahkan menyimpan array field dengan panjang yang berbeda untuk setiap dokumen. Hal ini memungkinkan pengguna untuk

mengadaptasi sistem database dengan lebih mudah saat memperbarui aplikasi atau menambah fitur baru. Selain itu, MongoDB juga dapat di-scaling secara horizontal dengan mudah. Hal ini dilakukan dengan cara membagi data ke beberapa server atau shard sehingga memungkinkan pengguna untuk mengelola dataset yang sangat besar dengan mudah.

3) Dapat mengelola query dengan lebih baik

Selain dua kelebihan di atas, MongoDB juga terkenal sebagai database yang memiliki kemampuan untuk menangani query dengan baik, query language yang kuat, dan fleksibel. Dalam query language ini, pengguna dapat melakukan pengolahan data seperti filtering, sorting, dan aggregation. MongoDB juga mendukung operasi join, meskipun dengan pendekatan yang berbeda dari database relasional tradisional.

4) Open-source dan memiliki komunitas yang aktif

MongoDB adalah sistem database open-source yang berarti bahwa pengguna dapat mengunduh dan menggunakannya secara gratis, baik untuk pengguna Windows, Linux, atau macOS. Selain itu, MongoDB juga memiliki komunitas pengguna yang aktif dan selalu siap membantu pengguna baru untuk memberikan dukungan teknis.

5) Bahasa pemrograman variative

MongoDB mendukung berbagai bahasa pemrograman, seperti C, C++, Java, Node.js, PHP, Perl, Python, Ruby, C# dan .NET, dan Mongoid.

6. Tailwind CSS

Tailwind CSS adalah utility-first framework CSS yang didesain untuk mempermudah dan mempercepat pembuatan aplikasi menggunakan desain custom. Utility-first classes merupakan utilitas tingkat rendah (low-level) untuk membuat desain khusus di dalam file

HTML. Developer hanya perlu memberi nama kelas utilitas sesuai tujuan yang dimaksud, sehingga nama kelas tersebut bisa dipahami banyak orang.

Dengan menggunakan kelas-kelas utilitas, developer dapat mengontrol layout, warna, spasi, tipografi, bayangan, dan lain-lain supaya desain komponennya bisa sesuai kebutuhan. Pekerjaan itu bisa dilakukan tanpa perlu meninggalkan kode HTML atau menulis CSS khusus. Konsep menarik dari Tailwind yaitu framework ini mengutamakan utilitas untuk membuat desain khusus. Tailwind tidak menawarkan style dan template tertentu. Sebagai gantinya, Tailwind menawarkan opinionated building blocks yang dikenal sebagai kelas utilitas untuk membantu mengatur style komponen website. Untuk membuat user interface yang menarik, developer bekerja dengan menggabungkan komponen-komponen kecil. Tailwind hanya mengambil file CSS mentah, memprosesnya melalui file konfigurasi, lalu menghasilkan output yang diinginkan.

Adapun Fungsi dari Tailwind CSS adalah sebagai berikut:

- a) Mengurangi penulisan CSS khusus: developer bisa mengatur tampilan elemen menggunakan kelas-kelas yang sudah ada langsung di HTML. Mereka juga dapat membuat desain kustom tanpa perlu menulis CSS.
- b) Bisa menjaga ukuran file CSS tetap kecil: jika tidak menggunakan framework seperti Tailwind, developer harus terus menulis CSS saat menambahkan fitur dan komponen baru. Akibatnya, file CSS akan semakin besar dan berat. Namun dengan menggunakan utilitas seperti flexbox dan padding di Tailwind, sebagian besar style dapat digunakan kembali sehingga tidak perlu menulis CSS baru.
- c) Tidak perlu menciptakan nama kelas sendiri: developer bisa memilih kelas-kelas dari sistem desain dengan nama yang sudah ditentukan sebelumnya. Mereka tidak perlu bingung mencari nama

kelas untuk style dan komponen tertentu, atau bahkan mengingat nama-nama yang rumit.

- d) Bisa melakukan perubahan dengan lebih aman: dalam pendekatan tradisional, saat melakukan perubahan pada CSS kemungkinan akan merusak halaman situs secara keseluruhan. Namun, kelas utilitas dalam HTML bersifat lokal yang dapat diubah tanpa khawatir merusak hal lain di website.

Berikut ini adalah beberapa perbedaan lain antara Tailwind dan Bootstrap dirangkum dari Tutorials Point:

- a) Tailwind CSS menawarkan pendekatan yang lebih custom dan fleksibel dalam desain user interface. Sedangkan Bootstrap menyediakan tema dan template siap pakai.
- b) Aplikasi dan website yang dibuat menggunakan Tailwind CSS memiliki tampilan yang unik dan fleksibel. Sementara aplikasi yang dibuat dengan Bootstrap umumnya memiliki tampilan yang serupa karena menggunakan template bawaan.
- c) Dibandingkan Tailwind CSS yang cenderung baru dan masih terus berkembang, Bootstrap sudah lebih lama eksis karena dikenal responsif serta efisien.
- d) Tailwind CSS memiliki ukuran file yang lebih kecil, berbeda dengan Bootstrap yang membutuhkan ukuran file lebih besar.
- e) Beberapa perusahaan terkenal yang menggunakan Tailwind CSS antara lain MAKE IT, Superchat, dan Livestorm. Bootstrap digunakan oleh perusahaan terkenal lainnya seperti Twitter, LinkedIn, Spotify, dan StackShare.

7. Google Chrome

Google Chrome adalah peramban web lintas platform yang dikembangkan oleh Google. Peramban ini pertama kali dirilis pada tahun 2008 untuk Microsoft Windows, kemudian diporting ke Android,

iOS, Linux, dan macOS yang menjadikannya sebagai peramban bawaan dalam sistem operasi. Peramban ini juga merupakan komponen utama Chrome OS, yang berfungsi sebagai platform untuk aplikasi web.

CEO Google, Eric Schmidt, awalnya menentang perkembangan zaman peramban web independen selama enam tahun, mengungkapkan kekhawatirannya terhadap potensi "perang peramban" yang merugikan. Schmidt, pada saat itu menggambarkan Google sebagai perusahaan kecil dan berpendapat bahwa terlibat dalam persaingan semacam itu tidaklah menguntungkan. Namun, pandangan ini berubah ketika pendiri Google, Sergey Brin dan Larry Page, merekrut beberapa pengembang Mozilla Firefox dan berhasil membuat demonstrasi yang mengesankan untuk Chrome. Schmidt mengakui bahwa perubahan ini dipicu oleh kecanggihan proyek yang diajukan oleh tim pengembang.

Pada bulan September 2004, muncul rumor tentang Google yang sedang mengembangkan peramban web. Kabar ini menyebar setelah Google mempekerjakan beberapa mantan pengembang web Microsoft. Saat itu, Mozilla Firefox baru saja merilis versi finalnya dan mendapatkan popularitas yang signifikan dengan mengambil pangsa pasar dari Internet Explorer, yang pada saat itu menghadapi masalah keamanan serius.

Proses pengembangan Chrome secara resmi dimulai pada tahun 2006 di bawah arahan Sundar Pichai, yang pada saat itu memimpin proyek tersebut. Lebih menarik lagi, sebagian besar pengembangan Chrome dilakukan di kantor Google di Kitchener. Selama waktu ini, Google berusaha menciptakan peramban web yang inovatif dan efisien untuk mengatasi tantangan dan kekurangan yang ada pada peramban web lainnya.

Fitur Google Chrome menampilkan antarmuka pengguna minimalis, dengan prinsip antarmuka pengguna yang kemudian diterapkan ke peramban lain. Misalnya, penggabungan bilah alamat dan bilah

penelusuran ke dalam omnibox. Chrome juga memiliki reputasi untuk kinerja peramban yang kuat.

Chrome terus meningkatkan keamanan penggunanya dengan secara berkala mengambil pembaruan dari dua daftar hitam, satu untuk phishing dan satu untuk malware. Pengguna diberi peringatan jika mereka mencoba mengunjungi situs yang dicurigai sebagai potensi berbahaya. Layanan ini juga dapat diakses oleh pihak lain melalui API publik gratis yang dikenal sebagai "Google Safe Browsing API."

Chrome menerapkan model alokasi proses ke dalam tab sandbox. Dengan prinsip least privilege, setiap proses tab diisolasi dari fungsi memori kritis dan proses tab lainnya, serupa dengan "Protected Mode" yang digunakan oleh Internet Explorer 9 atau yang lebih tinggi. Tim Sandbox Chrome telah berhasil menciptakan batas proses yang efektif, menciptakan suatu penjara virtual. Model keamanan ini menjalankan dua tingkat keamanan bertingkat, yaitu pengguna dan sandbox, di mana sandbox hanya merespons permintaan komunikasi yang diinisiasi oleh pengguna. Di Linux, sandboxing menggunakan mode seccomp.

Pada 2 September 2020, dengan rilis Chrome 85, Google memperluas dukungan untuk DNS Aman di Chrome untuk Android. DNS-over-HTTPS (DoH) diperkenalkan untuk meningkatkan keamanan dan privasi saat menjelajah web. Dengan pembaruan tersebut, Chrome secara otomatis beralih ke DNS-over-HTTPS (DoH) jika penyedia DNS saat ini mendukung fitur tersebut. Ini adalah langkah lebih lanjut dalam upaya Chrome untuk menyediakan pengalaman menjelajah web yang aman dan terjamin privasi bagi penggunanya.

8. Github

GitHub adalah sebuah website dan layanan berbasis cloud bagi para developer untuk menyimpan dan mengelola kode, serta mendokumentasikan dan mengontrol perubahannya. GitHub juga

merupakan salah satu storehouse online terbesar di dunia untuk pekerjaan kolaborasi. Dengan platform ini, Anda bisa bekerja bersama-sama dengan rekan dari berbagai belahan dunia untuk merencanakan proyek.

Berikut adalah beberapa komponen penting dari definisi GitHub:

a. Git

Git adalah sistem kontrol versi terdistribusi yang digunakan untuk melacak perubahan dalam kode sumber selama pengembangan perangkat lunak. Setiap perubahan atau revisi disebut "commit," dan Git memungkinkan pengembang untuk menyimpan, membagikan, dan mengelola riwayat revisi ini.

b. Repository

Repository adalah tempat penyimpanan untuk proyek perangkat lunak. Ini bisa berisi semua file, dokumen, dan sumber daya lain yang diperlukan untuk mengembangkan dan menyimpan proyek. Repositori di GitHub dapat bersifat publik (dapat diakses oleh siapa saja) atau pribadi (hanya dapat diakses oleh tim yang diundang).

c. Branch

Branch adalah cabang atau salinan independen dari repositori yang memungkinkan pengembang untuk bekerja pada fitur atau perbaikan tanpa mempengaruhi kode di cabang utama (biasanya disebut "master" atau "main"). Setelah pengembangan selesai, perubahan dapat digabungkan kembali ke cabang utama.

d. Pull Request Pull Request (PR) adalah permintaan untuk menggabungkan perubahan dari satu cabang ke cabang lainnya. Ini memungkinkan pengembang untuk memeriksa, mengomentari, dan meninjau perubahan sebelum disatukan ke dalam repositori utama.

e. Collaborators

Collaborators adalah orang atau tim yang diizinkan untuk berkontribusi pada repositori tertentu. Peserta memiliki hak akses

tertentu, tergantung pada peran Peserta (seperti read-only, write, atau admin).

f. Issues dan Projects

GitHub menyediakan fitur Issues untuk melacak tugas, perbaikan bug, dan diskusi terkait proyek. Projects memungkinkan pengembang untuk mengorganisir dan mengelola tugas dalam tata letak papan kerja (kanban) yang dapat disesuaikan.

g. Wiki

Wiki adalah bagian dari repositori yang dapat digunakan untuk menyimpan dokumentasi proyek. Ini membantu dalam menyimpan informasi yang relevan dan dapat diakses oleh semua kontributor

9. Netlify

Netlify adalah perusahaan web hosting infrastruktur dan teknologi otomatisasi yang berbasis di San Francisco. Netlify menyederhanakan proses bagi pengembang untuk menyebarkan dan meng-host situs web. Itu melakukan semua pekerjaan untuk Peserta yang Peserta tidak ingin menghabiskan terlalu banyak waktu atau usaha. Manfaat Netlify untuk Editor dan Pemasar Digital :

- 1) Netlify Kurang Mahal, dan Anda Mendapatkan Situs yang Lebih Cepat Membangun dan hosting situs web statis akan membantu Anda menghemat uang dalam jangka panjang. Dengan Netlify Edge, kecepatan situs secara drastis meningkat melalui distribusi global dan pra-render otomatis.
- 2) Netlify Build Memungkinkan Pengembang untuk Membangun Dengan Integrasi Apa Pun. Setiap kali editor ingin membuat pembaruan ke situs, itu memungkinkan Anda membuat cabang situs web Anda yang tidak terbatas.
- 3) Lebih Mudah Meluncurkan Situs Menggunakan Netlify. Netlify memiliki sertifikat manajemen DNS & SSL bawaan (dan GRATIS,

tidak kedaluwarsa). Ini berarti lebih sedikit kompleksitas dalam meluncurkan situs web Anda dan semuanya di-host di satu tempat.

B. Hardware

Dalam tugas frontend engineering ada beberapa perangkat keras yang diperlukan meliputi:

- 1) Server: Dibutuhkan server, baik berupa komputer fisik maupun mesin virtual di cloud, untuk menjalankan aplikasi web yang dibuat oleh frontend engineering.
- 2) Jaringan dan Router: Jaringan yang stabil diperlukan untuk menghubungkan server dengan pengguna. Router digunakan untuk mengarahkan lalu lintas jaringan antara server dan pengguna.
- 3) Storage (Penyimpanan): Ruang penyimpanan diperlukan untuk menyimpan data aplikasi seperti basis data dan file pengguna. Ini dapat menggunakan hard disk drive (HDD) atau solid-state drive (SSD) pada server.
- 4) Switch dan Kabel Jaringan: Digunakan untuk menghubungkan server dengan perangkat jaringan lainnya seperti router, komputer pengguna, dan perangkat klien.
- 5) Komputer atau Laptop Pengguna: Pengguna aplikasi akan mengaksesnya melalui perangkat seperti komputer atau laptop dengan akses internet. Komputer ini harus memenuhi persyaratan minimum sistem untuk menjalankan aplikasi web dengan baik.
- 6) Perangkat Masukan/Keluaran: Pengguna akan menggunakan perangkat seperti keyboard, mouse, monitor, dan perangkat masukan/keluaran lainnya saat mengakses dan menggunakan aplikasi web.

BAB IV

PELAKSANAAN KERJA PRAKTIK

Dalam Pelaksanaan kerja praktik yang diselenggarakan oleh PT Ruang Raya Indonesia, khususnya pada bidang Frontend Engineering, terdapat konsep final project yang dipecah menjadi beberapa bagian. Dalam konteks ini saya mengerjakan JSON Pada Basic Frontend Development.

IV.1 Input

Berikut adalah inputan dari final project pada materi JSON pada Basic Frontend Development yang disediakan oleh PT. Ruang Raya Indonesia (Ruang Guru) pada program Frontend Engineering. Dalam pengembangan aplikasi *Library App* ini melibatkan beberapa inputan pengolahan data diantaranya:

1. API Server sudah disediakan dalam folder server dengan nama file `library-db.json`. Server ini menggunakan `json-server` yang dapat dijalankan menggunakan perintah `"npm start:server"`. Server ini memiliki beberapa endpoint, antara lain:
 - a. GET `/books` (untuk mendapatkan seluruh data buku)
 - b. POST `/books` (untuk menambahkan data buku)
 - c. PUT `/books/id` (untuk mengubah data buku)
 - d. DELETE `/books/id` (untuk menghapus data buku berdasarkan id)
2. Skeleton code untuk client sudah disediakan dalam folder client dengan nama file `main.js`. Penulis hanya perlu mengimplementasikan kode yang diperlukan untuk menyelesaikan website ini. Beberapa fungsi sudah diberikan komentar yang menjelaskan apa yang harus dilakukan oleh masing-masing fungsi. Client dapat dijalankan menggunakan perintah `"npm start:client"`.

Web Application ini memiliki beberapa persyaratan umum sebagai berikut:

- 1) Saat halaman web di-render pertama kali, data buku dari database akan ditampilkan secara otomatis dalam sebuah tabel.

- 2) Ketika tombol "Add Book" diklik, tampilan akan berubah menjadi formulir untuk menambahkan data buku. Saat formulir di-submit, data buku akan ditambahkan ke dalam database dan data terbaru akan ditampilkan secara real-time tanpa perlu me-refresh halaman.
- 3) Ketika tombol "Edit" diklik, tampilan akan berubah menjadi formulir untuk mengubah data buku. Informasi buku yang akan diubah sudah akan muncul dalam input formulir. Saat formulir di-submit, data buku akan diubah dalam database dan data terbaru akan ditampilkan secara real-time tanpa perlu me-refresh halaman.
- 4) Ketika tombol "Hapus" diklik, data buku akan dihapus dari database dan data buku yang dihapus akan hilang secara otomatis tanpa perlu me-refresh halaman.
- 5) Aplikasi harus di-deploy ke Netlify, dan link deploy-nya harus dijadikan variabel `NetlifyDeployUrl` dalam file `deployData.js`. Jangan lupa menyertakan nama dan CAMPID pada file tersebut.

1. Kebutuhan perangkat Keras

Perangkat keras yang diperlukan untuk pembuatan aplikasi *Library App* ini mencakup penggunaan laptop, oleh karena itu, diperlukan sebuah komputer dengan spesifikasi yang dapat mendukung tugas-tugas pengembangan frontend

Tabel IV. 1 Kebutuhan Perangkat Keras

Item	Spesifikasi
Processor	Intel Core i7
RAM	16 GB
SSD	512GB
VGA	INVIDIA GeForce GTX 1660

2. Minimum kebutuhan perangkat keras

Spesifikasi minimum komputer yang dibutuhkan untuk pembuatan aplikasi *Library App* adalah sebagai berikut:

Tabel IV. 2 Minimum kebutuhan perangkat keras

Item	Spesifikasi
Processor	Intel Core i5
RAM	8 GB
SSD	256GB
VGA	Intel UHD Graphics 620

3. Kebutuhan perangkat lunak

Perangkat lunak mencakup serangkaian instruksi atau program komputer yang memiliki peran khusus dalam mengendalikan dan menjalankan berbagai kegiatan terkait pengembangan antarmuka pengguna (frontend) pada aplikasi atau situs web. Berikut adalah beberapa jenis perangkat lunak yang umumnya diterapkan dalam pembuatan *Library App* ini:

Tabel IV. 3 Kebutuhan perangkat lunak

Item	Spesifikasi
Sistem Operasi	Windows 11 64 bit
Kode Editor	Visual Studio Code
Browser	Chrome
Version Control	Git, GitHub
Package Manager	npm (Node Package Manager)
Framework dan Library	JSON, Tailwind CSS
Code Repository Hosting	GitHub

IV.2 Proses

Pada proyek Final Project ini, peserta ditugaskan untuk mengembangkan sebuah website perpustakaan yang memiliki kemampuan untuk melakukan operasi CRUD (Create, Read, Update, Delete) terhadap data buku. Komunikasi data antara website dan server akan dilakukan melalui API, dengan menggunakan API Server yang sudah disediakan dalam folder server dengan nama file `library-db.json`. Penting untuk diperhatikan bahwa peserta dilarang mengubah file `main.test.js`,

mengganti nama function yang diberikan, mengubah struktur skeleton kode, dan mengedit file initial.json di folder server. Sebelum memulai proyek, peserta juga diminta untuk menjalankan perintah "npm install" atau "pnpm install" guna menginstal dependensi yang diperlukan. Proyek ini memiliki beberapa tujuan, di antaranya adalah penerapan DOM untuk membuat website dan melakukan panggilan API. API Server menggunakan json-server dan dapat dijalankan dengan perintah "pnpm start:server", menyediakan endpoint-endpoint seperti GET, POST, PUT, dan DELETE untuk operasi CRUD pada data buku. Sebuah skeleton code client (main.js) juga sudah disediakan, dan peserta diharapkan untuk mengimplementasikan kode yang diperlukan sesuai dengan komentar yang telah diberikan. Persyaratan umum website mencakup tampilan otomatis data buku pada halaman pertama kali, formulir penambahan dan pengubahan data buku, serta operasi real-time tanpa perlu me-refresh halaman. Terakhir, website harus di-deploy ke Netlify, dengan link deploy dijadikan variabel NetlifyDeployUrl pada file deployData.js, yang juga harus menyertakan nama dan CAMPID peserta. Proyek ini memberikan kesempatan bagi peserta untuk eksplorasi dan perancangan perangkat lunak, mengimplementasikan operasi CRUD, serta melakukan deployment website secara efektif.

IV.2.1 Eksplorasi

Eksplorasi Perangkat Lunak pada Proyek Final Project:

1. Tujuan Penerapan DOM:

Proyek ini menetapkan tujuan untuk menggunakan Document Object Model (DOM) dalam pembuatan website, menandakan eksplorasi dalam struktur dan representasi data pada halaman web.

3. Penggunaan API Server:

Eksplorasi melibatkan penggunaan API Server dari file library-db.json di folder server. Penggunaan json-server dan endpoint-endpoint seperti GET, POST, PUT, dan DELETE menunjukkan pemahaman tentang komunikasi antara client dan server.

4. Implementasi Kode dengan Skeleton Code:

Pengguna diminta untuk menjalankan implementasi kode dalam skeleton code client (main.js). Fungsi-fungsi yang perlu diimplementasikan telah diberikan dengan komentar, mendorong eksplorasi dalam implementasi fitur CRUD dan interaksi real-time.

5. Panggilan API:

Dalam mencapai operasi CRUD, proyek ini memerlukan pemahaman dalam melakukan panggilan API, menunjukkan eksplorasi bagaimana data dikirim dan diterima antara client dan server.

Melalui fokus pada tujuan menggunakan DOM, interaksi dengan API Server, implementasi kode, dan panggilan API, proyek ini menggambarkan aspek eksplorasi dalam pengembangan perangkat lunak.

IV.2.2 Perancangan Perangkat Lunak

Perancangan Perangkat Lunak pada aplikasi *Library App* ini melalui beberapa tahapan diantaranya sebagai berikut:

1. Perencanaan

Tahap perencanaan proyek adalah langkah pengembangan perangkat lunak. Proses ini mencakup pemahaman mendalam terhadap kebutuhan pengguna, penentuan tujuan proyek secara jelas, penjadwalan tugas-tugas yang perlu dilakukan, dan alokasi sumber daya yang diperlukan. Selain itu, tahap ini juga mencakup identifikasi risiko-risiko potensial dan strategi, sehingga proyek dapat dijalankan secara efisien dan sesuai dengan target waktu yang ditetapkan. Perencanaan proyek yang baik memberikan landasan yang kuat untuk tahapan-tahap selanjutnya dalam siklus pengembangan, memastikan bahwa proyek berjalan sesuai rencana dan dapat memenuhi ekspektasi pengguna serta.

2. Analisis

Tahap analisis kebutuhan sistem adalah langkah awal dalam mengembangkan perangkat lunak. Pada tahap ini, berfokus untuk mengumpulkan, menganalisis, dan memahami kebutuhan sistem dengan

cermat. Proses ini melibatkan identifikasi masalah utama yang perlu dipecahkan oleh perangkat lunak yang akan dikembangkan. Selain itu, tujuan utama adalah merinci persyaratan fungsional dan non-fungsional yang harus dipenuhi agar perangkat lunak dapat memenuhi ekspektasi pengguna dan memenuhi standar kualitas yang ditetapkan.

3. Desain

Proses desain sistem merupakan tahap pengembangan perangkat lunak yang melibatkan penciptaan struktur dan arsitektur sistem. Berfokus pada perancangan antarmuka pengguna yang optimal, memastikan kebergunaan dan keterbacaan yang baik. Pemilihan teknologi yang sesuai juga menjadi bagian penting dari tahap desain ini. Hal ini mencakup penentuan bahasa pemrograman, jenis database, kerangka kerja, dan alat pengembangan yang paling cocok untuk memenuhi kebutuhan proyek.

a) Analisis kebutuhan Non fungsional

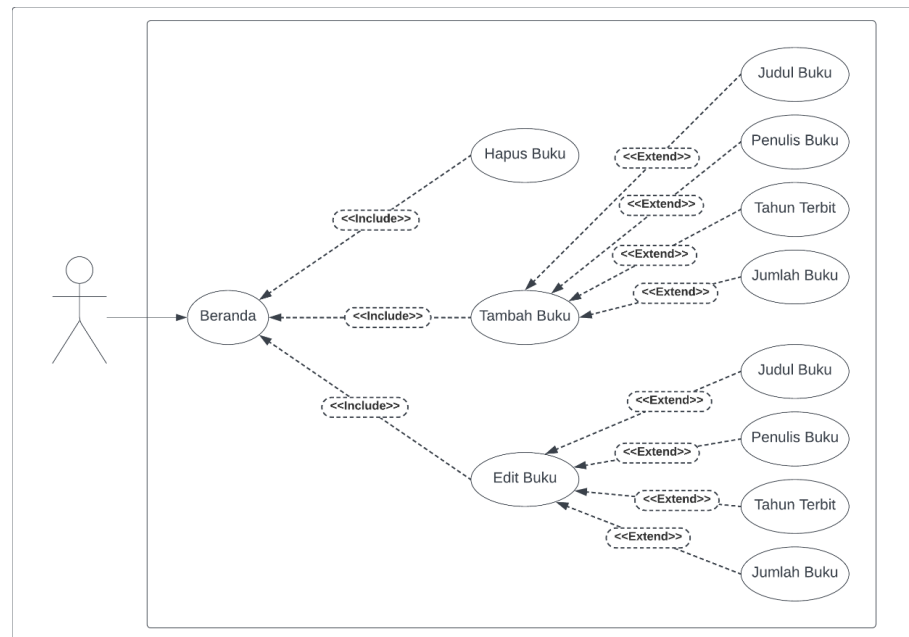
Analisis kebutuhan non-fungsional adalah tahap dalam menetapkan spesifikasi sistem secara menyeluruh. Spesifikasi ini mencakup elemen-elemen dan komponen-komponen esensial yang diperlukan dari awal hingga implementasi sistem. Analisis ini mendefinisikan spesifikasi masukan yang diperlukan oleh sistem, keluaran yang dihasilkan, dan proses yang diperlukan untuk memproses masukan tersebut sehingga dapat menghasilkan keluaran yang sesuai dengan tujuan. Selain itu, analisis kebutuhan non-fungsional juga membantu menetapkan parameter seperti keamanan, keandalan, dan kinerja sistem yang harus dicapai.

b) Analisis kebutuhan fungsional

1) Use Case Diagram

Use case diagram adalah salah satu jenis diagram dalam Unified Modeling Language (UML) yang digunakan untuk menggambarkan interaksi antara pengguna dengan sistem. Diagram ini menunjukkan fungsi, ruang lingkup, dan interaksi pengguna dengan sistem tersebut. Use case diagram terdiri dari beberapa komponen utama, seperti aktor, use case, sistem boundary, dan asosiasi. Aktor adalah pengguna sistem,

bisa orang atau organisasi yang menggunakan, atau orang lain yang bukan pengguna namun berinteraksi dengan sistem. Use case adalah komponen gambaran fungsional dalam sebuah sistem. Sistem boundary adalah batas antara sistem dan lingkungan luar.



Gambar IV 1 Use Case Diagram

Pada perancangan aplikasi *Library App* di PT Ruang Raya Indonesia memiliki actor atau user yang bisa melakukan hal hal seperti pada gambar *Gambar IV 1 Use Case Diagram* diatas ketika user sudah masuk ke aplikasi maka system akan menampilkan menu-menu yang ada pada dashboard aplikasi *Library App* diantaranya:

1. Menu Beranda

Beranda aplikasi menyediakan 4 fitur utama yang dapat diakses oleh pengguna. Fitur-fitur tersebut mencakup Data Buku, Tambah Buku, Edit Data Buku, dan Hapus Data Buku. Setiap fitur dirancang untuk memberikan akses yang mudah bagi pengguna dalam mengelola koleksi buku mereka.

2. Menu Data Buku

Menu ini memberikan akses kepada pengguna untuk bisa melihat semua data buku yang telah terekam sebelumnya di aplikasi *Library App*. Dengan tampilan yang sederhana, pengguna dapat dengan mudah melihat dan memeriksa seluruh koleksi buku yang ada, memberikan transparansi terhadap informasi yang tersimpan

3. Menu Tambah Buku

Pengguna dapat menambahkan buku baru melalui menu ini. Proses penambahan buku melibatkan pengisian empat data penting, termasuk nama buku, nama penulis, tahun terbit, dan jumlah buku. Langkah-langkah ini dirancang agar pengguna dapat dengan cepat dan akurat menambahkan buku ke dalam aplikasi.

4. Menu Edit Buku

Menu ini memungkinkan pengguna untuk mengubah data buku yang telah dimasukkan sebelumnya. Proses pengeditan mencakup kemampuan untuk merubah nama buku, nama penulis, tahun terbit, dan jumlah buku. Fungsi ini memberikan fleksibilitas dalam memperbarui informasi buku yang mungkin berubah seiring waktu.

5. Menu Hapus Buku

Pengguna dapat menghapus data buku yang sudah dimasukkan melalui menu ini, berguna saat data buku tidak lagi relevan atau dibutuhkan. Penghapusan dapat dilakukan dengan mudah, memberikan kendali kepada pengguna untuk menjaga relevansi dalam koleksi buku mereka.

2) Activity Diagram

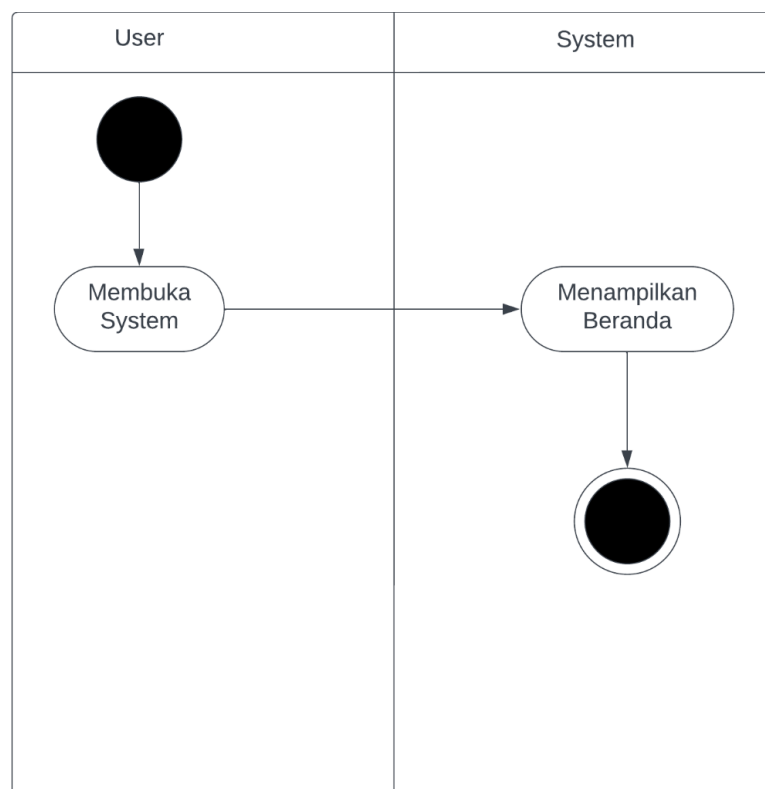
Berikut ini adalah activity diagram pada rekayasa perangkat lunak aplikasi *Library App*:

Activity diagram pertama menggambarkan langkah-langkah tampilan masuk ke *Library App*. Proses ini mencakup aktivitas-aktivitas seperti otentikasi pengguna, validasi informasi, dan

pengalihan ke halaman beranda setelah masuk. Diagram ini memberikan gambaran visual mengenai alur kerja saat pengguna mengakses aplikasi.

Activity diagram kedua fokus pada halaman beranda aplikasi. Di dalamnya, terdapat aktivitas-aktivitas yang mencakup tampilan menu-menu yang tersedia. Setiap menu, seperti Data Buku, Tambah Buku, Edit Data Buku, dan Hapus Data Buku, direpresentasikan sebagai aktivitas terpisah dengan urutan langkah-langkah yang jelas. Diagram ini memberikan gambaran visual tentang proses interaksi pengguna dengan menu-menu yang disediakan dalam aplikasi *Library App*. Berikut ini gambar dari activity diagram pada aplikasi *Library App* dan penjelasan lengkapnya:

1. Activity Diagram Library App



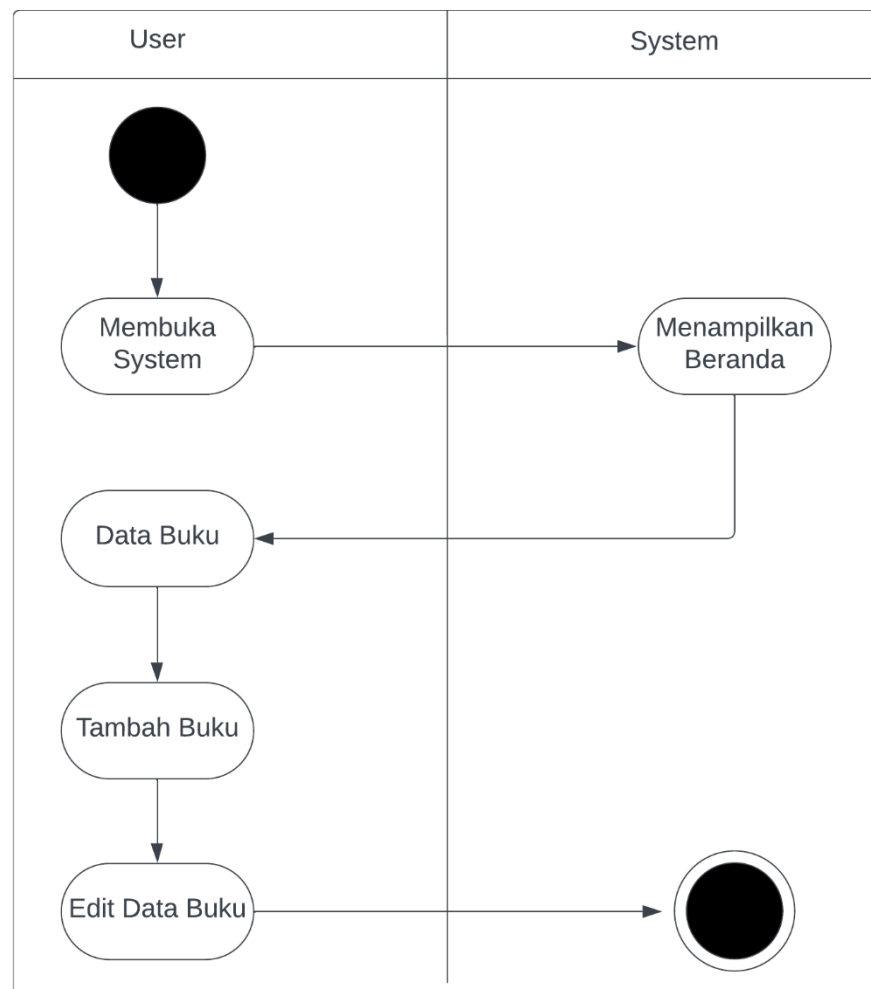
Gambar IV 2 Activity Diagram Library App

Pada aplikasi Library App, pengguna dapat mengakses aplikasi tanpa melalui proses login, yang berarti mereka langsung masuk ke dalam antarmuka aplikasi, dikarenakan tidak ada fitur login pada tahap awal, pengembang perlu memperhatikan bahwa keamanan dan privasi pengguna mungkin menjadi pertimbangan di masa mendatang. Pengembang akan menambahkan fitur login sebagai langkah untuk mengamankan akses pengguna, menyimpan preferensi personal, dan memberikan pengalaman yang lebih personalisasi. Dengan demikian, meski saat ini pengguna dapat langsung menjelajahi aplikasi, implementasi fitur login dapat menjadi peningkatan berarti dalam jangka waktu tertentu.

Dalam activity diagram library app, proses dimulai ketika pengguna membuka sistem atau library app. Setelah membuka aplikasi, sistem akan merespons dengan menampilkan halaman beranda. Pada halaman beranda ini, pengguna segera disajikan dengan informasi lengkap mengenai buku-buku yang telah diinputkan sebelumnya. Informasi ini mencakup detail seperti nama buku, tahun terbit, nama penulis, dan stok yang dimiliki oleh pengguna.

Halaman beranda tersebut dirancang untuk memberikan pandangan cepat dan komprehensif tentang koleksi buku yang dimiliki oleh pengguna dalam aplikasi library app. Dengan menyediakan informasi yang jelas dan terstruktur, pengguna dapat dengan mudah melihat dan mengelola data buku yang terkait dengan aktivitas perpustakaan mereka. Oleh karena itu, activity diagram tidak hanya mencerminkan navigasi pengguna melalui aplikasi, tetapi juga menyoroti elemen penting dari antarmuka pengguna yang dapat memfasilitasi manajemen efektif dan penggunaan koleksi buku.

2. Beranda



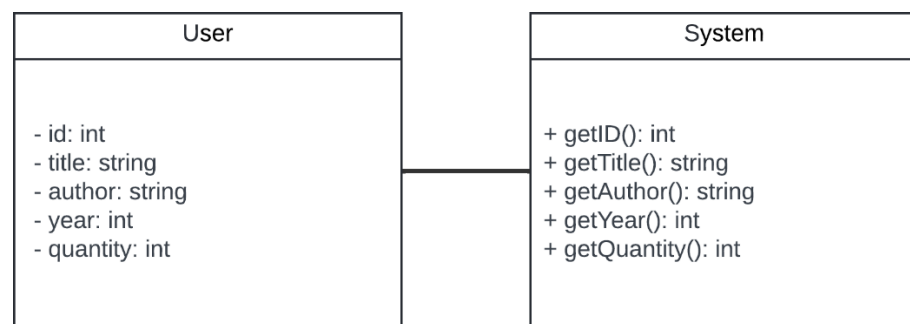
Gambar IV 3 Activity Diagram Beranda

Dalam activity diagram beranda, pengalaman pengguna diawali dengan melihat halaman beranda. Halaman ini menampilkan informasi tentang buku-buku yang sudah diinputkan sebelumnya, termasuk nama buku, tahun terbit, nama penulis, dan jumlah stok yang dimiliki oleh pengguna. Selain menampilkan informasi, pengguna juga memiliki akses untuk mengedit data buku jika terdapat kesalahan input, memastikan bahwa data buku yang dimasukkan menjadi valid.

Selain itu, pengguna juga diberikan akses untuk menghapus data buku yang dianggap tidak diperlukan lagi. Akses ini

memungkinkan pengguna untuk merapikan dan mengelola data buku sesuai dengan kebutuhan mereka. Dengan demikian, activity diagram tidak hanya memvisualisasikan tampilan halaman beranda, tetapi juga menggambarkan interaksi dinamis pengguna dalam mengelola dan memodifikasi data buku pada platform tersebut.

3) Class Diagram



Gambar IV 4 Class Diagram

Diagram kelas pada gambar tersebut menunjukkan hubungan antara kelas **User** dan kelas **System**. Kelas **User** mewakili pengguna sistem, sedangkan kelas **System** mewakili sistem itu sendiri. Kelas **User** memiliki atribut `id`, `title`, `author`, `year`, dan `quantity`. Atribut `id` adalah atribut unik yang digunakan untuk mengidentifikasi pengguna. Atribut `title`, `author`, dan `year` menyimpan informasi tentang buku yang sedang dibaca pengguna. Atribut `quantity` menyimpan jumlah buku yang telah dibaca pengguna.

Kelas **System** memiliki operasi `getID()`, `getTitle()`, `getAuthor()`, `getYear()`, dan `getQuantity()`. Operasi `getID()` mengembalikan nilai atribut `id`. Operasi `getTitle()`, `getAuthor()`, dan `getYear()` mengembalikan nilai atribut `title`, `author`, dan `year`. Operasi `getQuantity()` mengembalikan nilai atribut `quantity`.

Berdasarkan diagram kelas tersebut, dapat disimpulkan bahwa sistem ini memungkinkan pengguna untuk menyimpan data buku,

sistem ini akan menyimpan informasi tentang buku yang sudah dimasukan oleh pengguna, diantaranya judul, penulis, tahun terbit, dan jumlah buku yang dimiliki.

4) Database

Tabel IV. 4 Database di JSON

Nama Field	Type	Keterangan
id	int	Kode buku
title	String	Judul buku
author	String	Penulis buku
year	Int	Tahun terbit
quantity	Int	Jumlah stok buku

Database pada tabel tersebut merupakan untuk informasi buku, terdiri dari tabel tunggal bernama buku pada tabel ini memiliki lima kolom penting, masing-masing dengan atribut khusus:

1. *id* yang bertype data Integer untuk kode buku, berfungsi sebagai kunci primer dengan nilai unik.
2. *Title* bertype data string untuk judul buku.
3. *author* bertype data string untuk nama penulis buku.
4. *year* bertype data Integer untuk tahun terbit buku.
5. *quantity* bertype data Integer jumlah stok buku.

Kolom *id* berperan sebagai kunci primer untuk menjaga keunikan setiap baris dalam tabel. Dengan informasi ini, database ini dapat digunakan untuk berbagai keperluan praktis, termasuk pencarian buku berdasarkan judul, penulis, atau tahun terbit, menampilkan daftar buku yang tersedia, dan mengelola stok buku di perpustakaan.

IV.3 Pencapaian Hasil

Adapun hasil yang dicapai dari kerja praktik di PT Ruang Raya Indonesia ini berupa membuat rekayasa perangkat lunak aplikasi *Library App* menggunakan JSON untuk merekam data buku, yang dimana rekayasa perangkat lunak aplikasi ini

nantinya akan berfungsi untuk menangani hal-hal yang sudah disediakan oleh pembuat seperti hal berikut ini:

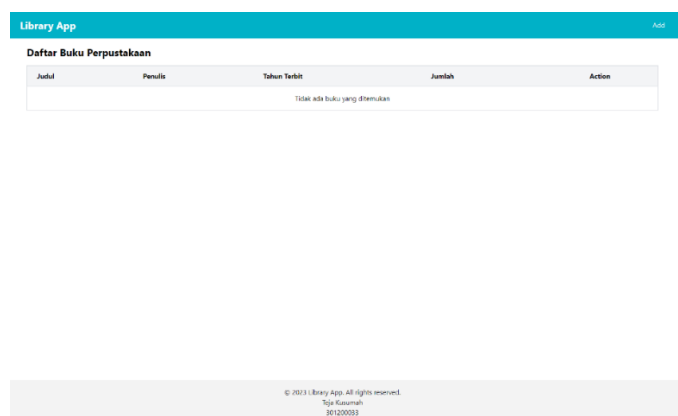
- 1) Pencatatan Data Buku: Aplikasi memungkinkan pengguna untuk memasukkan informasi buku ke dalam sistem, menciptakan basis data yang terstruktur.
- 2) Pengeditan Data Buku: Fasilitas pengeditan memungkinkan pengguna untuk memperbaiki informasi buku jika mengalami kesalahan dalam melakukan penginputan data buku, termasuk penyesuaian jumlah buku, nama penulis, dan tahun terbit.
- 3) Penghapusan Data Buku: Aplikasi memberikan kemampuan untuk menghapus data buku yang sudah tidak diperlukan lagi sehingga pada halaman utama *Library App* hanya akan ada buku-buku yang masih dimiliki oleh pengguna.

Kerja praktik ini juga menghasilkan beberapa hal yang diantaranya:

- 1) Use case diagram
- 2) Activity diagram
- 3) Class diagram
- 4) Antarmuka aplikasi

Berikut ini tampilan antarmuka dari rekayasa perangkat lunak aplikasi *Library App* yang sudah dibuat:

1. Tampilan Awal Beranda Library App



Gambar IV 5 Tampilan Beranda Library App sebelum ada data buku

Gambar diatas merupakan tampilan beranda awal pada aplikasi *Library App*, terlihat pada gambar masih menampilkan antarmuka yang masih kosong tanpa konten. Hal ini disebabkan oleh belum adanya data buku yang dimasukkan oleh pengguna ke dalam aplikasi. Pada tahap ini, antarmuka beranda memberikan kesan minimalis dan menunjukkan kepada pengguna bahwa koleksi buku mereka belum terisi. Melalui interaksi pengguna, aplikasi akan dapat membangun tampilan beranda dengan menampilkan secara terstruktur semua buku yang telah dimasukkan.

2. Tampilan Awal Fitur Tambah Buku

Library App Add

Tambah Buku

Judul Buku

Penulis Buku

Tahun Terbit

Jumlah Stok

Tambah Buku

© 2023 Library App. All rights reserved.
Teja Kusumah
301200033

Gambar IV 6 Tampilan pada fitur tambah buku sebelum diisi data

Tampilan pada gambar diatas merupakan antarmuka awal setelah pengguna mengklik fitur tambah buku. Pada tahap ini, elemen antarmuka masih kosong, karena pengguna belum menginputkan informasi buku secara manual. Fitur ini dirancang untuk memudahkan pengguna dalam menambahkan buku ke dalam koleksi aplikasi. Pengguna akan diminta untuk memasukkan data seperti judul buku, penulis buku, tahun terbit, dan jumlah buku yang dimiliki. Oleh karena itu pengguna dapat dengan mudah memulai proses penambahan buku dengan mengisi formulir yang disediakan.

3. Tampilan Akhir Fitur Tambah Buku

Library App

Tambah Buku

Judul Buku
Ayahku bukan pembohong

Penulis Buku
Tere Liye

Tahun Terbit
2018

Jumlah Stok
10

Tambah Buku

© 2023 Library App. All rights reserved.
Teja Kusumah
301200033

Gambar IV 7 Tampilan pada fitur tambah buku sesudah diisi data

Pada gambar di atas, terlihat tampilan akhir dari penggunaan fitur "tambah buku" dalam aplikasi Library App. Pada fase ini, formulir yang disediakan oleh aplikasi sudah diisi lengkap oleh pengguna dengan informasi seperti judul buku, penulis, tahun terbit, dan jumlah buku yang dimiliki. Setelah melengkapi formulir dengan data yang relevan, pengguna dapat melanjutkan ke langkah berikutnya dengan mengklik tombol "tambah buku" yang terletak di bagian bawah formulir.

Tombol "tambah buku" berfungsi untuk menyimpan data yang telah dimasukkan ke dalam sistem aplikasi. Setelah pengguna menyelesaikan langkah ini, data buku yang baru ditambahkan secara otomatis akan ada di beranda utama Library App begitupun seterusnya data-data buku yang sudah dimasukan oleh pengguna akan terekam dan tersimpan di halaman beranda utama sehingga pengguna bisa melihat data buku apa saja yang sudah dimasukan serta pengguna juga bisa mengecek kembali data-data buku yang sudah terekam dihalaman beranda apakah sudah benar atau ada yang salah ketika menginputkan data buku.

4. Tampilan Akhir Beranda Library App

Library App

Daftar Buku Perpustakaan

Judul	Penulis	Tahun Terbit	Jumlah	Action	
Laskar Pelangi	Andrea Hirata	2005	3	Edit	Hapus
Negeri 5 Menara	Ahmad Fuadi	2009	5	Edit	Hapus
Dilan: Dia adalah Dilan yang Tahu 1990	Pidi Baiq	2014	2	Edit	Hapus
Ayat-Ayat Cinta	Habiburrahman El Shirazy	2004	7	Edit	Hapus
Bumi Manusia	Pramoedya Ananta Toer	1980	4	Edit	Hapus
Laut Bercermin	Remy Sylado	1994	1	Edit	Hapus
Sang Pemimpi	Andrea Hirata	2006	6	Edit	Hapus
Pulang	Tere Liye	2016	9	Edit	Hapus
Perahu Kertas	Dee Lestari	2009	2	Edit	Hapus
Sepotong Hati yang Baru	Tere Liye	2010	3	Edit	Hapus
Katika Cinta Bertasbih	Habiburrahman El Shirazy	2004	4	Edit	Hapus

Gambar IV 8 Tampilan pada beranda Library App sesudah diisi data buku

Pada gambar diatas memperlihatkan tampilan akhir dari beranda Library App, yang kini telah diisi penuh dengan data buku yang sebelumnya dimasukkan oleh pengguna. Pada tampilan beranda ini, pengguna dapat dengan mudah meninjau dan memeriksa keakuratan data buku yang telah mereka input sebelumnya. Fungsionalitas ini memberikan kemudahan bagi pengguna untuk memverifikasi informasi dan mendeteksi potensi kekeliruan data.

Selain itu, beranda ini tidak hanya berfungsi memperlihatkan koleksi buku pengguna, tetapi juga sebagai pusat pengelolaan data. Pengguna dapat melakukan edit terhadap data buku jika menemui kesalahan atau perlu pembaruan informasi. Fitur ini memberikan fleksibilitas yang tinggi dalam pengelolaan koleksi buku sesuai dengan kebutuhan dan preferensi pengguna.

Tidak hanya itu, beranda Library App juga menyediakan opsi untuk menghapus data buku yang dianggap tidak diperlukan lagi. fitur ini memungkinkan pengguna untuk menjaga kebersihan dan relevansi koleksi buku mereka. Dengan demikian, beranda ini tidak hanya menjadi tampilan visual yang informatif tetapi juga menjadi pusat kendali yang efektif untuk manajemen dan penyempurnaan data buku dalam aplikasi Library App.

5. Tampilan Fitur Edit Buku

Library App Add

Edit Buku

Judul Buku
Laskar Pelangi

Penulis Buku
Andrea Hirata

Tahun Terbit
2005

Jumlah Stok
3

Simpan

© 2023 Library App. All rights reserved.
Teja Kusumah
301200033

Gambar IV 9 Tampilan pada fitur edit data buku

Pada gambar diatas menunjukkan fitur edit buku dalam aplikasi Library App. Fitur ini menjadi solusi praktis ketika pengguna melakukan kesalahan dalam memasukkan data buku, seperti kesalahan dalam nama buku, penulis, tahun terbit, atau jumlah stok buku. Fitur edit buku memberikan fleksibilitas kepada pengguna untuk merevisi dan memperbarui informasi yang telah dimasukkan sebelumnya.

Namun, fitur edit buku hanya muncul setelah pengguna berhasil memasukkan data buku dan informasi tersebut terekam di beranda utama Library App. Hal ini menekankan bahwa fitur ini menjadi relevan setelah data buku diinputkan oleh pengguna.

Dengan adanya fitur edit buku, pengguna dapat dengan mudah mengoreksi kesalahan atau melakukan pembaruan terhadap informasi buku tanpa harus menghapus dan memasukkan ulang data tersebut. Ini tidak hanya meningkatkan akurasi data di dalam aplikasi tetapi juga memberikan kontrol yang lebih besar kepada pengguna dalam mengelola dan menyelaraskan informasi buku mereka dengan kebenaran yang sebenarnya.

6. Tampilan Awal Fitur Hapus Buku

Bumi Manusia	Pramoedya Ananta Toer	1980	4	Edit	Hapus
Laut Bercermin	Remy Sylado	1994	1	Edit	Hapus
Sang Pemimpi	Andrea Hirata	2006	6	Edit	Hapus
Pulang	Tere Liye	2016	9	Edit	Hapus
Perahu Kertas	Dee Lestari	2009	2	Edit	Hapus
Sepotong Hati yang Baru	Tere Liye	2010	3	Edit	Hapus
Ketika Cinta Bertasbih	Habiburrahman El Shirazy	2004	4	Edit	Hapus
Cinta di Dalam Gelas	Andrea Hirata	2005	6	Edit	Hapus
Ayah	Andrea Hirata	2010	2	Edit	Hapus
Ronggeng Dukuh Paruk	Ahmad Tohari	1982	8	Edit	Hapus
Lelaki Harimau	Eka Kurniawan	2004	3	Edit	Hapus
Ayahku bukan pembohong	Tere Liye	2018	10	Edit	Hapus

© 2023 Library App. All rights reserved.
Teja Kusumah
301200033

Gambar IV 10 Tampilan fitur hapus data buku sebelum data buku dihapus

Dalam tampilan gambar diatas, kita dapat melihat tahapan awal sebelum data buku yang terletak di bagian paling bawah akan dihapus. Pada titik ini, gambar menunjukkan langkah-langkah sebelum penghapusan data buku dilakukan.

Pada tahapan awal ini terletak pada kemampuan aplikasi untuk pengguna terkait data buku yang akan dihapus. Sebelum keputusan penghapusan diambil, pengguna dapat secara jelas melihat dan memeriksa kembali informasi buku tersebut pada antarmuka. Proses ini memberikan kesempatan bagi pengguna untuk memvalidasi keputusan penghapusan dan memastikan bahwa data yang akan dihapus adalah yang diinginkan.

Tampilan awal ini menciptakan transparansi dalam proses pengelolaan data buku. Pengguna dapat merasa yakin dan memiliki kendali penuh terhadap keputusan yang diambil terkait dengan data koleksi buku mereka. Ini merupakan langkah proaktif yang dapat meningkatkan pengalaman pengguna dan memastikan bahwa proses penghapusan data buku dilakukan secara akurat dan sesuai dengan keinginan pengguna.

7. Tampilan Akhir Beranda Library App

Ayat-Ayat Cinta	Habiburrahman El Shirazy	2004	7	Edit	Hapus
Bumi Manusia	Pramoedya Ananta Toer	1980	4	Edit	Hapus
Laut Bercermin	Remy Sylado	1994	1	Edit	Hapus
Sang Pemimpi	Andrea Hirata	2006	6	Edit	Hapus
Pulang	Tere Liye	2016	9	Edit	Hapus
Perahu Kertas	Dee Lestari	2009	2	Edit	Hapus
Sepotong Hati yang Baru	Tere Liye	2010	3	Edit	Hapus
Ketika Cinta Bertasbih	Habiburrahman El Shirazy	2004	4	Edit	Hapus
Cinta di Dalam Gelas	Andrea Hirata	2005	6	Edit	Hapus
Ayah	Andrea Hirata	2010	2	Edit	Hapus
Ronggeng Dukuh Panuk	Ahmad Tohari	1982	8	Edit	Hapus
Lelaki Harimau	Eka Kumilawan	2004	3	Edit	Hapus

© 2023 Library App. All rights reserved.
Tjia Kusumah
301200033

Gambar IV 11 Tampilan fitur hapus data buku sesudah data buku dihapus

Dalam tampilan gambar di atas, kita dapat memerhatikan tahapan akhir setelah data buku yang terletak di bagian paling bawah dihapus. Dengan dihapusnya data tersebut, pengguna dapat mengamati perubahan signifikan pada tampilan beranda di aplikasi Library App. Data buku yang sebelumnya terpampang pada antarmuka utama akan secara instan menghilang, meninggalkan hanya data buku yang masih relevan dengan koleksi milik pengguna.

Perubahan ini memberikan kesan yang bersih dan terorganisir pada beranda aplikasi, memastikan bahwa pengguna hanya melihat data buku yang mereka pertahankan. Kejelasan ini penting untuk membantu pengguna dengan mudah mengidentifikasi dan menavigasi melalui koleksi buku mereka.

BAB V

PENUTUP

V.1 Kesimpulan dan Saran Mengenai Pelaksanaan

Berdasarkan uraian yang telah disampaikan pada bab-bab sebelumnya, dapat diambil kesimpulan bahwa:

V.1.1 Kesimpulan Pelaksanaan Kerja Praktik

1. Mahasiswa memiliki kesempatan untuk mengaplikasikan pengetahuan yang telah diperoleh selama kuliah dan kerja praktik dalam menangani masalah-masalah dunia nyata.
2. Mahasiswa dapat memperoleh pemahaman mendalam terkait pengetahuan dan keterampilan yang relevan untuk beradaptasi dengan lingkungan kerja global, seperti hal berikut ini:
 - a. Kemampuan berkomunikasi dan bekerja sama dalam kelompok
 - b. Pengetahuan fundamental dalam bidang khusus seperti informatika dan pemrograman
 - c. Keahlian dalam menganalisis masalah untuk menemukan solusi.
3. Mahasiswa juga diharapkan memiliki ilmu pengetahuan umum dan kemampuan untuk memperoleh pemahaman tentang hal-hal baru dalam waktu yang relatif singkat.
4. Ketika melibatkan diri dalam kerja praktik, mahasiswa menjadi lebih sadar akan pentingnya etos kerja yang baik, disiplin, dan tanggung jawab dalam menyelesaikan tugas.
5. Kolaborasi dalam tim, baik dengan sesama peserta kerja praktik maupun dengan mentor dan pengajar di PT Ruang Raya Indonesia, memungkinkan mereka mengembangkan keterampilan interpersonal dan memahami dinamika kerja tim.

6. Melalui kerja praktik di PT Ruang Raya Indonesia, mahasiswa mendapatkan pengetahuan tambahan yang tidak selalu dapat diperoleh melalui perkuliahan. Mereka mendapatkan wawasan lebih mengenai:
 - a. Dasar-dasar bahasa pemrograman sebagai seorang frontend engineering
 - b. Penerapan JSON pada Basic Frontend Web Development
 - c. Pengembangan aplikasi library app menggunakan JSON.

Hal Ini memberikan pengalaman langsung yang berharga untuk melengkapi pemahaman akademis mereka dengan praktik nyata di dunia industri.

V.1.2 Saran Pelaksanaan Kerja Praktik

1. Mahasiswa yang sedang menjalani kerja praktik di PT Ruang Raya Indonesia diharapkan untuk secara proaktif meningkatkan kualitas diri mereka. Hal ini tidak hanya bertujuan agar pengalaman kerja praktik tidak terbuang sia-sia, tetapi juga untuk memastikan bahwa pengetahuan dan keterampilan yang diperoleh dapat terus berkembang dan bermanfaat di masa depan.
2. Selain itu, setelah menyelesaikan kerja praktik, mahasiswa disarankan untuk terus melatih diri dengan menggunakan studi kasus. Dengan melakukan ini, mereka dapat mengaplikasikan pengetahuan yang telah mereka peroleh dalam situasi nyata di dunia industri, sehingga memperdalam pemahaman mereka dan meningkatkan keterampilan mereka.
3. Mahasiswa juga diharapkan untuk aktif berkomunikasi dan bertanya kepada dosen, rekan sekelas, dan profesional di bidang yang relevan. Bertukar pikiran dan meminta bimbingan merupakan cara efektif untuk memecahkan masalah, mendapatkan sudut pandang baru, dan memperluas pengetahuan mereka dalam bidang yang mereka geluti.

4. Selain itu, bimbingan intensif selama kerja praktik sangat penting bagi mahasiswa. Bimbingan ini membantu mereka memahami tugas dan tanggung jawab dengan lebih baik, sehingga dapat mengoptimalkan kontribusi mereka dalam lingkungan kerja dan mencegah terjadinya hambatan dalam proyek-proyek yang sedang berjalan.
5. Terakhir, mahasiswa juga diharapkan untuk memiliki sikap yang terbuka terhadap pembelajaran dan perbaikan diri. Menerima kritik dan saran dengan lapang dada, belajar dari kesalahan, dan selalu mencari cara untuk meningkatkan diri adalah kunci kesuksesan dalam karier profesional maupun akademis. Dengan mempertahankan sikap ini, mahasiswa dapat terus berkembang dan menjadi profesional yang tangguh di masa depan.

V.2 Kesimpulan dan Saran Mengenai Substansi

Dari observasi yang dilakukan selama kerja praktik di PT Ruang Raya Indonesia, dapat diambil kesimpulan dan saran mengenai substansi sebagai berikut:

V.2.1 Kesimpulan

Setelah menyelesaikan program kerja praktik yang melibatkan serangkaian tahapan, mulai dari aspek pengenalan terhadap lingkungan kerja hingga penerapan materi yang telah dipelajari, mahasiswa menjalani pengalaman praktik langsung dalam menyelesaikan sebuah studi kasus yang diberikan. Dari rangkaian aktivitas ini, terciptalah sebuah aplikasi berbasis website yang diberi nama "*Library App*" oleh PT Ruang Raya Indonesia. Aplikasi ini dibangun dengan menggunakan teknologi JSON untuk mengelola dan merekam data buku, dengan tujuan utama untuk memenuhi beberapa fungsi, antara lain:

1. Memudahkan akses data buku dari jarak jauh yang artinya dalam era digital yang terus berkembang, kemudahan akses menjadi hal yang sangat diharapkan oleh pengguna. Dengan "*Library App*", pengguna dapat mengakses informasi tentang koleksi buku dari mana pun mereka berada. Baik itu dari rumah,

perjalanan, atau bahkan di tempat lain yang jauh dari lokasi fisik perpustakaan. Keleluasaan ini memberikan fleksibilitas bagi pengguna untuk mengelola koleksi buku tanpa batasan tertentu.

2. Memudahkan integrasi dengan sistem lain yang artinya keberhasilan sebuah aplikasi tidak hanya terletak pada fungsionalitasnya sendiri, tetapi juga pada kemampuannya untuk berinteraksi dengan sistem lain. "*Library App*" didesain dengan fokus pada kemudahan integrasi dengan sistem-sistem lain yang digunakan dalam konteks perpustakaan atau entitas terkait lainnya. sehingga aplikasi ini mengoptimalkan efisiensi operasional dan memperluas jangkauan fungsionalitasnya.
3. Mempermudah analisis data buku untuk pengambilan keputusan yang artinya melalui informasi-informasi yang disediakan pada aplikasi ini, "*Library App*" diharapkan mampu memberikan wawasan yang berharga bagi pengguna dalam mengelola koleksi buku. Analisis yang mendalam tentang informasi data buku, preferensi pengguna, dan kebutuhan pembaca dapat membantu pihak terkait mengambil keputusan yang lebih tepat dan strategis. Aplikasi ini juga menjadi sumber informasi berharga untuk pengembangan dan pengelolaan koleksi buku yang lebih efektif dan responsif terhadap kebutuhan pengguna.

V.2.2 Saran

Dari hasil kegiatan kerja praktik ini, mahasiswa yang mengikuti program di PT Ruang Raya Indonesia ingin menyampaikan beberapa saran kepada pihak mitra. Berikut adalah beberapa saran yang diajukan:

1. Mahasiswa berharap agar mitra akan memberikan respons yang lebih cepat terhadap pertanyaan yang diajukan oleh peserta. Respon yang cepat ini sangat penting dalam menanggapi aspek-aspek rinci terkait pengerjaan laporan kerja praktik dan elemen-elemen lain yang membutuhkan klarifikasi atau panduan lebih lanjut. Dengan ketersediaan respon yang cepat, diharapkan mahasiswa

dapat melanjutkan proyek tanpa hambatan berarti, dan mengoptimalkan efisiensi dalam mencapai hasil akhir.

2. Terkait materi pembelajaran yang disampaikan, mahasiswa berharap agar mendapatkan informasi yang lebih terperinci dan relevan dengan studi kasus yang diberikan. Dengan demikian, diharapkan mahasiswa tidak akan merasa kebingungan dan dapat memahami materi secara lebih mendalam saat mengikuti kegiatan kerja praktik.
3. Selain itu, mahasiswa berharap agar informasi terkait studi kasus yang diberikan menjadi lebih rinci. Hal ini bertujuan untuk menghindari kebingungan mahasiswa dalam menjalankan instruksi yang diberikan oleh pihak mitra. Dengan detail yang lebih lengkap, mahasiswa dapat memahami tugas dengan lebih baik dan menghasilkan solusi yang lebih tepat sesuai dengan kebutuhan proyek.
4. Mahasiswa juga berharap agar sistem yang digunakan untuk pengujian aplikasi ditingkatkan kualitasnya. Hal ini akan mencegah terjadinya kesalahan dan masalah teknis saat mahasiswa melakukan pengujian. Dengan sistem pengujian yang lebih baik, diharapkan proses pengujian aplikasi dapat berjalan lancar dan efisien, tanpa mengalami hambatan atau error yang disebabkan oleh kelemahan sistem yang ada.

DAFTAR PUSTAKA

- A., F. (2023). Apa Itu GitHub? Kenali Pengertian dan Fungsinya. Retrieved from <https://www.hostinger.co.id/tutorial/apa-itu-github>
- Apa Itu MongoDB? Pengertian, Keunggulan, dan Cara Installnya. (1969). Retrieved from <https://www.biznetgio.com/news/mongodb-adalah>
- Apa itu Tailwind? Pengertian dan contoh 2023. (n.d.). Retrieved from <https://revou.co/kosakata/tailwind>
- Apa itu Visual Studio Code? Pengertian dan contoh 2023. (n.d.). Retrieved from <https://revou.co/kosakata/visual-studio-code>
- Kemendikbud.2022. Buku Panduan Merdeka Belajar - Kampus Merdeka. Jakarta: Direktorat Jenderal Pendidikan Tinggi Kemendikbud R .
- Komputer, U. S. & T. (2008). Google Chrome. Retrieved from https://p2k.stekom.ac.id/ensiklopedia/Google_Chrome
- Mengenal Apa Itu CSS: Pengertian, Fungsi, dan Jenis-jenisnya. (1969). Retrieved from <https://www.biznetgio.com/news/apa-itu-css>
- SemiColonWeb. (n.d.). Berita. Retrieved from <https://if.unpas.ac.id/berita/apa-itu-javascript-pengertian-sejarah-dan-bagaimana-cara-kerjanya/>
- Tutorial Node.js: Apa itu Node.js dan dasar-dasar Node.js. (2021). Retrieved from <https://www.dicoding.com/blog/apa-itu-node-js/>
- Varty, J. (2023). Netlify & How it Works: Benefits for Users: Agility. Retrieved from <https://agilitycms.com/resources/posts/what-is-netlify-and-why-should-you-care-as-an-editor>

LAMPIRAN A.
TERM OF REVERENCE (TOR)

Tabel 1 Term Of Reverence

No	Judul Kegiatan	Detail Kegiatan	Alokasi Waktu	Tujuan Pembelajaran	SKS Didapat
1	General Competence	<p>Peserta akan melakukan pembelajaran asynchronous dan synchronous dengan rinciannya sebagai berikut:</p> <ul style="list-style-type: none"> • Mentoring • Membaca materi • Mengerjakan pre-test dan post-test • Mengerjakan exercise/assignment/mid-test • Mengerjakan final course project 	90 Jam	<p>Pembelajaran tambahan yang dapat mendukung kemampuan peserta yang meliputi:</p> <ol style="list-style-type: none"> 1) Communication and Presentation Skill 2) Analytical & Strategic Thinking Skill 	2 SKS
2	Software Engineering	<p>Peserta akan melakukan pembelajaran asynchronous dan synchronous dengan rinciannya sebagai berikut:</p> <ul style="list-style-type: none"> • Mentoring • Membaca materi • Mengerjakan pre-test dan post-test • Mengerjakan exercise/assignment/mid-test • Mengerjakan final course project 	135 Jam	<p>Menerapkan proses pengembangan software dengan menggunakan Code Editor Visual Studio Code, Terminal, version control, dan basic networking</p>	3 SKS
3	Pemograman Frontend Dasar w/ Javascript	<p>Peserta akan melakukan pembelajaran asynchronous dan synchronous dengan</p>	180 Jam	<p>Memanfaatkan ekosistem development Node JS untuk mengembangka</p>	4 SKS

		<p>rinciannya sebagai berikut:</p> <ul style="list-style-type: none"> • Mentoring • Membaca materi • Mengerjakan pre-test dan post-test • Mengerjakan exercise/assignment/mid-test • Mengerjakan final course project 		<p>n aplikasi berbasis Javascript serta memahami dan menerapkan konsep konsep pemrograman pada bahasa pemrograman Javascript</p>	
4	Pemograman Frontend Lanjutan w/ Javascript	<p>Peserta akan melakukan pembelajaran asynchronous dan synchronous dengan rinciannya sebagai berikut:</p> <ul style="list-style-type: none"> • Mentoring • Membaca materi • Mengerjakan pre-test dan post-test • Mengerjakan exercise/assignment/mid-test • Mengerjakan final course project 	90 Jam	Memahami konsep dasar OOP serta penerapan proses Asynchronous di Javascript dan memahami fundamental arsitektur RESTful API dan menerapkannya dalam pengembangan website di sisi frontend development.	2 SKS
5	Basic Frontend Development	<p>Peserta akan melakukan pembelajaran asynchronous dan synchronous dengan rinciannya sebagai berikut:</p> <ul style="list-style-type: none"> • Mentoring • Membaca materi • Mengerjakan pre-test dan post-test • Mengerjakan exercise/assignment/mid-test 	180 Jam	Memahami penulisan HTML dan penggunaan CSS dalam Web Development dasar serta memahami bagaimana aplikasi website berjalan dan ditampilkan dari sisi klien	4 SKS

		<ul style="list-style-type: none"> • Mengerjakan final course project 			
6	Dasar Pemogram an Frontend dengan React	<p>Peserta akan melakukan pembelajaran asynchronous dan synchronous dengan rinciannya sebagai berikut:</p> <ul style="list-style-type: none"> • Mentoring • Membaca materi • Mengerjakan pre-test dan post-test • Mengerjakan exercise/assignm en t/mid-test • Mengerjakan final course project 	135 Jam	Memahami fundamental library ReactJS (JSX, Component, Props, State, LifeCycle, Handling Event) dan menggunakan ReactJS untuk membuat Single Page Application	3 SKS
7	Dasar Pemogram an Frontend dengan React Lanjutan	<p>Peserta akan melakukan pembelajaran asynchronous dan synchronous dengan rinciannya sebagai berikut:</p> <ul style="list-style-type: none"> • Mentoring • Membaca materi • Mengerjakan pre-test dan post-test • Mengerjakan xercise/assignme n t/mid-test • Mengerjakan final course project 	90 Jam	Memahami dan menerapkan konsep state management pada ReactJS serta mempelajari penggunaan library Chakra UI. Selain itu juga memahami cara deployment aplikasi ReactJS yang sudah dibuat sehingga bisa diakses online dengan menggunakan layanan Netlify / Vercel / Github Pages.	2 SKS
Jumlah SKS					20 SKS

a. Struktur Data:

Implementasi struktur data yang efisien untuk merepresentasikan informasi buku. Penggunaan DOM untuk memanipulasi dan merepresentasikan data pada halaman web.

b. Komunikasi Client-Server:

Perancangan mekanisme komunikasi yang efektif antara client dan server menggunakan API Server. Pemahaman yang baik terhadap endpoint-endpoint seperti GET, POST, PUT, dan DELETE untuk operasi CRUD.

c. Implementasi Operasi CRUD:

Rancangan fungsi-fungsi untuk setiap operasi CRUD (Create, Read, Update, Delete) dengan memastikan integritas data dan respons real-time.

d. Antarmuka Pengguna:

Perancangan antarmuka pengguna yang responsif dan intuitif. Termasuk transisi antara tampilan tabel dan formulir untuk menambah atau mengubah data buku.

e. Keamanan dan Validasi:

Penerapan validasi input pada formulir dan pertimbangan keamanan data saat melakukan operasi CRUD, termasuk manajemen error dengan baik.

f. Optimalisasi Kinerja:

Menerapkan strategi optimalisasi kinerja untuk memastikan respons cepat dan efisiensi penggunaan sumber daya.

g. Deployment:

Merencanakan proses deployment ke Netlify dengan memastikan konfigurasi yang tepat, dan memasukkan informasi penting seperti nama dan CAMPID pada file `deployData.js`.

h. Uji Coba dan Debugging:

Perancangan rencana uji coba yang mencakup skenario penggunaan untuk memastikan fungsionalitas yang baik. Menyiapkan strategi debugging untuk mengatasi potensi masalah.

Dengan memperhatikan aspek-aspek di atas, perancangan perangkat lunak ini mencakup elemen-elemen kunci dalam mengembangkan website perpustakaan dengan fitur CRUD

LAMPIRAN B.
LOG ACTIVITY

Tabel 2 Log Activity

Minggu/Tgl	Kegiatan	Hasil
1/ 08-05- 2023 sd 10- 05-2023	mempelajari HTML, CSS mengerjakan assignment 1 dan 2 di materi pada basic frontend development	memahami dan terus mempelajari materi dasar pengembangan frontend seperti HTML, dan CSS. Mengerjakan assignment 1 dan 2
2/ 13-05- 2023 sd 15- 05-2023	mempelajari DOM mempelajari Browser API	memahami dan terus mempelajari materi dasar pengembangan frontend seperti DOM dan Browser API
3/ 17-05- 2023 sd 21- 05-2023	Mempelajari JSON	memahami dan terus mempelajari materi dasar pengembangan frontend seperti JSON
4/ 23-05- 2023 sd 29- 05-2023	mengerjakan final project basic frontend development	memahami dan terus mempelajari materi dasar pengembangan frontend dan mengerjakan final project aplikasi web library app menggunakan JSON