

## IMPLEMENTASI FRONTEND DINAMIS HALAMAN DASHBOARD DAN LOGIN PADA APLIKASI PEMINJAMAN RUANGAN

I. P. A. Wijaya<sup>1</sup>, I. G. S. Astawa<sup>2</sup>, G. A. V. M. Giri<sup>3</sup>

### ABSTRAK

Peminjaman ruangan kebanyakan dilakukan secara manual, misalnya dengan surat menyurat. Dengan sistem manual tersebut, banyak hal menjadi tidak efisien, mulai dari masalah keaslian surat, pencatatan yang tidak terorganisir, terjadinya miskomunikasi, hingga tabrakan waktu peimjaman dengan peminjam yang lain. Contohnya adalah peminjaman ruangan di universitas untuk pembelajaran tatap muka, acara seminar, ataupun dipinjam oleh salah satu organisasi mahasiswa untuk berkegiatan. Hal ini yang menjadi permasalahan/ide bagi Avatar Solution, sebuah perusahaan yang bergerak di bidang jasa pembuatan software, untuk membuat website peminjaman ruangan yang bersifat online, *realtime* dengan menggunakan teknologi React.js yang langsung memperbaharui data secara otomatis pada setiap komponennya dengan harapan peminjaman ruangan dapat dilakukan dengan cepat dan efisien.

**Kata kunci :** peminjaman, ruangan, *website*, *reactjs*, *frontend*.

### ABSTRACT

*Room borrowing is mostly done manually, for example by correspondence. With this manual system, many things become inefficient, starting from issues of authenticity of letters, disorganized record keeping, miscommunication, to collisions with other borrowers. An example is borrowing a room at a university for face-to-face learning, seminars, or being borrowed by a student organization for activities. This is a problem (idea) for Avatar Solution, a company engaged in the field of software development services, to create an online, real-time room lending website using React.js technology which automatically updates data on each of its components in the hope that room borrowing can be done quickly and efficiently.*

**Keywords:** *lending, space, website, reactjs, frontend.*

---

<sup>1</sup> Informatika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Udayana, Jalan Raya Kampus Unud Jimbaran, 80361, Badung-Indonesia, anantaw81@gmail.com

<sup>2</sup> Informatika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Udayana, Jalan Raya Kampus Unud Jimbaran, 80361, Badung-Indonesia, santi.astawa@cs.unud.ac.id

<sup>3</sup> Informatika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Udayana, Jalan Raya Kampus Unud Jimbaran, 80361, Badung-Indonesia, vida@unud.ac.id

Submitted: 7 November 2022

Revised: 25 November 2022

Accepted: 27 November 2022

Submitted:

## 1. PENDAHULUAN

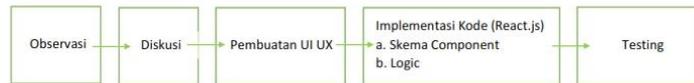
Avatar Solutions adalah perusahaan yang berfokus pada solusi manajemen informasi komprehensif yang menggabungkan paket perangkat keras dan perangkat lunak lengkap dengan dukungan teknis yang andal. Didirikan pada tahun 2009, perusahaan ini memberikan jasa pembuatan *software/website* sesuai dengan permintaan klien atau membuat aplikasi yang nantinya bisa ditawarkan ke klien. Website dapat diartikan sebagai kumpulan halaman yang berisi informasi data digital baik berupa teks, gambar, animasi, suara dan video atau gabungan dari semuanya yang disediakan melalui jalur koneksi internet (Abdulloh, 2018). Perusahaan ini membuka *internship* untuk bergabung dan masuk pada banyak pilihan divisi, saya salah satunya masuk ke web developer. Web Developer bertanggung jawab dalam mengembangkan bagian tampilan (*front-end*). Proyek yang saya bantu buat adalah sistem peminjaman ruangan dengan penampilan data yang dinamis, lebih spesifiknya pada pembuatan halaman login dan dashboard.

Peminjaman ruangan adalah suatu proses kegiatan meminjam atau meminjamkan ruangan untuk digunakan oleh orang yang melakukan peminjaman dalam waktu yang telah ditentukan (Dhanardana, N. A. D., et.al., 2020). Seringkali kita temui proses peminjaman ruangan yang masih manual, contohnya menggunakan surat. Proses peminjamannya pun relatif terbilang lama dengan menunggu pengesahan pada surat dan mengecek ulang kembali surat yang telah dibuat. Selain itu, banyak juga kendala seperti keaslian surat, peminjaman yang tidak terorganisir, terjadinya miskomunikasi. Hal itu dapat diselesaikan dengan perkembangan teknologi. Salah satunya dengan membuat website peminjaman, yang memiliki kecepatan, kehandalan, dan data yang mudah dikelola. Salah satu kasusnya adalah universitas, dimana pasti memiliki ruangan yang dikelola. Ruangan tersebut biasa digunakan untuk pembelajaran tatap muka, acara seminar, ataupun dipinjam oleh organisasi mahasiswa untuk berkegiatan. Implementasi peminjaman ruangan ini masih terbilang manual. Berbagai pihak pun terlibat secara tatap muka hanya untuk proses ini, mulai dari staf kebersihan, dosen pembina, mahasiswa (Raka, A. A., 2018). Proses pencatatan pun sedikit menyita waktu dikarenakan harus membuat surat manual terlebih dahulu. Alhasil, waktu dari proses hingga bisa meminjam ruangan melebihi 24 jam.

Pada penelitian mengenai Perancangan Sistem Informasi Peminjaman dan Pengembalian Berkas Rekam Medis Rawat Jalan di Rumah Sakit Khusus Gigi dan Mulut Kota Bandung oleh (Khalifatullah, A. J., et. al., 2013), berhasil mempermudah peminjaman barang dengan menggunakan teknologi dan meningkatkan sistem kerja dari rumah sakit. Oleh karena itu, permasalahan peminjaman ruangan yang telah disebutkan di atas dibuat rancangan *front-end* untuk implementasi sistem serta ditekankan untuk penampilan data yang dinamis dengan menggunakan React.js. Proyek ini nantinya akan diserahkan ke Avatar Solution dengan sebelumnya menunjukkan hasil manual testing website sehingga bisa menunjukkan apakah website berjalan lancar.

## 2. METODE PELAKSANAAN

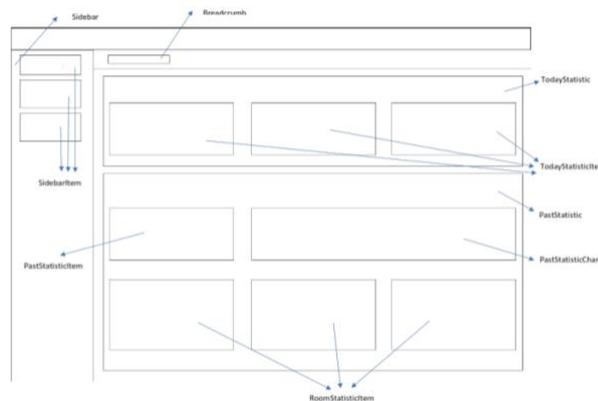
Website dibuat menggunakan metode observasi, diskusi, implementasi, testing. Diagram metode pelaksanaan dapat dilihat pada Gambar 2.1. Pertama-tama, akan dilakukan observasi terkait fitur aplikasi, skema aplikasi, dan tampilan yang akan dibuat. Observasi yaitu metode penelitian yang menggunakan cara melihat secara langsung (mengamati) proses yang sedang berjalan saat ini (Romindo, et.al., 2022). Setelah itu, dilakukan diskusi dengan seluruh tim terkait untuk penambahan atau pengurangan fitur untuk mencapai target aplikasi yang minimal. Setelah itu, tim UI UX membuat desain di Figma. Figma adalah aplikasi desain berbasis cloud dan alat prototyping untuk proyek digital (Pramudita, R., et.al., 2021). Implementasi kode dilakukan menggunakan framework React.js (javascript). Konsep React.js adalah komponen, jadi dibuatkan dahulu skema komponen yang menangani masing-masing bagian di tampilan. Setelah itu, ditambahkan logic dengan menghubungkan ke API. Setelah kedua langkah tersebut selesai, dilakukan testing untuk mengecek fitur sudah bisa berjalan bersama Avatar Solution.



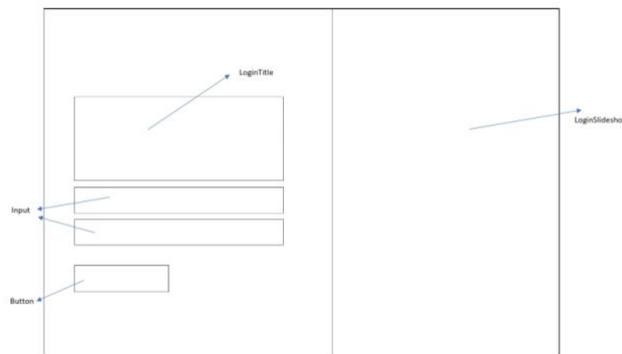
**Gambar 2.1.** Diagram Metode Pelaksanaan

### 3. HASIL DAN PEMBAHASAN

Front-End website dikerjakan melalui aplikasi Visual Studio Code dengan menggunakan ReactJS serta framework Bootstrap sebagai pendukung *styling*. Bootstrap mengadopsi teknik grid system untuk mengatur ukuran tampilan layar (Putra, 2020). Pada tampilan admin, setelah melakukan login ke dalam sistem maka akan dialihkan ke halaman dashboard. Kedua halaman ini bersifat dinamis karena dibuat dengan React.js. Pada proses pembuatan, karena React.js menggunakan konsep komponen, pertama harus dibuat dahulu pembagian komponennya seperti pada gambar 3.1 dan gambar 3.2. Pada gambar 3.1, halaman ‘Dashboard’ memiliki anak komponen, yaitu Sidebar dengan anak komponennya SidebarItem, Breadcrumb, TodayStatistic dengan anak komponennya TodayStatisticItem, PastStatistic dengan anak komponennya PastStatisticItem dan PastStatisticChart, RoomStatisticItem. Pada gambar 3.2, halaman ‘Login’ memiliki anak komponen, yaitu LoginSlideshow, LoginTitle, Input, Button.



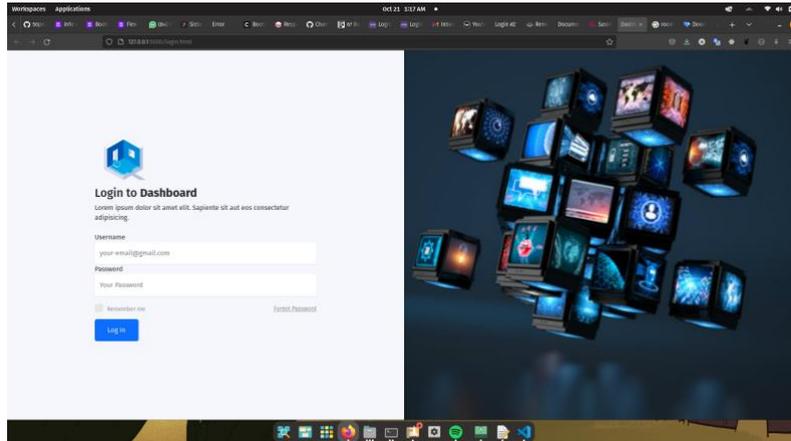
**Gambar 3.1.** Pembagian Komponen Dashboard



**Gambar 3.2.** Pembagian Komponen Login

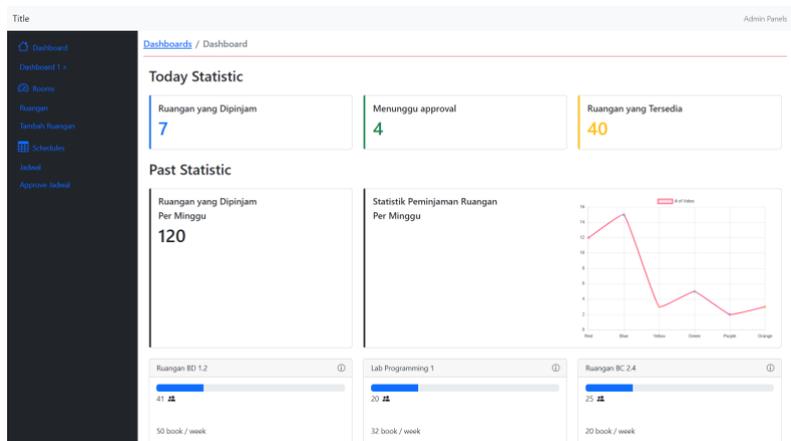
### Implementasi Frontend Dinamis Halaman Dashboard dan Login Pada Aplikasi Peminjaman Ruangan

Setelah itu, implementasinya dilakukan dengan membuat setiap file-file komponen dalam setiap folder page. Jadi, terdapat dua folder yaitu 'Dashboard' dan 'Login'. Konsep komponen ini akan membuat perulangan di setiap induk komponen saat ada anak komponen yang harus ditampilkan berkali-kali. Kelebihan konsep komponen ini adalah jika terdapat perubahan API ke satu komponen, maka hanya komponen itu dan anak komponen yang dipanggil di dalamnya yang akan terjadi render (*refresh*). Dengan adanya hal ini, React.js membuat pengguna tidak perlu melakukan klik pada tombol refresh di browser sehingga performa yang ada terbilang cepat. Selain itu React JS berbasis komponen yang bisa digunakan kembali jika dibutuhkan hal ini dapat menghemat waktu pengembangan startup (Nasution, 2021).



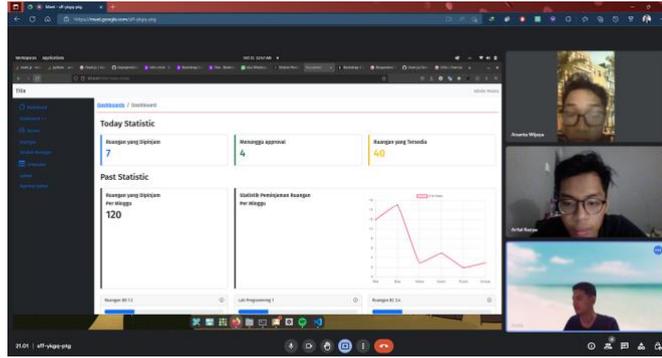
Gambar 3.3. Halaman Login

Hasil implementasi kode terlihat dari (Gambar 3.3). Halaman login perlu memasukkan username dan password sehingga pengguna bisa masuk ke halaman dashboard.

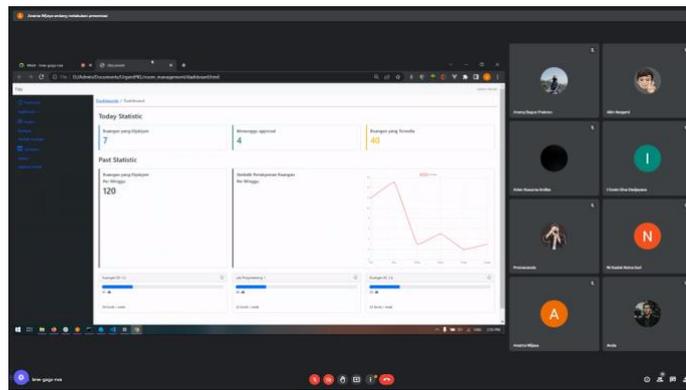


Gambar 3.4. Halaman Dashboard

Pada halaman Dashboard, ditampilkan secara singkat statistik data yang ada pada sistem ini. Statistik yang ditampilkan adalah Today Statistic dengan Past Statistic. Today Statistic menampilkan data ruangan yang dipinjam, memerlukan approval, dan ruangan yang tersedia pada hari itu. Past Statistic menampilkan ruangan yang dipinjam tiap minggu, serta diagram chartnya, setiap ruangan yang ada dengan kapasitasnya dan jumlah peminjam setiap minggu. Setelah itu, dibuat logic untuk mengintegrasikan data dari API menggunakan pustaka axios.



Gambar 3.4. Testing Program Bersama Tim Front-End



Gambar 3.5. Manual Testing Bersama Mentor dan Tim Lain

Test Scenario ID : Login-1							Test Scenario ID : Dashboard-1								
Test Case Description : Login-Negative test case							Test Case Description : Dashboard-Negative test case								
Pre-Requsite : NA							Pre-Requsite : Login success								
Post Requisite : NA							Post Requisite : NA								
S.No	Action	Inputs	Expected Output	Actual Output	Test Browser	Test Result	Test Comments	S.No	Action	Inputs	Expected Output	Actual Output	Test Browser	Test Result	Test Comments
1	Memasukkan email yang invalid dan password apapun. Setelah itu, klik tombol login.	Email id : invalid@xyz.com Password: *****	Alamat email yang dimasukkan tidak cocok dengan akun apapun. Sign up untuk membuat akun	Alamat email yang dimasukkan tidak cocok dengan akun apapun. Sign up untuk membuat akun	Chrome 61-106	Pass	[Ananta 11/21/2022] : Percobaan login gagal.	1	Today statistic dan past statistic menampilkan nilai yang sebenarnya.	Data diambil dari API	Data yang didapatkan dari API ditampilkan dengan benar.	Data yang didapatkan dari API ditampilkan dengan benar.	Chrome 61-106	Pass	[Ananta 11/21/2022]: Data sesuai dengan API
2	Memasukkan email yang valid dan password yang salah. Setelah itu klik tombol login.	Email id: valid@xyz.com Password: *****	Password yang dimasukkan salah. Lupa password?	Password yang dimasukkan salah. Lupa password?	Chrome 61-106	Pass	[Ananta 11/21/2022] : Percobaan login gagal.	2	Chart pada past statistic menampilkan diagram yang sesuai.	Data diambil dari API	Chart menampilkan data sesuai.	Chart menampilkan data sesuai.	Chrome 61-106	Pass	[Ananta 11/21/2022]: Chart sudah sesuai dengan API
3	Memasukkan email yang valid dan password yang valid. Setelah itu klik tombol login.	Email id: valid@xyz.com Password: *****	Login berhasil, mengalihkan ke laman. Laman dialihkan ke dashboard.	Login berhasil, mengalihkan ke laman. Laman dialihkan ke dashboard.	Chrome 61-106	Pass	[Ananta 11/21/2022] : Percobaan login berhasil.	3	Komponen diperbaharui saat data berubah tanpa memperbaharui keseluruhan laman.	Data dari API berubah.	Komponen yang ditampilkan memperbaharui data.	Komponen yang ditampilkan memperbaharui data.	Chrome 61-106	Pass	[Ananta 11/21/2022]: Komponen dengan cepat memperbaharui data.
4	Mengubah ukuran laman ke mobile	Laman diubah ke ukuran lebar <768px	Gambar slide hilang, form login masih tetap ada.	Gambar slide hilang, form login masih tetap ada.	Chrome 61-106	Pass	[Ananta 11/21/2022] : Tampilan sesuai diharapkan.	4	Mengubah ukuran laman ke mobile	Laman diubah ke ukuran lebar <768px di browser	Sidebar menghilang, hanya tersedia tombol. Item card berjejer vertical.	Sidebar menghilang, hanya tersedia tombol. Item card berjejer vertical.	Chrome 61-106	Pass	[Ananta 11/21/2022]: Tampilan sesuai diharapkan.

Gambar 3.6. Test Cases dalam Manual Testing Website

Testing dilakukan dua kali, pertama dengan menunjukkan fitur yang telah selesai kepada satu tim yang membuat frontend. Pada bagian ini, diberikan saran serta masukan jika ada yang belum berhasil. Testing kedua dilakukan di akhir yaitu *manual testing* bersama mentor (gambar 3.5) dari Avatar Solution serta Tim selain Web Developer. *Manual testing* adalah sebuah teknik testing

dimana tester menyiapkan *test cases* secara manual dan mengeksekusi *test cases* untuk mengidentifikasi defect di software (Min, J. L., 2020). Daftar *test cases*-nya ditunjukkan gambar 3.6. Seluruh hasil testing sesuai dengan fitur yang ada dikatakan berhasil berjalan dengan baik. Setelah itu, proyek diberikan kepada Avatar Solution dan diterima dengan baik.

#### **4. KESIMPULAN**

Pembuatan sistem ini dengan tujuan memudahkan pengaturan ruangan, dimana awalnya kita harus mengecek ke ruangan langsung untuk memantau apakah ruangan tersebut digunakan, menjadi pengecekan secara online dengan data yang *realtime*. Pembuatan halaman dashboard dan login, digunakan sebuah framework React.js yang bisa membuat sebuah situs website bersifat dinamis dengan menggunakan konsep komponen. Website berhasil berjalan dengan baik sesuai hasil akhir testing bersama mentor dari Avatar Solution.

#### **UCAPAN TERIMA KASIH**

Terima kasih penulis ucapkan kepada tim dan sub-tim yang sudah bekerja sama. Selain itu, penulis mengucapkan terimakasih kepada mentor dari Avatar Solutions yang telah menerima dan memberikan pengalaman berharga selama kegiatan praktek kerja lapangan yang berlangsung selama dua bulan.

#### **DAFTAR PUSTAKA**

- Abdulloh. (2018). *Pemrograman Web Untuk Pemula*. Jakarta: PT Elex Media Komputindo.
- Dhanardana, N. A. D., Rahayu, T. (2020). *APLIKASI PEMINJAMAN RUANGAN BERBASIS WEB PADA DIREKTORAT JENDERAL HAK ASASI MANUSIA*. **Vol. -: No. -**, 211-221.
- Hartanti, D., Pratama, B., and Handayani, D. (2021). Perancangan Aplikasi Pengendalian Ruang Kelas Berbasis Mobile. **Vol. 2: No. 1**, 1-10.
- Khalifatullah, A. J., Yanuar, Y. (2021). *Perancangan Sistem Informasi Peminjaman dan Pengembalian Berkas Rekam Medis Rawat Jalan di Rumah Sakit Khusus Gigi dan Mulut Kota Bandung*. **Vol. 4: No. 1**, 1-11.
- Min, J. L., Istiqomah, A., and Rahmani, Ani. (2020). *Evaluasi Penggunaan Manual dan Automated Software Testing pada Pelaksanaan End-to-end Testing*. **Vol. 6: No. 1**, 18-25.
- Nasution and Iswari, L. (2021). *Penerapan React JS Pada Pengembangan FrontEnd Aplikasi Startup Ubaform*. **Vol. 2: No. 2**, -.
- Pramudita, R., Arifin, R. W., Alfian, A. N., Safitri, N., and Anwariya, S. D. (2021). *Penggunaan Aplikasi Figma dalam Membangun UI/UX yang Interaktif pada Program Studi Teknik Informatika Stmik Tasikmalaya*. **Vol. 3: No. 1**, 149-154.
- Putra, M. Y. (2020). *Responsive Web Design Menggunakan Bootstrap dalam Merancang Layout Website*. **Vol. 5: No. 2**, 61-70.
- Raka, A. A., Maulana, M. H., Andini, C. D., Nadziroh, F. (2018). *SISTEM PEMINJAMAN RUANGAN ONLINE (SPRO) DENGAN METODE UML (UNFIELD MODELING LANGUAGE)*. **Vol. 1: No. 1**, 1-8.
- Romindo and Christine. (2022). *Penerapan Model SDLC Terhadap Sistem Informasi Penjualan dan Persediaan Bangunan pada Cv. Nilafa*. **Vol. 7: No. 1**, 62-72.