

Rancang Bangun *Website* Peminjaman Ruangan Dengan *Framework* Laravel Di Telkom University Purwokerto

1st Nabila Afifatul Fajri
Directorate of Campus Banyumas
Telkom University
Purwokerto, Indonesia
nabilaafifatulfajri@student.telkomuniversity.ac.id

2nd Yohani Setiya Rafika Nur
Directorate of Campus Banyumas
Telkom University
Purwokerto, Indonesia
yohanin@telkomuniversity.ac.id

3rd Shintia Dwi Alika
Directorate of Campus Banyumas
Telkom University
Purwokerto, Indonesia
shintiaa@telkomuniversity.ac.id

Abstrak—Telkom University Purwokerto merupakan salah satu perguruan tinggi di Indonesia yang berfokus pada bidang teknologi informasi dan komunikasi. Namun, pengelolaan peminjaman ruangan masih sering dilakukan secara konvensional, yaitu melalui sistem pemesanan manual yang melibatkan proses administratif yang mengharuskan mahasiswa untuk menemui pihak logistik dan akademik terlebih dahulu. Meskipun proses tersebut relatif cepat, sistem manual memiliki keterbatasan seperti ketergantungan pada jam kerja dan belum adanya dokumentasi historis peminjaman yang terpusat. Selain itu, *user* juga tidak dapat secara langsung mengetahui ketersediaan ruangan tanpa menghubungi pihak terkait. Sebuah *website* diperlukan untuk mengelola peminjaman ruangan di kampus agar dapat terorganisir dengan baik. Sistem peminjaman ruangan yang dapat diakses melalui situs *web* melayani *user* yang ingin menggunakan ruangan untuk tujuan akademik dan non-akademik. Tujuannya adalah merancang *website* peminjaman ruangan yang berfungsi untuk mengelola peminjaman ruang di Telkom University Purwokerto. *Website* peminjaman ruangan ini telah dibuat dengan menggunakan metode *prototype* dan *framework* *Laravel*. Hasilnya adalah *website* peminjaman ruangan di Telkom University Purwokerto yang memiliki fitur untuk melihat daftar ruangan yang tersedia, proses peminjaman ruangan, serta riwayat dan daftar peminjam ruangan. Berdasarkan hasil pengujian *blackbox testing* didapatkan total presentase keberhasilan fungsi sebesar 96,45%.

Kata kunci— *framework* *Laravel*, manajemen ruangan, peminjaman ruangan, *prototype*, *website*

I. PENDAHULUAN

Telkom University Purwokerto merupakan salah satu perguruan tinggi di Indonesia yang berfokus pada bidang teknologi informasi dan komunikasi. Kampus ini dilengkapi dengan infrastruktur dan fasilitas pendidikan modern, seperti laboratorium komputer, ruang kuliah berbasis teknologi, perpustakaan digital, dan akses internet yang cepat untuk mendukung pembelajaran berbasis teknologi. Seringkali perguruan tinggi dan kampus memiliki banyak ruang yang tersebar di berbagai gedung dan lokasi, salah satunya pada Telkom University

Purwokerto. Kegiatan akademik dan non akademik berlangsung di ruangan. Termasuk perkuliahan, seminar, diskusi kelompok, pertemuan organisasi mahasiswa, dan acara kampus lainnya.

Pengelolaan peminjaman ruangan masih sering dilakukan secara konvensional, yaitu melalui sistem pemesanan manual yang melibatkan proses administratif dan mengharuskan mahasiswa untuk menemui pihak logistik dan akademik terlebih dahulu. Berdasarkan hasil wawancara yang dilakukan oleh kepala pelayanan akademik Telkom University Purwokerto, proses ini dapat memakan waktu 10 hingga 15 menit atau bahkan lebih tergantung pada tingkat kecepatan respon dari pihak yang terlibat dalam proses peminjaman ruangan dan juga mengenai ketersediaan ruangnya. Meskipun proses tersebut relatif cepat, sistem manual memiliki keterbatasan seperti ketergantungan pada jam kerja dan belum adanya dokumentasi historis peminjaman yang terpusat. Selain itu, *user* juga tidak dapat secara langsung mengetahui ketersediaan ruangan tanpa menghubungi pihak terkait. Telkom University Purwokerto terkadang memiliki banyak acara yang menyebabkan dibutuhkan ruangan untuk melakukan kegiatan tersebut sehingga sistem peminjaman ruangan ini diperlukan untuk menghemat waktu. *Website* peminjaman ruangan ini telah dibuat untuk semua program studi yang ada di Telkom University Purwokerto.

Berdasarkan hasil wawancara singkat yang sudah dilakukan dengan pihak pengelola akademik bahwa di Telkom University Purwokerto memiliki ruangan sebanyak 44 ruangan pada 5 gedung (Gedung DC, Gedung DSP, Gedung IOT, Gedung Rektorat, dan Gedung TT) yang belum memiliki sistem peminjaman ruangan yang efisien. Hal ini dibuktikan bahwa masih banyak mahasiswa yang harus menggunakan cara manual dengan melakukan proses administratif yang kompleks yaitu meminjam yang ingin meminjam ruangan harus melewati beberapa tahapan sebelum meminjam ruangan, seperti menemui pihak logistik untuk meminta jadwal ruangan yang kosong, lalu ke pihak akademik dan

selanjutnya memastikan apakah ruangan yang akan dipinjam benar kosong atau tidak. Sebuah *website* diperlukan untuk mengelola peminjaman ruangan di kampus agar dapat terorganisir dengan baik.

Sistem peminjaman ruangan yang dapat diakses melalui *website* ini melayani *user* yang ingin menggunakan ruangan untuk keperluan akademik maupun non akademik [1]. *Website* adalah media informasi digital yang dapat diakses dengan mudah dan dibangun menggunakan sintaks atau *script* seperti *HyperText Markup Language (HTML)* dan *Cascading Style Sheets (CSS)* [2]. *Website* peminjaman ruangan ini mempermudah proses peminjaman ruangan dengan memungkinkan peminjaman ruangan, menampilkan data daftar ruangan, mengolah data ruangan, dan daftar peminjaman ruangan [3]. Pengembangan *website* ini akan menggunakan bahasa pemrograman *PHP* dengan *framework* *Laravel*.

Framework *Laravel* yang digunakan untuk mengembangkan *website* peminjaman ruangan ini memiliki ciri khas yaitu penekanan pada fleksibilitas desain. Hal tersebut mempermudah dalam proses pembuatan *website* dalam waktu yang singkat dan kode yang rapi [4]. Metode *prototype* digunakan pada pengembangan sistem ini untuk memahami kebutuhan *user*, validasi konsep, dan mendapatkan umpan balik *user*. Proses pengembangan sistem *iterative* yang mengubah kebutuhan menjadi sistem yang terus diperbaiki melalui kerja sama *user* dan analis [5]. Penggunaan metode *prototype* memungkinkan penyesuaian kebutuhan *user*, sekaligus menghemat waktu dan biaya. Hasil uji menggunakan metode *blackbox testing*, yaitu suatu pengujian yang lebih berfokus pada spesifikasi fungsi-fungsi yang dikembangkan. *Blackbox testing* menggunakan perspektif *user*, sehingga fokusnya hanya pada pengujian fungsional sistem [6].

II. KAJIAN TEORI

I. Rancang Bangun

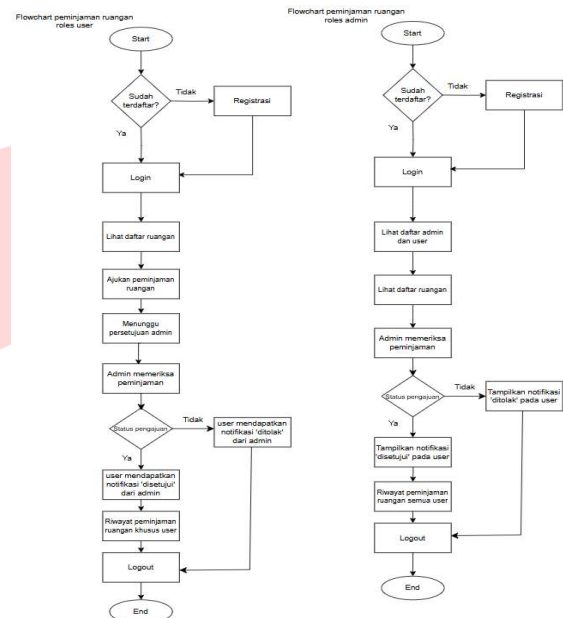
Proses desain melibatkan serangkaian langkah yang bertujuan untuk menerjemahkan hasil analisis sistem ke dalam bahasa pemrograman, sementara juga menawarkan penjelasan rinci tentang fungsionalitas yang terkait dengan setiap komponen sistem. Proses pembangunan melibatkan konstruksi komprehensif dari sistem baru atau modifikasi dan perbaikan sistem yang sudah ada secara keseluruhan. Proses rancang dan bangun melibatkan transformasi hasil analisis ke dalam perangkat lunak, yang kemudian mengarah pada pembangunan atau peningkatan sistem [7].

B. Website

Sebuah situs web terdiri dari kumpulan halaman digital dengan berbagai bentuk konten. Halaman-halaman ini dapat diakses melalui internet untuk pengguna dengan konektivitas. Sebuah situs web menggunakan beberapa bahasa pemrograman untuk fungsinya *HTML*, *PHP*, *CSS*, dan *JAVASCRIPT* [8].

C. Peminjaman Ruangan

Peminjaman melibatkan tindakan menyediakan sumber daya kepada pihak lain. Peminjaman ruangan melibatkan proses peminjaman atau penyediaan ruangan kepada peminjam untuk jangka waktu tertentu. Dapat terjadi di banyak tempat, seperti sekolah, kampus, perkantoran, atau tempat umum seperti gedung pertemuan atau pusat komunitas [3]. Gambar 1 di bawah ini menggambarkan diagram alir sistem peminjaman ruangan baik untuk *user* maupun *admin*:



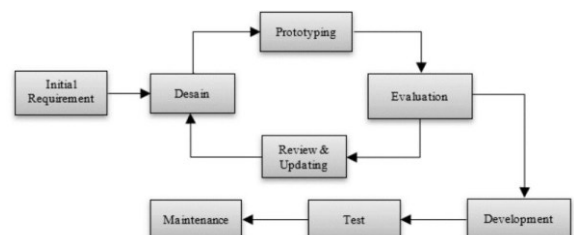
GAMBAR 1.

(FLOWCHART SISTEM PEMINJAMAN RUANG)

D. Metode Prototype

Metode *prototype* merupakan fase awal dari sistem perangkat lunak, menawarkan garis besar konsep yang luas, memungkinkan uji coba desain, mengungkapkan banyak masalah, dan mengidentifikasi solusi potensial untuk kesulitan ini. Tujuan dari metode *prototype* yang digunakan dalam sistem ini adalah untuk mendapatkan penggambaran pemodelan situs web yang dimaksud.

Desain situs web dimulai sebagai prototipe yang dapat dievaluasi pengguna. Setelah evaluasi pengguna, prototipe berfungsi sebagai pedoman bagi pengembang perangkat lunak dalam desain situs web. [9]. Metode prototipe mengikuti paradigma Siklus Hidup Pengembangan Perangkat Lunak (SDLC). Gambar 2 menggambarkan tahapan teknik *prototype* :



GAMBAR 2.
(TAHAP SDLC METODE PROTOTYPE)

E. Framework Laravel

Framework Laravel adalah alat pengembangan web berdasarkan arsitektur MVC (*Model View Controller*), dibuat dengan PHP. Kerangka kerja MVC meningkatkan kualitas perangkat lunak dengan mengurangi biaya pengembangan dan pemeliharaan, sementara juga meningkatkan pengalaman pengguna melalui aplikasi dengan sintaks yang jelas, menghasilkan efisiensi waktu. [10]. Aplikasi dalam MVC terdiri dari tiga komponen utama yaitu *Model*, *View*, dan *Controller*.

F. Blackbox Testing

Pengujian *Blackbox* mengevaluasi fungsionalitas perangkat lunak sesuai dengan spesifikasi, tanpa memeriksa desain atau kodenya. Metode ini berusaha untuk memverifikasi bahwa input, output, dan fungsi perangkat lunak selaras dengan kriteria yang ditentukan. [11]. Evaluasi ini terjadi pasca-pengembangan perangkat lunak untuk menilai kemampuan operasional perangkat lunak.

III. METODE

Metode *prototype* digunakan untuk mengembangkan sistem ini, sehingga proses pengembangannya melibatkan model awal yang diuji dan disesuaikan berdasarkan umpan dari pengguna. *Website* ini, yang Sistem ini dikembangkan menggunakan *Framework* Laravel, yang mencakup beberapa fitur penting, termasuk katalog kamar yang tersedia untuk pinjaman, formulir aplikasi pinjaman, catatan riwayat pinjaman pengguna, dan dasbor administratif untuk mengelola permintaan pinjaman. Sistem ini dirancang untuk meningkatkan efektivitas dan transparansi pengelolaan fasilitas kampus melalui antarmuka yang responsif dan ramah pengguna.

A. Initial Requirement

Initial requirement merupakan tahap awal dari proses *prototype*, Tahap analisis kebutuhan menggambarkan struktur perangkat lunak, mengidentifikasi semua komponen yang diperlukan, dan menentukan sistem yang akan dikembangkan.

TABEL 1.
(KEBUTUHAN FUNGSIONAL)

No.	Kebutuhan fungsional
1.	Sistem menyediakan fitur <i>login</i> untuk <i>user</i> dan <i>admin</i> menggunakan <i>email</i> dan <i>password</i> .
2.	Sistem menampilkan daftar ruangan yang tersedia di Telkom University Purwokerto.
3.	Sistem memungkinkan <i>user</i> (mahasiswa) untuk mengisi formulir peminjaman ruangan.
4.	Sistem memungkinkan <i>admin</i> untuk melihat dan memverifikasi permintaan peminjaman.
5.	Sistem menampilkan status peminjaman (disetujui atau ditolak) kepada <i>user</i> .
6.	Sistem mencatat dan menampilkan riwayat peminjaman untuk masing-masing <i>user</i> .

B. Desain

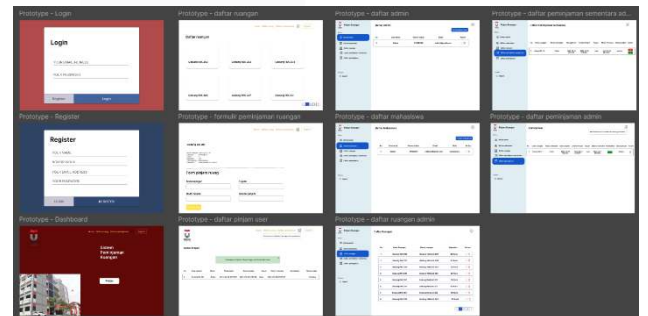
Tahap ini dilakukan pembuatan desain sederhana yang telah memberi gambaran singkat mengenai sistem yang ingin dibuat. Desain ini dibuat berdasarkan kebutuhan sistem yang telah diidentifikasi sebelumnya diterjemahkan dalam bentuk *prototype* atau model awal dari perangkat lunak. Pada fase ini, sangat penting untuk mengembangkan *wireframe*, atau desain *low-fidelitas*, yang berfungsi sebagai struktur visual awal.



GAMBAR 3.
(DESAIN LOW-FIDELITY SISTEM)

C. Prototyping

Tahap *prototyping* merupakan sebuah model awal atau representasi dari sistem perangkat lunak yang diusulkan dibuat dan diperbaiki melalui iterasi berulang. Tahap *prototype*, pengembangan desain akan berkonsentrasi pada membuat rancangan tampilan web versi *high-fidelity*. Versi *high-fidelity* ini merupakan penyempurnaan dari *wireframe* atau desain *low-fidelity* yang telah dibuat pada tahap sebelumnya.



GAMBAR 4.
(DESAIN HIGH-FIDELITY SISTEM)

D. Evaluation

Tahap evaluasi, *user* dapat menilai *prototype* untuk memastikan bahwa ia memenuhi kebutuhan *user* atau tidak. Tujuan utama dari tahap evaluasi adalah untuk menemukan titik lemah *prototype* dan area yang perlu diperbaiki atau ditingkatkan. Umpan balik yang diterima selama tahap evaluasi sangat penting untuk menyempurnakan dan mengoptimalkan *prototype* sebelum dibawa ke tahap pengembangan akhir atau implementasi.

E. Review and updating

Tahap ini *prototype* dievaluasi dan diperbarui berdasarkan hasilnya. Tujuan utama dari tahap ini adalah untuk memastikan bahwa *prototype* terus disempurnakan melalui umpan balik langsung dari *user* sesuai dengan kebutuhan dan keinginan *user* sebelum dikembangkan lebih lanjut.

F. Development

Tahap *development* merupakan tahap di mana *prototype* yang telah dirancang dan dievaluasi pada tahap sebelumnya dikembangkan lebih lanjut menjadi perangkat lunak yang lebih lengkap dan berfungsi. *Prototype* awal berkembang menjadi versi yang lebih canggih, sehingga mencapai produk akhir yang diinginkan, dan memiliki fitur serta kemampuan yang lebih lengkap.

G. Test

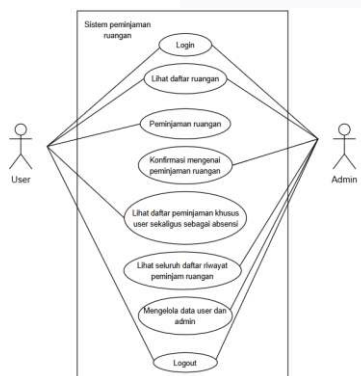
Tahap selanjutnya yaitu tahap test di mana *prototype* yang telah dikembangkan pada tahap sebelumnya diuji secara menyeluruh untuk memastikan bahwa sistem tersebut memenuhi standar kualitas yang diharapkan dan memenuhi kebutuhan *user*. Tujuannya di sini adalah untuk menemukan dan memperbaiki *bug*, masalah, atau kekurangan dalam *prototype* sebelum sistem digunakan oleh pengguna. Tahap *testing* ini dilakukan melalui *blackbox testing* yang merupakan pengujian dari sudut pandang user tanpa harus mengetahui struktur kode dari perangkat lunak.

H. Maintenance

Tahap *maintenance*, *prototype* perangkat lunak yang telah dikembangkan dan diuji akan diperbaiki dan ditingkatkan seiring waktu. Fase ini memainkan peran penting dalam proses pengembangan perangkat lunak karena menjamin fungsionalitas yang berkelanjutan dari sistem yang ada, sesuai dengan kebutuhan *user*, dan beradaptasi dengan persyaratan atau lingkungan baru.

I. Use Case Diagram

Hal ini menunjukkan interaksi antara aktor dan sistem yang akan dibuat. Berikut ini adalah skema yang menunjukkan *use case* peminjaman ruangan, seperti yang terlihat pada Gambar 5:

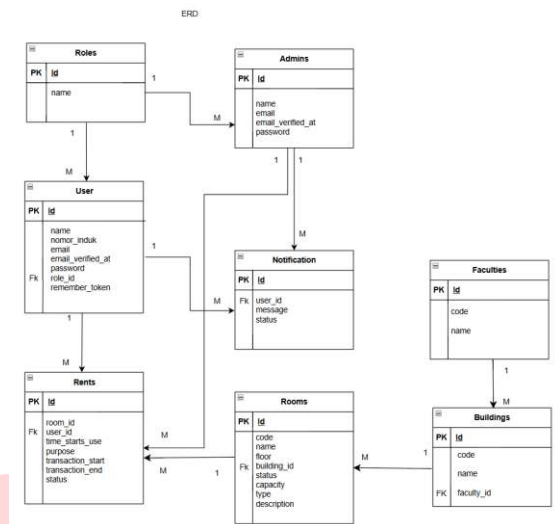


GAMBAR 5.

(USE CASE DIAGRAM PEMINJAMAN RUANG)

J. Entity Relationship Diagram (ERD)

ERD mewakili entitas, atribut, dan hubungan antar-entitas dalam struktur database, menggambarkan koneksi satu-ke-banyak (1:M), menandakan bahwa satu entitas mungkin sesuai dengan banyak entitas lain. Berikut ini adalah ERD untuk peminjaman ruangan, seperti pada Gambar 6:



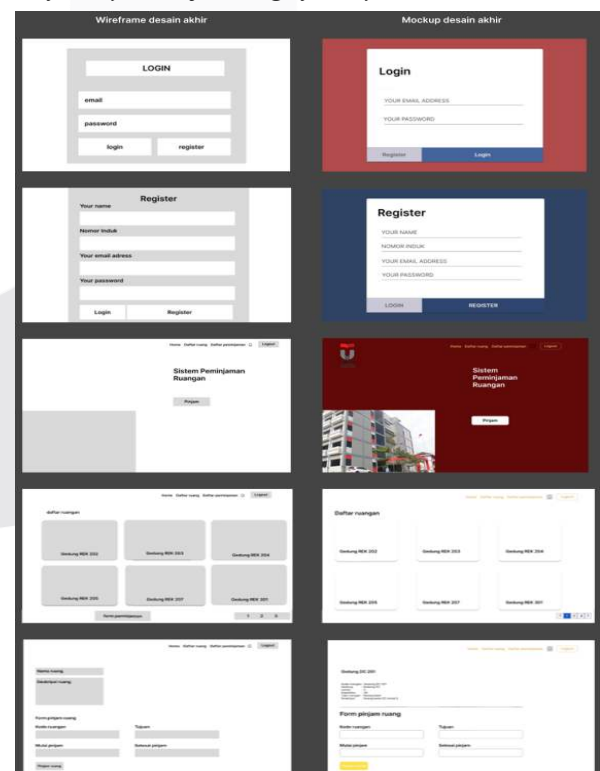
GAMBAR 6.

(ERD PEMINJAMAN RUANGAN)

IV. HASIL DAN PEMBAHASAN

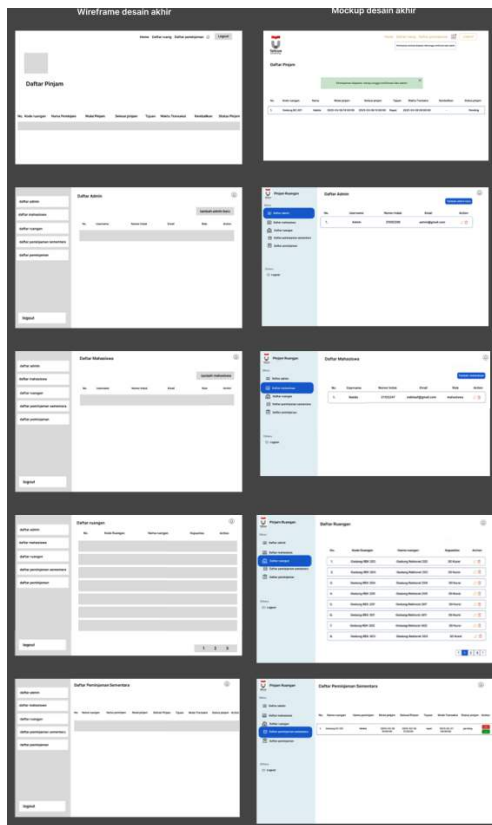
A. Evaluation and updating

Pada tahap *evaluation and updating* didapatkan hasil dari iterasi 2 yang dilakukan perubahan sesuai dengan umpan balik dari *user* mengenai rancangan *low fidelity* atau *wireframe*. *High fidelity prototype* atau *mockup* dibuat berdasarkan hasil desain terakhir pada iterasi 2. Berikut ini merupakan perubahan desain dari *low fidelity* menuju ke *high fidelity*:



GAMBAR 7.

(PERUBAHAN DESAIN WIREFRAME & MOCKUP)



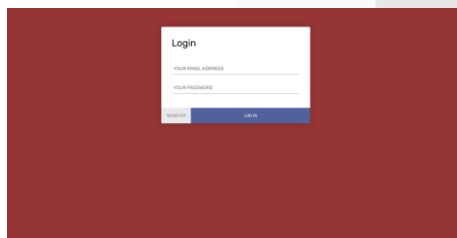
GAMBAR 8.

(PERUBAHAN DESAIN WIREFRAME & MOCKUP (2))

B. Development

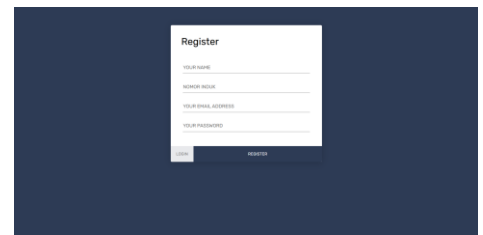
Hasil dari fase pengembangan ini adalah situs web penyewaan kamar fungsional dengan fitur-fitur penting yang disesuaikan untuk memenuhi persyaratan *user* dan *admin*. Tampilan dan fiturnya telah disesuaikan sesuai dengan rancangan awal, termasuk halaman *login* dan *register*, daftar ruangan, formulir peminjaman, daftar peminjaman atau riwayat peminjaman, pengelolaan data oleh *admin*, persetujuan peminjaman oleh *admin*, dan *logout*. Berikut ini merupakan tampilan antarmuka (UI) dari *website* peminjaman ruangan yang telah dibuat:

1) Halaman login

GAMBAR 9.
(HALAMAN LOGIN)

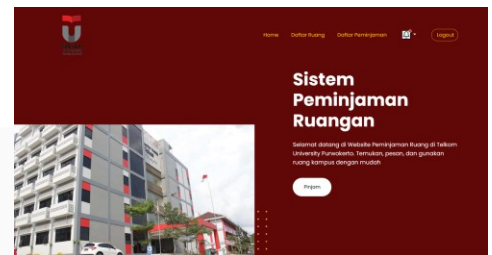
Halaman *login website* untuk *user* yang sudah memiliki akun dapat langsung mengakses sistem dengan mengisi *email* dan *password* pada form *login* yang disediakan.

2) Halaman registrasi

GAMBAR 10.
(HALAMAN REGISTRASI)

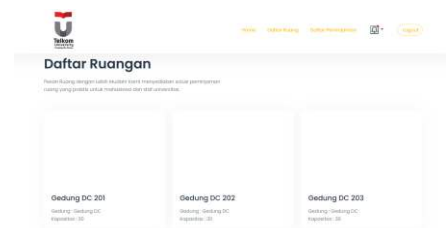
Halaman registrasi ini memungkinkan *user* yang belum memiliki akun untuk mendaftarkan diri dengan mengisi formulir yang tersedia. Formulir tersebut mencakup data seperti nama lengkap, *email*, dan *password* yang akan digunakan untuk *login* ke dalam sistem.

3) Halaman dashboard

GAMBAR 11.
(DASHBOARD USER)

Dashboard user merupakan halaman utama setelah *user* berhasil *login*, yang menampilkan beberapa fitur navigasi penting seperti *home*, daftar ruangan, daftar peminjaman, notifikasi peminjaman, tombol pinjam, dan *logout*. Melalui fitur *home* atau beranda, *user* dapat melihat ringkasan atau informasi umum terkait sistem.

4) Halaman daftar ruangan user

GAMBAR 12.
(HALAMAN DAFTAR RUANGAN USER)

Halaman daftar ruangan menampilkan seluruh ruangan yang tersedia untuk dipinjam oleh *user*. Setiap ruangan ditampilkan dengan informasi yang jelas, seperti nama ruangan, lokasi, kapasitas,

serta deskripsi singkat mengenai ruangan tersebut.

5) Formulir peminjaman ruangan

GAMBAR 13.
(FORMULIR PEMINJAMAN RUANG)

Halaman ini memuat *user* yang telah memilih ruangan yang ingin dipinjam dan akan diarahkan untuk mengisi formulir peminjaman ruangan yang tersedia. Formulir ini mencakup informasi penting seperti tujuan peminjaman, waktu mulai, dan waktu selesai peminjaman.

6) Halaman daftar peminjaman user

GAMBAR 14.
(DAFTAR PEMINJAMAN USER)

Halaman ini menampilkan seluruh riwayat peminjaman ruangan yang telah dilakukan oleh *user*.

7) Halaman daftar admin

GAMBAR 15.
(HALAMAN DAFTAR ADMIN)

Halaman daftar *admin* ini memungkinkan *admin* untuk menambahkan, mengedit, maupun menghapus *admin*.

8) Halaman daftar mahasiswa

GAMBAR 16.
(HALAMAN DAFTAR MAHASISWA)

Halaman daftar mahasiswa ini memungkinkan *admin* untuk menambahkan, mengedit, maupun menghapus data mahasiswa yang terdaftar.

9) Halaman daftar ruangan admin

GAMBAR 17.
(HALAMAN DAFTAR RUANG ADMIN)

Halaman daftar ruangan ini memungkinkan *admin* untuk mengedit maupun menghapus ruangan yang terdaftar.

10) Halaman daftar peminjaman sementara admin

GAMBAR 18.
(HALAMAN DAFTAR PINJAM SEMENTARA)

Halaman ini mencakup peminjaman sementara jika ada *user* yang mengajukan peminjaman.

11) Halaman daftar peminjaman keseluruhan admin

GAMBAR 19.
(HALAMAN DAFTAR PINJAM ADMIN)

Halaman ini memuat seluruh daftar peminjaman atau riwayat peminjaman ruangan yang telah dilakukan oleh *user*,

memberikan gambaran lengkap mengenai status peminjaman yang sedang berlangsung maupun yang telah selesai.

- 12) Notifikasi user saat mengajukan permintaan kepada admin



GAMBAR 20.
(NOTIFIKASI USER)

Fitur notifikasi untuk *user* akan muncul setiap kali terdapat pembaruan terkait status peminjaman.

- 13) Notifikasi admin saat menerima permintaan dari user



GAMBAR 21.
(NOTIFIKASI ADMIN)

Fitur notifikasi untuk *admin* akan muncul setiap kali ada pengajuan peminjaman baru dari *user*.

C. Analisis Pengujian

Tahap pengujian mengikuti tahap desain dan pengembangan untuk memastikan fungsionalitas sistem yang tepat. Metodologi pengujian yang digunakan adalah pengujian *blackbox*, yang menilai fungsionalitas sistem dari perspektif pengguna, terlepas dari arsitektur pengujian dan tanpa memerlukan pengetahuan tentang struktur kode perangkat lunak. Pada pengujian ini, diambil 20 responden mahasiswa sebagai *user* untuk menguji *website* peminjaman ruangan.

Hasil pengujian pada *blackbox* yang melibatkan 20 mahasiswa sebagai *user* dengan masing-masing *user* melakukan 8 pola pengujian dan 1 *admin* yang melakukan 9 pola pengujian, didapatkan hasil perhitungan *blackbox testing* seperti pada Tabel 2:

TABEL.2
(REKAPITULASI HASIL UJI *BLACKBOX*)

Role	Jumlah Responden	Pola Pengujian	Total uji coba	Berhasil
Mahasiswa (user)	20	8	160	154
Staf akademik (admin)	1	9	9	9
Total	21	17	169	163

Dari rekapitulasi tersebut, lalu dihitung menggunakan teknik analisis deskriptif, yaitu:

$$\text{Presentase kelayakan} = \frac{\text{Total uji coba berhasil}}{\text{Total uji coba}} \times 100\%$$

$$\text{Presentase kelayakan} = \frac{163}{169} \times 100\% = 96,45\%$$

Berdasarkan hasil perhitungan *blackbox testing* diatas didapatkan total presentase keberhasilan fungsi sebesar 96,45%.

V. KESIMPULAN

Berdasarkan hasil perancangan sistem, telah berhasil dikembangkan sebuah *website* peminjaman ruangan yang dirancang untuk meminjam ruangan yang ada di Telkom University Purwokerto. *User* dapat melakukan proses peminjaman secara *online* dan melihat informasi ketersediaan ruangan secara *real-time* melalui sistem ini. Sistem ini juga membantu pengelola mengatur jadwal penggunaan ruang secara lebih terorganisir.

Website ini telah berhasil menerapkan *framework* Laravel dan metode *prototype* secara optimal. Sementara Laravel memudahkan pengembangan sistem berbasis arsitektur MVC serta mendukung keamanan dan skalabilitas sistem, metode *prototype* memungkinkan pengembangan berulang dengan melibatkan *user* melalui pengujian awal, yang memungkinkan sistem disesuaikan dengan kebutuhan nyata. Hasilnya adalah *website* peminjaman ruangan di Telkom University Purwokerto yang memiliki fitur untuk melihat daftar ruangan yang tersedia, proses peminjaman ruangan, serta riwayat dan daftar peminjam ruangan. Berdasarkan hasil pengujian *blackbox testing* didapatkan total presentase keberhasilan fungsi sebesar 96,45%.

REFERENSI

- [1] A. S. Putro, F. Ismuharyanti, and A. M. Saktiwi, "Analisis pemanfaatan sistem peminjaman ruangan berbasis web di Resource Centre Universitas Amikom Yogyakarta," *Pustaka Karya : Jurnal Ilmiah Ilmu Perpustakaan Dan Informasi*, vol. 10, no. 2, pp. 104, December 2022.
- [2] A. Apriliando, "Implementasi Framework Laravel pada Rancang Bangun Website IAKN Palangka Raya dengan Metode Prototype," *Jurnal Sains Komputer Dan Teknologi Informasi*, vol. 3, no. 2, pp. 87-96, May 2021.
- [3] N. A. D. Dhanardana, and T. Rahayu, "Aplikasi Peminjaman Ruangan Berbasis Web Pada Direktorat Jenderal Hak Asasi Manusia," *Seminar Nasional Mahasiswa Bidang Ilmu Komputer Dan Aplikasi (SENAMIKA)*, vol. 1, no. 2, pp. 211–221, August 2020.
- [4] A. Nurhidayah, and S. Kosasi, "Perancangan Perangkat Lunak Penjualan Berbasis Website Dengan Framework Laravel Pada Emiracase," *E-Jurnal JUSITI (Jurnal Sistem Informasi Dan*

- Teknologi Informasi*), vol. 11, no. 1, pp. 24–35, April 2022.
- [5] N. Nabilla, and A. Ichwani, “Sistem Informasi Layanan E-Konseling Psikologi Untuk Mahasiswa Berbasis *Website* Dengan Metode *Prototype*,” *Jurnal Mnemonic*, vol. 5, no. 2, pp. 191–198, September 2022.
- [6] N. L. G. P. Suwirmayanti, I. K. A. A. Aryanto, I. G. A. N. W. Putra, N. K. Sukerti, and R. Hadi, “Penerapan *Helpdesk System* dengan Pengujian *Blackbox Testing*,” *Jurnal Ilmiah Intech : Information Technology Journal of UMUS*, vol. 2, no. 2, pp. 55–64, November 2020.
- [7] N. A. Putri, and Amaliyah, “Rancang Bangun Sistem Reservasi Ruang Rapat Berbasis *Website*,” *IKRAITH-Teknologi*, vol. 7, no. 2, pp. 22–34, July 2023.
- [8] A. O. Sari, A. Abdilah, and Sunarti, *Web Programming*, edisi ke-1, Yogyakarta: Graha Ilmu, 2020.
- [9] N. L. A. M. R. Dewi, R. S. Hartati, and Y. Divayana, “Penerapan Metode *Prototype* dalam Perancangan Sistem Informasi Penerimaan Karyawan Berbasis *Website* pada Berlian Agency,” *Majalah Ilmiah Teknologi Elektro*, vol. 20, no. 1, pp. 147, January-June 2021.
- [10] A. Yunisa, and R. Amalia, “Perancangan Sistem Informasi Manajemen Jasa Fotografi Berbasis *Website* Menggunakan *Framework* Laravel (Studi Kasus: Haydey Moment)”, *Jurnal Informatika MULTI*, vol. 1, no. 1, pp. 25–36, January 2023.
- [11] M. Syarif, and E. B. Pratama, “Analisis Metode Pengujian Perangkat Lunak *Blackbox Testing* Dan Pemodelan Diagram Uml Pada Aplikasi Veterinary Services Yang Dikembangkan Dengan Model *Waterfall*,” *Jurnal Teknik Informatika Kaputama (JTİK)*, vol. 5, no. 2, pp. 253–258, July 2021.