

SISTEM MONITORING SKRIPSI BERBASIS PROGRESSIVE WEB APPLICATION DENGAN PUSH NOTIFICATION

Herdiesel Santoso¹, Wahyu Widodo^{2*}, Muhammad Alvian Rizky³

¹Sistem Informasi STMIK El Rahma,

^{2,3}Informatika STMIK El Rahma

e-mail: [1herdiesel.santoso@stmikelrahma.ac.id](mailto:herdiesel.santoso@stmikelrahma.ac.id), [2 wahyu@stmikelrahma.ac.id](mailto:wahyu@stmikelrahma.ac.id), [3 alvianrizky62@gmail.com](mailto:alvianrizky62@gmail.com)

Abstract

The Covid-19 pandemic requires thesis guidance activities to be carried out online. If the tertiary institution does not have a proper thesis monitoring system, the guidance process will not run smoothly and on time. This study aims to create a thesis monitoring system using Progressive Web Application (PWA) technology with a software development method using mobile-D. The development stage begins with the construction of a system architecture in the form of use cases and activity diagrams. Then the initialize stage is translating the system architecture design into an application interface design. Next is the stage of translating the design into an actual application using lumens and mariaDB on the server side and javascript and indexedDB on the client side. After each module is completed, the system functionality modules are integrated into a whole application into a thesis monitoring system. Then the software functionality testing is carried out, especially when it is run on a smartphone. The test results indicate that there are no bugs or errors found when running on laptop or mobile devices.

Keywords— *Mobile-D, Push Notification, Progressive Web Apps*

PENDAHULUAN

Teknologi pada era industri 4.0 telah memberikan banyak manfaat dan kemajuan diberbagai aspek. Terlebih dimasa pandemi Covid – 19 telah terjadi banyak perubahan pola belajar di lingkungan perguruan tinggi dengan memprioritaskan penggunaan teknologi informasi [1]. Sebagian besar kegiatan akademik harus dilakukan dari rumah dan dilaksanakan secara *online* atau daring, termasuk kegiatan bimbingan skripsi. Perguruan tinggi diharapkan masih dapat menghasilkan karya mahasiswa yang berkualitas dibawah arahan dosen pembimbingnya. Proses bimbingan tidak akan dapat berjalan lancar dan tepat waktu jika perguruan tinggi tidak memiliki sistem yang tepat, yaitu sistem yang dapat merekam dan memotoring kemajuan penelitian yang dilakukan oleh mahasiswanya secara daring [2]–[4].

Selama pandemi proses pendaftaran skripsi di STMIK El Rahma masih dilakukan secara semi daring. Pertama mahasiswa yang sudah mengambil matakuliah skripsi di KRS-nya mendaftar di bagian akademik dengan mengumpulkan dokumen persyaratan skripsi. Pengumpulan dokumen bisa dilakukan dengan *email* atau *whatsapp*. Selanjutnya dokumen tersebut diserahkan ke Ketua Program Studi (Kaprodi) untuk diperiksa terlebih dahulu apakah mahasiswa memenuhi persyaratan untuk mengambil skripsi atau tidak. Jika memenuhi maka kaprodi akan membuatkan Surat Keputusan (SK) Pembimbing Skripsi. Selanjutnya SK tersebut didistribusikan menggunakan *whatsApp* ke mahasiswa. Mahasiswa akan menghubungi dosen pembimbing masing-masing. Proses bimbingan selama pandemi dilakukan dengan menggunakan grup *whatsApp* dan email.

Proses pengajuan dan bimbingan skripsi yang berjalan saat ini masih memungkinkan kontak fisik, dan hal tersebut tidak sesuai dengan protokol kesehatan. Penggunaan *whatsApp* dan *email* masih memiliki kelemahan yaitu masih kesulitan untuk merekam dokumen mahasiswa yang melakukan bimbingan skripsi. Semakin banyaknya

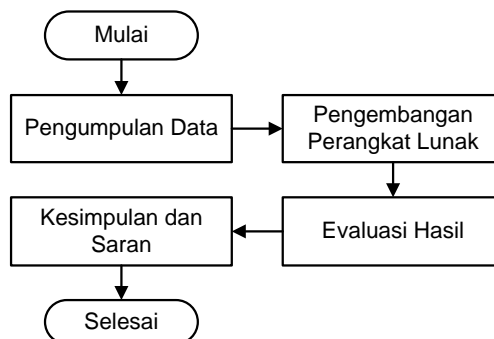
jumlah mahasiswa bimbingan, menjadikan dosen kesulitan untuk memantau perkembangan mahasiswa bimbingannya. Padahal dosen pembimbing diharuskan bisa memonitor semua aktifitas mahasiswa bimbingan [5]. Semua aktifitas yang berkaitan dengan bimbingan harus bisa masuk ke notifikasi di *smartphone* dan laptop dosen pembimbing, sehingga dosen pembimbing bisa memberi respon dengan cepat di manapun mereka berada dan kapanpun itu datang.

Melalui perkembangan teknologi perangkat *mobile*, notifikasi atau pemberitahuan dapat langsung dikirim dan diterima dari satu perangkat ke perangkat lainnya terutama melalui perangkat *smartphone* dan laptop. *Progressive Web Application* (PWA) adalah teknologi yang dikembangkan oleh Google untuk mengatasi keterbatasan teknologi *native* dan *website* [6]. Penelitian [7]–[9] menunjukkan bahwa PWA menawarkan fitur-fitur yang dapat menjembatani antara aplikasi *native* dengan aplikasi *web* seperti *offline support*, sinkronisasi latar belakang, *push notification*, dan pemasangan aplikasi di beranda *smartphone*. Walaupun tidak ada jaringan *internet*, PWA dapat berjalan dengan baik dan cara menjalankannya sama seperti aplikasi *native* yaitu dengan melakukan klik ikon aplikasi di layar *smartphone* [10]. Pengembangan sistem dilakukan menggunakan metode *agile* yaitu dengan *mobile-D*. Penelitian [11], [12] menunjukkan metode *mobile-D* cocok diterapkan untuk pengembangan aplikasi *mobile* karena memiliki kelebihan yaitu dapat diterapkan untuk pengembangan perangkat lunak yang memiliki lingkungan sangat dinamis oleh tim kecil menengah, pendekatan berorientasi objek, dan siklus pengembangan yang relatif singkat.

Tujuan penelitian ini adalah membuat sistem monitoring skripsi menggunakan teknologi *Progressive Web Application* (PWA) dengan metode pengembangan perangkat lunak menggunakan *mobile-D*. Sistem monitoring tersebut mulai dari pendaftaran, distribusi dosen dan SK pembimbing, proses bimbingan dan kemajuan serta pendaftaran ujian skripsi. Sistem dilengkapi dengan fasilitas *chat* sehingga dosen bisa langsung memberikan komentar kemajuan skripsi. Notifikasi membantu pengguna pengguna mengeloa tugas dan perubahan informasi dalam lingkup aplikasi. Sistem *push notification* ini juga dapat dimanfaatkan untuk memberikan pesan jika ada mahasiswa yang mengirimkan laporan kemajuan kepada dosen maupun jika dosen telah memeriksa laporan skripsi mahasiswa.

METODE PENELITIAN

Metode penelitian yang digunakan adalah metode kualitatif dengan mengambil studi kasus di STMIK El Rahma Yogyakarta [13]. Adapun langkah-langkah penelitian ditunjukkan pada Gambar 1.

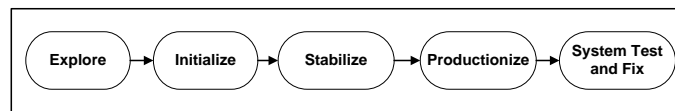


Gambar 1. Langkah-langkah penelitian

Observasi dilakukan dengan cara melakukan pengamatan langsung mulai dari proses pengajuan proposal dan dosen pembimbing sampai dengan menunggu jadwal

ujian. Studi dokumen dilakukan dengan mengumpulkan dokumen pendukung yang berhubungan dengan skripsi. Wawancara dilakukan dengan mahasiswa, dosen pembimbing dan kaprodi mengenai proses administrasi pengajuan skripsi hingga mahasiswa mendapat jadwal ujian.

Metode pengembangan perangkat lunak dalam penelitian ini menggunakan Mobile-D. Mobile-D merupakan metodologi pengembangan perangkat lunak yang dikhususkan untuk pembuatan aplikasi *mobile* [14]. Metode ini memungkinkan tim pengembangan membuat perangkat lunak dalam periode waktu yang sangat pendek [15]. Hal ini dikarenakan *mobile-D* mengadopsi paradigma pengembangan perangkat lunak secara tangkas (*agile*) dan efisien. Metode *Mobile D* memiliki karakteristik yaitu dapat diterapkan untuk pengembangan perangkat lunak yang memiliki lingkungan sangat dinamis oleh tim kecil menengah, pendekatan berorientasi objek, dan siklus pengembangan yang relatif singkat. Tahapan pengembangan perangkat lunak dengan *Mobile-D* ditunjukkan pada Gambar 2 [12].



Gambar 2. Tahapan pengembangan perangkat lunak dengan *Mobile-D*

Penjelasan setiap tahapan pengembangan adalah sebagai berikut:

1. *Explore*, pada tahap ini diawali dengan menjabarkan pengguna beserta kebutuhan sistem. Selanjutnya spesifikasi kebutuhan sistem tersebut digambarkan dalam pemodelan sistem dalam bentuk *use case* dan *activity diagram*.
2. *Initialize*, pada tahap ini selanjutnya akan menterjemahkan rancangan arsitektur sistem menjadi rancangan antarmuka aplikasi sistem monitoring skripsi.
3. *Productionize*, tahapan ini merupakan tahapan menterjemahkan rancangan ke dalam bentuk aplikasi yang sebenarnya (*coding*). Bahasa pemrograman yang digunakan adalah HTML 5 dan CSS 3 untuk UI dan UX, *lumen* dan basis data *mysqli* di sisi *server* serta *javascript* dan *indexedDB* di sisi *client*.
4. *Stabilize*, pada tahapan ini dipastikan bahwa fungsionalitas sistem yang wujudkan dalam bentuk modul-modul sudah harus disatukan menjadi satu kesatuan aplikasi sistem secara utuh menjadi sistem monitoring skripsi.

System test and fix, tahapan ini dilakukan pengujian fungsionalitas perangkat lunak khususnya ketika dijalankan pada *smartphone*. Fitur-fitur utama seperti *login*, *offline support*, sinkronisasi latar belakang, *push notification* dan fasilitas bimbingan harus berjalan dengan baik. Pengujian dilakukan dengan metode *black box testing* [16].

HASIL DAN PEMBAHASAN

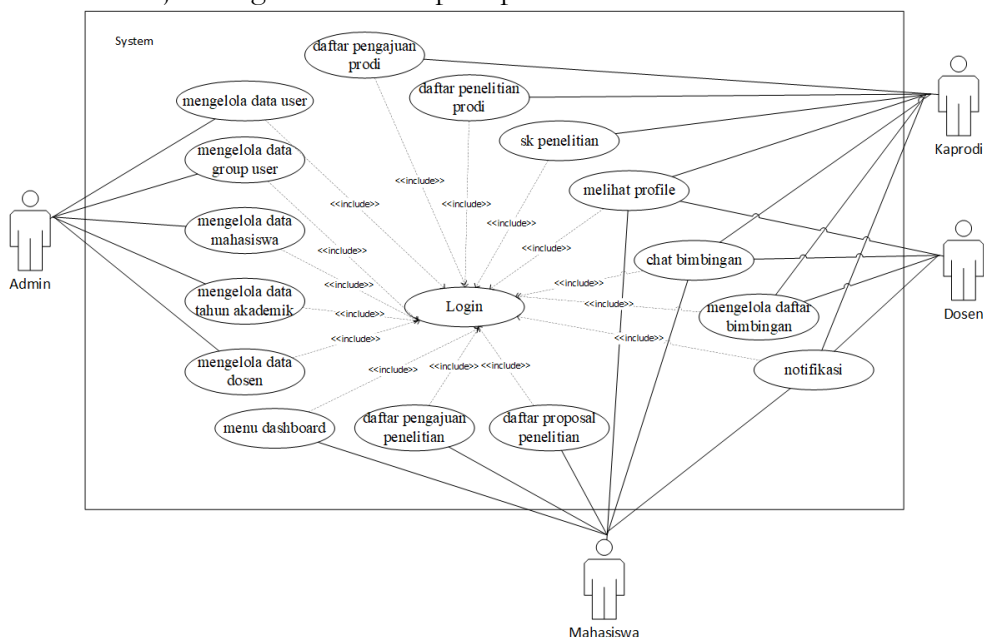
Setelah melakukan identifikasi dan pengumpulan data, selanjutnya dilakukan pengembangan sistem mengikuti metode pengembangan *mobile-D*. Hasil dan pembahasan penelitian adalah sebagai berikut.

Kebutuhan sistem didasarkan pada dokumentasi pengumpulan data yang selanjutnya fungsi tersebut tersebut dibagi berdasarkan hak akses penggunaannya. Pengguna sistem informasi monitoring skripsi dibedakan menjadi admin akademik, ketua program studi, dosen dan mahasiswa. Tabel 1 merupakan tanggung jawab dan fungsionalitas untuk setiap pengguna.

Tabel 1. Pengguna sistem beserta fungsionalitasnya

Kategori Pengguna	Tugas
Admin Akademik	Mengelola data pengguna dan hak akses Mengelola data mahasiswa dan data tahun akademik Mengelola data dosen dan ketua program studi Mengelola data jadwal seminar skripsi
Ketua Program Studi	Mengelola data pengajuan skripsi program studi Mengelola data dosen pembimbing skripsi Mengelola data progres bimbingan skripsi dosen dan mahasiswa Mengelola Surat Keputusan pembimbing skripsi Mengelola notifikasi ketua program studi
Dosen	Melihat dan mencetak Surat Keputusan pembimbing skripsi Mengelola data progres bimbingan skripsi setiap mahasiswa Mengelola notifikasi dosen
Mahasiswa	Proses pengajuan proposal skripsi Melihat dan mencetak Surat Keputusan pembimbing skripsi Proses pengajuan laporan skripsi Mendaftar seminar skripsi Mengelola notifikasi mahasiswa

Setelah kebutuhan fungsional sistem teridentifikasi, selanjutnya adalah memodelkan menjadi diagram *use case* seperti pada Gambar 3.

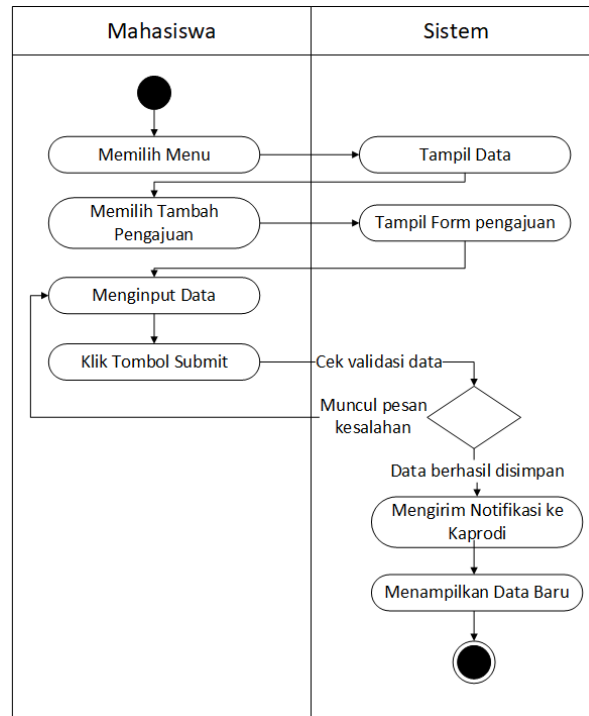


Gambar 3. Use case diagram sistem informasi monitoring skripsi

Aktor atau pengguna sistem ini terdiri dari mahasiswa, dosen, ketua program studi dan admin. Pengguna harus melakukan *login* terlebih dahulu agar dapat menggunakan menu atau fungsi yang tersedia pada sistem. Menu yang dapat diakses oleh mahasiswa antara lain menu *dashboard*, menu profile, menu pengajuan, menu proposal, menu pesan dan menu list notifikasi. Menu untuk dosen antara lain menu profil, menu pesan, menu daftar bimbingan dan menu list notifikasi. Menu untuk ketua program studi yaitu menu profil, menu daftar bimbingan, menu pesan dan menu list notifikasi. Terakhir menu untuk admin antara lain menu mengelola hak akses pengguna mulai dari mahasiswa, dosen, ketua program studi dan admin, menu mengelola data mahasiswa, data dosen dan ketua program studi serta menu mengelola data tahun akademik.

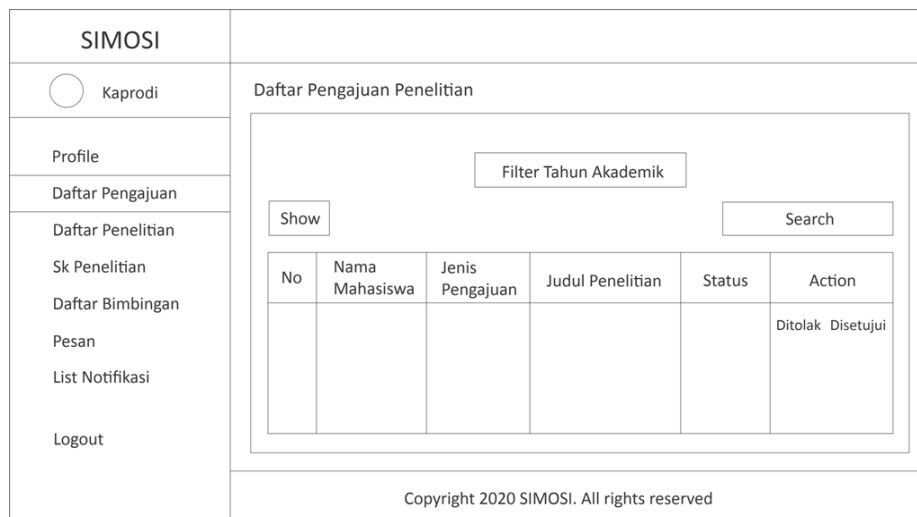
Setelah kebutuhan fungsional sistem teridentifikasi, selanjutnya adalah memodelkan menjadi diagram *use case*. Gambar 3 menunjukkan rancangan diagram *use case* sistem monitoring skripsi. Aktor atau pengguna sistem ini terdiri dari mahasiswa, dosen, ketua program studi dan admin. Pengguna harus melakukan *login* terlebih dahulu agar dapat menggunakan menu atau fungsi yang tersedia pada sistem. Menu yang dapat diakses oleh mahasiswa antara lain menu *dashboard*, menu profile, menu pengajuan, menu proposal, menu pesan dan menu list notifikasi. Menu untuk dosen antara lain menu profil, menu pesan, menu daftar bimbingan dan menu list notifikasi. Menu untuk ketua program studi yaitu menu profil, menu daftar bimbingan, menu pesan dan menu list notifikasi. Terakhir menu untuk admin antara lain menu mengelola hak akses pengguna mulai dari mahasiswa, dosen, ketua program studi dan admin, menu mengelola data mahasiswa, data dosen dan ketua program studi serta menu mengelola data tahun akademik.

Pada Gambar 4 menggambarkan *activity diagram* proses pengajuan judul skripsi mahasiswa yang dimulai dari mahasiswa memilih menu pengajuan, kemudian sistem akan menampilkan data riwayat pengajuan skripsi mahasiswa. Mahasiswa dapat mengajukan skripsi dengan cara memilih tombol tambah data dan selanjutnya melengkapi isian *form* pengajuan skripsi. Ketika data pengajuan skripsi sudah lengkap, maka mahasiswa dapat menekan tombol *submit* data. Sistem akan memproses data yang diinputkan mahasiswa, jika data sesuai maka sistem akan menyimpan data dan mengirimkan notifikasi ke ketua program studi, akan tampil pesan data berhasil disimpan. Namun jika proses terdapat kesalahan maka sistem akan menampilkan pesan kesalahan dan mahasiswa harus meninjau kembali proses pengajuan.



Gambar 4. Activity diagram pengajuan judul skripsi

Selanjutnya rancangan arsitektur dan model sistem yang telah dibuat diterjemahkan menjadi rancangan antarmuka perangkat lunak. Gambar 5 merupakan rancangan *dashboard* sistem monitoring skripsi untuk ketua program studi. Pada rancangan tersebut terdapat menu yang digunakan misalkan menu untuk melihat daftar pengajuan dan SK yang telah dibagikan.



Gambar 5. Rancangan dashboard system monitoring skripsi

Gambar 6 merupakan desain notifikasi pada tampilan *mobile*. Menu notifikasi akan menampilkan *list* daftar notifikasi yang sudah dikirim oleh sistem. Pada menu ini mahasiswa, dosen dan ketua program studi dapat melihat isi lengkap notifikasi dengan mengklik detail salah satu notifikasi tersebut.

Notifikasi Bimbingan Online

Judul Notifikasi

Tanggal

Isi Notifikasi

Judul Notifikasi

Tanggal

Isi Notifikasi

Judul Notifikasi

Tanggal

Isi Notifikasi

Judul Notifikasi

Tanggal

Isi Notifikasi

Gambar 6. Rancangan menu notifikasi

Perangkat lunak sistem monitoring memerlukan *server hosting* yang sesuai sebagai lingkungan penerapan sistem agar dapat diakses secara daring dengan baik. Spesifikasi *server hosting* untuk meletakkan perangkat lunak adalah prosesor Intel® Xeon Silver 4215 (*cache* 11 M dan frekuensi 2.50 GHz), memory RAM 64 -128 GB, *hardisk solid state drive* 128 GB, kecepatan transfer 1 Gbps. *Server* mendukung versi PHP 7.x, MariaDB, PHPMyAdmin, SSH, GIT, Akses POP3, IMAP, dan SMTP.

Setelah membuat desain perangkat lunak dan menentukan lingkungan sistem, selanjutnya adalah mengimplementasikan rancangan ke dalam bentuk perangkat lunak yang sebenarnya. Proses implementasi terdiri dari implementasi di sisi *server* dan implementasi di sisi *client*. Teknologi yang digunakan untuk membangun *server* adalah HTML 5 dan CSS 3 untuk UI dan UX, PHP sebagai bahasa pemrograman menggunakan *framework* laravel. MariaDB digunakan sebagai media penyimpanan data di *server*.

Teknologi yang digunakan untuk membangun notifikasi di sisi *client* adalah HTML 5 dan CSS 3 untuk UI dan UX, *javascript* sebagai bahasa pemrograman serta *indexedDB* sebagai basis data. *IndexedDB* digunakan sebagai media penyimpanan data pada *browser* secara permanen, sehingga aplikasi dapat berjalan tanpa harus terhubung dengan jaringan internet. Aplikasi dapat berjalan secara cepat karena tidak perlu meminta informasi basis data di *server*. Sistem monitoring skripsi ini menggunakan pertukaran data menggunakan REST API dengan format data berbentuk JSON antara *server* dan *client*. Lumen digunakan sebagai *framework* untuk membuat REST API menangani pertukaran data tersebut.

Tahapan stabilize merupakan tahapan untuk memastikan bahwa fungsionalitas sistem yang berbetuk modul-modul sudah harus disatukan menjadi satu kesatuan aplikasi sistem secara utuh menjadi sistem monitoring skripsi. Juga dilakukan pemeriksaan terhadap modul-modul yang sudah dibuat apakah sudah selesai dan sesuai belum dengan kebutuhan fungsional pengguna. Jika sudah sesuai, maka langkah selanjutnya akan diambil. Namun jika tidak, modul akan direvisi dengan mengulang langkah-langkah sebelumnya. Modul yang sudah jadi, diintegrasikan dengan sistem pengiriman notifikasi menggunakan REST API. Karena itu juga perlu dipastikan apakah modul pertukaran data serta perangkat keras *server hosting* mampu menangani permintaan dalam jumlah banyak dan dalam waktu bersamaan. Pengujian dilakukan dengan metode *black box testing* yaitu dengan melihat fungsi masukan dan keluaran perangkat lunak sudah sesuai dengan kebutuhan pengguna. Tabel 2 merupakan hasil pengujian perangkat lunak sistem monitoring skripsi. Berdasarkan hasil pengujian menggunakan *back box testing* menunjukkan bahwa tidak ditemukan adanya bug atau error baik di sisi *server (website)* dan *client (mobile)*.

Tabel 2. Hasil pengujian perangkat lunak dengan *black box testing*

No	Skenario Pengujian	Hasil yang Diharapkan	Kesimpulan
1	<i>Login</i> dengan nim atau npp dan <i>password</i> yang kosong	Menampilkan dialog " <i>The Email / Username field is required.</i> "	Valid
2	<i>Login</i> dengan nim atau npp dan <i>password</i> yang salah	Menampilkan dialog " <i>Incorrect Login.</i> "	Valid
3	<i>Login</i> dengan <i>captcha</i> kosong atau salah	Menampilkan dialog " <i>Captcha Is Wrong.</i> "	Valid
4	<i>Input</i> pengajuan judul skripsi mahasiswa	Berhasil input data pengajuan dan menampilkan pada tabel pengajuan mahasiswa	Valid
5	Input pengajuan judul skripsi mahasiswa yang kedua saat status masih <i>onproses</i> pengajuan yang pertama	Menampilkan dialog "maaf anda sudah melakukan pengajuan judul skripsi dengan status <i>onproses</i> "	Valid
6	Memilih (klik) tombol disetujui pengajuan judul skripsi kaprodi	Menampilkan <i>pop up</i> "yakin ingin menyetujui judul ini" jika ya maka menampilkan "judul berhasil disetujui"	Valid
7	Memilih (klik) tombol ditolak pengajuan judul skripsi ketua program studi	Menampilkan <i>pop up</i> "yakin ingin menolak judul ini" jika ya maka menampilkan "judul berhasil ditolak"	Valid
8	<i>Upload</i> data bimbingan saat belum melakukan pengajuan judul mahasiswa	Menampilkan dialog "maaf data skripsi anda belum terdaftar"	Valid
9	Mencetak SK Skripsi	Menampilkan dialog "maaf data skripsi anda belum ada"	Valid
10	Dosen pembimbing memilih tombol (klik) selesai saat bimbingan belum mencapai minimal 12 kali bimbingan	Menampilkan dialog "maaf bimbingan mahasiswa ini belum mencapai batas yang ditentukan yaitu minimal 12 kali bimbingan"	Valid

KESIMPULAN

Penelitian ini menghasilkan sistem monitoring skripsi yang memiliki fungsionalitas mulai dari pendaftaran, distribusi dosen dan SK pembimbing, proses bimbingan dan kemajuan serta pendaftaran ujian skripsi menggunakan teknologi *Progressive Web Application* (PWA). Jika ada pembaharuan informasi berkaitan dengan skripsi, sistem dapat mengirimkan notifikasi kepada penggunanya. Sistem yang dibuat dapat diakses dengan baik melalui perangkat laptop maupun *mobile* (*smartphone* atau *tablet*). Hasil pengujian fungsionalitas sistem menggunakan *black box testing* tidak ditemukan adanya bug atau error ketika dijalankan pada perangkat *web* laptop maupun *mobile*.

SARAN

Adapun saran untuk pengembangan selanjutnya adalah memperbaiki keamanan sistem dan menambahkan metode pengujian keamanan sistem, agar dapat dipastikan data terlindung dengan baik. Selain itu kedepannya perlu dilakukan evaluasi usability untuk mengukur seberapa baik antar muka sistem dimata pengguna aplikasi

DAFTAR PUSTAKA

- [1] N. B. Argaheni, "Sistematik Review: Dampak Perkuliahan Daring Saat Pandemi COVID-19 Terhadap Mahasiswa Indonesia," *PLACENTUM J. Ilm. Kesehat. dan Apl.*, vol. 8, no. 2, p. 99, 2020, doi: 10.20961/placentum.v8i2.43008.
- [2] M. R. Ramadhan, L. E. Nugroho, and S. Sulisty, "Perancangan Sistem Informasi Monitoring Skripsi," in *Proceedings of Conference on Information Technology and Electrical Engineering*, 2017, pp. 290–295.
- [3] B. E. Kusuma, "Sistem Informasi Bimbingan Skripsi Berbasis Web Di Universitas Pelita Harapan," *J. ISD*, vol. 3, no. 1, pp. 71–78, 2018.
- [4] N. M. Neva Satyahadewi, "Sistem Informasi Monitoring Tugas Akhir (SIMTA)," *J. Comput. Eng. Syst. Sci.*, vol. 4, no. 1, pp. 83–87, 2019.
- [5] Y. Wiyatmo, "Efektivitas Bimbingan Tugas Akhir Skripsi (TAS) Mahasiswa Jurusan Pendidikan Fisika FMIPA UNY," in *Prosiding Seminar Nasional Penelitian, Pendidikan dan Penerapan MIPA*, 2010, pp. 405–414.
- [6] C. Rojas, *Building Progressive Web Applications with Vue.js: Reliable, Fast, and Engaging Apps with Vue.js*. New York: Apress Media LLC, 2019.
- [7] Nurwanto, "Penerapan Progressive Web Application (PWA) pada E-Commerce," *Techno.Com*, vol. 18, no. 3, pp. 227–235, 2019, doi: 10.33633/tc.v18i3.2400.
- [8] A. Setiawan, D. S. Rusdianto, and A. P. Kharisma, "Pengembangan Sistem Pelaporan Gangguan Berbasis Web dengan menggunakan Teknologi Progressive Web Application (Studi Kasus : Unit IT PT . Kereta Api Indonesia Daerah Operasi III Cirebon)," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, no. 5, pp. 4194–4203, 2019.

- [9] G. L. Dewi and S. Tjandra, "Pemanfaatan Progressive Web Apps Pada Web Akuntansi," *Teknika*, vol. 9, no. 1, pp. 38–47, 2020, doi: 10.34148/teknika.v9i1.252.
- [10] T. Ater, *Building Progressive Web Apps*. Sebastopol, California: O'Reilly Media, 2017.
- [11] A. Yulianto and D. P. Wigandi, "Implementasi Mobile-D Dalam Pengembangan Aplikasi Mobile Berbasis Android," in *Seminar Nasional Inovasi dan Tren (SNIT)*, 2018, no. August, pp. 1–6.
- [12] R. Andri, N. A. O. Saputri, and M. Akbar, "Sistem Notifikasi Tugas Akhir Universitas Bina Darma Berbasis Mobile," *Sistemasi*, vol. 9, no. 1, p. 155, 2020, doi: 10.32520/stmsi.v9i1.630.
- [13] S. P. Mukherjee, *A Guide to Research Methodology: An Overview Of Research Problems, Tasks and Methods*. New York: Taylor & Francis Group, LLC, 2020.
- [14] G. P. P, R Hamsini, and Smitha G R, "Agile Development Methodology and Testing for Mobile Applications - A Survey," *Int. J. New Technol. Res.*, vol. 2, no. 9, pp. 98–101, 2012.
- [15] H. K.Flora, S. V. Chande, and X. Wang, "Adopting an Agile Approach for the Development of Mobile Applications," *Int. J. Comput. Appl.*, vol. 94, no. 17, pp. 43–50, 2014, doi: 10.5120/16454-6199.
- [16] S. R. Jan, S. T. U. Shah, Z. U. Johar, Y. Shah, and F. Khan, "An Innovative Approach to Investigate Various Software Testing Techniques and Strategies," *IJSRSET*, vol. 2, no. 2, pp. 682–689, 2016.