

PENERAPAN PROGRESSIVE WEB APP PADA PEMBUATAN WEBSITE MAGANG STUDI KASUS PRODI INFORMATIKA UNSIKA

Muhammad Adif Fikri¹, Aji Primajaya, S.Si., M.Kom.², Mohamad Jajuli, M.Si.³

^{1,2,3}Program Studi Informatika, Fakultas Ilmu Komputer, Universitas Singaperbangsa Karawang

Email: muhammad.adif19101@student.unsika.ac.id¹, aji.primajaya@staff.unsika.ac.id²,

mohamad.jajuli@staff.unsika.ac.id³

ABSTRACT

Internship is one of the graduation requirements for students of the Faculty of Computer Science, Singaperbangsa University Karawang (Fasilkom Unsika), but the process of finding internship vacancies is still difficult because at Fasilkom Unsika there is no platform that provides information on internship vacancies for Fasilkom Unsika students. The internship registration process is also not effective because there is no system that takes care of student internship registration and still uses Google Form. Therefore, a website is needed that can provide information on internship vacancies and take care of internship registration. From the results of interviews with students, the website must also be able to run both on desktop and mobile devices so that the Progressive Web App (PWA) will be implemented on the website to be created. The internship website will be created using the Waterfall development method with 5 stages that are done systematically. Using Black Box testing and Lighthouse testing, it was found that the internship website created can run well on desktop and mobile devices and has good PWA compatibility.

Keywords: Internship Website, Progressive Web App, Waterfall, Black Box, Lighthouse.

ABSTRAK

Magang merupakan salah satu syarat kelulusan mahasiswa Fakultas Ilmu Komputer Universitas Singaperbangsa Karawang (Fasilkom Unsika), namun proses mencari lowongan magang masih sulit karena di Fasilkom Unsika belum ada platform yang menyediakan informasi lowongan magang untuk mahasiswa Fasilkom Unsika. Proses pendaftaran magang juga belum efektif karena belum adanya sistem yang mengurus pendaftaran magang mahasiswa dan masih menggunakan *Google Form*. Oleh karena itu, dibutuhkan sebuah *website* yang dapat menyediakan informasi lowongan magang dan mengurus pendaftaran magang. Dari hasil wawancara dengan mahasiswa, *website* juga harus dapat berjalan baik pada perangkat desktop dan *mobile* sehingga *Progressive Web App* (PWA) akan di implementasikan pada *website* yang akan dibuat. *Website* magang akan dibuat menggunakan metode pengembangan *Waterfall* dengan 5 tahapan yang dikerjakan dengan sistematis. Menggunakan *Black Box testing* dan *Lighthouse testing*, didapatkan bahwa *website* magang yang dibuat dapat berjalan dengan baik pada perangkat desktop dan *mobile* serta memiliki kompatibilitas dengan PWA yang baik.

Kata Kunci: Website Magang, Progressive Web App, Waterfall, Black Box, Lighthouse.

Riwayat Artikel :

Tanggal diterima : 16-10-2023

Tanggal revisi : 23-10-2023

Tanggal terbit : 24-10-2023

DOI :

<https://doi.org/10.31949/infotech.v9i2.7059>

INFOTECH journal by Informatika UNMA is licensed under CC BY-SA 4.0

Copyright © 2023 By Author



1. PENDAHULUAN

1.1. Latar Belakang

Perubahan sosial, budaya, dunia kerja, dan kemajuan teknologi memaksa perguruan tinggi dan pemerintah untuk menyiapkan gaya edukasi baru agar lulusannya dapat beradaptasi dengan baik di dunia kerja serta siap menghadapi masa depan yang sangat cepat berubah. Kampus Merdeka, wadah yang disiapkan pemerintah bekerja sama dengan perguruan tinggi dan berbagai instansi dari segala sektor diharapkan dapat menjadi jembatan penghubung antara mahasiswa dan industri di berbagai sektor (Kemendikbud., 2022).

Dengan kebijakan Merdeka Belajar, mahasiswa dibebaskan untuk belajar dan mencari pengalaman baru di luar kampus. Magang, Studi Independen, Kampus Mengajar, Pertukaran Mahasiswa, dan beberapa program lainnya merupakan berbagai macam program yang dapat diambil mahasiswa untuk dapat belajar dan mencari pengalaman di luar kampus (Kemendikbud., 2022). Magang menjadi salah satu program terpenting terutama pada mahasiswa Fakultas Ilmu Komputer Universitas Singaperbangsa Karawang karena mahasiswanya diwajibkan untuk magang setidaknya satu kali sebagai syarat kelulusan.

Pada Fakultas Ilmu Komputer atau yang disingkat Fasilkom, proses mencari lowongan magang masih sulit karena di Fasilkom belum ada platform yang menyediakan informasi lowongan magang untuk mahasiswa Fasilkom. Dari hasil wawancara dengan salah satu mahasiswa program studi Informatika Fasilkom yang pernah mengikuti magang mandiri didapatkan bahwa untuk mencari tempat magang, ia harus bolak-balik ke berbagai instansi untuk menanyakan ketersediaan lowongan magang. Beliau juga menambahkan bahwa tidak semua instansi mau menerima peserta magang karna waktu magang yang relatif singkat. Bukan hanya itu, instansi yang menerima peserta magang juga belum tentu memiliki lowongan yang cocok dengan mahasiswa Informatika sehingga membuat mencari tempat magang semakin sulit. Beberapa mahasiswa lain yang diwawancarai juga setuju terkait kesulitan mencari lowongan magang dan berharap dihadirkan platform yang dapat membagikan informasi lowongan magang untuk mahasiswa Fasilkom.

Pada wawancara dengan bapak Muhammad Rizki Fauzi Suharto, S.Pd., salah satu staf tata usaha (TU) yang juga merupakan tim magang. Beliau menyebutkan kalau proses pendaftaran magang juga belum efektif karena belum adanya sistem yang mengurus pendaftaran magang mahasiswa dan masih mengandalkan Google Form. Beliau melanjutkan bahwa pada alur pendaftaran saat ini Google Form harus diisi terlebih dahulu oleh mahasiswa dan setelah diterima, TU harus mengisi kembali secara manual Google Form untuk dapat mengirimkan surat yang diperlukan ke surel mahasiswa.

Beliau juga mengatakan bahwa Fasilkom sudah memiliki banyak kerja sama dengan berbagai perusahaan namun masih bermasalah pada komunikasi dengan perusahaan terkait apakah perusahaan masih menerima mahasiswa magang, berapa mahasiswa magang yang diperlukan, dan kualifikasi apa yang dibutuhkan. Ia berharap bahwa pembuatan *website* magang ini dapat menyederhanakan alur pendaftaran magang dan sekaligus memudahkan perusahaan dalam mengkomunikasikan jumlah dan kriteria mahasiswa magang yang diperlukan.

Dari hasil wawancara juga didapatkan bahwa optimalisasi *website* untuk mendukung perangkat *mobile* diperlukan. Dua dari tiga mahasiswa yang di wawancara mengatakan lebih sering membuka halaman *website* yang berkaitan dengan kegiatan perkuliahan menggunakan *smartphone*. Sedangkan satu mahasiswa yang lebih sering menggunakan perangkat *desktop* mengeluhkan pengalaman menggunakan *smartphone* yang tidak optimal membuatnya hanya akan membuka *website* perkuliahan dengan *smartphone* saat terpaksa dan tidak ada pilihan lain.

Progressive Web App (PWA) merupakan konsep yang menggabungkan pengalaman terbaik web dan bagian terbaik aplikasi *native* dengan memanfaatkan teknologi web browser seperti *app manifest* dan *service worker*. PWA dapat meningkatkan efisiensi pembuatan sistem yang *multi-platform* dengan membuat sebuah *website* dapat berjalan layaknya aplikasi *native* pada perangkat Android, IOS, Windows, Linux, macOS, dan ChromeOS. Hal ini sangat berguna terutama pada pengembang yang bekerja sendiri atau tim pengembang kecil yang hanya berisikan beberapa orang dimana mereka dapat fokus pada pengembangan web dan membiarkan PWA menangani pengguna dari berbagai platform (Maximiliano et al., n.d).

Dengan mengimplementasikan konsep PWA, *website* akan dapat berjalan di berbagai platform seperti *website*, *desktop*, dan *mobile*. Beberapa keunggulan dari PWA yaitu memiliki sifat aplikasi *native* layaknya Android, tetapi tidak memerlukan spesifikasi *hardware* yang tinggi dalam membuatnya, dapat mempercepat proses pembuatan *website* dan mengurangi beban server, dapat berfungsi secara *offline*. Tidak seperti *website* yang tidak dapat diakses jika tidak ada koneksi internet, dan sudah banyak digunakan di perusahaan besar seperti Twitter, Facebook, Bukalapak, Tokopedia, dan sebagainya (Aripin, S., Somantri, 2021).

Berdasarkan permasalahan di atas, didapatkan kriteria *website* yang akan dibuat pada penelitian ini yaitu sebuah *website* magang dengan fitur utama: dapat menampilkan informasi lowongan magang, dapat melakukan administrasi pendaftaran magang, dan perusahaan dapat langsung membagikan informasi lowongan magang beserta kriteria mahasiswa yang dibutuhkan. Implementasi PWA juga diperlukan pada pembuatan *website* magang

agar *website* dapat berjalan dengan baik pada perangkat desktop dan *mobile*.

1.2. Tinjauan Pustaka

1. Magang

Magang merupakan pembelajaran bagi mahasiswa dengan tujuan mempersiapkan mahasiswa tersebut untuk dapat menjadi SDM profesional yang siap menghadapi dunia kerja (Wisnumurti, Faulina, & Novari, 2023).

2. Waterfall

Sekitar tahun 1970, Winston Royce memperkenalkan model pengembangan perangkat lunak baru dengan nama *Waterfall*. Metode ini terus dipakai sampai sekarang hingga menjadi model paling populer dalam *Software Engineering* (Wahid, 2020).

Karena menggambarkan pendekatan yang sistematis dan berurut dalam proses mengembangkan perangkat lunak, metode waterfall juga dapat disebut "*Linear Sequential Model*" (Wahid, 2020).

3. Website

Website merupakan metode dalam membagikan informasi di internet dalam bentuk gambar, tulisan, juga video yang dapat diakses menggunakan browser. Kelebihan dari website adalah kemampuan untuk mengaitkan (link) satu dokumen dengan dokumen lainnya (*hypertext*) (Romadhon, Yudhistira, & Mukrodin, 2021).

4. Progressive Web App

Progressive Web Apps (PWA) merupakan konsep yang dikemukakan oleh Alex Russell dan Frances Berriman pada 2015 dengan tujuan menggabungkan pengalaman terbaik web dan bagian terbaik aplikasi *native* menggunakan teknologi web browser yaitu *service workers* dan *app manifest*. Dengan penerapan konsep PWA yang baik, akan didapatkan sebuah *website* yang dapat diandalkan, cepat, dan menarik (Adi, Akbar, & Khotimah, 2017).

5. Web Responsif

Web responsif merupakan serangkaian teknik dalam pengembangan situs web dengan tujuan dapat menyesuaikan dirinya sendiri saat digunakan oleh berbagai perangkat dan mampu membentuk kembali dirinya sendiri pada berbagai ukuran layar dan resolusi (Wasino, Herwindiati, Setyawan, & Maupa (2023).

6. UML

Unified Modelling Language (UML) adalah bahasa dalam pembuatan model sistem atau model perangkat lunak dengan berorientasi objek. UML digunakan untuk memudahkan pemahaman konsep sebuah sistem yang akan dibuat dengan menampilkannya sebagai sebuah diagram (Suendri, 2018).

7. Use Case Diagram

Use case diagram adalah gambaran dari interaksi sistem dan aktor dengan tujuan menampilkan beragam fungsi yang dimiliki sistem serta bagaimana fungsi akan digunakan (Mukhtar, & Asmasari, 2021).

8. Activity Diagram

Activity diagram adalah diagram yang digunakan untuk menggambarkan objek, aktifitas, keadaan, transisi *state*, dan kejadian. Dengan menggunakan *activity diagram* kita dapat menggambarkan aktivitas yang mampu dilakukan oleh sistem (Suendri, 2018).

9. Class Diagram

Class diagram adalah diagram yang digunakan untuk menggambarkan struktur sistem *database* yang akan digunakan suatu program (Suryadin, 2022).

10. Black Box Testing

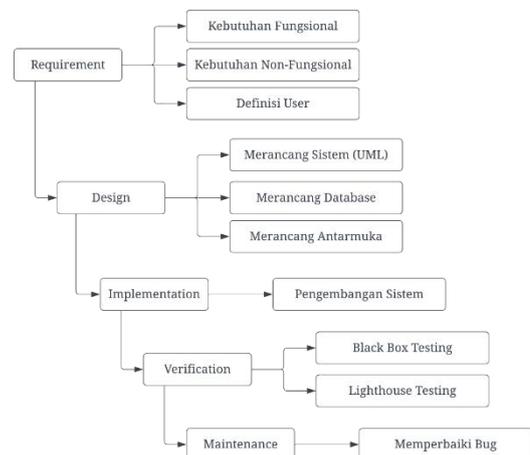
Black Box testing adalah tes yang dilakukan pada perangkat lunak untuk mengevaluasi semua kegunaan elemen yang dapat berinteraksi dengan pengguna seperti tampilan, tombol, dan elemen lainnya. Black Box memiliki beberapa metode antara lain *equivalence partitioning*, dan *boundary value analysis* (Hidayat, & Muttaqin, 2020).

11. Lighthouse Testing

Lighthouse merupakan alat bantu yang disediakan Chrome untuk meningkatkan kualitas aplikasi web. Lighthouse testing akan menjalankan serangkaian pengujian secara otomatis pada halaman yang ditentukan, lalu menghasilkan sebuah laporan pengujian halaman tersebut (Faizah, Aji, & Hartiyah, 2023).

1.3. Metodologi Penelitian

Metodologi yang akan digunakan pada penelitian ini adalah *Software Development Life Cycle* (SDLC) *Waterfall*. SDLC *waterfall* mengharuskan pengembangan berjalan secara sistematis dan berurutan mulai dari tahap awal *requirement* hingga tahapan akhir *maintenance*. Setiap tahapan harus diselesaikan sebelum dapat lanjut ke tahapan berikutnya dan tidak diperbolehkan untuk kembali atau mengulangi tahap sebelumnya.



Gambar 1. Rancangan Penelitian

1. Requirement
 - 1.1. Kebutuhan Fungsional
Tahap ini akan dilakukan penentuan kebutuhan fungsional dari *website* magang.
 - 1.2. Kebutuhan Non-Fungsional
Tahap ini akan dilakukan penentuan kebutuhan non-fungsional dari *website* magang.
 - 1.3. Definisi User
Tahap ini akan dilakukan penentuan *user* yang menggunakan *website* magang.
2. Design
Pada tahap design akan dilakukan perancangan terhadap berbagai aspek sistem untuk memudahkan implementasi pada tahapan selanjutnya. Tahap *design* pada penelitian ini akan dibagi menjadi tiga, yaitu :
 - 2.1. Merancang Sistem
Tahap ini akan menentukan alur dari sistem dengan menggunakan *Unified Modelling Language* (UML). Beberapa diagram UML yang akan digunakan adalah *use case* diagram untuk menggambarkan fungsi yang dimiliki dari sistem serta interaksinya dengan aktor, *activity* diagram untuk menggambarkan lebih detail aktivitas yang mampu dilakukan oleh sistem, dan *class* diagram untuk menggambarkan struktur sistem *database* dari sistem. Alat yang akan digunakan adalah Astah UML.
 - 2.2. Merancang Database
Pada tahap ini akan dibuatkan rancangan *database* untuk digunakan sistem sesuai dengan kebutuhan yang telah ditentukan. Alat yang akan digunakan adalah DBAver.
 - 2.3. Merancang Antarmuka
Pada tahap ini akan dibuatkan rancangan antarmuka dari sistem yang akan dibuat agar dapat mengakomodasi penggunaan sistem pada perangkat desktop juga perangkat *mobile*. Alat yang akan digunakan adalah Balsamiq Mockup.
3. Implementation
 - 3.1. Pengembangan Sistem
Pada tahap ini sistem akan dibuat mengikuti rancangan yang telah dibuat sebelumnya. Alat yang akan digunakan adalah Visual Studio Code, Laragon, dan PHP 8.1.
4. Verification
 - 4.1. Black Box Testing
Pada tahap ini akan dilakukan pengujian dengan metode Black Box. Dengan pengujian ini akan didapatkan apakah semua fungsi luar seperti tampilan dan aksi *inputan* berfungsi dengan baik.
 - 4.2. Lighthouse Testing
Pada tahap ini akan dilakukan pengujian dengan Lighthouse. Tujuan pengujian ini

untuk mengetahui bagaimana kualitas halaman web serta memverifikasi apakah PWA berjalan dengan baik.

5. Maintenance
 - 5.1. Memperbaiki Bug
Pada tahap ini akan dilakukan perbaikan terhadap *bug* (kesalahan) yang ditemukan pada sistem saat tahap *verification*.

2. HASIL PENELITIAN

2.1. Requirement

2.1.1. Definisi User

User dari *website* magang akan di definisikan sebagai berikut :

Tabel 1. Definisi User

User	Keterangan
Staf TU / Admin	Akun yang akan digunakan sebagai admin dan memiliki akses mengelola dokumen pendaftaran magang, data lowongan magang, data mahasiswa, dan data perusahaan.
Verifikator	Akun yang akan digunakan untuk memonitor dan memverifikasi permintaan pembuatan dokumen.
Mahasiswa	Akun yang akan digunakan mahasiswa dan memiliki akses membuat pengajuan dokumen pendaftaran magang, melihat, mencari, menambahkan, dan mendaftar pada lowongan magang tertentu.
Perusahaan	Akun yang akan digunakan perusahaan dan memiliki akses mengelola lowongan magang pada perusahaan, dan melihat pelamar pada lowongan perusahaan.

2.1.2. Kebutuhan Fungsional

Kebutuhan fungsional dari *website* magang yang didapatkan dari hasil wawancara dengan mahasiswa dan staf TU serta dari hasil observasi peneliti yaitu:

Tabel 2. Daftar Fitur Kebutuhan Fungsional

Kode	Fitur	Deskripsi
KF001	Login	Dapat melakukan login
KF002	Register	Dapat melakukan register
KF003	Ubah profil	Dapat mengubah data profil
KF004	Kelola user	Dapat menambah, melihat, mengubah, dan menghapus user
KF005	Kelola lowongan magang	Dapat menambah, melihat, mengubah, dan menghapus lowongan magang
KF006	Lihat lowongan magang	Dapat melihat lowongan magang yang tersedia
KF007	Ajukan surat permohonan magang	Dapat mengajukan pembuatan surat permohonan magang
KF008	Unggah surat permohonan magang	Dapat mengunggah surat permohonan magang

Kode	Fitur	Deskripsi
KF009	Unduh surat permohonan magang	Dapat mengunduh surat permohonan magang
KF010	Lamar lowongan magang	Dapat melamar lowongan magang
KF011	Lihat pelamar lowongan magang	Dapat melihat pelamar pada lowongan magang
KF012	Ajukan surat plot pembimbing	Dapat mengajukan pembuatan surat plot pembimbing
KF013	Verifikasi surat plot pembimbing	Dapat menyetujui atau menolak pengajuan surat plot pembimbing
KF014	Unggah surat plot pembimbing	Dapat mengunggah surat plot pembimbing yang telah disetujui
KF015	Unduh surat plot pembimbing	Dapat mengunduh surat plot pembimbing
KF016	Monitor pendaftaran magang	Dapat memonitor kegiatan pendaftaran magang

Tabel 3. Fitur Kebutuhan Fungsional Berdasarkan User

User	Fitur
Admin	KF001, KF003, KF004, KF005, KF008, KF011, KF014
Verifikator	KF001, KF003, KF013, KF016
Mahasiswa	KF001, KF003, KF005, KF006, KF007, KF009, KF010, KF012, KF015
Perusahaan	KF001, KF002, KF003, KF005, KF011

2.1.3. Kebutuhan Non-Fungsional

Kebutuhan non-fungsional dari website magang adalah perangkat keras dan perangkat lunak yang akan digunakan pada pembuatan website yaitu :

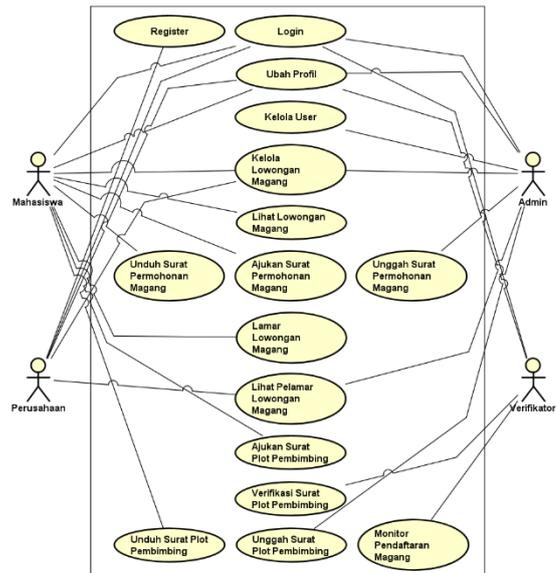
Tabel 4. Kebutuhan Non-Fungsional

Kategori	Keterangan
Perangkat Keras	CPU Intel I3-12100F, GPU Radeon RX 6600, RAM 16 GB, dan SSD 512 GB.
Perangkat Lunak	Google Chrome, Visual Studio Code, DBaver, Astah UML, Balsamiq Mockup, Laragon, PHP 8.1, dan MySQL.

2.2. Design

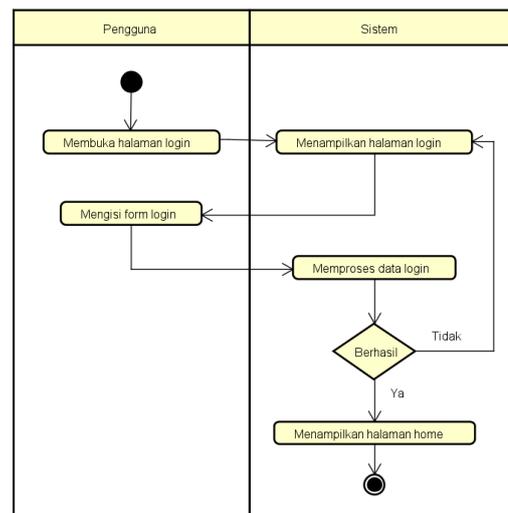
2.2.1. Merancang Sistem

Use case diagram akan digunakan untuk menggambarkan fungsi yang dimiliki dari sistem serta interaksinya dengan aktor seperti sebagai berikut:

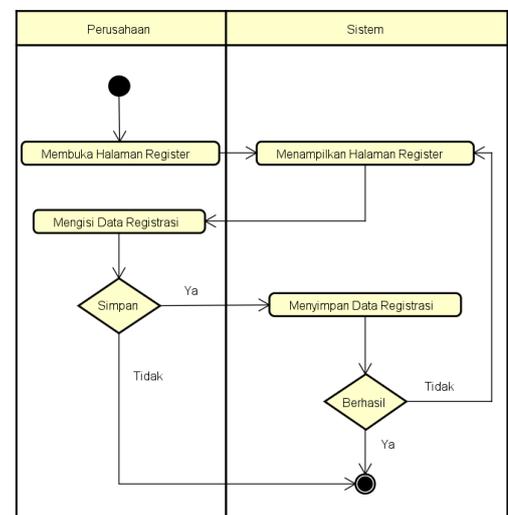


Gambar 2. Use Case Diagram

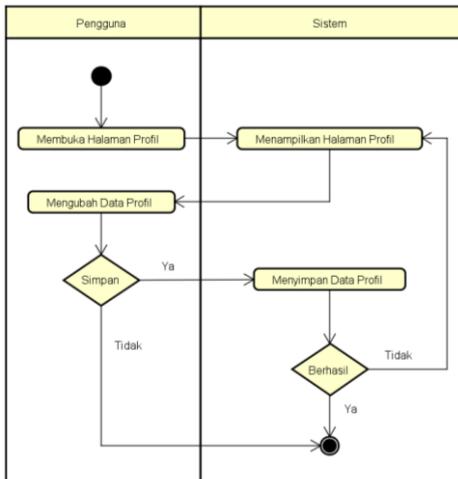
Activity diagram akan digunakan untuk menggambarkan lebih detail aktivitas yang mampu dilakukan oleh sistem seperti sebagai berikut:



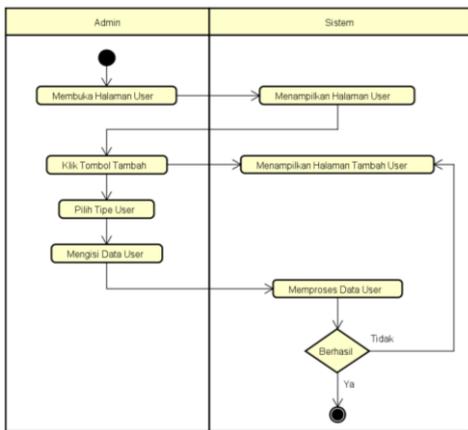
Gambar 3. Activity Diagram Login



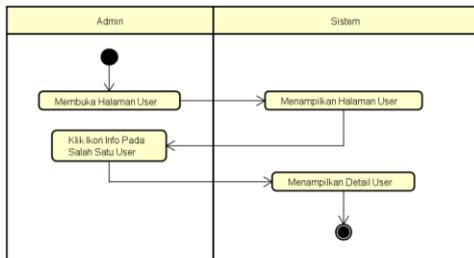
Gambar 4. Activity Diagram Register Perusahaan



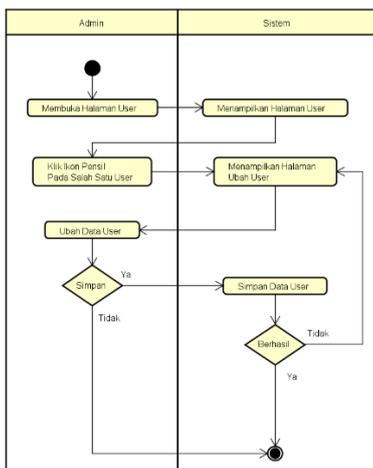
Gambar 5. Activity Diagram Profile



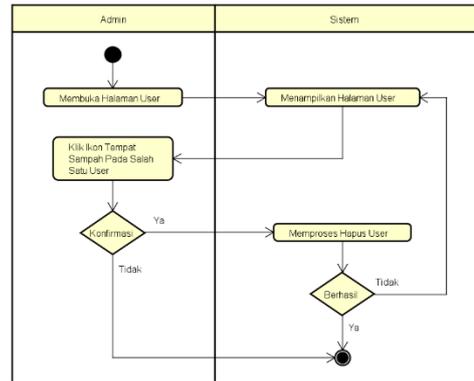
Gambar 6. Activity Diagram Tambah User



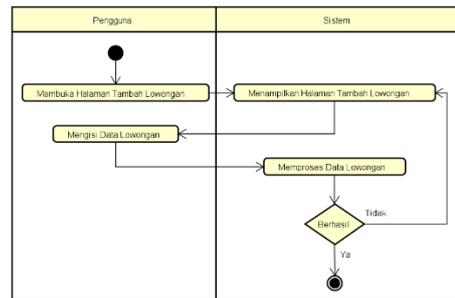
Gambar 7. Activity Diagram Lihat User



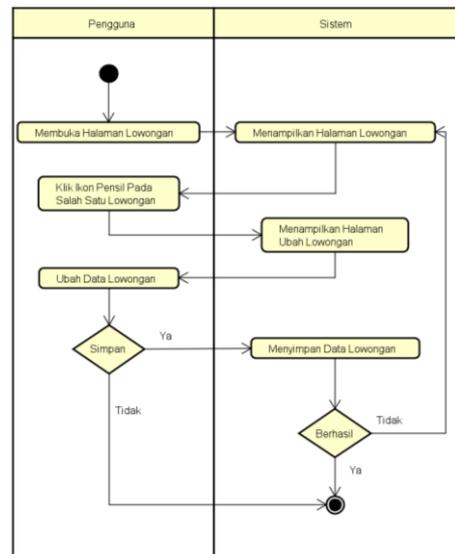
Gambar 8. Activity Diagram Ubah User



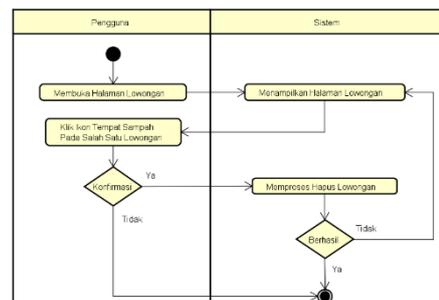
Gambar 9. Activity Diagram Hapus User



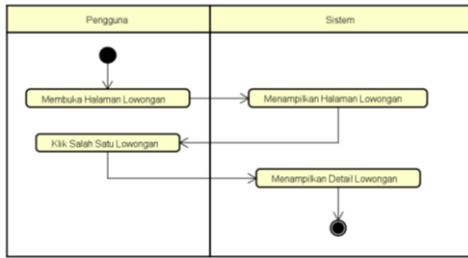
Gambar 10. Activity Diagram Tambah Lowongan Magang



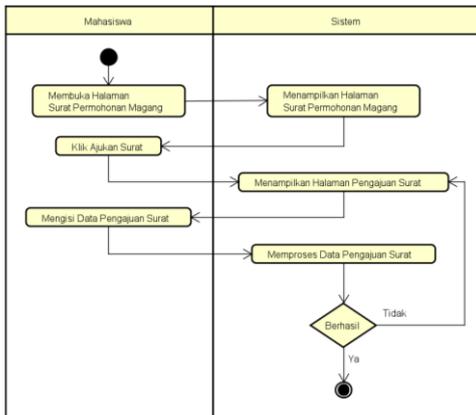
Gambar 11. Activity Diagram Ubah Lowongan Magang



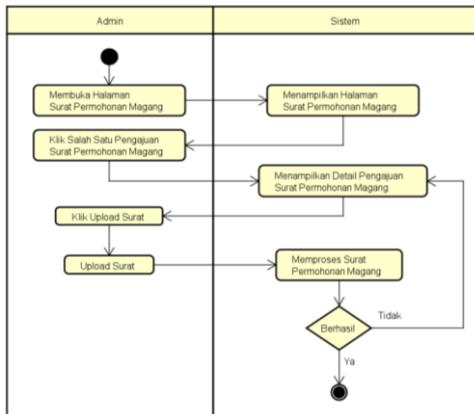
Gambar 12. Activity Diagram Hapus Lowongan Magang



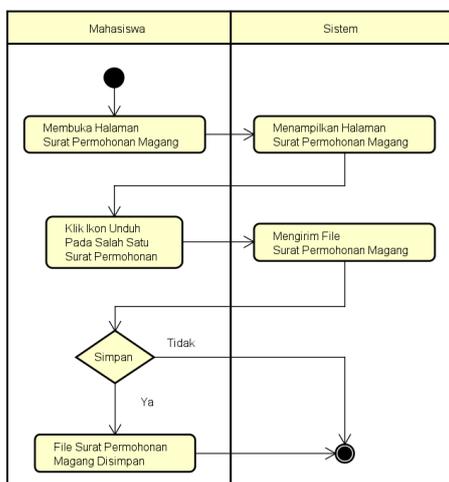
Gambar 13. Activity Diagram Lihat Lowongan Magang



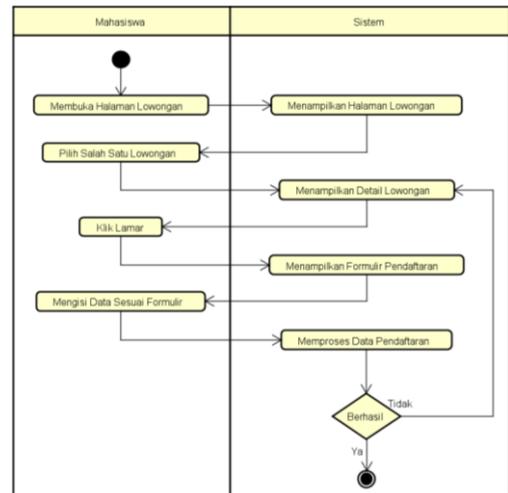
Gambar 14. Activity Diagram Ajukan Surat Permohonan Magang



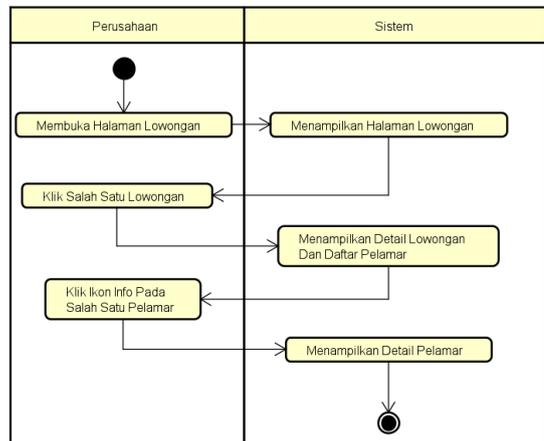
Gambar 15. Activity Diagram Unggah Surat Permohonan Magang



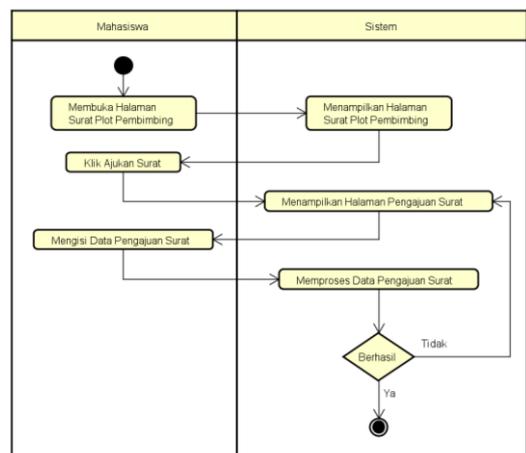
Gambar 16. Activity Diagram Unduh Surat Permohonan Magang



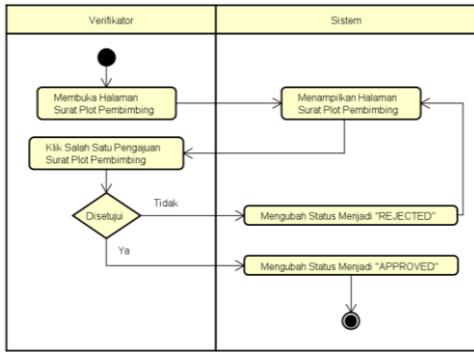
Gambar 17. Activity Diagram Lamar Lowongan Magang



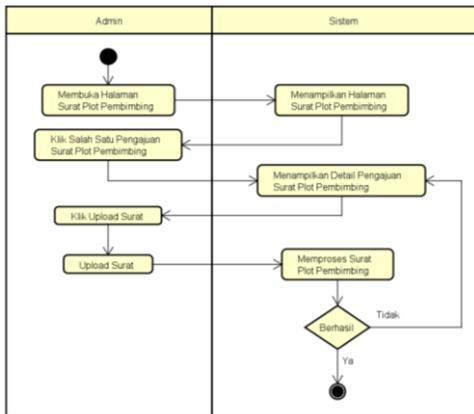
Gambar 18. Activity Diagram Lihat Pelamar Lowongan Magang



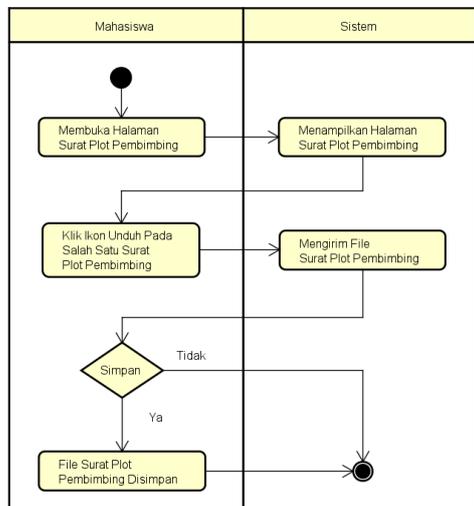
Gambar 19. Activity Diagram Ajukan Surat Plot Pembimbing



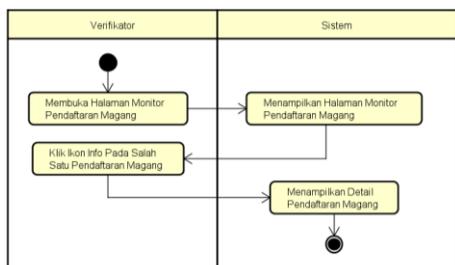
Gambar 20. Activity Diagram Verifikasi Surat Plot Pembimbing



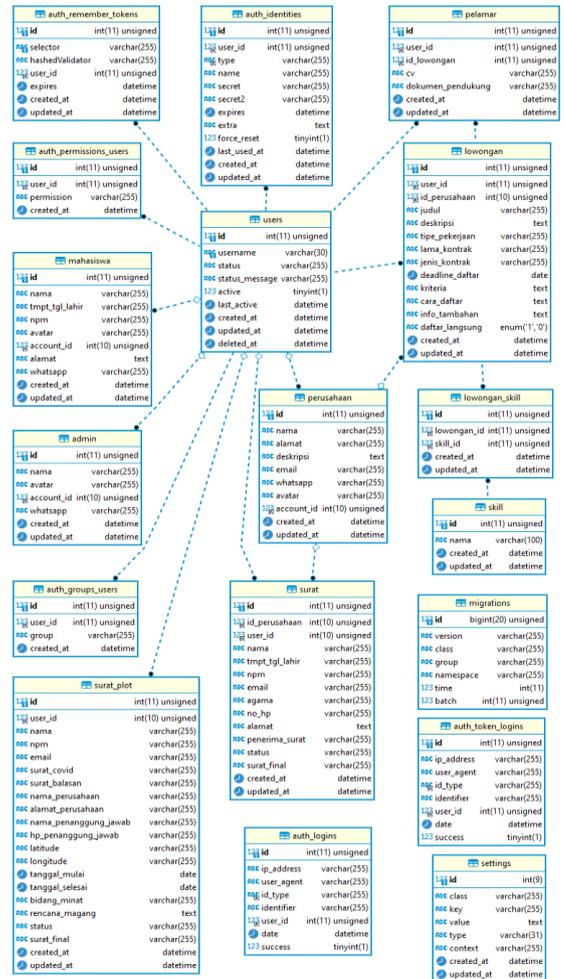
Gambar 21. Activity Diagram Uunggah Surat Plot Pembimbing



Gambar 22. Activity Diagram Unduh Surat Plot Pembimbing



Gambar 23. Activity Diagram Monitor Pendaftaran



Gambar 24. Class Diagram

2.2.2. Merancang Database

MySQL akan digunakan untuk menangani database dari website yang akan dibuat. Database akan dijalankan server lokal menggunakan Laragon dan tabel yang akan digunakan dalam website ini adalah :

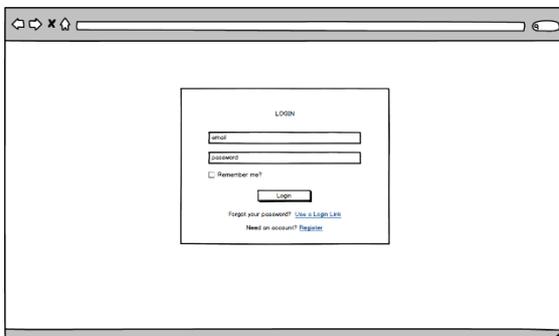
Tabel 5. Daftar Tabel Pada Website Magang

Nama Tabel	Deskripsi
admin	Digunakan untuk menyimpan data profil admin
auth_groups_users	Tabel bawaan Codeigniter Shield. Digunakan untuk menyimpan jenis user dari suatu akun
auth_identities	Tabel bawaan Codeigniter Shield. Digunakan untuk menyimpan data kredensial akun
auth_logins	Tabel bawaan Codeigniter Shield. Digunakan untuk mencatat kegiatan login dalam sistem
auth_permissions_users	Tabel bawaan Codeigniter Shield. Digunakan untuk menyimpan izin akses akun
auth_remember_tokens	Tabel bawaan Codeigniter Shield. Digunakan untuk menyimpan data login "remember me"

Nama Tabel	Deskripsi
auth_token_logins	Tabel bawaan Codeigniter Shield. Digunakan untuk menyimpan data access token
lowongan	Digunakan untuk menyimpan data lowongan magang
lowongan_skill	Digunakan untuk menyimpan data kemampuan yang diperlukan pada lowongan magang
mahasiswa	Digunakan untuk menyimpan data profil mahasiswa
migrations	Tabel bawaan Codeigniter. Digunakan untuk menyimpan data migrasi
pelamar	Digunakan untuk menyimpan data pelamar pada lowongan magang
perusahaan	Digunakan untuk menyimpan data profil perusahaan
settings	Tabel bawaan Codeigniter. Digunakan untuk menyimpan data pengaturan Codeigniter
skill	Digunakan untuk menyimpan data skill
surat	Digunakan untuk menyimpan data surat permohonan magang
surat_plot	Digunakan untuk menyimpan data surat plot pembimbing
users	Tabel bawaan Codeigniter Shield. Digunakan untuk menyimpan data akun pengguna

2.2.3. Merancang Antarmuka

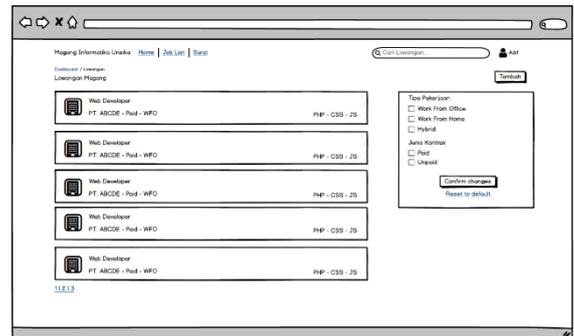
Balsamic Mockup digunakan untuk membuat rancangan antarmuka. Dua tipe rancangan antarmuka dibuat yaitu untuk ukuran layar desktop dan ukuran layar *smartphone*. Berikut beberapa contoh hasil rancangan antarmuka :



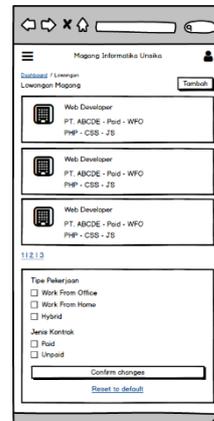
Gambar 25. Rancangan Antarmuka Halaman Login (Desktop)



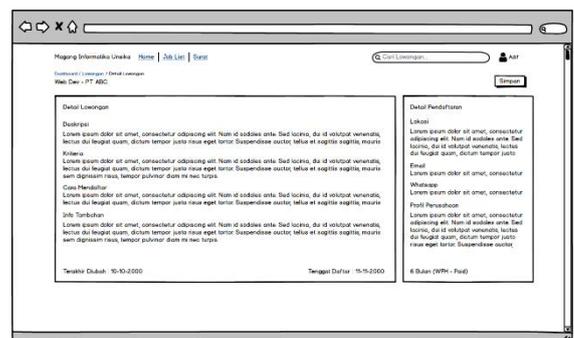
Gambar 26. Rancangan Antarmuka Halaman Login (Smartphone)



Gambar 27. Rancangan Antarmuka Halaman Lowongan Magang (Desktop)



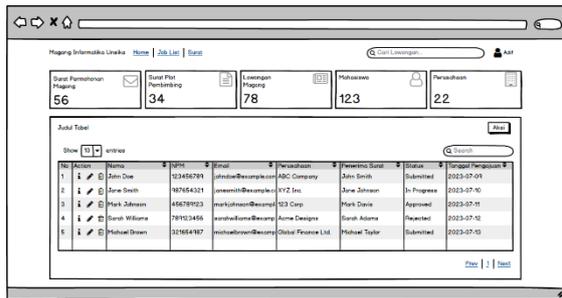
Gambar 28. Rancangan Antarmuka Halaman Lowongan Magang (Smartphone)



Gambar 29. Rancangan Antarmuka Halaman Detail Lowongan (Desktop)



Gambar 30. Rancangan Antarmuka Halaman Detail Lowongan (Smartphone)



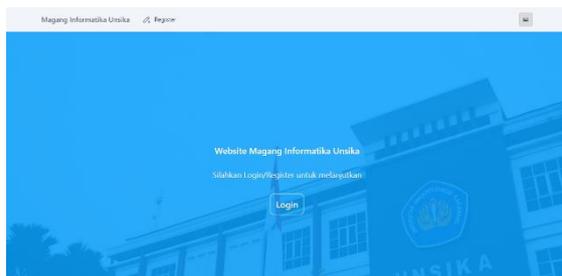
Gambar 31. Rancangan Antarmuka Halaman Dashboard Pengelola (Desktop)



Gambar 32. Rancangan Antarmuka Halaman Dashboard Pengelola (Smartphone)

2.3. Implementation

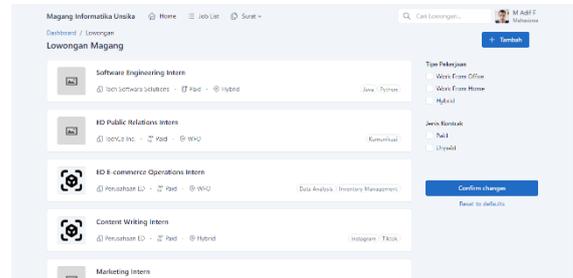
Pada tahap ini akan dilakukan pembuatan *website* magang. Berikut adalah beberapa contoh hasil tampilan dari *website* magang yang dibuat :



Gambar 33. Antarmuka Halaman Homepage (Desktop)



Gambar 34. Antarmuka Halaman Homepage (Smartphone)



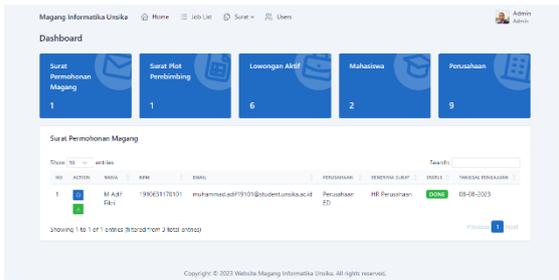
Gambar 35. Antarmuka Halaman Lowongan Magang Mahasiswa (Desktop)



Gambar 36. Antarmuka Halaman Lowongan Magang Mahasiswa (Smartphone)



Gambar 37. Antarmuka Halaman Detail Lowongan Mahasiswa (Desktop)



Gambar 38. Antarmuka Halaman Dashboard Admin (Dekstop)



Gambar 39. Antarmuka Halaman Dashboard Admin (Smartphone)

2.4. Verification

Pada tahap ini, *website* magang yang telah dibuat akan di tes menggunakan dua metode yaitu black box testing dan lighthouse testing.

2.4.1. Black Box Testing

Black box testing dilakukan untuk menguji tampilan dan aksi inputan *website* magang yang telah dibuat. Terdapat 349 tes yang akan dilakukan pada 41 halaman dan 4 panel navigasi dari masing-masing aktor. Berikut adalah beberapa contoh hasil pengesanan :

Tabel 6. Black Box Testing Halaman Homepage

Kode	Deskripsi	Hasil yang diharapkan	Hasil pengujian	Status
A001	Tata letak	Tata letak antarmuka sesuai pada perangkat desktop dan <i>mobile</i>	Tata letak sesuai	Sukses
A002	Tombol Menu pada perangkat <i>mobile</i>	Dapat menampilkan menu <i>login</i> dan register	Menampilkan menu <i>login</i> dan register	Sukses
A003	Tombol <i>Login</i>	Dapat mengarahkan ke halaman <i>login</i>	Berpindah ke halaman <i>login</i>	Sukses

A004	Tombol Register	Dapat mengarahkan ke halaman register	Berpindah ke halaman register	Sukses
A005	Gambar profile	Dapat menampilkan menu <i>login</i> pada dropdown	Menampilkan menu <i>login</i> pada dropdown	Sukses
A006	Link Magang Informatika Unsika	Dapat mengarahkan ke halaman <i>homepage</i>	Berpindah ke halaman <i>homepage</i>	Sukses

Tabel 7. Hasil Black Box Halaman Lowongan Magang (Mahasiswa)

Kode	Deskripsi	Hasil yang diharapkan	Hasil pengujian	Status
BB01	Tata letak	Tata letak antarmuka sesuai pada perangkat desktop dan <i>mobile</i>	Tata letak sesuai	Sukses
BB02	Link Dashboard	Dapat mengarahkan ke halaman dashboard	Berpindah ke halaman dashboard	Sukses
BB03	Tombol Tambah	Dapat mengarahkan ke halaman tambah lowongan magang	Berpindah ke halaman tambah lowongan magang	Sukses
BB04	Link Judul lowongan	Dapat mengarahkan ke halaman detail lowongan magang	Berpindah ke halaman detail lowongan magang	Sukses
BB05	Tombol Filter confirm changes	Dapat menampilkan lowongan magang sesuai filter yang dipilih	Lowongan magang yang tampil sesuai dengan filter yang dipilih	Sukses
BB06	Tombol Reset to default	Dapat mereset filter dan menampilkan semua lowongan yang tersedia	Filter di reset dan semua lowongan ditampilkan	Sukses
BB07	Pencarian pada tabel	Dapat mencari data sesuai kata kunci yang dimasukkan	Data tampil sesuai kata kunci yang ditulis	Sukses
BB08	Tombol halaman pada lowongan	Dapat mengubah lowongan yang tampil sesuai halaman yang dipilih	Halaman lowongan berubah sesuai dengan pilihan	Sukses

Tabel 8. Hasil Black Box Halaman Dashboard (Admin)

Kode	Deskripsi	Hasil yang diharapkan	Hasil pengujian	Status
AA01	Tata letak	Tata letak antarmuka sesuai pada perangkat desktop dan mobile	Tata letak sesuai	Sukses
AA02	Pencarian pada tabel	Dapat mencari data sesuai kata kunci yang dimasukkan	Data tampil sesuai kata kunci yang ditulis	Sukses
AA03	Dropdown Show entries pada tabel	Dapat menentukan berapa banyak data yang ditampilkan pada tabel	Jumlah maksimal data yang tampil mengikuti angka	Sukses
AA04	Ikon Detail pada tabel	Dapat mengarahkan ke halaman detail surat permohonan magang	Berpindah ke halaman detail surat permohonan magang	Sukses
AA05	Ikon Reject pada tabel	Dapat menampilkan pesan peringatan dan konfirmasi reject surat permohonan magang. Hanya muncul saat status pending	Menampilkan pesan peringatan dan konfirmasi reject surat permohonan magang. Mengubah status menjadi rejected saat menekan Ya	Sukses
AA06	Ikon Download pada tabel	Dapat mengunduh surat permohonan magang yang telah dibuat. Hanya muncul saat status done	Sistem menanyakan dimana ingin menyimpan surat dan mulai mengunduh	Sukses
AA07	Tombol halaman pada tabel	Dapat mengubah isi tabel sesuai halaman yang dipilih	Halaman tabel berubah sesuai dengan pilihan	Sukses

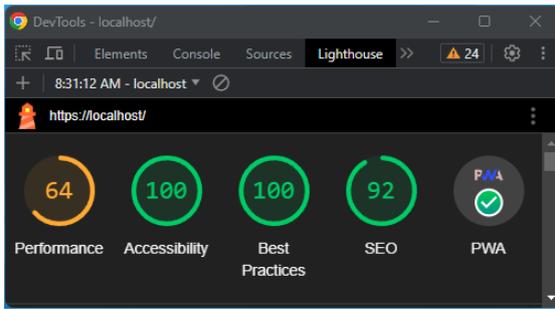
2.4.2. Lighthouse Testing

Lighthouse testing dilakukan untuk melihat kualitas, performa dan kompatibilitas PWA dari *website* magang yang telah dibuat. Berikut adalah hasil pengetesan pada 41 halaman *website* :

Tabel 9. Hasil Pengujian Lighthouse

Halaman	Performa	Best Practices	PWA
Homepage	64	100	Ya

Halaman	Performa	Best Practices	PWA
Login	100	100	Tidak
Register	100	100	Tidak
Dashboard (Admin)	98	100	Ya
Lowongan Magang (Admin)	90	100	Ya
Tambah Lowongan (Admin)	86	100	Ya
Detail Lowongan (Admin)	98	100	Ya
Ubah Lowongan (Admin)	85	100	Ya
Detail Pelamar (Admin)	90	92	Ya
Surat Permohonan Magang (Admin)	96	100	Ya
Detail Surat Permohonan Magang (Admin)	98	100	Ya
Surat Plot Pembimbing (Admin)	96	100	Ya
Detail Surat Plot Pembimbing (Admin)	91	92	Ya
Kelola User (Admin)	88	100	Ya
Tambah User (Admin)	98	100	Ya
Detail User (Admin)	97	100	Ya
Ubah User (Admin)	98	100	Ya
Profile (Admin)	98	100	Ya
Dashboard (Mahasiswa)	98	100	Ya
Lowongan Magang (Mahasiswa)	98	100	Ya
Tambah Lowongan (Mahasiswa)	96	100	Ya
Detail Lowongan (Mahasiswa)	98	100	Ya
Ubah Lowongan (Mahasiswa)	87	100	Ya
Surat Permohonan Magang (Mahasiswa)	91	100	Ya
Ajukan Surat Permohonan Magang (Mahasiswa)	98	100	Ya
Detail Surat Permohonan Magang (Mahasiswa)	98	100	Ya
Surat Plot Pembimbing (Mahasiswa)	91	100	Ya
Ajukan Surat Plot Pembimbing (Mahasiswa)	98	100	Ya
Detail Surat Plot Pembimbing (Mahasiswa)	94	92	Ya
Profile (Mahasiswa)	98	100	Ya
Dashboard (Verifikator)	98	100	Ya
Surat Plot Pembimbing (Verifikator)	96	100	Ya
Detail Surat Plot Pembimbing (Verifikator)	93	92	Ya
Profile (Verifikator)	98	100	Ya
Dashboard (Perusahaan)	96	100	Ya
Lowongan Magang (Perusahaan)	90	100	Ya
Tambah Lowongan (Perusahaan)	88	100	Ya
Detail Lowongan (Perusahaan)	98	100	Ya
Ubah Lowongan (Perusahaan)	88	100	Ya
Detail Pelamar (Perusahaan)	92	92	Ya
Profile (Perusahaan)	100	100	Ya



Gambar 40. Hasil Lighthouse Halaman Homepage

Pada hasil lighthouse testing halaman homepage terlihat bahwa skor performa hanya mendapatkan 64 poin yang masuk kedalam kategori *needs improvement*. Hal ini disebabkan pada banyaknya konten pendukung seperti aset CSS dan JavaScript yang dibutuhkan saat pertama kali memuat website. Namun setelah aset dimuat, data akan disimpan pada cache sehingga dapat digunakan kembali tanpa mengunduh ulang dan skor performa akan kembali naik. Hasil pengetesan juga memperlihatkan bahwa PWA sudah berjalan dengan baik dan mendapat centang pengetesan PWA dari lighthouse.

2.5. Maintenance

Dari hasil 349 tes yang dilakukan pada tahap blackbox testing, 5 diantaranya tidak memenuhi hasil yang diharapkan dan dinyatakan gagal. Pada tahap maintenance ini akan dilakukan analisa permasalahan dan perbaikan agar fitur dapat berjalan sesuai hasil yang diharapkan. Fitur yang akan diperbaiki adalah :

Tabel 10. Fitur Yang Butuh Perbaikan

Kode	Deskripsi	Hasil yang diharapkan	Hasil pengujian Black Box
BG07	Kolom input data perusahaan baru pada modal perusahaan	Dapat memasukkan data perusahaan baru	Data perusahaan baru berhasil dimasukkan. Namun data tidak dapat disimpan karena terdapat error
CD04	Tombol Hapus avatar	Dapat menampilkan pesan peringatan dan konfirmasi hapus avatar	Menampilkan pesan peringatan dan konfirmasi hapus avatar. Setelah menekan Ya, muncul peringatan error dan avatar tidak terhapus
DA07	Ikon Hapus pada tabel	Dapat menampilkan pesan peringatan dan konfirmasi hapus data	Menampilkan pesan peringatan dan konfirmasi hapus lowongan. Setelah menekan Ya, muncul peringatan error dan lowongan tidak terhapus
DB07	Ikon Hapus pada tabel	Dapat menampilkan pesan peringatan	Menampilkan pesan peringatan dan konfirmasi

Kode	Deskripsi	Hasil yang diharapkan	Hasil pengujian Black Box
		dan konfirmasi hapus data	hapus lowongan. Setelah menekan Ya, muncul peringatan error dan lowongan tidak terhapus
DG04	Tombol Hapus avatar	Dapat menampilkan pesan peringatan dan konfirmasi hapus avatar	Menampilkan pesan peringatan dan konfirmasi hapus avatar. Setelah menekan Ya, muncul peringatan error dan avatar tidak terhapus

Berikut adalah tabel hasil analisa, perbaikan, dan pengetesan blackbox ulang dari fitur – fitur yang diperbaiki :

Tabel 11. Hasil Perbaikan Fitur

Kode	Deskripsi	Permasalahan	Perbaikan	Status pengetesan ulang
BG07	Kolom input data perusahaan baru pada modal perusahaan	Kolom tersembunyi "id_perusahaan" memiliki atribut wajib diisi sehingga data tidak dapat disimpan saat menambahkan perusahaan baru	Menghapus atribut wajib diisi pada kolom tersembunyi "id_perusahaan"	Sukses
CD04	Tombol Hapus avatar	Link pada tombol mengarah ke "/admin" membuat gagal menghapus avatar verifikasi	Mengubah link tombol menjadi "/verifikator"	Sukses
DA07	Ikon Hapus pada tabel	Controller hapus belum ditambahkan di server	Menambahkan controller hapus	Sukses
DB07	Ikon Hapus pada tabel	Controller hapus belum ditambahkan di server	Menambahkan controller hapus	Sukses
DG04	Tombol Hapus avatar	Link pada tombol mengarah ke "/mahasiswa" membuat gagal menghapus avatar perusahaan	Mengubah link tombol menjadi "/perusahaan"	Sukses

3. PEMBAHASAN

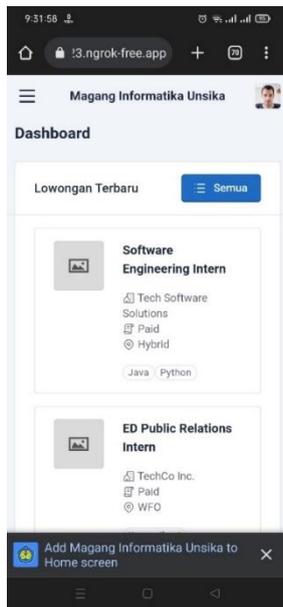
Berdasarkan pengujian yang dilakukan menggunakan black box testing dan lighthouse testing, *website* magang dan implementasi PWA secara keseluruhan dapat berjalan dengan baik. Beberapa masalah yang ditemukan saat pengujian black box sudah diperbaiki pada tahap maintance.

3.1. Implementasi Progressive Web App

Progressive Web App di implementasikan pada *website* magang dengan bantuan Workbox, kumpulan *libraries* JavaScript untuk PWA. Dengan memanfaatkan teknologi *service worker*, *website* magang mendapatkan tambahan fitur seperti : dapat di pasang pada perangkat *mobile* dan desktop, kemampuan menyimpan *cache* (*caching*) menggunakan konsep *precaching* dan *runtime caching*, serta dapat digunakan saat perangkat *offline*.

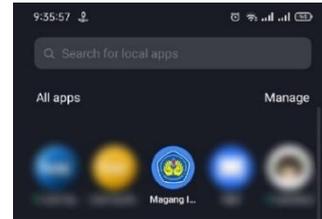
3.1.1. Fitur Install

Implementasi PWA membuat *website* magang dapat di pasang pada perangkat *mobile* dan desktop. Dengan memasang pada perangkat, *website* magang akan berjalan sebagai aplikasi sendiri terpisah dari web browser. Sistem juga akan memperlakukan *website* magang layaknya aplikasi pada umumnya yaitu berjalan seperti aplikasi sendiri, memiliki ikon aplikasi, dapat dicari melalui pencarian sistem, dan juga muncul pada halaman *recent apps*.

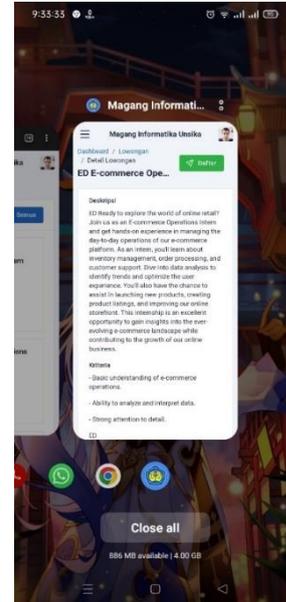


Gambar 41. Tampilan Pesan Untuk Memasang Website Magang

Untuk memasang *website* magang pada perangkat, pengguna cukup membuka *website* magang menggunakan web browser yang mendukung PWA lalu akan muncul pesan untuk menambahkan *website* magang ke *home screen* atau dapat menekan ikon titik tiga dan menekan tambahkan ke *home screen*. Jika berhasil, *website* magang akan muncul pada daftar aplikasi dan siap untuk dijalankan.



Gambar 42. Tampilan Ikon Website Magang Pada Daftar Aplikasi



Gambar 43. Tampilan Website Magang Berjalan Pada Halaman Recent Apps

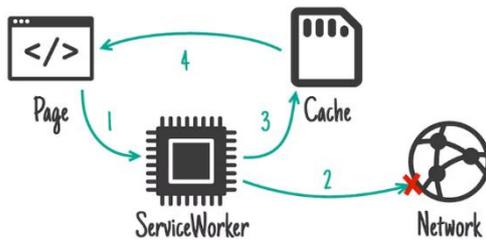
3.1.2. Precaching dan Runtime Caching

Precaching dan *runtime caching* merupakan dua konsep menyimpan *cache* (*caching*) yang dimiliki *service worker*. *Precaching* memproses penyimpanan *cache* dengan melakukan *caching* sebelum data tersebut dibutuhkan, biasanya saat pemasangan *service worker*. Sedangkan *runtime caching* dilakukan sembari berjalannya *website*. *Website* magang menggunakan kedua konsep ini untuk melakukan *caching*.

Precaching digunakan untuk menyimpan beragam aset yang hampir selalu digunakan seperti CSS untuk tampilan *website*, *library* JavaScript yang sering digunakan, halaman yang akan ditampilkan saat perangkat *offline*, dan lain-lain.

Runtime caching digunakan untuk menyimpan data halaman yang sudah dikunjungi agar tetap dapat dipakai saat perangkat *offline*, serta data untuk tabel dari server.

Selain konsep yang digunakan untuk *caching*, terdapat juga strategi yang digunakan untuk *caching*. *Website* magang menggunakan beberapa strategi berbeda dalam *caching* yaitu : strategi “*Network first*” digunakan untuk menyimpan data halaman dan tabel data, strategi “*Stale while revalidate*” digunakan untuk menyimpan data gambar, dan strategi “*Cache First*” digunakan untuk menyimpan dokumen CSS dan JavaScript.

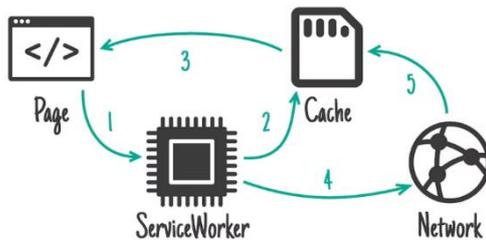


Gambar 44. Strategi Network First

(Sumber :

<https://developer.chrome.com/docs/workbox/caching-strategies-overview/>)

Strategi *caching network first* merupakan strategi dimana permintaan halaman akan diarahkan menggunakan jaringan, lalu data respons ditampilkan dan disimpan ke dalam *cache* dan jika permintaan ke jaringan gagal, data dari *cache* akan ditampilkan. Strategi ini cocok digunakan pada halaman dan tabel data *website* magang karena selalu menampilkan data terbaru saat jaringan tersedia, dan akan menampilkan data terakhir dari jaringan yang telah disimpan oleh *cache* saat perangkat tidak memiliki akses jaringan.

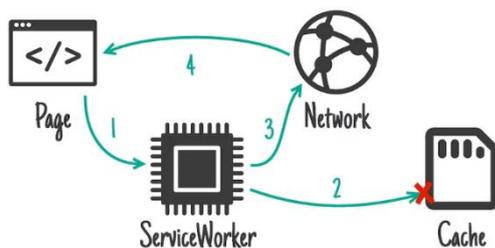


Gambar 45. Strategi Stale While Revalidate

(Sumber :

<https://developer.chrome.com/docs/workbox/caching-strategies-overview/>)

Strategi *caching stale while revalidate* merupakan strategi dimana permintaan akan ditampilkan menggunakan data dalam *cache* terlebih dahulu, sembari melakukan permintaan ke jaringan untuk memperbarui data dalam *cache*. Jika data tidak tersedia dalam *cache*, maka permintaan akan menunggu respons dari jaringan sebelum ditampilkan. Strategi ini cocok digunakan pada data gambar seperti foto profil dari pengguna *website* magang karena data hanya akan berubah saat pengguna mengubah foto profil sehingga data versi terbaru tidak sepenuhnya diperlukan pada setiap permintaan.



Gambar 46. Strategi Cache First

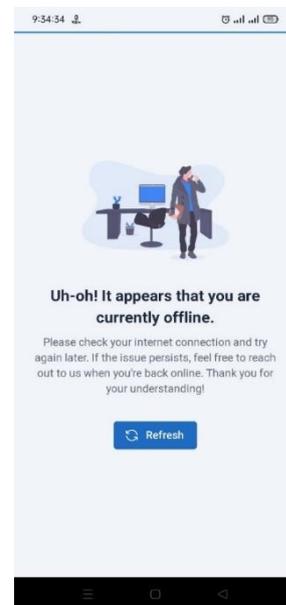
(Sumber :

<https://developer.chrome.com/docs/workbox/caching-strategies-overview/>)

Strategi *caching cache first* merupakan strategi dimana permintaan akan ditampilkan menggunakan data dalam *cache*, dan jika data tidak tersedia dalam *cache* maka akan dilakukan permintaan data ke jaringan. Lalu respons dari jaringan akan ditambahkan ke dalam *cache* dan data akan ditampilkan. Strategi ini cocok digunakan pada aset-aset statis *website* magang seperti dokumen CSS dan JavaScript karena data tidak akan berubah sehingga menampilkan data dari *cache* akan mempercepat waktu memuat halaman.

3.1.3. Pemakaian Saat Offline

Dengan kombinasi *service worker* dan kemampuan *caching* membuat *website* magang dapat digunakan walaupun perangkat sedang *offline*. Untuk dapat menggunakan *website* magang secara *offline*, pengguna hanya perlu menggunakan *website* magang dengan normal dan *website* magang secara otomatis akan menyimpan halaman yang telah ditampilkan serta aset-aset yang diperlukan ke penyimpanan *cache* perangkat untuk dapat digunakan mengurangi waktu memuat halaman dan juga saat perangkat sedang *offline*. Jika pengguna mengunjungi halaman yang tidak tersimpan dalam *cache*, maka halaman khusus akan ditampilkan untuk menginformasikan bahwa perangkat sedang *offline*. Pemakaian *offline* ini berlaku untuk penggunaan pada web browser maupun penggunaan melalui aplikasi PWA.



Gambar 47. Tampilan Halaman Offline Website Magang

4. KESIMPULAN

Berdasarkan hasil penelitian penerapan *Progressive Web App* pada pembuatan *website* magang dapat disimpulkan :

1. Penelitian ini berhasil menerapkan *Progressive Web App* pada pembuatan *website* magang. Implementasi PWA yang diterapkan membuat *website* magang dapat di install pada perangkat *mobile* dan desktop, dan juga dapat tetap digunakan walaupun perangkat sedang *offline*.
2. Pengujian *website* magang yang dilakukan dengan black box testing dan lighthouse testing menunjukkan bahwa *website* magang dapat berjalan dengan baik dan juga memiliki kompatibilitas PWA yang baik. Dari hasil ini dapat disimpulkan bahwa *website* magang telah memenuhi tujuan dari penelitian ini.

Saran pengembangan selanjutnya untuk *website* ini adalah dengan memperluas fitur *website* agar dapat digunakan dalam pencatatan kegiatan selama magang berlangsung sampai pengumpulan laporan saat kegiatan magang selesai.

PUSTAKA

- Adi, L., Akbar, R. J., Khotimah, W. N. (2017). Platform E-Learning Untuk Pembelajaran Pemrograman Web Menggunakan Konsep Progressive Web Apps, *Jurnal Teknik ITS*, 6(2).
- Aripin, S., Somantri, (2021). Implementasi Progressive Web Apps (PWA) pada Repository E-Portofolio Mahasiswa, *Jurnal Eksplorasi Informatika*, 10(2), 148-158.
- Faizah, Y. L. N., Aji, R. A., Hartiyah, S., (2023). Aplikasi Akuntansi Berbasis *Website* Menggunakan Bahasa Pemrograman Python, *Jurnal Akuntansi, Manajemen & Perbankan Syariah*, 3(1), 71-81.
- Hidayat, T., Muttaqin, M., (2020). Pengujian Sistem Informasi Pendaftaran dan Pembayaran Wisuda Online menggunakan Black Box Testing dengan Metode Equivalence Partitioning dan Boundary Value Analysis, *Jurnal Teknik Informatika UNIS*, 6(1), 25-29.
- Kemendikbud, (2022). *Latar Belakang Kampus Merdeka*. Diakses pada tanggal 13 Maret 2023. <https://kampusmerdeka.kemdikbud.go.id/web/about/latar-belakang>.
- Maximiliano, F., Rachel, A., Adriana, J., Pete, L., Sam, R., Joe, M., (n.d). *Progressive Web Apps*. Diakses pada tanggal 15 Mei 2023. <https://web.dev/learn/pwa/progressive-web-apps/>.
- Mukhtar, Asmasari, N., (2021). Perancangan dan implementasi sistem informasi tracer study amik mahaputra riau berbasis web, *Jurnal Intra Tech*, 5(2), 24-35.
- Romadhon, M. H., Yudhistira, Y., Mukrodin (2021). Sistem Informasi Rental Mobil Berbasis Android Dan *Website* Menggunakan Framework Codeigniter 3 Studi Kasus : CV Kopja Mandiri, *Jurnal Sistem Informasi dan Teknologi Peradaban (JSITP)*, 2(1), 30-36.
- Suendri, (2018). Implementasi Diagram UML (Unified Modelling Language) Pada Perancangan Sistem Informasi Remunerasi Dosen Dengan Database Oracle (Studi Kasus: UIN Sumatera Utara Medan), *Jurnal Ilmu Komputer dan Informatika*, 3(1).
- Suryadin, I. T., (2022). Perancangan Sistem Pemesanan Menu Cafe Pada Ruang Ngopi Berbasis Android, *Jurnal Ekonomi Dan Teknik Informatika*, 10(1), 74-83.
- Wahid, A. A. (2020). Analisis Metode Waterfall Untuk Pengembangan Sistem Informasi, *Jurnal Ilmu-ilmu Informatika dan Manajemen STMIK*, 2-3. https://www.researchgate.net/publication/346397070_Analisis_Metode_Waterfall_Untuk_Pengembangan_Sistem_Informasi.
- Wasino, Herwindiati, D. E., Setyawan, I. R., Maupa, H., (2023). Desain situs web yang responsif berdasarkan strategi agile sebagai pendukung pemasaran destinasi wisata, *Jurnal Teknik Informatika dan Sistem Informasi*, 10(1), 526-540.
- Wisnumurti, W., Faulina, S. T., Novari, S. (2023). Pelatihan optimalisasi Microsoft Office untuk meningkatkan kegiatan mahasiswa mahasiswi pada pengabdian masyarakat di universitas mahakarya Asia Baturaja. *BERNAS: Jurnal Pengabdian Kepada Masyarakat*, 4(1), 410-415.