

# SISTEM INFORMASI PENGELOLAAN DANA AMAL POLIBATAM BERBASIS WEB

*Web-Based Polibatam Charity Fund Management Information System*

Budi Prayoga<sup>1</sup>, Yeni Rokhayati<sup>2</sup>

\*Teknik Informatika, Politeknik Negeri Batam

[budiprayoga5103@gmail.com](mailto:budiprayoga5103@gmail.com)<sup>1</sup>, [yeni@polibatam.ac.id](mailto:yeni@polibatam.ac.id)<sup>2</sup>

## Abstrak

Dana Amal Polibatam (DAP) adalah sebuah program amal dari dan untuk Politeknik Negeri Batam (Polibatam), dimana setiap bulannya ada iuran tetap dari karyawan, lalu dana yang terkumpul digunakan untuk kegiatan sosial di lingkup keluarga karyawan juga. Proses pengajuan yang dilakukan melalui email lalu direkap menggunakan Ms.Excel menyebabkan prosesnya lama dan memerlukan tenaga serta waktu yang tidak sedikit serta rentan terjadi kesalahan dalam perekapan datanya dan hal tersebut juga menyulitkan dari segi keterbukaan saldo dana amal serta pemakaiannya kepada karyawan. Penelitian ini mengusulkan sebuah sistem informasi berbasis website yang bisa memproses pengajuan DAP, perekapan, dan memfasilitasi keterbukaan *update* dananya. Dengan menggunakan metode pengembangan *Waterfall* yang merupakan salah satu pendekatan dalam pengembangan perangkat lunak yang mengikuti serangkaian tahapan yang dilakukan secara berurutan dan selesai satu per satu sebelum melanjutkan ke tahap berikutnya. Adapun teknologi yang digunakan antara lain MySQL sebagai *database*, JavaScript sebagai Bahasa pemrograman dengan *ReactJs* sebagai *library* untuk Front-end dan *ExpressJs* sebagai *Framework* untuk Back-end. Hasil dari sistem ini dapat memfasilitasi proses pengajuan DAP, mempermudah rekapitulasi, dan meningkatkan transparansi pemakaian saldo dana amal kepada karyawan.

Kata kunci: Sistem Informasi Dana Amal, JavaScript; Polibatam; Waterfall

## Abstract

*The Polibatam Charity Fund (DAP) is a charity program from and for the Batam State Polytechnic (Polibatam), where every month there is a fixed contribution from employees, then the funds collected are used for social activities within the employee's family as well. The application process, which is carried out via email and then recapitulated using Ms. Excel, causes the process to take a long time and requires a lot of energy and time and is prone to errors in recording the data and this also makes it difficult in terms of disclosing the balance of charity funds and their use to employees. This research proposes a website-based information system that can process DAP applications, record them, and facilitate open funding updates. By using the Waterfall development method, which is an approach to software development that follows a series of stages which are carried out sequentially and completed one by one before moving on to the next stage. The technology used includes MySQL as a database, JavaScript as a programming language with ReactJs as a library for the front-end and ExpressJs as a framework for the back-end. The results of this system can facilitate the DAP application process, simplify recapitulation, and increase transparency in the use of charity fund balances for employees.*

Keywords: Charity Fund Information System, JavaScript; Polibatam; Waterfall

## 1. PENDAHULUAN

Politeknik Negeri Batam (Polibatam) adalah salah satu perguruan tinggi di Indonesia yang terletak di Kota Batam, Kepulauan Riau. Polibatam dikenal sebagai lembaga pendidikan yang memberikan beragam program studi di berbagai bidang, termasuk teknik, bisnis, dan lainnya. Dalam lingkungan kampus seperti Polibatam, staf dan karyawan adalah aset berharga yang mendukung berbagai aspek operasional perguruan tinggi. Dalam upaya untuk mendukung lingkungan pendidikan yang berkelanjutan, Polibatam telah menginisiasi program Dana Amal Polibatam (DAP). Latar belakang munculnya program Dana Amal Polibatam didasarkan pada pengamatan dan perhatian terhadap kondisi karyawan di Polibatam. Program ini bertujuan untuk menggalang dana amal dari karyawan yang kemudian juga akan digunakan untuk kegiatan sosial di lingkup keluarga karyawan Polibatam.

Akan tetapi program Dana Amal Polibatam memiliki sejumlah permasalahan yang signifikan yang perlu diatasi. Seperti proses pengajuan dalam program DAP saat ini memakan waktu yang sangat lama. Pengajuan dana dan alokasi sumber daya memerlukan langkah-langkah manual yang kompleks dan terkadang berbelit-belit. Hal ini mengakibatkan penundaan dalam pengambilan keputusan dan pelaksanaan kegiatan sosial. Dan rekapitulasi data terkait DAP juga dilakukan secara manual, yang tidak hanya memakan waktu lama, tetapi juga meningkatkan risiko kesalahan pengolahan data. Proses manual ini berpotensi mengurangi akurasi dan keteraturan dalam pelaporan atau pengelolaan dana amal. Yang terakhir saat ini, tidak ada keterbukaan yang memadai dalam hal saldo dan penggunaan dana amal. Karyawan dan pihak-pihak yang berkontribusi mungkin merasa tidak yakin tentang bagaimana dan di mana dana amal digunakan, yang dapat mengurangi kepercayaan dan motivasi untuk berpartisipasi aktif dalam program ini.

Salah satu penelitian terkait yang dilakukan oleh [1] dalam judul “Perancangan Sistem Informasi Manajemen Badan Amal Menggunakan Metode Waterfall”. Dalam penelitian tersebut disebutkan bahwa kurang berminatnya Masyarakat untuk melakukan donasi yang diiklankan lewat brosur dan sering terjadi permasalahan dalam manajemen data pemberian dana, sehingga pengelolaan dana kurang terorganisir dan pembuatan laporan memakan waktu yang lama. Oleh karena itu dibuatlah sistem informasi untuk manajemen data donasi untuk memberikan informasi yang cepat dan tepat kepada para *donator* serta dapat memanajemen data dana agar tersusun dengan baik.

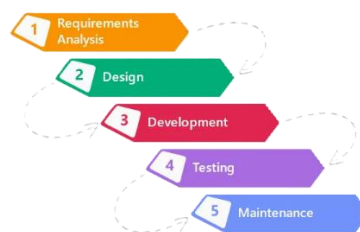
Untuk membuat sistem informasi ini menjadi kenyataan, peneliti memerlukan metode pengembangan perangkat lunak *Waterfall*. Kelebihan dari metode *Waterfall* ini adalah Metode ini memiliki struktur tahapan yang terorganisir, dengan setiap tahap yang harus diselesaikan sebelum melanjutkan ke tahap berikutnya. Hal ini membantu dalam mengelola proyek dengan lebih terstruktur dan memastikan bahwa setiap aspek telah diperhitungkan dengan baik. Karena setiap tahap harus diselesaikan sebelum melanjutkan ke tahap berikutnya, metode *Waterfall* cenderung menghasilkan perangkat lunak dengan kualitas yang tinggi dan konsistensi yang baik dikarenakan metode ini dikerjakan secara bertahap[2]. Adapun juga kelebihan dari menggunakan *library ReactJs* pada sisi *Front-end* adalah dengan pendekatan berbasis komponen, *ReactJS* memungkinkan pembangunan *user interface* yang modular dan dapat digunakan Kembali. Selain itu, penggunaan *Virtual DOM* oleh *ReactJS* membantu meningkatkan kinerja dengan melakukan pembaruan hanya pada bagian-bagian yang diperlukan dari *DOM*, menghindari pembaruan yang tidak perlu. *ReactJS* juga memisahkan logika *user interface* ke dalam komponen yang memudahkan pemeliharaan dan pengembangan[3]. Dan adapun juga kelebihan dari menggunakan *ExpressJs* pada sisi *Back-end* adalah karena keringanan, kehandalan, dan dukungan komunitas yang besar. Dengan sistem *middleware* yang kuat, struktur *routing* yang sederhana, *ExpressJS* memberikan solusi yang efisien dan terpercaya untuk pengembangan aplikasi web modern [4].

Berdasarkan dari permasalahan yang telah dijelaskan, dapat diidentifikasi beberapa masalah seperti, proses pengajuan dana menggunakan email setelah itu direkap menggunakan Ms.Excel menyebabkan proses pencairan dana memerlukan tenaga serta waktu yang tidak

sebentar dan juga rentan terjadinya kesalahan dalam perekapan datanya. Kurangnya juga keterbukaan saldo dana amal serta pemakaiannya kepada karyawan. Oleh karena itu dibutuhkan sebuah Sistem Informasi Dana Amal Polibatam Berbasis Web yang nantinya diharapkan dapat membantu permasalahan – permasalahan yang ada sebelumnya.

## 2. METODE PENELITIAN

Metode pengembangan produk yang digunakan dalam pembuatan sistem informasi ini adalah *Waterfall*. Metode *Waterfall* merupakan salah satu Metode *SDLC* yang sering digunakan dalam pengembangan sistem informasi atau perangkat lunak, Model ini menggunakan pendekatan sistematis dan berurutan. Kelebihan menggunakan metode *waterfall* dalam pengembangan sistem informasi ini adalah kualitas dari sistem yang dihasilkan akan baik karena pelaksanaannya dilakukan secara bertahap. Tahapan dalam metode ini terdiri dari *requirements*, *design*, *development*, *testing*, dan yang terakhir ada *maintenance* dan dilakukan secara bertahap[5] seperti terlihat pada Gambar 1.



Gambar 1. Metode Waterfall

Tahapan *Requirement* adalah langkah awal dari metode *Waterfall* di mana peneliti berusaha memahami kebutuhan dari pemangku kepentingan, termasuk kebutuhan fungsional dan non-fungsional, melalui komunikasi, wawancara, observasi, atau analisis dokumen, sehingga memastikan pemahaman menyeluruh tentang apa yang diinginkan dari perangkat lunak. Pada tahapan ini akan dilakukan wawancara kepada pengurus DAP untuk mengumpulkan informasi terkait apa saja kebutuhan website sistem dana amal polibatam, fitur apa yang akan dirancang serta mengetahui siapa saja pengguna yang akan terlibat. Luaran dari tahapan ini berupa deskripsi umum sistem, kebutuhan fungsional dan non-fungsional, serta diagram *use case* [6].

Pada tahap *design* bertujuan untuk memberikan gambaran lengkap tentang apa yang harus dikerjakan dan bagaimana tampilan dari sebuah sistem yang diinginkan, sehingga membantu peneliti dalam mendefinisikan arsitektur sistem yang akan dibuat secara keseluruhan. Pada tahapan ini peneliti melakukan proses *design* sistem meliputi *design interface* dan perancangan *database*, dengan membuat *mockup* menggunakan *software* pendukung Figma. Luaran dari tahapan ini berupa *design user interface* dan *Entity Relationship Diagram (ERD)* [7].

Pada tahap *development* ini merupakan proses pengcodingan dari *design* sistem yang telah dibuat pada tahapan sebelumnya. Pada tahapan ini melakukan pembuatan website menggunakan bahasa pemrograman *JavaScript* dengan menggunakan *Framework ReactJs* dan *ExpressJs* [8].

Tahapan *testing* merupakan tahapan pengujian di mana perangkat lunak yang telah dibuat diuji secara menyeluruh untuk memastikan bahwa itu memenuhi kebutuhan dan spesifikasi yang telah ditentukan sebelumnya. Tahapan ini merupakan tahap penting untuk mengidentifikasi dan memperbaiki bug atau masalah dalam perangkat lunak. Pemilihan cara pengujian menggunakan metode *Blackbox testing* yang digunakan pada Metode *waterfall* dianggap mampu mengurangi resiko terjadinya kesalahan dalam fungsionalitas [9].

Pada tahapan terakhir yaitu *maintenance*, perangkat lunak yang sudah jadi dan melewati tahapan *testing* akan masuk ketahapan *maintenance* ini. Pada tahapan ini akan dilakukan pemeliharaan sistem termasuk dalam memperbaiki kesalahan yang tidak ditemukan pada Langkah sebelumnya [2]. Setelah pengujian selesai dan berjalan sebagaimana mestinya, website akan diberikan kepada UPT TI polibatam dan unit tersebut yang akan melakukan tahapan *maintenance* ini selanjutnya.

### 3. HASIL DAN PEMBAHASAN

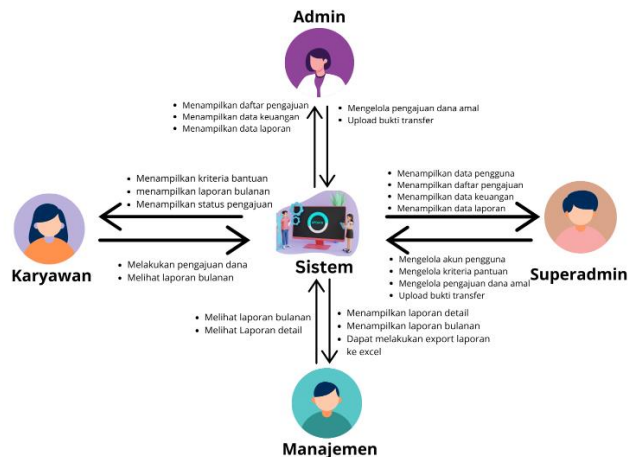
#### 3.1. Requirement Analysis

##### 3.1.1. Gambaran Umum Sistem

Sistem Informasi Dana Amal Polibatam ini dibuat dengan menggunakan arsitektur web yang dapat diakses melalui web browser dan perangkat mobile. Pada tahapan *requirement analysis* sudah dilakukan wawancara kepada *stakeholder* atau *client* adapun hasil wawancaranya berupa :

- Pengguna dapat melakukan *login* ke dalam website.
- Kemudahan untuk melakukan pengajuan pencairan dana sesuai dengan kriteria yang terdapat pada DAP.
- Setelah karyawan mengajukan klaim dana maka admin akan menerima email berisi berupa informasi seputar pengajuan dana yang diajukan.
- Karyawan dapat melihat informasi status pengajuan dana yang diajukannya.

Pada sistem ini memiliki 4 *roles* yaitu ada Karyawan, Admin, Superadmin, Manajemen. Untuk *roles* karyawan nantinya akan digunakan oleh seluruh karyawan Polibatam yg menjadi anggota DAP, Admin dan Superadmin nantinya digunakan oleh pengurus harian dari sistem informasi dana amal polibatam ini dan *roles* Manajemen nantinya akan digunakan oleh pudir 2 polibatam.



Gambar 2. Gambaran Umum Sistem

##### 3.1.2. Kebutuhan Fungsional

Kebutuhan Fungsional adalah kebutuhan yang berisi proses atau layanan yang nantinya akan disediakan oleh sistem[10]. Berikut adalah hasil analisis mengenai Kebutuhan Fungsional pada Sistem Informasi Dana Amal Polibatam:

Table 1. Kebutuhan fungsional

No	Kebutuhan Fungsional
F001	Pengguna dapat melakukan <i>login</i>
F002	Super admin dapat melakukan <i>input/update/delete</i> data pengguna
F003	Super admin dapat melakukan <i>block/unblock</i> data pengguna
F004	Pengguna dapat melakukan <i>update profile</i>
F005	Karyawan dapat melihat laporan bulanan
F006	Super admin, Admin, dan Manajemen bisa melihat laporan detail
F007	Karyawan dapat melakukan pengajuan pencairan DAP
F008	Karyawan dapat melihat setatus pengajuan seperti ditangguhkan, ditolak dan selesai
F009	Superadmin dan Admin dapat melakukan pengelolaan pengajuan DAP seperti membuat pengajuan ditolak, ditangguhkan atau selesai
F010	Super admin, Admin dapat melakukan <i>input/update/delete</i> jumlah dana DAP setiap bulan
F011	Sistem dapat membatasi jumlah pengajuan dana DAP jika sudah melebihi kriteria yang sudah ditentukan
F012	Superadmin dapat melakukan <i>input/update/delete</i> kriteria penerimaan dana DAP
F013	Sistem dapat memberikan notifikasi via email kepada SuperAdmin dan Admin tentang adanya pengajuan klaim dana DAP dari karyawan
F014	Karyawan dapat melakukan pengajuan ulang jika status dari pengajuannya ditangguhkan

### 3.1.3. Kebutuhan Non Fungsional

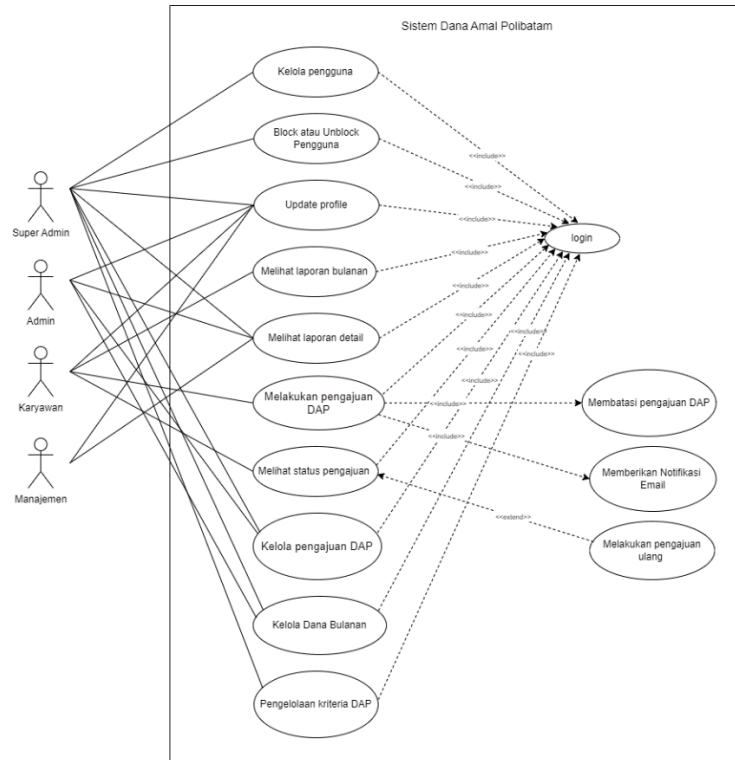
*Non Functional Requirement* adalah kebutuhan yang menitik beratkan pada *property* pendukung yang dimiliki oleh sistem [10].

Table 2. Kebutuhan fungsional

Kriteria	Parameter
Availability	<ul style="list-style-type: none"> <li>• Sistem dapat di akses apabila terhubung ke jaringan internet</li> <li>• Sistem mampu berjalan selama 24 jam non-stop, kecuali apabila ada perawatan sistem atau pembaharuan sistem</li> <li>• Sistem dapat dijalankan dimana saja dan kapan saja</li> </ul>
Ergonomy	Sistem harus dapat digunakan dengan mudah atau user friendly
Bahasa	Menggunakan Bahasa Indonesia (optional Bahasa Inggris)
Safety	Sistem dapat memastikan bahwa data yang digunakan harus terlindungi dari akses yang tidak berwenang

### 3.1.4. Diagram *Use Case*

*Use Case* adalah alat atau teknik yang digunakan dalam rekayasa perangkat lunak untuk mendefinisikan, menggambarkan, dan memodelkan interaksi antara sistem perangkat lunak dengan pengguna atau entitas eksternal lainnya. *Use case* digunakan untuk memahami dan mendokumentasikan bagaimana pengguna akan berinteraksi dengan sistem serta apa yang diharapkan dari sistem tersebut [11].

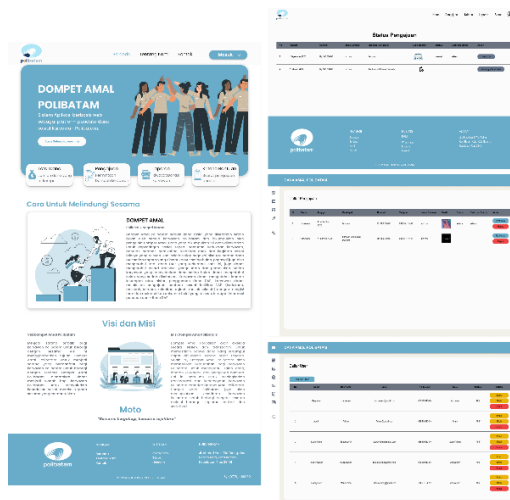


Gambar 3. Use case diagram

### 3.2. Design

#### 3.2.1. Design User Interface

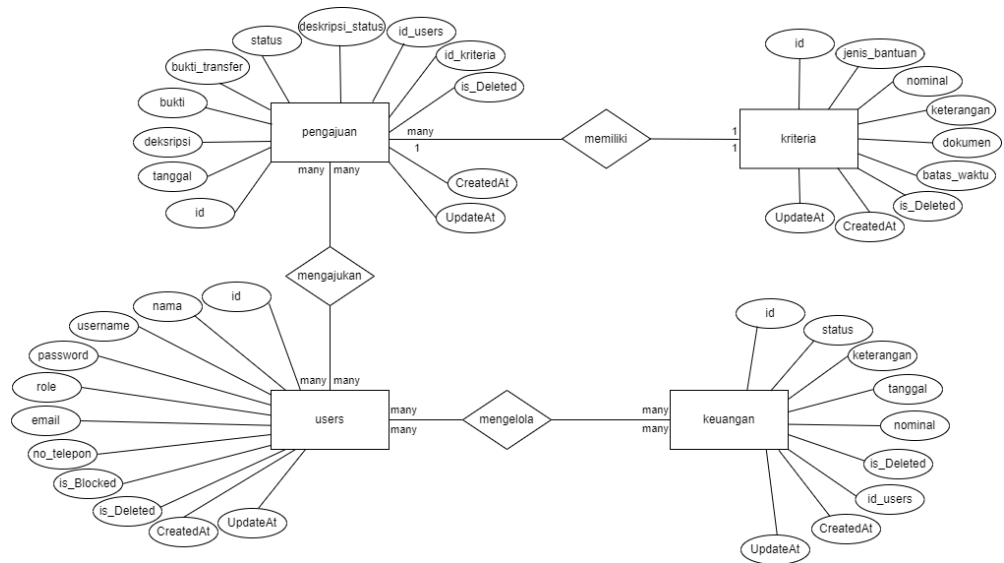
*Design User Interface (UI)* adalah proses merancang elemen-elemen visual dan interaktif dari suatu sistem atau produk, dengan tujuan utama untuk meningkatkan pengalaman pengguna[12]. *User Interface* sistem ini dapat dilihat lebih lengkap pada link figma berikut: <https://bit.ly/42UwosY>



Gambar 4. Design User Interface

#### 3.2.2. Entity Relationship Diagram

*Entity Relationship Diagram (ERD)* adalah alat visual yang digunakan dalam rekayasa perangkat lunak dan desain sistem *database*. *ERD* digunakan untuk menggambarkan dan memodelkan hubungan antara table pada *database* [13].



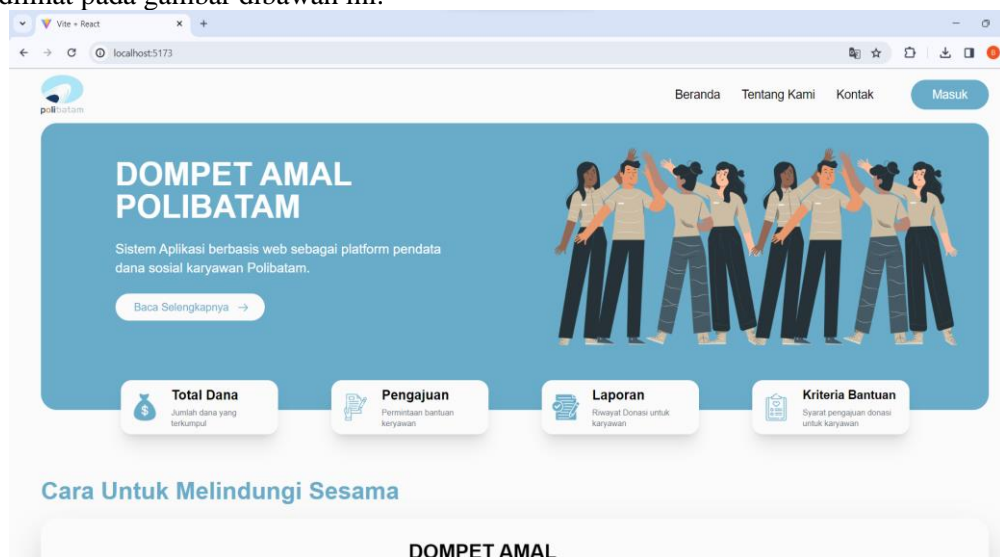
Gambar 5. Entity Relationship Diagram

### 3.3. Development

#### 3.3.1. Implementasi Antar Muka

##### 1. Implementasi halaman *landing page*

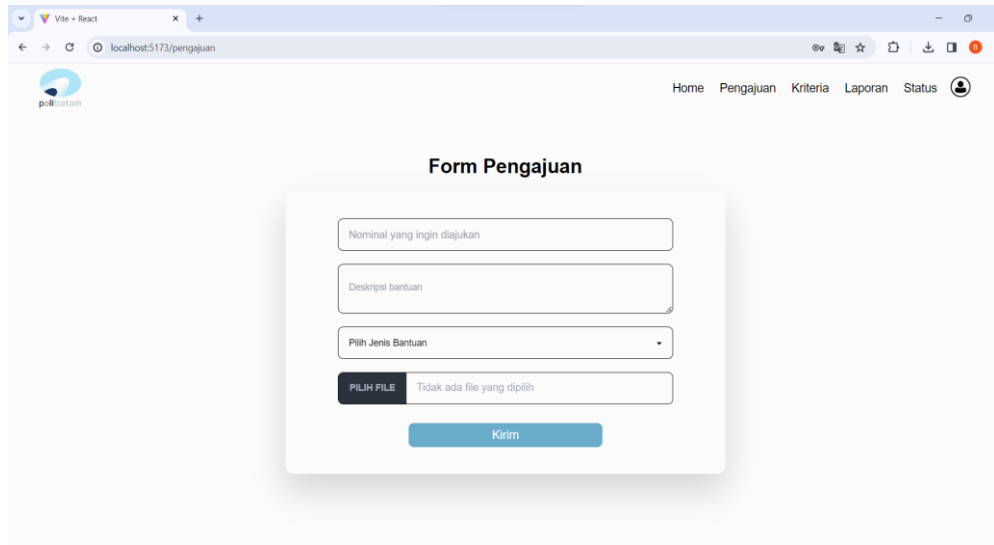
Halaman *landing page* adalah halaman yang dapat di akses secara bebas oleh semua orang tanpa harus *login*. Pada halaman ini menampilkan seputar informasi umum tentang dana amal polibatam Implementasi halaman *landing page* ini dapat dilihat pada gambar dibawah ini.



Gambar 6. Halaman landing page

##### 2. Implementasi halaman pengajuan

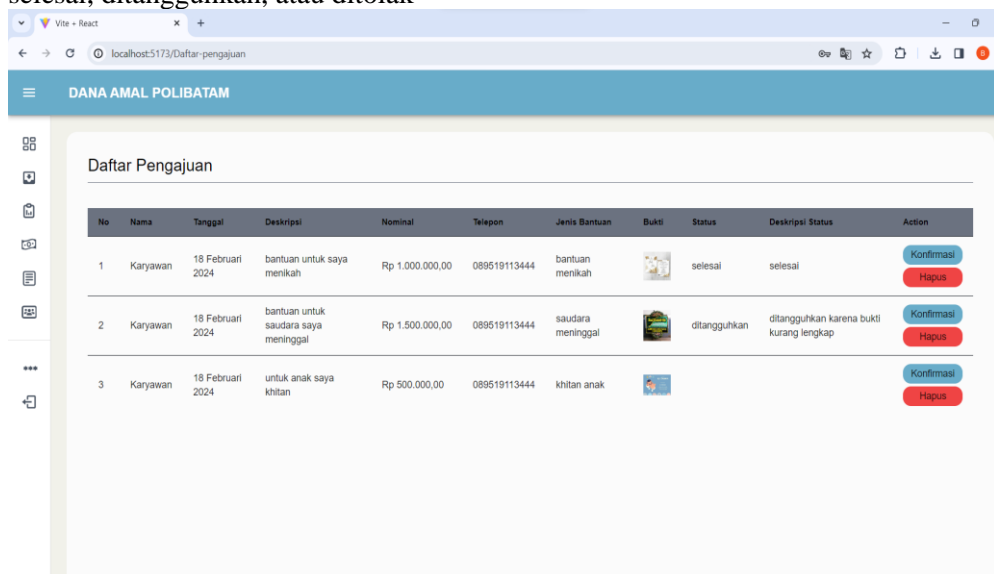
Halaman pengajuan ini hanya dapat diakses ketika karyawan sudah melakukan *login*. Pada halaman ini karyawan dapat mengisi *form* dan mengklik tombol kirim untuk melakukan pengajuan pencairan dana amal.



Gambar 7. Halaman Pengajuan

3. Implementasi halaman daftar pengajuan

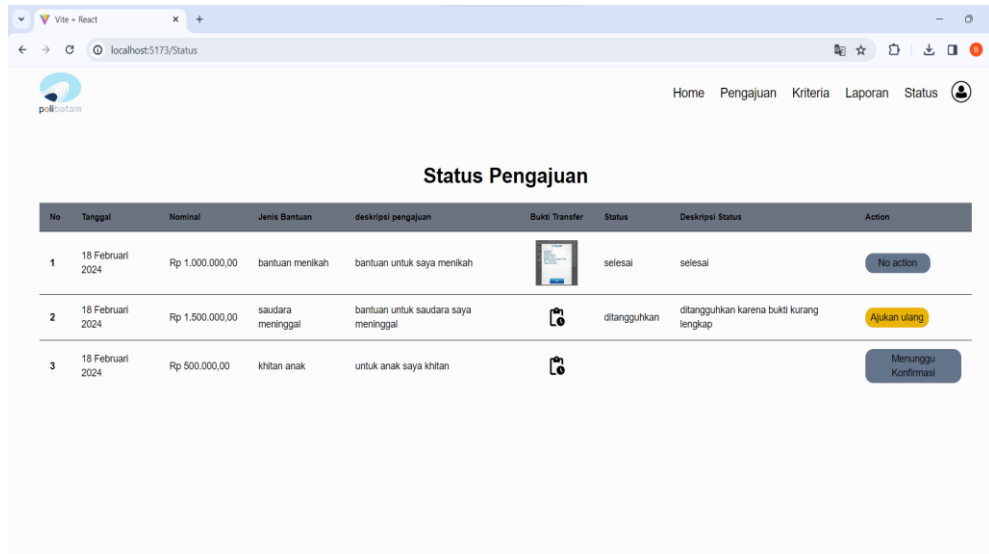
Halaman daftar pengajuan ini adalah fitur yang hanya dapat digunakan oleh Superadmin dan Admin setelah *login*, pada halaman ini Admin dapat melakukan verifikasi pengajuan dari karyawan dan membuat status dari pengajuan tersebut selesai, ditangguhkan, atau ditolak



Gambar 8. Halaman daftar pengajuan

4. Implementasi halaman status pengajuan

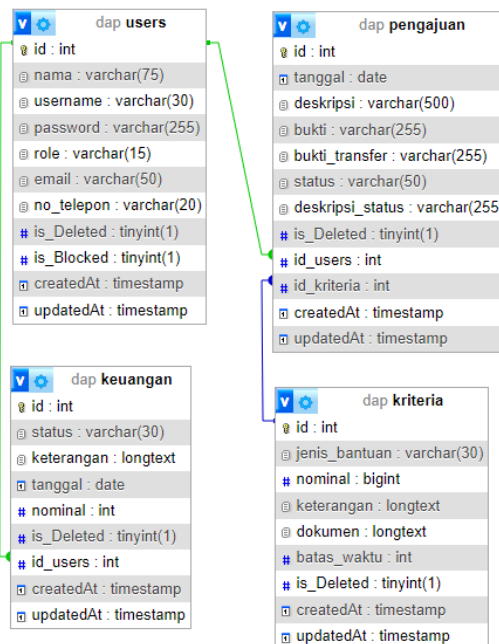
Halaman status pengajuan ini berfungsi untuk karyawan melihat semua status dari pengajuan pencairan dana amal nya, pada halaman ini karyawan dapat melihat status dari pengajuannya seperti selesai, ditangguhkan, ditolak, atau menunggu konfirmasi



Gambar 9. Halaman status pengajuan

### 3.3.2. Implementasi Basis Data

Pada tahap implementasi basis data ini dapat dilihat tipe dan panjang data yang digunakan dan terdapat juga relasi antar table.



Gambar 10. Implementasi Basis Data

### 3.4. Testing

Tujuan dari testing pada sistem ini adalah untuk memastikan bahwa sistem yang telah dibangun dapat beroperasi dengan baik sesuai dengan kebutuhan fungsionalnya. Proses pengujian sistem ini menggunakan metode *black box*, yang berfokus pada penilaian kualitas perangkat lunak dengan mengevaluasi fungsionalitasnya untuk mengidentifikasi ketidaksesuaian fungsi, kesalahan antarmuka, dan potensi kesalahan lainnya [14]. Berikut hasil dari *testing* pada sistem ini :

Table 3. Testing

No	Fungsional (Role)	Detail Fungsional	Skenario Pengujian	Indikator Keberhasilan	Hasil Pengujian
1	Login (Karyawan, Admin, Super Admin, Manajemen)	Pengguna dapat melakukan login	Pengguna memasukkan <i>username</i> benar dan <i>password</i> benar	Sistem memberikan pesan Selamat datang dan Pengguna berhasil login	Sesuai
			Pengguna memasukkan <i>username</i> benar dan <i>password</i> salah	Sistem memberikan pesan Password salah	Sesuai
			Pengguna memasukkan <i>username</i> salah dan <i>password</i> benar	Sistem memberikan pesan Anda tidak terdaftar sebagai anggota DAP	Sesuai
			Pengguna memasukkan <i>username</i> salah dan <i>password</i> salah	Sistem memberikan pesan Anda tidak terdaftar sebagai anggota DAP	Sesuai
2	Kelola Pengguna (Super Admin)	Input dan Update data pengguna	Superadmin memasukkan data user baru yang ingin dibuat	Sistem akan menampilkan pesan User berhasil dibuat	Sesuai
			Superadmin memasukkan data user dengan <i>username</i> yang sudah digunakan dan email baru	Sistem akan menampilkan pesan <i>Username</i> sudah digunakan	Sesuai
			Superadmin memasukkan data <i>user</i> dengan <i>username</i> baru dan <i>email</i> yang sudah digunakan	Sistem akan menampilkan pesan Email sudah digunakan	Sesuai
			Superadmin memasukkan data <i>user</i> dengan <i>username</i> dan <i>email</i> yang sudah digunakan	Sistem akan menampilkan pesan <i>Username</i> dan <i>email</i> sudah digunakan	Sesuai
		Delete data pengguna	Menekan tombol hapus pada user yang ingin dihapus	Sistem akan menampilkan pesan Akun berhasil dihapus	Sesuai

3	Block atau Unblock data pengguna (Super Admin)	Block data pengguna	Menekan tombol <i>Block</i> pada pengguna yang ingin diblock	Sistem akan menampilkan pesan Akun berhasil diblock	Sesuai
		Unblock data pengguna	Menekan tombol <i>Unblock</i> pada pengguna yang ingin dibuka blocknya	Sistem akan menampilkan pesan Akun berhasil di unblock	Sesuai
4	Update profile (Admin, Karyawan, Manajemen,)	Pengguna melakukan <i>update profile</i>	Pengguna memasukan data baru	Sistem akan menampilkan pesan Akun berhasil diupdate	Sesuai
			Pengguna memasukan <i>username</i> yang sudah digunakan oleh <i>user</i> lain	Sistem akan menampilkan pesan <i>Username</i> sudah digunakan	Sesuai
			Pengguna memasukan email yang sudah digunakan oleh <i>user</i> lain	Sistem akan menampilkan pesan Email sudah digunakan	Sesuai
5	Melihat laporan bulanan (Karyawan)	Karyawan dapat melihat laporan bulanan	Karyawan mengklik tombol Laporan pada <i>navbar</i>	Sistem akan menampilkan seluruh data laporan bulanan	Sesuai
6	Melihat laporan detail (Super admin, Admin, Manajemen)	Pengguna dapat melihat laporan detail	Pengguna mengklik tombol Laporan pada <i>Sidebar</i>	Sistem akan menampilkan seluruh data laporan detail	Sesuai
7	Melakukan pengajuan DAP (Karyawan)	Karyawan dapat melakukan pengajuan pencairan DAP	Memasukan seluruh data di form pengajuan	Sistem akan menampilkan pesan Pengajuan berhasil dikirim	Sesuai
8	Melihat status pengajuan (Karyawan)	Karyawan dapat melihat setatus pengajuan seperti, Ditungguhkan, ditolak dan selesai	Karyawan belum pernah melakukan pengajuan	Sistem tidak menampilkan data pengajuan	Sesuai
			Karyawan sudah pernah melakukan pengajuan	Sistem dapat menampilkan seluruh pengajuan yang sudah pernah diajukan oleh karyawan tersebut beserta statusnya	Sesuai

9	Kelola pengajuan DAP (Super admin, Admin)	Menyelesaikan pengajuan yang diajukan oleh karyawan	Membuat status menjadi selesai, mengisi deskripsi status dan mengupload bukti Transfer	Sistem menampilkan pesan berhasil dikonfirmasi	Sesuai
		Menangguhkan pengajuan yang diajukan oleh karyawan	Membuat status menjadi ditangguhkan dan mengisi deskripsi statusnya	Sistem menampilkan pesan berhasil dikonfirmasi	Sesuai
		Menolak pengajuan yang diajukan oleh karyawan	Membuat status menjadi ditolak dan mengisi deskripsi statusnya	Sistem menampilkan pesan berhasil dikonfirmasi	Sesuai
10	Kelola Dana bulanan (Super Admin, Admin)	<i>Input</i> data pemasukan bulanan	Memasukan data pemasukan bulanan	Sistem menampilkan pesan pemasukan berhasil ditambahkan	Sesuai
		<i>Update</i> data bulanan	Mengubah data pemasukan bulanan	Sistem menampilkan pesan pemasukan berhasil diupdate	Sesuai
		<i>Delete</i> data pemasukan bulanan	Mengklik tombol Hapus pada data yang ingin dihapus	Sistem menampilkan pesan Keuangan berhasil dihapus	sesuai
11	Membatasi pengajuan DAP (Karyawan)	Sistem dapat membatasi jumlah pengajuan dana DAP jika sudah melebihi kriteria yang sudah ditentukan	Karyawan memilih jenis bantuan yang sudah pernah diajukan sebelumnya dan masih dalam rentan batas waktu yang ditentukan	Sistem akan menampilkan pesan pengajuan ditolak karena masih dalam <i>cooldown</i>	sesuai
12	Pengelolaan kriteria DAP (Super Admin)	<i>Input</i> data kriteria penerima DAP	Mengisi seluruh data yang dibutuhkan	Sistem akan menampilkan pesan kriteria berhasil dibuat	sesuai
		<i>Update</i> data kriteria penerima DAP	Mengubah data yang ingin diubah pada kriteria DAP	Sistem akan menampilkan pesan Kriteria berhasil diupdate	sesuai
		<i>Delete</i> data kriteria penerima DAP	Menghapus data yang ingin dihapus pada kriteria DAP	Sistem akan menampilkan pesan Kriteria berhasil dihapus	sesuai
13	Memberikan notifikasi Email (Karyawan)	Sistem dapat memberikan notifikasi email kepada	Pengajuan yang dilakukan karyawan berhasil	Admin dan Super admin akan menerima email berupa data	Sesuai

		role Admin dan Superadmin jika ada yang mengajukan dana DAP		pengajuan yang dilakukan oleh karyawan	
14	Melakukan pengajuan ulang (Karyawan)	Karyawan dapat melakukan pengajuan ulang jika status dari pengajuannya ditangguhkan	Menekan tombol ajukan ulang dan mengisi seluruh data yang diminta	Sistem menampilkan pesan pengajuan berhasil diajukan ulang	Sesuai

### 3.5. Maintenance

Pada tahapan ini dilakukan pemeliharaan sistem termasuk dalam memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya setelah pengujian selesai dan berjalan sebagaimana mestinya, website akan diberikan kepada UPT TI Polibatam dan unit tersebut yang akan melakukan tahapan *maintenance* ini selanjutnya.

## 4. KESIMPULAN DAN SARAN

Berdasarkan dari hasil perancangan dan implementasi telah dibangun Sistem Informasi Pengelolaan Dana Amal Polibatam Berbasis Web dengan Metode *Waterfall* dapat disimpulkan bahwa sistem ini memberikan kemudahan bagi pengguna DAP untuk melakukan pengajuan dana amal, pengelolaan data untuk pengurus dan juga keterbukaan total dana kepada seluruh pengguna. Berdasarkan hasil pengujian dengan metode *Black box testing* bahwa sistem ini sudah sesuai dan dapat digunakan dengan baik.

## DAFTAR PUSTAKA

- [1] A. S. Kirsan, N. N. Arisa, and A. H. D. Putra, "PERANCANGAN SISTEM INFORMASI MANAJEMEN BADAN AMAL MENGGUNAKAN METODE WATERFALL," *Antivirus : Jurnal Ilmiah Teknik Informatika*, vol. 16, no. 1, pp. 63–77, May 2022, doi: 10.35457/antivirus.v16i1.2110.
- [2] A. Abdul Wahid Sekolah Tinggi Manajemen Informatika dan Komputer Sumedang, "Analisis Metode Waterfall Untuk Pengembangan Sistem Informasi." [Online]. Available: <https://www.researchgate.net/publication/346397070>
- [3] P. Rawat and A. N. Mahajan, "ReactJS: A Modern Web Development Framework," 2020. [Online]. Available: [www.ijisrt.com](http://www.ijisrt.com)
- [4] N. Hussain, "Full Stack Hybrid Web Application for an Auto Repair Company."
- [5] S. Supiyandi, M. Zen, C. Rizal, and M. Eka, "Perancangan Sistem Informasi Desa Tomuan Holbung Menggunakan Metode Waterfall," *JURIKOM (Jurnal Riset Komputer)*, vol. 9, no. 2, p. 274, Apr. 2022, doi: 10.30865/jurikom.v9i2.3986.
- [6] V. Olindo and A. Syaripudin, "Perancangan Sistem Informasi Absensi Pegawai Berbasis Web Dengan Metode Waterfall (Studi Kasus : Kantor Dbpr Tangerang Selatan)," *OKTAL : Jurnal Ilmu Komputer dan Science*, vol. 1, no. 01, 2022.
- [7] N. Hidayati, "Penggunaan Metode Waterfall Dalam Rancang Bangun Sistem Informasi Penjualan," 2019.

- [8] Frensia Tanaga Anaclaudia, Dian Pramana, and I Made Arya Budhi Saputra, "Reactjs and Expressjs Implementation In PMK ITB STIKOM Bali Activity Management," *Aptisi Transactions on Technopreneurship (ATT)*, vol. 5, no. 3, pp. 1–11, Mar. 2023, doi: 10.34306/att.v5i3.313.
- [9] 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 (JTJK)*, vol. 5, no. 2, 2021.
- [10] B. Johansson and M. Lahtinen, "Requirement Specification in Government IT Procurement," *Procedia Technology*, vol. 5, pp. 369–377, 2012, doi: 10.1016/j.protcy.2012.09.041.
- [11] W. Shen and S. Liu, "Formalization, Testing and Execution of a Use Case Diagram."
- [12] S. Sridevi, "USER INTERFACE DESIGN." [Online]. Available: [www.researchpublish.com](http://www.researchpublish.com)
- [13] N. ANISAH MOHD SAAD and M. MUNIANDI, "The Reflections on the using of Oracle Data Modeler in Creating Entity Relationship Diagram (ERD)," *International Journal of Research Publications*, vol. 66, no. 1, Dec. 2020, doi: 10.47119/ijrp1006611220201601.
- [14] S. Sutiah and S. Supriyono, "Software testing on e-learning Madrasahs using Blackbox testing," *IOP Conf Ser Mater Sci Eng*, vol. 1073, no. 1, p. 012065, Feb. 2021, doi: 10.1088/1757-899x/1073/1/012065.