Project Title: Crowdfunding Platform for Social Causes

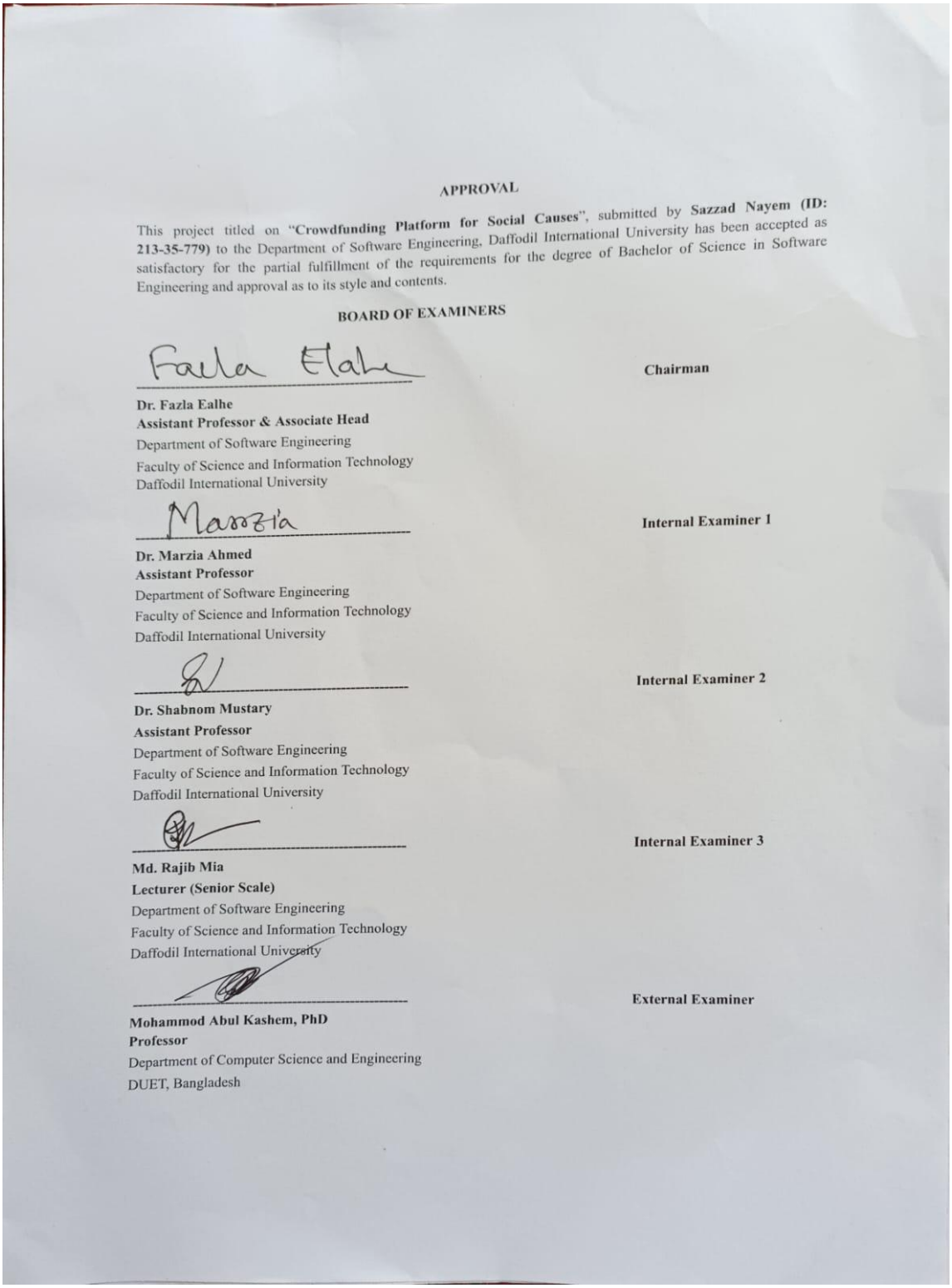
Final Year Project/Thesis/Internship
Course Code: SE431

Submitted To
Sazzad Nayem
ID: 213-35-779
Department of Software Engineering
Daffodil International University

Supervised By
Dr. Md. Fazla Elahe
Assistant Professor & Associate Head
Department of Software Engineering
Daffodil International University

This Project report has been submitted in fulfillment of the requirements for the Degree of **Bachelor of Science in Software Engineering**

APPROVAL



APPROVAL

This project titled on "**Crowdfunding Platform for Social Causes**", submitted by **Sazzad Nayem (ID: 213-35-779)** to the Department of Software Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science in Software Engineering and approval as to its style and contents.

BOARD OF EXAMINERS

Chairman

Dr. Fazla Ealhe
Assistant Professor & Associate Head
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Internal Examiner 1

Dr. Marzia Ahmed
Assistant Professor
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Internal Examiner 2

Dr. Shabnom Mustary
Assistant Professor
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Internal Examiner 3

Md. Rajib Mia
Lecturer (Senior Scale)
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

External Examiner

Mohammod Abul Kashem, PhD
Professor
Department of Computer Science and Engineering
DUET, Bangladesh

DECLARATION

Department of Software Engineering

Faculty of Science and Information Technology

Supervisor Approval Form

Fall 2025	B.Sc. In SWE	Campus: DSC

Student Name	Student ID
Sazzad Nayem	213-35-779

Project/Thesis Information	
Project/Thesis Title	Crowdfunding Platform for Social Causes
Type of work	Project

Supervisor information	
Supervisor Name	Dr. Md. Fazla Elahe
Supervisor Initial	Assistant Professor & Associate Head
Completed Credit till now	139
How many credits in this semester	
Supervisor Consent	✓ Yes ☐ No

Fazla Elahe 26.11.25

Supervisor Signature

ACKNOWLEDGEMENTS

In the name of the Almighty Allah the greatest thanks to him is that he blessed me through his mercy and made me complete my final project. It is my great privilege to mention my honoured supervisor, Dr. Md. Fazla Elahe, the Assistant Professor, Assistant Head, Department of Software Engineering, Daffodil International University who was of great help to me by offering me constant support, good advice and wise ideas in all stages of my project. His advice has enabled me to become a better technical worker and become capable of creating confidently and thinking critically.

I am also grateful to the entire well-known body of the faculty members of the Department of Software Engineering because of the support, knowledge, and inspiration they gave me during my academic career. Secondly, I would like to express my gratitude to my friends and other students who provided me with valuable ideas, candor feedback and moral encouragement in developing and reviewing this system. Lastly, I would like to express my deep gratitude to my parents and family who have given me their unconditional support, patience, and love which I consider as the strongest power in my life and constant motivational support. Without them, I would not have achieved this because they have been my main support and source of inspiration.

DEDICATION

The majority of this work would not have been possible without Dr. Md. Fazla Elahe. During my educational journey, he provided me with great motivation and advice to make sure that I completed my degree. I appreciate everything that Dr. Fazla Elahe has accomplished and how helpful he has been to me.

This document is dedicated to my parents for being my greatest support while I was attending college, and for all the sacrifices and encouragement that they gave me in order for me to be who I am today.

The majority of this project was developed under the supervision of Dr. Md. Fazla Elahe, a Professor and Associate Head of Software Engineering at Daffodil International University, and will not be submitted anywhere else whole or in part, as this project is an original piece.

ABSTRACT

Web-based Crowdfunding for community social causes. This platform provides fund raising and easy way for people and groups to collect financial support for their social/community work. Many traditional fundraising methods suffer from limited reach, lengthy processing, lack of transparency and inefficiency. To tackle these issues, the platform provides a comprehensive and easily navigable digital online product that allows Campaign Creators to set-up, manage and track a fundraising campaign, while Donors can make easy contributions via secure online payments.

The design of the platform has combined the Roles of Campaign Creators, Donors, and Admins into one organized workflow (Campaign Approval, Donation Processing, E-mail Notification, and Transparent Tracking). SSLCommerz is used to enable fast and secure payment transactions, and Multilingual support will enable people to use the crowdfunding platform in any language.

Future Development of the platform will include an analytics dashboard to track campaigns, reports detailing contributions and donors, feedback systems, and a blockchain-based method of proving transparency within the system. The ultimate goal of the platform is to improve the means that communities in Bangladesh and abroad can raise funds for the causes that they believe in, and promote social change through increased access, efficiency, and trust.

TABLE OF CONTENT

Table 1: Declaration Title Page

LIST OF TABLES

Table 2: List of Tables

LIST OF FIGURES

Table 3: List of Figures

CHAPTER 1 INTRODUCTION

1.1 Background

1.1.1 Context and Relevance

The social welfare, charity and fundraising sectors in Bangladesh have experienced dramatic changes as a result of the emergence of digital technologies over the past several years. Traditional donation channels have suffered from a lack of transparency, limited reach and inefficient management of funds. With the growing popularity of the internet and smartphones, many people and organisations are now turning to online crowdfunding websites as a viable option for obtaining funding for causes they support. By providing a place for individuals and organisations to connect with each other through a social cause based **crowdfunding platform**, it is possible to create an environment in which all parties benefit.

1.1.2 Problem Identification

The Usefulness of Understanding the Problem Crowdfunding continues to grow in popularity around the globe; however, in Bangladesh, there are no dedicated, organized or automated sites to solely aid individuals during times of medical emergencies, provide educational assistance, and provide disaster relief assistance. Many of the existing providers do not provide sufficient means for campaign legitimacy verification and for establishing a means of building donor confidence and also do not provide a secure, reliable means of making electronic donations. Therefore, there are many campaigns that are legitimate, but do not receive the required level of assistance from the public because of a lack of platform and because of donor concern about trustworthiness and authenticity of campaigns.

When there is no clear and well-designed platform for connecting donors and organizers, both parties experience challenges and difficulties.

1.1.3 Purpose and Justification

The purpose and justification of this initiative is to provide users with a safe, secure, easy-to-use, online crowdfunding platform exclusively for social causes. By utilizing a fully verified and transparent campaign process, donors will have access to current campaign information and be able to view their contributions through real-time contributions. Additionally, campaign creators will benefit from improved efficiency when creating campaigns and receiving contributions. By providing an alternative way for individuals to contribute to their community, the project creates value for the donor and has a positive impact on the social responsibility of the donor and the vulnerability of the targeted community. It also creates greater access to financial support for individuals and businesses who need it.

1.1.4 Scope

The scope of this project encompasses the design and development of a fully integrated crowdfunding platform where any user may create a fundraising campaign; support projects through donations with security; and view their donation activity and tracking information. The key components of this project are authentication of donors, administrative approval of all fundraising campaigns, the history of donor contributions and payment processing via the SSLCommerz gateway, and a moderation administrative interface for the main application. This system will only support charitable fundraising (i.e., not for-profit investments or personal gain).

1.2 Project Planning and Initiation

Feasibility Study (Step-by-Step)

Phase 1: Preliminary Review & Project Scoping This stage determines that there is a requirement for the development of a pedagogy approach to social cause crowdfunding. The aim is to identify system boundaries, target stakeholders and necessary capabilities. Initial reports provide indications of an increased need for transparent donation systems in Bangladesh.

Phase 2: Market Analysis If there were a trusted platform, so many people would be willing to invest back in social causes. The use of online donation is increasing, also promoted by mobile financial services including bKash and Nagad as well as digital banking. This suggests strong market potential.

Phase 3: Technical Feasibility Analysis The system may be built in terms of the up-to-date web platform technologies, including React, Next.js, Node.js, Express.js, and Prisma. SSLCommerz is integrated to provide security in payment. Cloud platform hosting is possible. Technical tools and resources required are readily available.

Phase 4: Financial Feasibility Analysis The development cost is low because the platform is academic and is at prototype level. Hosting, domain, payment gateway and security features are expenses in case it is extended commercially. The ratio of benefits to costs is good because of high usability and the social impact.

1.3 Target User Profile and Tentative Elicitation Process

1.3.1 Target User

- Donors
- Campaign Creators
- NGOs & Social Organizations
- Admin/Moderator
- General Public seeking help for emergencies

1.3.2 User profile

Table 4: User profile

Table 4.1: User Profile for Donor

User Class	**Note on Characteristics**
Type of user	Occasional/Frequent Donor
Age range	18–60
Frequency of use	Moderate
Mandatory	No
Computer experience	Basic–Intermediate
Education	Varies
goal	Donate to trusted causes
Language skills	Bangla/English
Number of users	Large
Training	Not required
Others system use	Mobile banking
Way of working	Simple, quick interactions

Table 4.2: User Profile for Campaign Creator

User Class	**Note on Characteristics**
Type of user	Campaign Creator
Age range	18–60
Frequency of use	High during campaign period
Mandatory	Yes
Computer experience	Intermediate
Education	College level or above
goal	Raise funds for causes
Language skills	Bangla/English
Number of users	Moderate
Training	Minimal

Others system use	Document upload
Way of working	Detailed form submission

Table 4.3: User Profile for Admin

User Class	**Note on Characteristics**
Type of user	Admin (System moderator)
Age range	25–50
Frequency of use	Regular
Mandatory	Yes
Computer experience	High
Education	Professional level
goal	Ensure authenticity & quality control
Language skills	Bangla/English
Number of users	Few
Training	Required
Others system use	Dashboard tools
Way of working	Review & decision-making

1.3.3 Elicitation Process

The informal interviews, online survey, and observation of the current donation systems were used to gather the user requirements. The need to be transparent was raised by the donors, and an easy submission process was needed by the campaign creators. Admins laid stress on the verification and fraud prevention.

1.4 Project Block Diagram

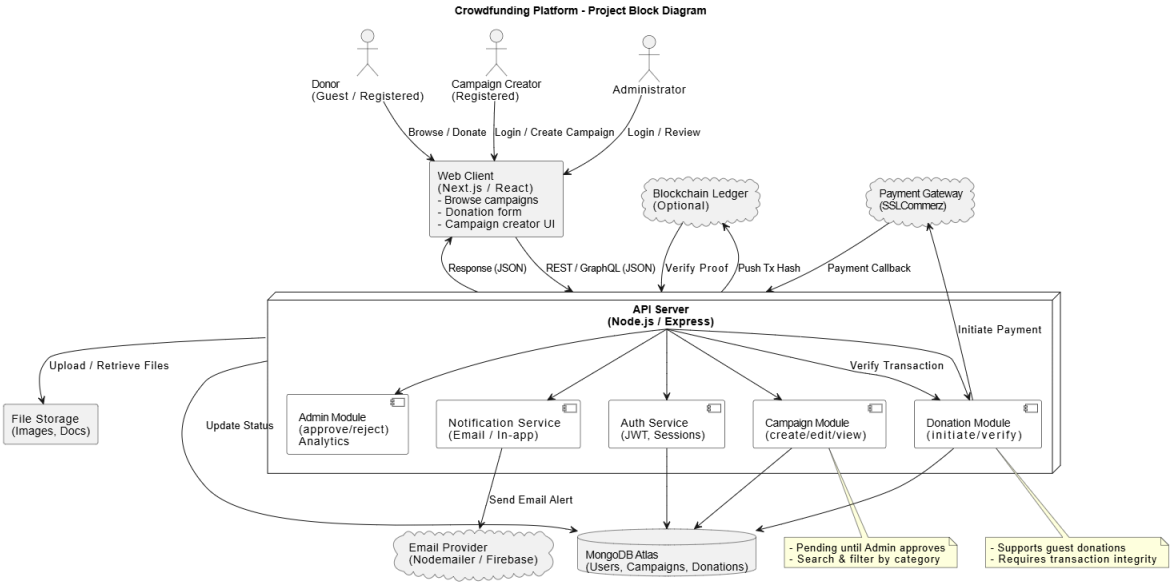


Figure 1: System Block Diagram

1.5 System Requirements

1.5.1 Hardware Requirements

• Any modern PC or laptop

• Minimum 4GB RAM

• Stable internet connection

1.5.2 Software Requirements

• Operating System: Windows/Linux/Mac

• Django, React/Next.js

• Database: PostgreSQL/MySQL

• Payment Gateway API

• Code Editor (VS Code)

1.5.3 Constraints and Dependencies

• Dependent on SSLCommerz payment integration

• Requires stable internet connectivity

• Verification depends on admin judgment

• Hosting uptime affects functionality

1.6 Project Scheduling

The process involved in the project schedule is planning, research, design, development, testing, and documentation. A simplified Gantt-like plan can be tracked over a number of weeks with the risk management approach in terms of delays, integration and requirement changes.

1.7 Summary

The social-cause-based crowdfunding platform, its context, problem statement, importance, scope, feasibility, user profile, block diagram, requirements, and scheduling were presented in this chapter. It preconditions the comprehension of the way the system works and why it is required.

CHAPTER 2 DESIGN AND IMPLEMENTATION

2.1 Introduction

This chapter dwells on the general design and execution of the Crowdfunding Platform of the Social Causes. It describes the transformation of the system requirements into organized models and technical elements. Both functional and non-functional requirements are also contained in the chapter, and then the chapter is followed by the detailed UML diagram like Use Case, Activity, Sequence, Class, and ER Diagrams which show the internal workflow of the system.

Moreover, this chapter describes an object-oriented design methodology employed to guarantee scalability, readability, and maintainability. Lastly, the coding structure is also described by means of sample implementations that are obtained in the attachment, Appendix A. In general, this chapter provides the technical base, on which the system is implemented.

2.2 Functional Requirements

Table 5 : Functional Requirements

FR01	**User Registration & Authentication**
Description	The homepage is publicly accessible. **Campaign Creators** and **Admins** must register and log in securely, while **Donors** can donate directly without registration through a form.
Stakeholder	Campaign Creator, Admin

FR02	**Campaign Creation & Management**
Description	**Campaign Creators** can create, edit, or delete campaigns with details like title, goal amount, and deadline. New campaigns remain pending until admin approval.
Stakeholder	Campaign Creator

FR03	Campaign Approval
Description	**Admins** review submitted campaigns and approve or reject them before publishing on the homepage.
Stakeholder	Admin

FR04	Browse & View Campaigns
Description	All users can view approved campaigns with goal amount, raised funds, and description. Campaigns can be filtered by category or keyword.
Stakeholder	Donor, Campaign Creator, Admin

FR05	Donation Processing
Description	**Donors** can donate directly through **SSLCommerz** by filling a short form. Payment details are processed securely, and confirmation messages are sent automatically.
Stakeholder	Donor

FR06	Email Notifications
Description	The system sends automatic email updates for important actions such as registration, campaign approval, and donation confirmation.
Stakeholder	Donor, Campaign Creator, Admin

FR07	Donation Tracking
Description	**Campaign Creators** can track all donations for their campaigns, view donor information, and analyze total funds raised.
Stakeholder	Campaign Creator

FR08	**Feedback & Review System** (Future Enhancement)
Description	**Donors** will be able to submit ratings and written feedback for campaigns they supported, helping maintain transparency and trust.
Stakeholder	Donor, Campaign Creator

FR09	**Admin Dashboard & Reporting** (Future Enhancement)
Description	The **Admin Dashboard** displays overall system stats like users, campaigns, and donations. Future upgrades will include analytics and detailed reporting.
Stakeholder	Admin

FR10	**Payment Dashboard** (Future Enhancement)
Description	A dedicated dashboard for **Admins** to view all transactions, including successful, failed, and refunded payments.
Stakeholder	Admin

FR11	**Multilingual Support**
Description	The platform supports both **Bangla** and **English**, allowing users to switch between languages easily.
Stakeholder	Donor, Campaign Creator, Admin

2.3 Non-Functional Requirements

In this section, the non-functional features of Social Causes Crowdfunding Platform are discussed. These are necessary to ensure system quality, performance, reliability, portability and general usability. What these actually represent is how the system should behave, not what features it should have.

2.3.1 Performance

There should be an optimal performance of the system to fulfill the requests of high number of users and heavy donation activities.

- User facing pages such as Home, Campaign Listing, CampaignDetails have to load within 3 seconds in average network conditions
- The system must provide a response time of at least 500 concurrent users before significant degradation occurs.
- Database queries should be optimized for campaign search, filtering to show results within 1 second optimising any dashboard-like page.
- Donation transaction API calls should be performed, which can be added within 2–4 seconds to help integrate payment gateway (like SSLCommerz).
- It should be able to work with large datsets (thousands of campaigns and donations) and display results without waiting too long.
- Both images and videos should be compressed and cached to increase load time.

2.3.2 Reliability

Stability: is the platform available to use or does it crash reliably and not lose your data/income.

- The system should have at least a 99% availability (planned maintenances not included).
- Anytime donations happen there should be a fail-safe so that people don't get charged twice due to network issues.
- The system shall recover from small crashes automatically triggered somehow by server restart scripts or container level health checks.

- It is necessary to prevent data from being compromised by utilizing daily automated backups and transactional database integrity policies.
- The system has to be ACID for all financial transactions, in order to preserve consistency and correctness.
- If the payment gateway goes down, donations should be recoverable and donors given a chance to try again without having to re-enter what was previously provided.
- Sessions need to survive securely (for authenticated users at least) while your tokens expire, right?

2.3.3 Portability

Portability makes the system capable of running on various platforms, devices and environmental settings without significant modifications.

- The platform needs to be fully responsive and be able to be accessed using mobile (smartphone), tablet, desktop with the same UI/UX.
- The technology should perform without a hitch in all major browsers such as Chrome, Firefox, Edge and Safari.
- The backend should easily deploy in different server environments as Linux, Cloud VM or Docker containers.
- The app must be deployable to a cloud provider such as AWS, Vercel, Render or DigitalOcean.

2.3.4 Security

- All user passwords need to be hashed by well-defined commercial hash algorithms (e.g., menu).
- The application should be able to back sessions with token-based authentication (JWT).
- You'd need to implement a Secure payment processing, following certain steps provided by SSLCommerz.

2.3.5 Usability

- It should have a user-friendly interface that is easy for anyone to use, even if said person is just starting out with this type of program.
- Navigation has to be intuitive, design should be consistent and visual hierarchy of elements satisfactory.
- Validation and help text + error message(s) to all the formats.

2.3.6 Maintainability

- The resolution will be a modular code structure (MVC / API Layering) to allow further development in the future.
- Naming and explanation, comment this.creationalStatement = new Keywordexplanation python should be consistent among code.
- All functionality is accompanied with unit/integration tests Core functionality should be tested (automated)

2.3.7 Scalability

- It should be able to be scaled horizontally or vertically according to user amount.
- The database should be scaleable (just enough) for a constantly increasing campaigns and donations count without losing performance.

2.4 Object-oriented System design using UML

2.4.1 Use Case Diagram

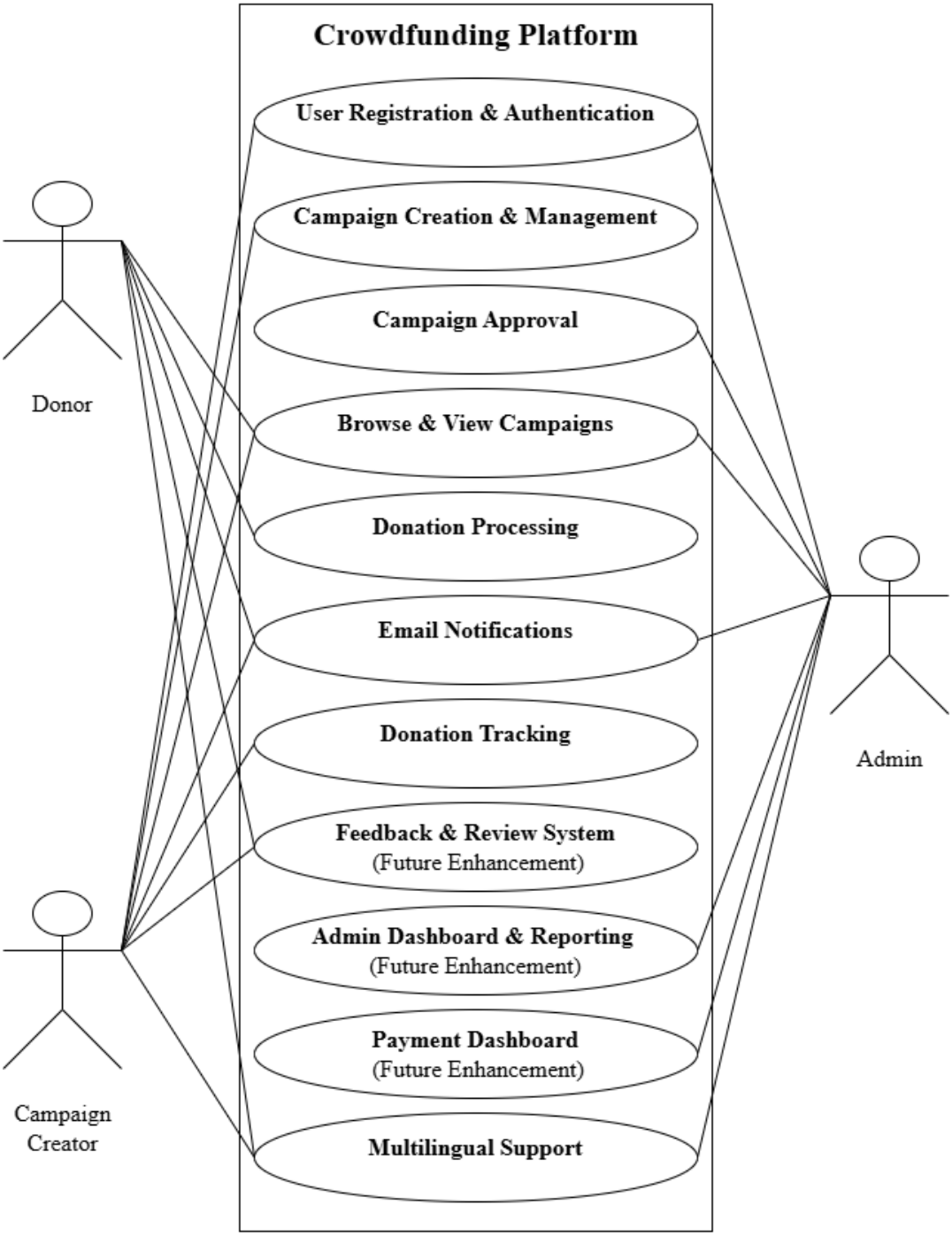


Figure 2: Use case Diagram

2.4.2 Case Description

Table 6 : Case Description

Case Description-6.1: User Registration & Authentication (FR01)

<table>
<tr><td>Use Case</td><td>User Registration & Authentication</td></tr>
<tr><td>Goal</td><td>To allow Campaign Creators and Admins to securely create an account and log in to the system.</td></tr>
<tr><td>Precondition</td><td>● The user must have access to the Crowdfunding Website.
● The user must provide valid information.</td></tr>
<tr><td>Success End Condition</td><td>● System shows: “Successfully Registered!”
● OR (for login) “Login Successful!”</td></tr>
<tr><td>Failed End Condition</td><td>● System shows: “Registration Failed” or “Invalid Login Credentials”</td></tr>
<tr><td>Primary Actors</td><td>● Campaign Creator
● Admin</td></tr>
<tr><td>Secondary Actors</td><td>● Authentication Service
● Database</td></tr>
<tr><td>Trigger</td><td>The user clicks the “Register” or “Login” button to access restricted features.</td></tr>
<tr><td>Main Success Scenario (MSS)</td><td>
<table>
<tr><td>1.</td><td>The user clicks on the “Register” button.</td></tr>
<tr><td>2.</td><td>The system displays the registration form.</td></tr>
<tr><td>3.</td><td>The user enters required information (Name, Email, Password).</td></tr>
<tr><td>4.</td><td>The user clicks the “Submit” button.</td></tr>
<tr><td>5.</td><td>The system validates the information.</td></tr>
<tr><td>6.</td><td>The system stores user information in the database.</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><td></td><td>
<table>
<tr><td>7.</td><td>System displays a message:
“Successfully Registered!”</td></tr>
<tr><td>8.</td><td>Users can now proceed to Login.</td></tr>
</table>
</td></tr>
<tr><td>Alternative Flows</td><td>
<table>
<tr><td>1.1</td><td>System Error</td></tr>
<tr><td></td><td>1.1.a System shows: “Try Again!”</td></tr>
<tr><td>4.1</td><td>Missing Input Fields</td></tr>
<tr><td></td><td>The user clicks Submit without completing the form.</td></tr>
<tr><td></td><td>4.1.a System checks and shows:
“Please fill out all required fields.”</td></tr>
<tr><td>5.1</td><td>Validation Failure</td></tr>
<tr><td></td><td>Email already exists / password too weak.</td></tr>
<tr><td></td><td>5.1.a System shows: “Invalid Information Provided.”</td></tr>
<tr><td>6.1</td><td>Database Error</td></tr>
<tr><td></td><td>The system fails to save user details.</td></tr>
<tr><td></td><td>6.1.a System shows: “Details Not Saved.”</td></tr>
<tr><td>7.1</td><td>Email Verification Pending (Optional enhancement)</td></tr>
<tr><td></td><td>7.1.a System shows:
“Please verify your email to complete registration.”</td></tr>
</table>
</td></tr>
<tr><td>Quality Requirements</td><td>

The registration process should complete within 30 seconds on a normal internet connection.
The system must validate all mandatory fields automatically.
User passwords must be encrypted before storing in the database.

</td></tr>
</table>

Case Description-6.2: Campaign Creation & Management (FR02)

<table>
<tr><td>Use Case</td><td>Campaign Creation & Management</td></tr>
<tr><td>Goal</td><td>Allow Campaign Creators to create, edit, or delete campaigns.</td></tr>
<tr><td>Precondition</td><td>● Users must be logged in as Campaign Creator.
● The system must be available.</td></tr>
<tr><td>Success End Condition</td><td>● Campaign saved as Pending for Approval.</td></tr>
<tr><td>Failed End Condition</td><td>● System shows: “Campaign Submission Failed”</td></tr>
<tr><td>Primary Actors</td><td>● Campaign Creator</td></tr>
<tr><td>Secondary Actors</td><td>● Database
● File Storage (for images/documents)</td></tr>
<tr><td>Trigger</td><td>The user selects “Create Campaign” from the dashboard.</td></tr>
<tr><td>Main Success Scenario (MSS)</td><td>
<table>
<tr><td>1.</td><td>User clicks Create Campaign.</td></tr>
<tr><td>2.</td><td>System shows campaign creation form.</td></tr>
<tr><td>3.</td><td>The user enters details (title, goal amount, description).</td></tr>
<tr><td>4.</td><td>Users upload images/documents.</td></tr>
<tr><td>5.</td><td>User clicks Submit.</td></tr>
<tr><td>6.</td><td>The system validates inputs.</td></tr>
<tr><td>7.</td><td>System stores campaign in database.</td></tr>
<tr><td>8.</td><td>System displays: “Campaign Submitted for Approval”.</td></tr>
</table>
</td></tr>
<tr><td>Alternative Flows</td><td>
<table>
<tr><td>4.1</td><td>Invalid or unsupported file type</td></tr>
</table>
</td></tr>
</table>

	<table><tr><td></td><td>4.1.a System shows: **“Invalid File Format”**</td></tr><tr><td>6.1</td><td>Validation Failed</td></tr><tr><td></td><td>6.1.a System shows: **“Please fill all required fields”**</td></tr><tr><td>7.1</td><td>Database Error</td></tr><tr><td></td><td>7.1.a System shows: **“Campaign Not Saved”**</td></tr></table>
Quality Requirements	● Form submission must be completed within 10 seconds. ● Images max 2MB per file.

Case Description-6.3: Campaign Approval (FR03)

Use Case	Campaign Approval
Goal	Allow Admin to approve or reject campaigns.
Precondition	● Users must be logged in as Admin. ● Campaign must exist in “Pending” state.
Success End Condition	● Campaign becomes **Approved** and visible to the public.
Failed End Condition	● Admin action not saved.
Primary Actors	● Admin
Secondary Actors	● Database
Trigger	Admin opens **Pending Campaigns List**.
Main Success Scenario (MSS)	<table><tr><td>1.</td><td>Admin clicks **Review Campaign**.</td></tr><tr><td>2.</td><td>The system shows all campaign details.</td></tr></table>

<table>
<tr><td></td><td><table>
<tr><td>3.</td><td>Admin clicks Approve.</td></tr>
<tr><td>4.</td><td>System updates campaign status.</td></tr>
<tr><td>5.</td><td>System displays: “Campaign Approved Successfully”.</td></tr>
</table></td></tr>
<tr><td>Alternative Flows</td><td><table>
<tr><td>3.1</td><td>Admin selects Reject</td></tr>
<tr><td></td><td>3.1.a System saves status as Rejected</td></tr>
<tr><td></td><td>3.1.b System notifies Campaign Creator.</td></tr>
<tr><td>4.1</td><td>Database issue</td></tr>
<tr><td></td><td>4.1.a System shows: “Approval Failed. Try Again.”</td></tr>
</table></td></tr>
<tr><td>Quality Requirements</td><td>● Admin decisions should update instantly (<3 sec).</td></tr>
</table>

Case Description-6.4: Browse & View Campaigns (FR04)

Use Case	Browse & View Campaigns
Goal	Allow users to view all approved campaigns.
Precondition	● Campaigns must be approved & published.
Success End Condition	● Campaign details displayed correctly.
Failed End Condition	● Error displaying campaign list.
Primary Actors	● Donor ● Campaign Creator ● Admin
Secondary Actors	● Database ● Search/Filter Engine

<table>
<tr><td>Trigger</td><td>The user visits the homepage or search bar.</td></tr>
<tr><td>Main Success Scenario (MSS)</td><td><table>
<tr><td>1.</td><td>User opens Campaign List.</td></tr>
<tr><td>2.</td><td>System fetches approved campaigns.</td></tr>
<tr><td>3.</td><td>The user clicks a specific campaign.</td></tr>
<tr><td>4.</td><td>The system shows details, progress bars, and donations.</td></tr>
</table></td></tr>
<tr><td>Alternative Flows</td><td><table>
<tr><td>2.1</td><td>Database fetch error</td></tr>
<tr><td></td><td>2.1.a System displays: “Unable to load campaigns.”</td></tr>
<tr><td>3.1</td><td>Campaign deleted</td></tr>
<tr><td></td><td>3.1.a System shows: “Campaign Not Available”</td></tr>
</table></td></tr>
<tr><td>Quality Requirements</td><td>● Results must load within 5 seconds.
● Support keyword/category filters.</td></tr>
</table>

Case Description-6.5: Donation Processing (FR05)

Use Case	Donation Processing via SSLCommerz
Goal	Allow donors to complete secure payments.
Precondition	● Campaign must be active. ● Donors must fill a donation form.
Success End Condition	● Payment marked as **Successful**.
Failed End Condition	● Payment shows **Failed / Cancelled**.
Primary Actors	● Donor

<table>
<tr><td>Secondary Actors</td><td>SSLCommerz Payment Gateway
Database</td></tr>
<tr><td>Trigger</td><td>User clicks Donate Now.</td></tr>
<tr><td>Main Success Scenario (MSS)</td><td><table>
<tr><td>1.</td><td>Donor clicks Donate.</td></tr>
<tr><td>2.</td><td>The system shows a donation form.</td></tr>
<tr><td>3.</td><td>Donor enters amount & details.</td></tr>
<tr><td>4.</td><td>System redirects to SSLCommerz.</td></tr>
<tr><td>5.</td><td>Donor completes payment.</td></tr>
<tr><td>6.</td><td>Payment Success returned.</td></tr>
<tr><td>7.</td><td>The system records the donation.</td></tr>
<tr><td>8.</td><td>System shows: “Thank you for your donation!”</td></tr>
</table></td></tr>
<tr><td>Alternative Flows</td><td><table>
<tr><td>5.1</td><td>Payment Failed</td></tr>
<tr><td></td><td>5.1.a System shows: “Payment Unsuccessful”</td></tr>
<tr><td>6.1</td><td>Callback Missing</td></tr>
<tr><td></td><td>6.1.a System shows: “Payment Verification Error”</td></tr>
</table></td></tr>
<tr><td>Quality Requirements</td><td>Payment should process within 10 seconds.
Must use HTTPS encryption.</td></tr>
</table>

Case Description-6.6: Email Notifications (FR06)

<table>
<tr><td>Use Case</td><td>Email Notifications</td></tr>
<tr><td>Goal</td><td>Automatically send emails based on user action</td></tr>
<tr><td>Precondition</td><td>● Email service must be active.</td></tr>
<tr><td>Success End Condition</td><td>● Email delivered successfully.</td></tr>
<tr><td>Failed End Condition</td><td>● System displays: “Email Not Sent”</td></tr>
<tr><td>Primary Actors</td><td>● Donor
● Campaign Creator
● Admin</td></tr>
<tr><td>Secondary Actors</td><td>● Email Server</td></tr>
<tr><td>Trigger</td><td>Events like registration, campaign approval, donation.</td></tr>
<tr><td>Main Success Scenario (MSS)</td><td><table>
<tr><td>1.</td><td>Events occur (e.g., donation).</td></tr>
<tr><td>2.</td><td>The system generates email content.</td></tr>
<tr><td>3.</td><td>The system sends email.</td></tr>
<tr><td>4.</td><td>The user receives the email.</td></tr>
</table></td></tr>
<tr><td>Alternative Flows</td><td><table>
<tr><td>3.1</td><td>Server not responding</td></tr>
<tr><td></td><td>3.1.a System logs: “Email Error”</td></tr>
</table></td></tr>
<tr><td>Quality Requirements</td><td>● Email should be sent within 1–5 seconds.</td></tr>
</table>

Case Description-6.7: Donation Tracking (FR07)

<table>
<tr><td>Use Case</td><td>Donation Tracking</td></tr>
<tr><td>Goal</td><td>Allow campaign creators to view all donation details.</td></tr>
<tr><td>Precondition</td><td>● Users must be logged in.
● Campaign must exist.</td></tr>
<tr><td>Success End Condition</td><td>● Donation list displayed.</td></tr>
<tr><td>Failed End Condition</td><td>● Tracking unavailable.</td></tr>
<tr><td>Primary Actors</td><td>● Campaign Creator</td></tr>
<tr><td>Secondary Actors</td><td>● Database
● Analytics Engine (optional)</td></tr>
<tr><td>Trigger</td><td>The user opens the Donation History tab.</td></tr>
<tr><td>Main Success Scenario (MSS)</td><td>
<table>
<tr><td>1.</td><td>Creator clicks Donation Tracking.</td></tr>
<tr><td>2.</td><td>System retrieves donation list.</td></tr>
<tr><td>3.</td><td>The system displays donor info & amounts.</td></tr>
</table>
</td></tr>
<tr><td>Alternative Flows</td><td>
<table>
<tr><td>2.1</td><td>Database connection error</td></tr>
<tr><td></td><td>2.1.a Show: “Unable to load donation data.”</td></tr>
</table>
</td></tr>
<tr><td>Quality Requirements</td><td>● Must load results within 4 seconds.</td></tr>
</table>

Case Description-6.8: Feedback & Reviews (FR08)

<table>
<tr><td>Use Case</td><td>Feedback & Review System</td></tr>
<tr><td>Goal</td><td>Allow donors to submit feedback about campaigns.</td></tr>
<tr><td>Precondition</td><td>● Users must have donated.</td></tr>
<tr><td>Success End Condition</td><td>● Feedback successfully saved.</td></tr>
<tr><td>Failed End Condition</td><td>● Feedback not submitted.</td></tr>
<tr><td>Primary Actors</td><td>● Donor</td></tr>
<tr><td>Secondary Actors</td><td>● Database</td></tr>
<tr><td>Trigger</td><td>User clicks “Write Feedback”.</td></tr>
<tr><td>Main Success Scenario (MSS)</td><td>
<table>
<tr><td>1.</td><td>Donor clicks Feedback button.</td></tr>
<tr><td>2.</td><td>The system shows a feedback form.</td></tr>
<tr><td>3.</td><td>Donor writes reviews.</td></tr>
<tr><td>4.</td><td>Donor submits.</td></tr>
<tr><td>5.</td><td>The system saves feedback.</td></tr>
<tr><td>6.</td><td>System shows: “Thank you for your feedback!”</td></tr>
</table>
</td></tr>
<tr><td>Alternative Flows</td><td>
<table>
<tr><td>3.1</td><td>Empty Feedback</td></tr>
<tr><td></td><td>3.1.a Show: “Feedback cannot be empty”</td></tr>
<tr><td>5.1</td><td>Database error</td></tr>
<tr><td></td><td>5.1.a Show: “Failed to save feedback”</td></tr>
</table>
</td></tr>
<tr><td>Quality Requirements</td><td>● Feedback must be saved within 2 seconds.</td></tr>
</table>

Case Description-6.9: Admin Dashboard & Reporting (FR09)

<table>
<tr><td>Use Case</td><td>Admin Dashboard</td></tr>
<tr><td>Goal</td><td>View system statistics & manage users/campaigns.</td></tr>
<tr><td>Precondition</td><td>● The user must be Admin.</td></tr>
<tr><td>Success End Condition</td><td>● Dashboard loads successfully.</td></tr>
<tr><td>Failed End Condition</td><td>● Dashboard fails to load.</td></tr>
<tr><td>Primary Actors</td><td>● Admin</td></tr>
<tr><td>Secondary Actors</td><td>● Database
● Analytics Engine</td></tr>
<tr><td>Trigger</td><td>Admin clicks Dashboard.</td></tr>
<tr><td>Main Success Scenario (MSS)</td><td><table><tr><td>1.</td><td>Admin opens the dashboard.</td></tr><tr><td>2.</td><td>The system retrieves stats.</td></tr><tr><td>3.</td><td>The system displays metrics & graphs.</td></tr></table></td></tr>
<tr><td>Alternative Flows</td><td><table><tr><td>2.1</td><td>Analytics unavailable</td></tr><tr><td></td><td>2.1.a Basic data only shown.</td></tr></table></td></tr>
<tr><td>Quality Requirements</td><td>● Dashboard loads within 5 seconds.</td></tr>
</table>

Case Description-6.10: Payment Dashboard (FR10)

<table>
<tr><td>Use Case</td><td>Payment Dashboard</td></tr>
<tr><td>Goal</td><td>Allow Admin to view payment transactions.</td></tr>
<tr><td>Precondition</td><td>● Payment records must exist.</td></tr>
<tr><td>Success End Condition</td><td>● Transactions displayed successfully.</td></tr>
<tr><td>Failed End Condition</td><td>● Payment list not loaded.</td></tr>
<tr><td>Primary Actors</td><td>● Admin</td></tr>
<tr><td>Secondary Actors</td><td>● Database
● SSLCommerz Log API (optional)</td></tr>
<tr><td>Trigger</td><td>Admin selects Payments tab.</td></tr>
<tr><td>Main Success Scenario (MSS)</td><td><table>
<tr><td>1.</td><td>Admin opens Payment Dashboard.</td></tr>
<tr><td>2.</td><td>The system retrieves all transactions.</td></tr>
<tr><td>3.</td><td>The system displays status (success/failed/refund).</td></tr>
</table></td></tr>
<tr><td>Alternative Flows</td><td><table>
<tr><td>2.1</td><td>Transaction fetch failed</td></tr>
<tr><td></td><td>2.1.a Show error: “Unable to load payments”</td></tr>
</table></td></tr>
<tr><td>Quality Requirements</td><td>● Must support 1,000+ transactions without lag.</td></tr>
</table>

Case Description-6.11: Multilingual Support (FR11)

<table>
<tr><td>Use Case</td><td>Multilingual Support</td></tr>
<tr><td>Goal</td><td>Allow users to switch between Bangla & English.</td></tr>
<tr><td>Precondition</td><td>● Language files must exist.</td></tr>
<tr><td>Success End Condition</td><td>● Language changed instantly.</td></tr>
<tr><td>Failed End Condition</td><td>● Language switch not applied.</td></tr>
<tr><td>Primary Actors</td><td>● Donor
● Campaign Creator
● Admin</td></tr>
<tr><td>Secondary Actors</td><td>● Localization Engine
● UI Renderer</td></tr>
<tr><td>Trigger</td><td>The user clicks Language Toggle.</td></tr>
<tr><td>Main Success Scenario (MSS)</td><td><table><tr><td>1.</td><td>The user selects a language.</td></tr><tr><td>2.</td><td>System loads selected language pack.</td></tr><tr><td>3.</td><td>All UI text updates immediately.</td></tr></table></td></tr>
<tr><td>Alternative Flows</td><td><table><tr><td>2.1</td><td>Translation file missing</td></tr><tr><td></td><td>2.1.a System shows default English text.</td></tr></table></td></tr>
<tr><td>Quality Requirements</td><td>● Language switch must occur within 1 second.</td></tr>
</table>

2.4.3 Activity Diagram

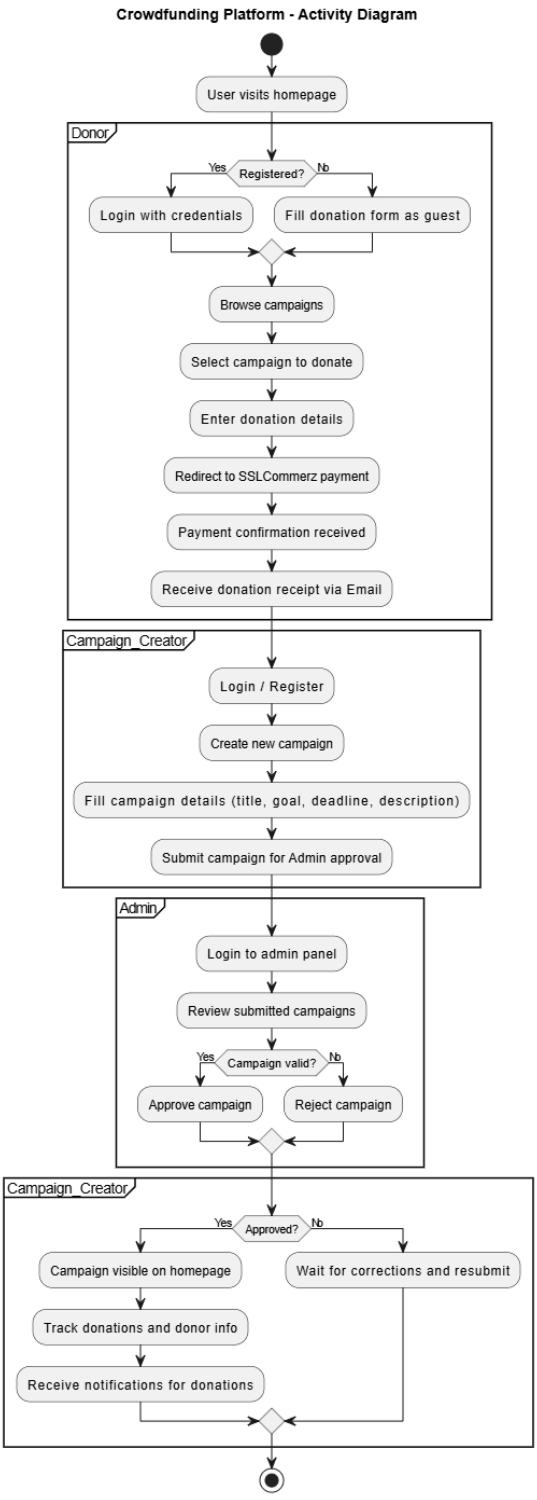


Figure 3: Activity Diagram

2.4.3.1 Donor Flow Activity Diagram

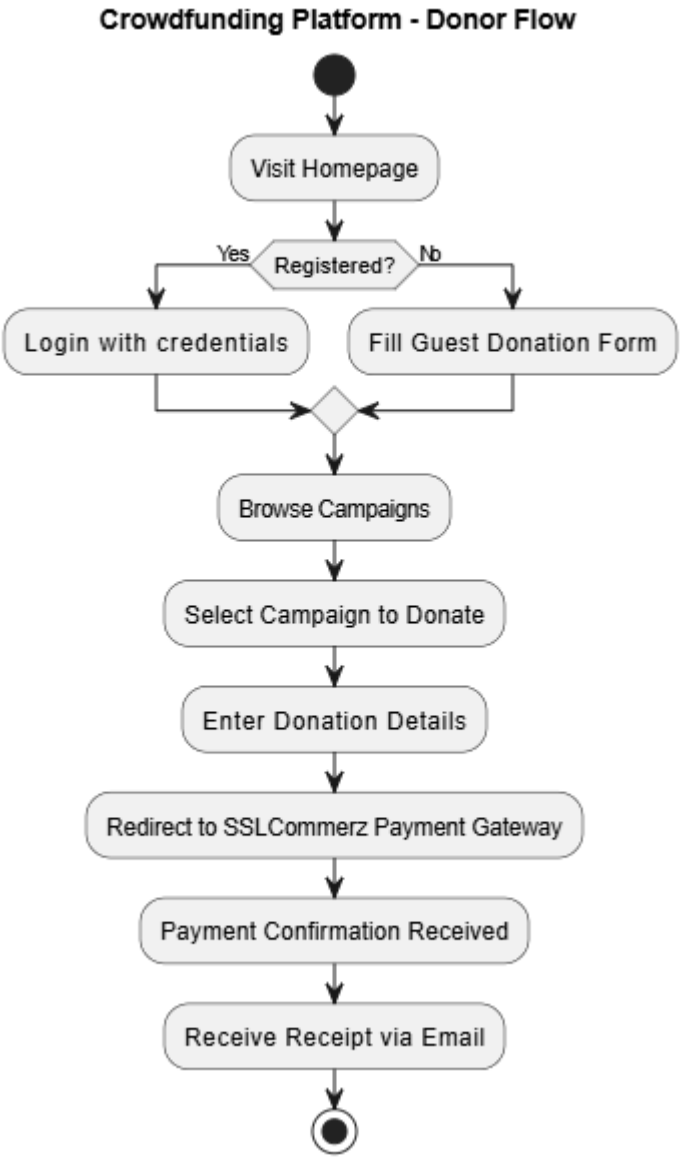


Figure 3.1: Activity Diagram

2.4.3.2 Campaign Creator Flow (Submission & Approval Waiting) Activity Diagram

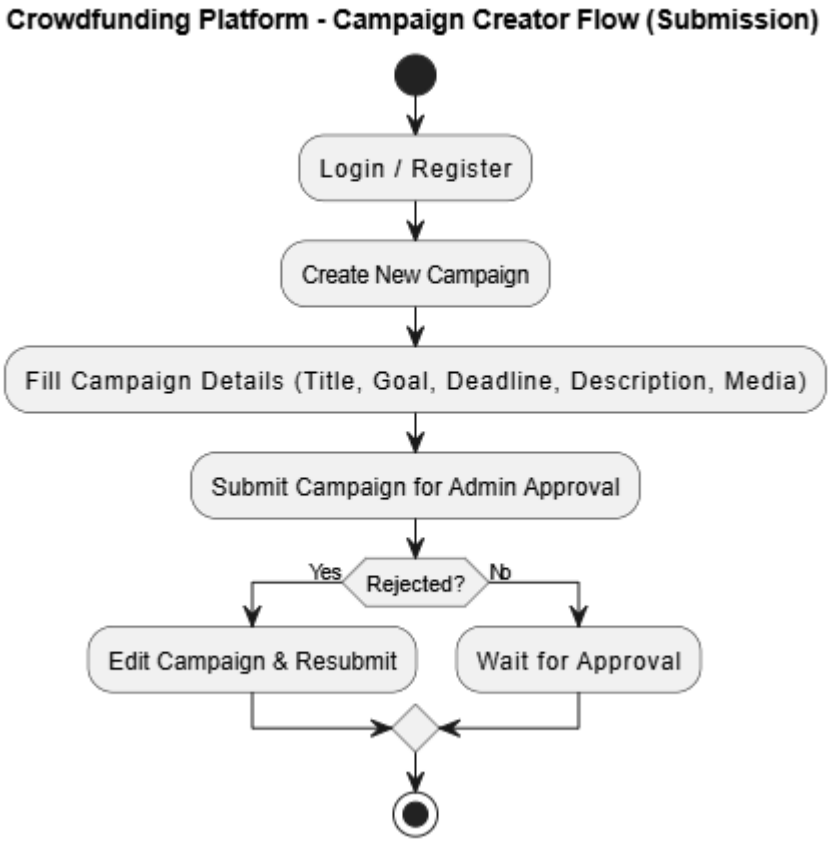


Figure 3.2: Activity Diagram

2.4.3.3 Admin Flow (Approval / Rejection) Activity Diagram

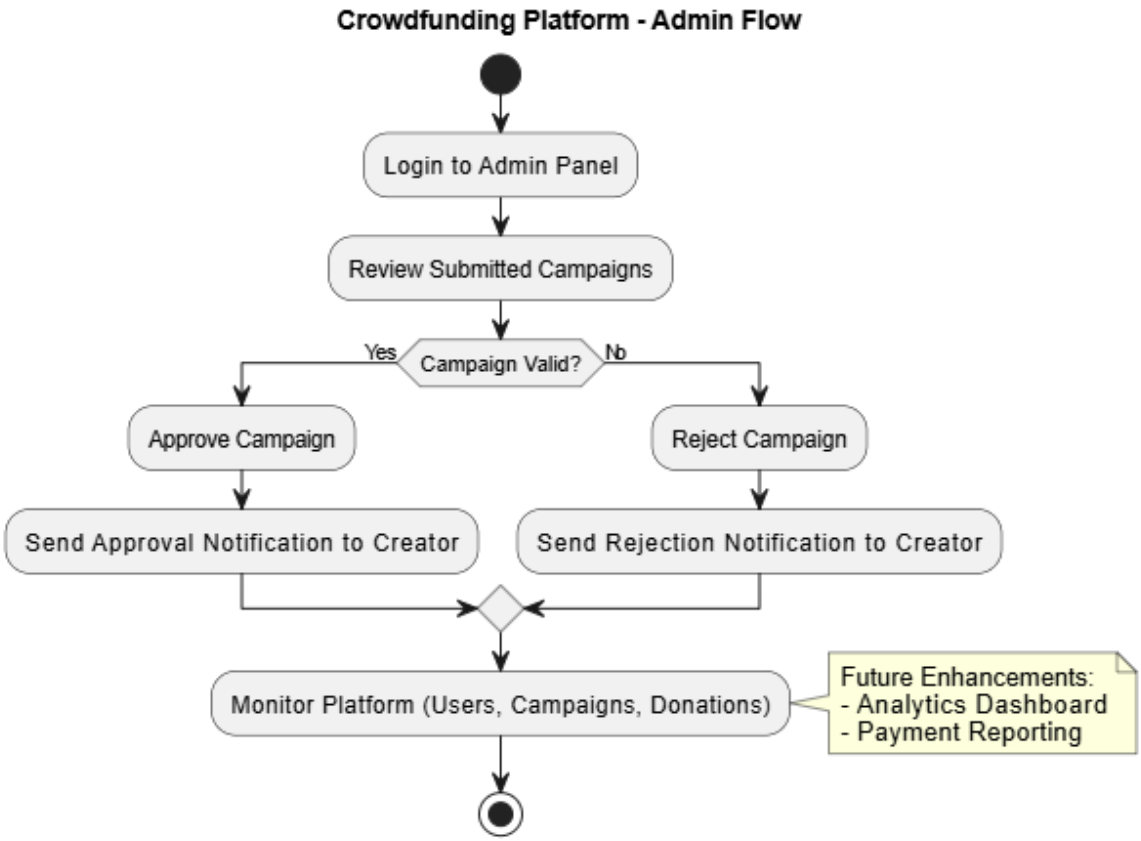


Figure 3.3: Activity Diagram

2.4.3.4 Campaign Creator Flow (Approved Campaign & Tracking / Notifications) Activity Diagram

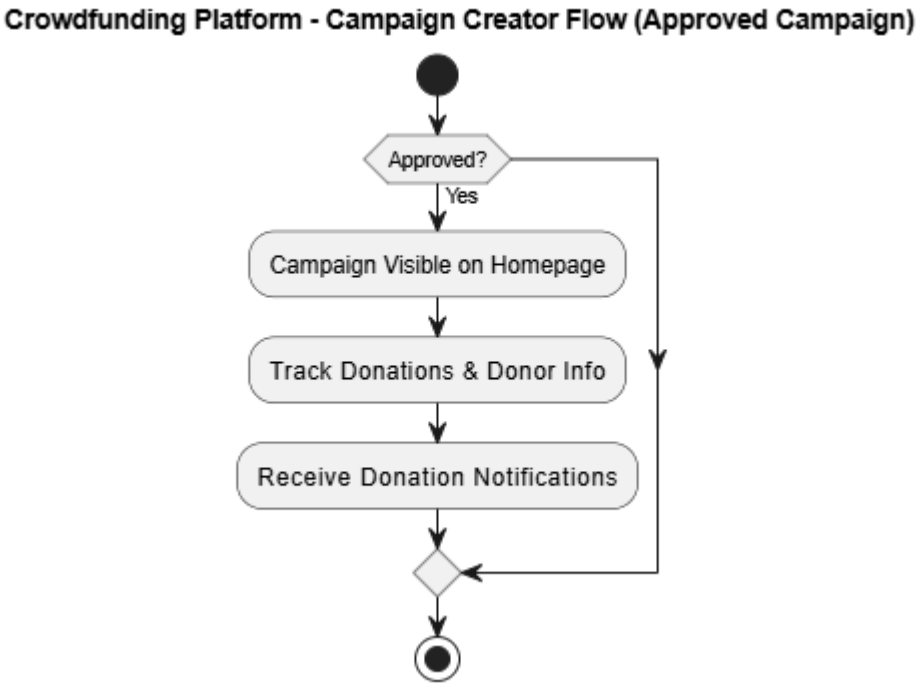


Figure 3.4: Activity Diagram

2.4.4 Sequence Diagram

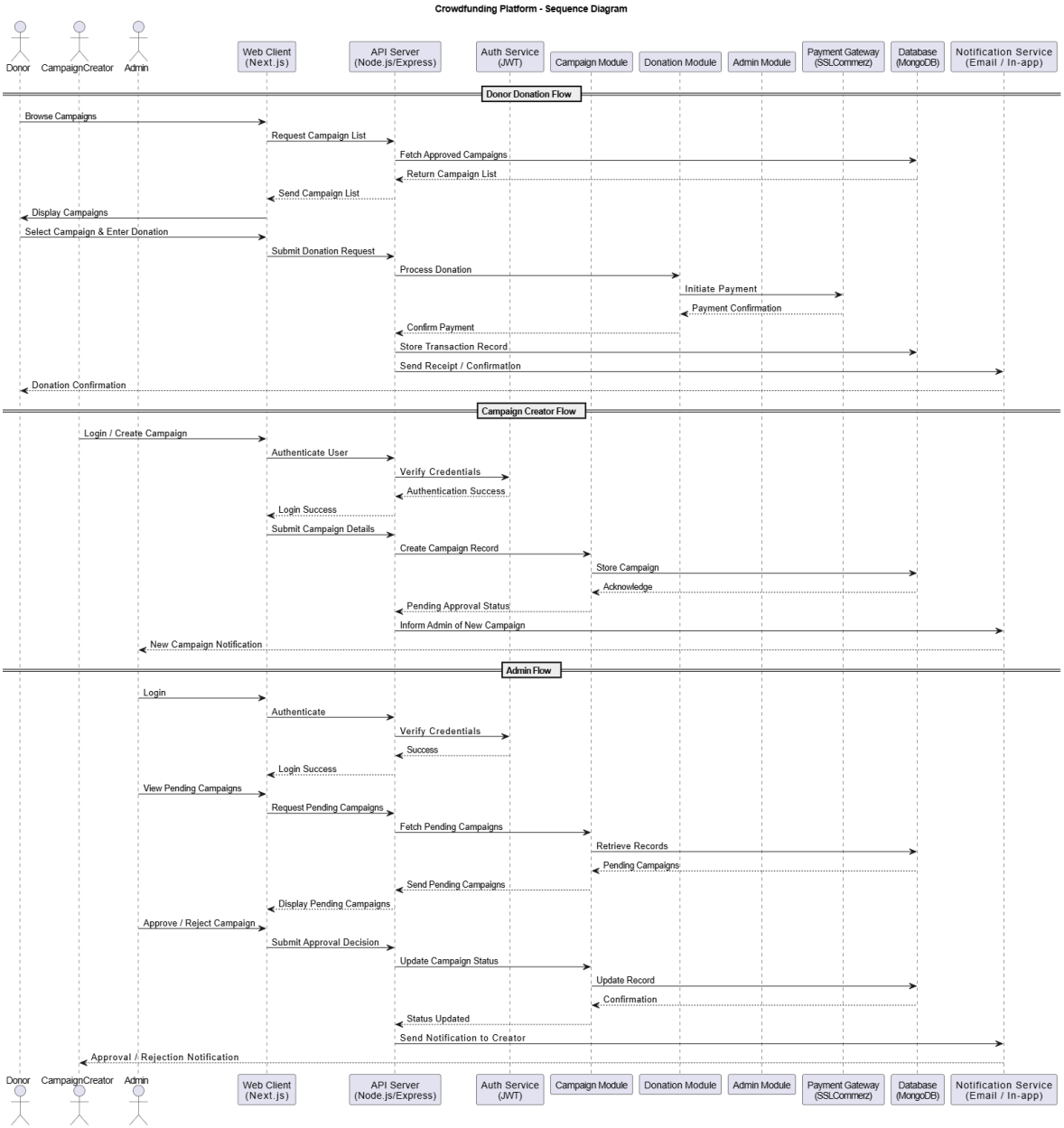


Figure 4: Sequence Diagram

2.4.4.1 Donor Flow Sequence Diagram

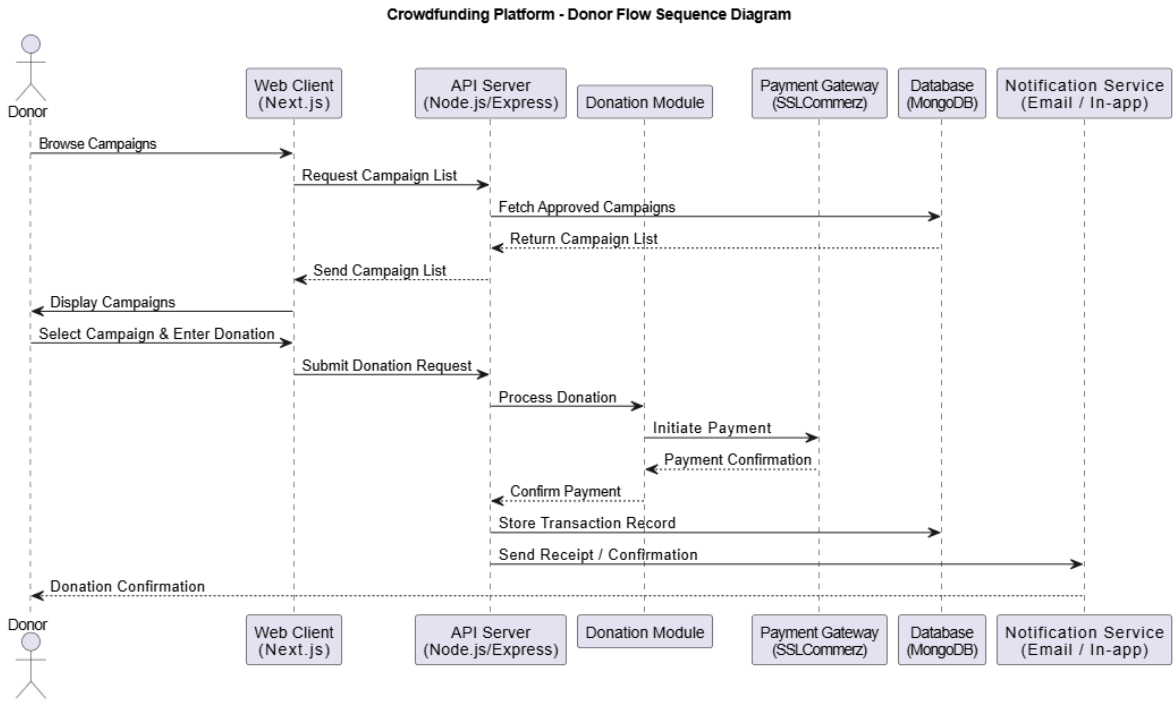


Figure 4.1: Sequence Diagram

2.4.4.2 Campaign Creator Flow (Submission & Approval Waiting) Sequence Diagram

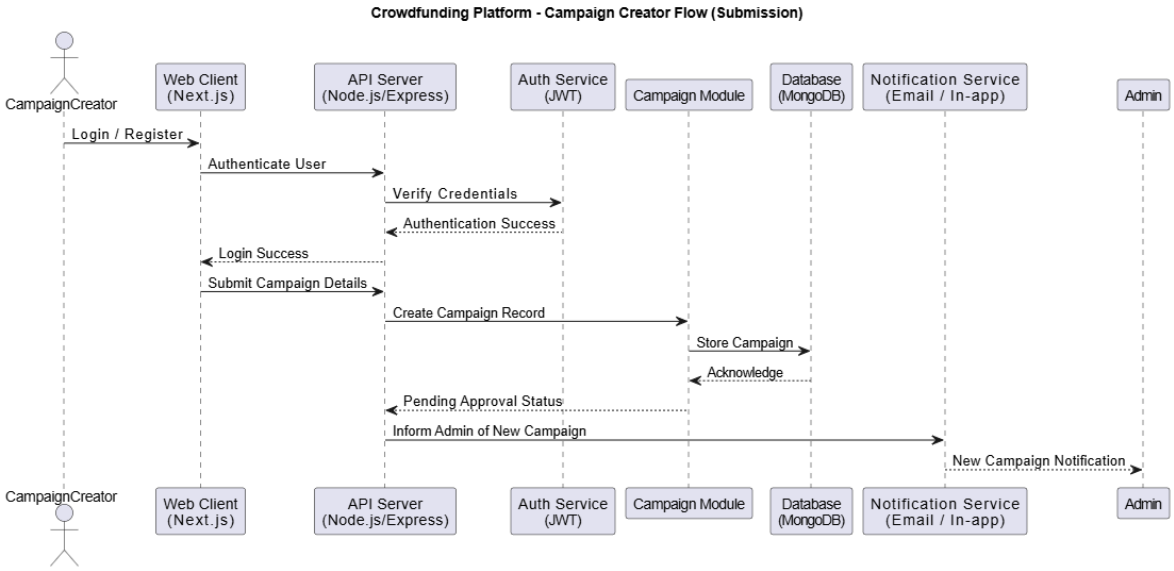


Figure 4.2: Sequence Diagram

2.4.4.3 Admin Flow (Approval / Rejection) Sequence Diagram

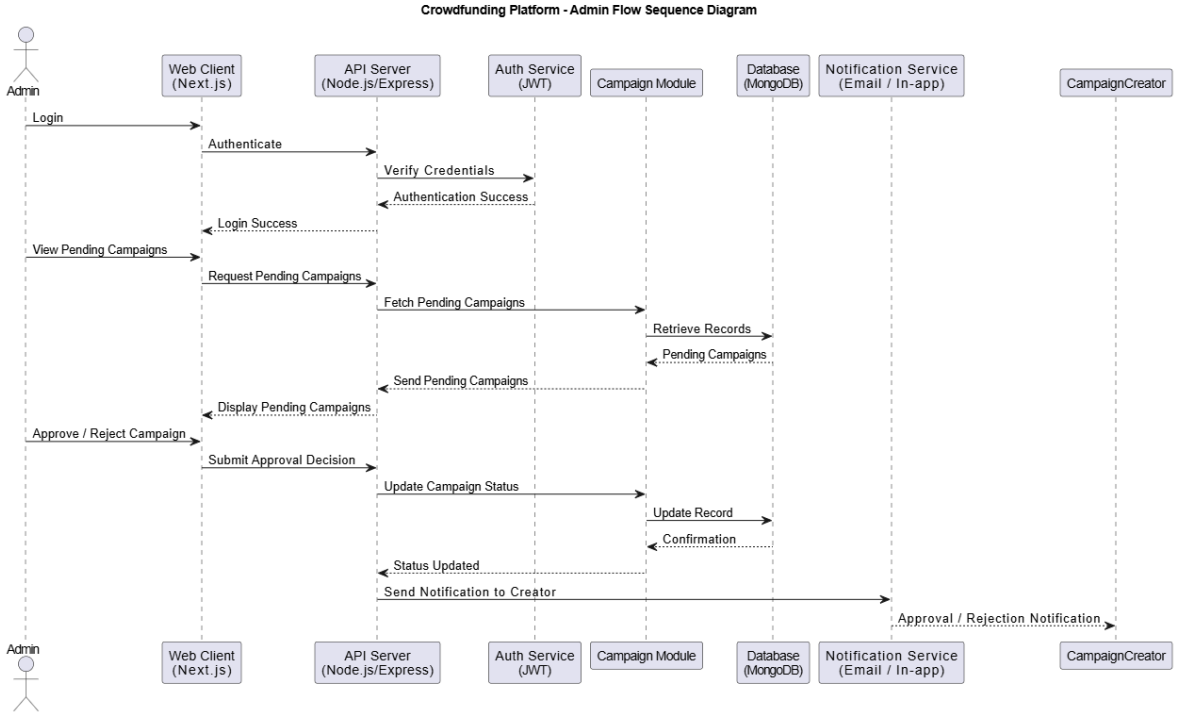


Figure 4.3: Sequence Diagram

2.4.5 Class Diagram

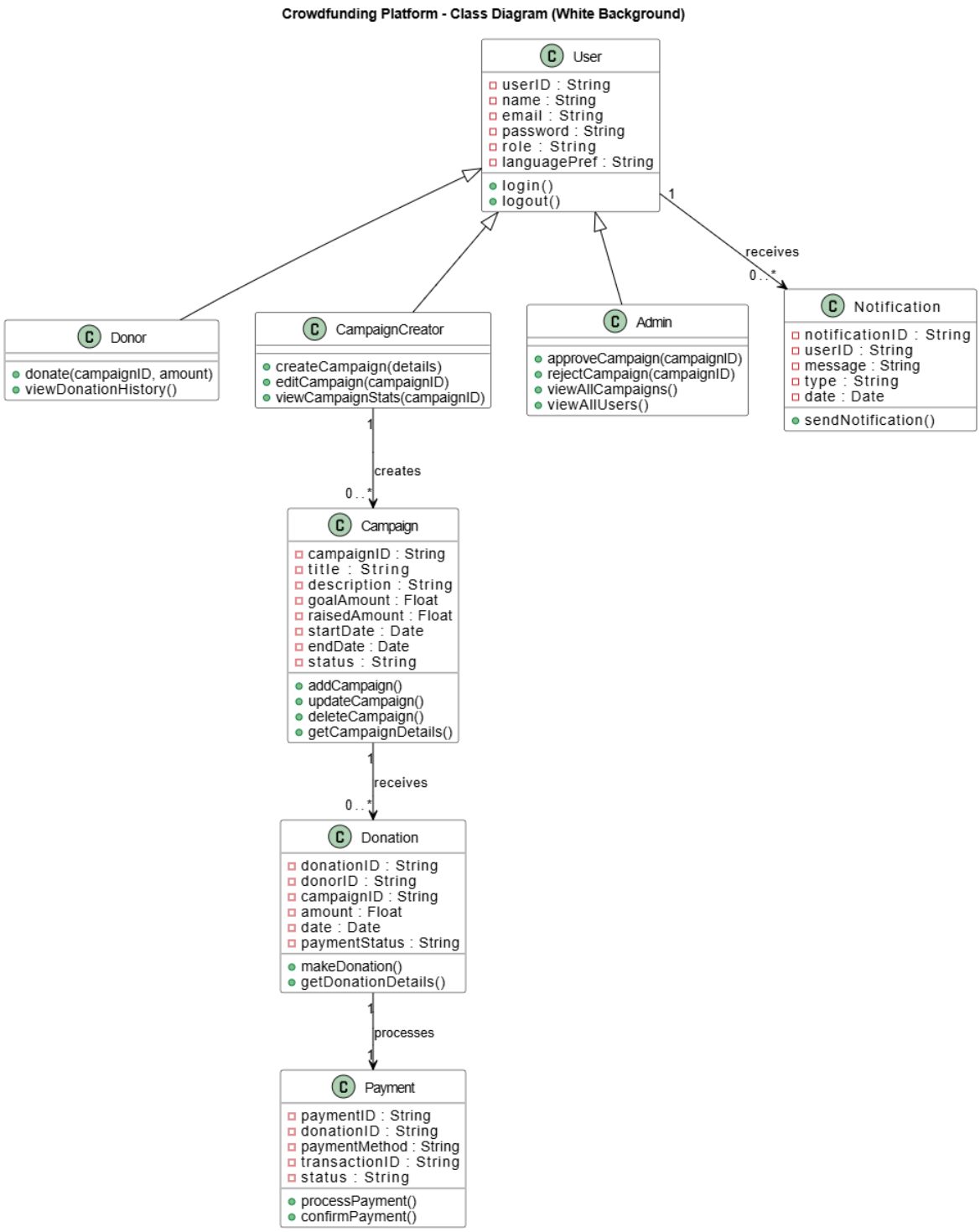


Figure 5: Class Diagram

2.4.6 ER Diagram

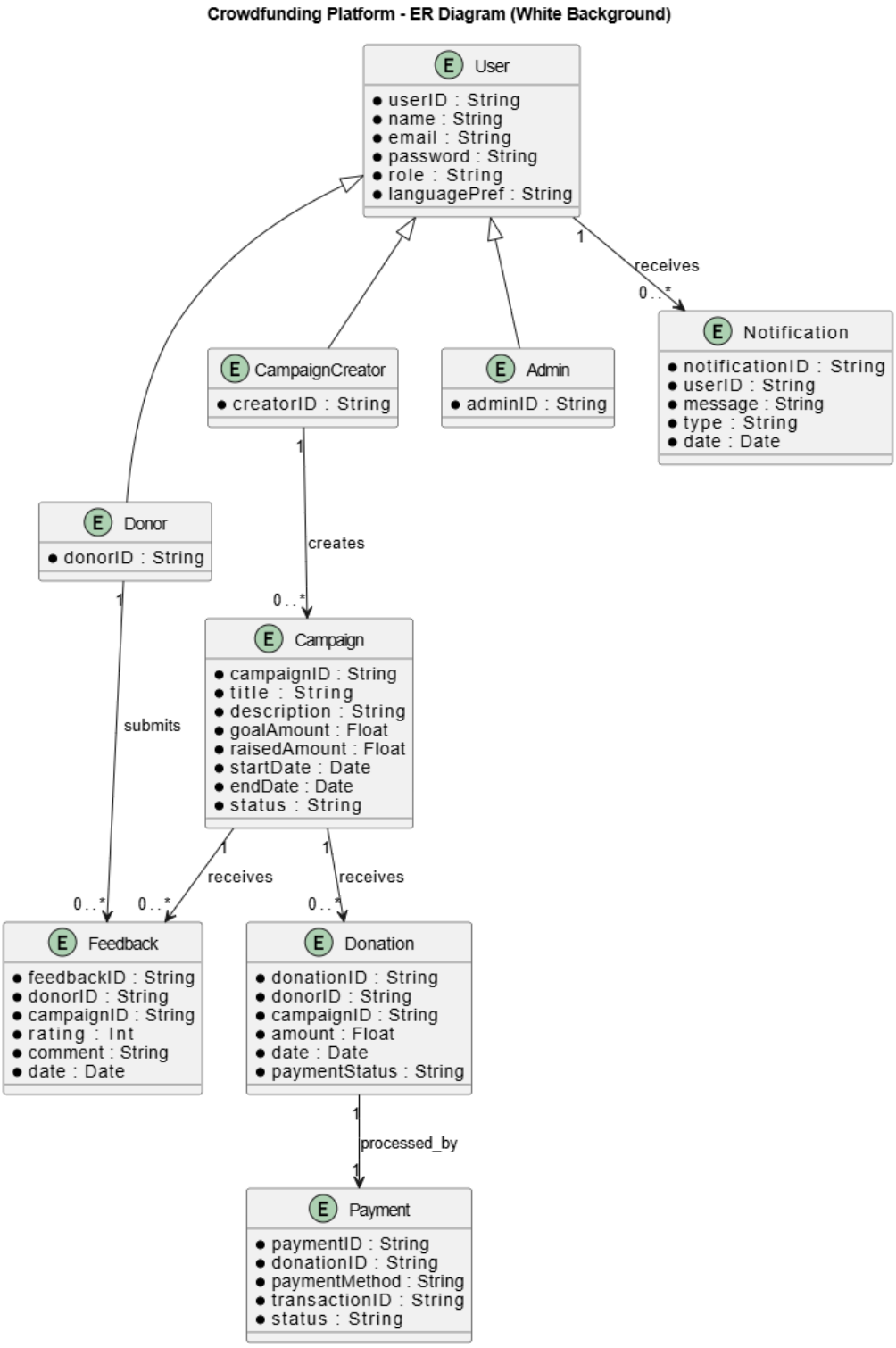


Figure 6: ER Diagram

2.6 Summary

This chapter provides: A detailed description of the full design and implementation aspects of the Crowdfunding Platform. It specified the system's functionalities, presented its logic by means of UML diagrams and implemented its data structure with Class Diagrams and E-R Diagrams. These design principles help with a clear understanding of the interaction between users and the platform, as well as what is happening to data in the system.

Support those examples in the Appendix A with some real code for building the simplest features. Overall, the chapter gives a neat and comprehensive introduction to the system's design, and provides an easy transition for development and testing in later chapters.

Chapter 3 Software Testing

3.1 Introduction

Software testing makes Crowdfunding and Donation Management Platform works properly, reliably and safely before it is prepared for its use. As the platform is handling confidential data of donors, campaigns and online payment, the testing is oriented towards functionality, security, usability and performance as next step.

In this regard, this chapter presents the key testing concepts and operational practices like important testing features, testing types, strategies for executing tests as well as system test cases. This testing will help us to confirm that components such as like User Registration, User Login Page, Campaign Creation now flows Donation Processing Admin/Reviewer Panel & Payment Confirmation are free from any bugs and behaving the right way.

3.2 Testing Features

3.2.1 Feature to Be Tested

1. User Registration

- Users must be able to create an account with valid information.

2. User Login

- Registered users must log in securely using their credentials.

3. View Campaign List

- Users can see all active, pending, and completed donation campaigns.

4. Campaign Details View

- Users can open a single campaign and view description, goal amount, progress bar, organizer info, and images.

5. Donation Submission

- Users can donate by submitting a form with name, phone, email, and amount.
- System redirects to **SSLCommerz / Payment Gateway**.

6. Payment Confirmation

- Payment success/fail response stored in the database.

7. Campaign Creation (Organizer)

- Campaign creators can create new campaigns with title, category, images, budget, description, etc.

8. Campaign Edit / Update

- Organizers can update the previously created campaign.

9. Campaign Delete

- Organizers can delete their campaigns.

10. Admin Login

- Admin can securely log in to the dashboard.

11. Admin Dashboard

- Admin can view all campaigns, donors, and users.

12. Admin Review Process

- Admin can **Approve, Reject, or Hold** campaigns.

13. User Logout / Admin Logout

- Secure logout process must terminate all active sessions.

14. Responsive UI Testing

- Ensure all pages work smoothly on mobile, tablet, and desktop.

15. Database Connectivity Testing

- All inputs and payments must be saved successfully.

3.3 Testing Strategies

3.3.1 Test Approach

The testing methodology of this project is the following: System Testing + Functional Testing + Black Box Testing. All the modules were tested separately and later on integrated for complete works compatibility.

Key strategies include:

- **Unit Testing:**
  Every function and UI control (like API endpoints, UI forms and validation logic) was tested in complete isolation.
- **Integration Testing:**
  The user interface form is linked to backend APIs and databases, hence all data flow was seamless from end-to-end.

- **Black Box Testing:**

  Testers provided inputs and analyzed outputs without accessing code.

- **Functional Testing:**

  All required operations such as registration, login, donation, campaign approvals, etc. were validated.

- **Usability Testing:**

  Checking whether users can easily understand the interface.

- **Security Testing:**
  - Password hashing
  - JWT authentication
  - Payment gateway verification
  - Prevention of invalid input attacks

- **Performance Testing:**

  The system's response time and behavior under load were observed.

3.3.2 Pass/Fail Criteria

Table 7: Testing Strategies

Criteria	Description
Pass	When expected output matches the actual output, input validation works properly, database saves correctly, and no errors occur.
Fail	When the system crashes, invalid output appears, the database does not update, input validation fails, or any error interrupts execution.
Critical Fail	Payment gateway issues, registration/login failure, or admin approval failure.

3.4 System Testing (Test Cases with Report)

Table 8: System Testing

Test Case ID	Feature Objective	Preconditions	Test Steps	Expected Result	Actual Result	Status
TC-01	Campaign Creator Registration	None	1. Navigate to Registration 2. Enter valid details 3. Submit	Account created & confirmation message shown	Account created successfully	Pass
TC-02	Campaign Creator Login	Valid creator account must exist	1. Go to login page 2. Enter email & password 3. Login	Dashboard loaded	Login successful	Pass
TC-03	Admin Login	Admin account required	1. Open admin login 2. Enter valid credentials	Admin dashboard displayed	Login successful	Pass
TC-04	Campaign Creation	Creator must be logged in	1. Click Create Campaign 2. Fill fields 3. Submit	Campaign saved as Pending	Created & pending	Pass
TC-05	Campaign Edit	Campaign exists	1. Open dashboard 2. Click Edit 3. Update 4. Save	Campaign updated	Updated successfully	Pass
TC-06	Campaign Delete	Campaign exists	1. Click Delete 2. Confirm	Campaign removed	Deleted successfully	Pass
TC-07	Admin Approves Campaign	Pending campaign must exist	1. Admin → Pending list	Status = Approved	Approved successfully	Pass

			2. Approve			
TC-08	Admin Rejects Campaign	Pending campaign	1. Open campaign 2. Reject	Status = Rejected	Rejected successfully	Pass
TC-09	View Approved Campaigns	Approved campaigns available	1. Visit homepage 2. View campaigns list	Approved campaigns displayed	Displayed correctly	Pass
TC-10	View Campaign Details	Campaign exists	1. Click campaign card	Full details displayed	Works correctly	Pass
TC-11	Donation Form Submission	Campaign must exist	1. Open campaign 2. Enter info 3. Donate	Redirect to SSLCommerz	Redirect successful	Pass
TC-12	SSLCommerz Payment	Valid donation form	1. Enter payment details 2. Pay	Payment success message	Payment completed	Pass
TC-13	Donation Confirmation Email	Successful payment	1. Complete donation 2. Check email	Email received	Received	Pass
TC-14	Donation Tracking	At least one donation exists	1. Dashboard 2. Open donations	Donor info visible	Works correctly	Pass
TC-15	Logout	Logged-in user	1. Click Logout	Redirect to homepage	Works	Pass

3.5 Summary

This chapter summarizes testing of the most important functions and verifies that all basic functionalities are behaving as expected and the system can be sent to production.

Chapter 4 Deployment and Maintenance

4.1 Introduction

Deploying and maintaining the Crowdfunding Platform When you deploy a system continuously, it will be delivered to end users in a stable, secure and optimised state. After successful testing, it was then launched on the production host and available for real user to use. In this chapter, the way in which the system is deployed and how it gets updated maintaining the software after the SRLC, are detailed.

4.2 Try to follow the SRLC (software release life cycle)

1. Pre-Alpha Stage

- Initial development started
- Basic UI and backend structure created

2. Alpha Release

- Core features implemented: registration, login, campaign creation
- Internal testing begins

3. Beta Release

- Donation module, admin approval, dashboard added
- External testing with sample users
- Bug fixes performed

4. Release Candidate (RC)

- All major bugs resolved
- System becomes stable
- Prepared for deployment

5. Production Release

- Website deployed to hosting/server
- Real donors and campaign creators start using the system

6. Maintenance Phase

- Fix bugs reported by users
- Add new features (e.g., mobile app, advanced analytics dashboard)
- Optimize performance

Chapter 5 User Manual

5.1 Introduction

This user manual explains how to use the Donation and Crowdfunding Management System. It describes the basic features of the system such as Admin activities, user operations, and navigation throughout the system.

5.2 Project Functionalities

5.2.1 Homepage

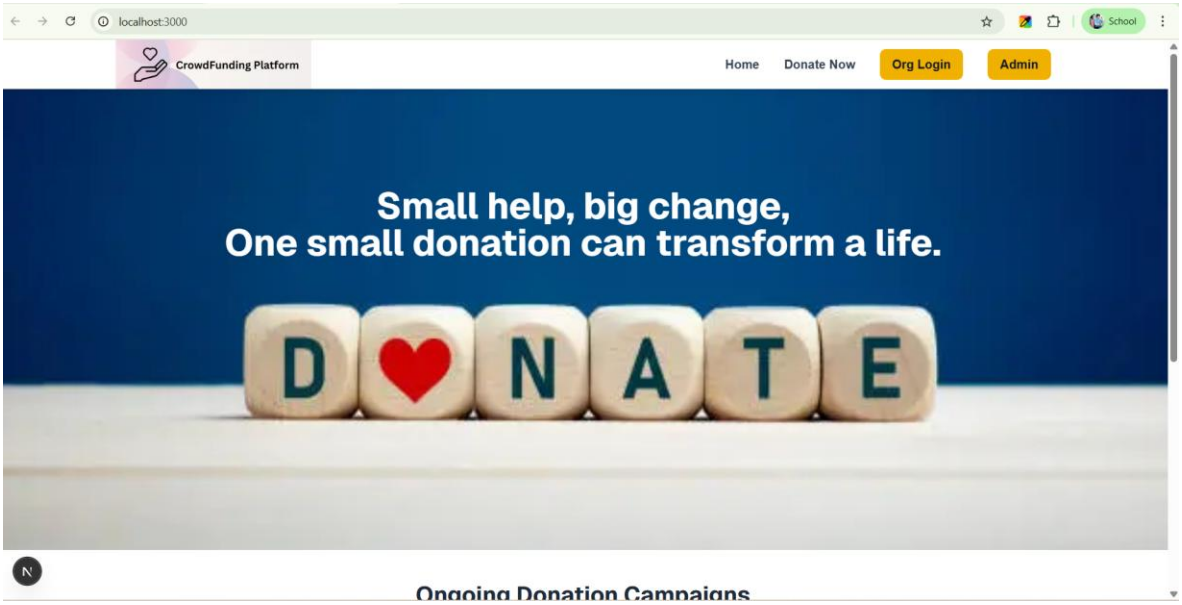


Figure 7.1: Homepage

5.2.2 SignUp for Campaign creates

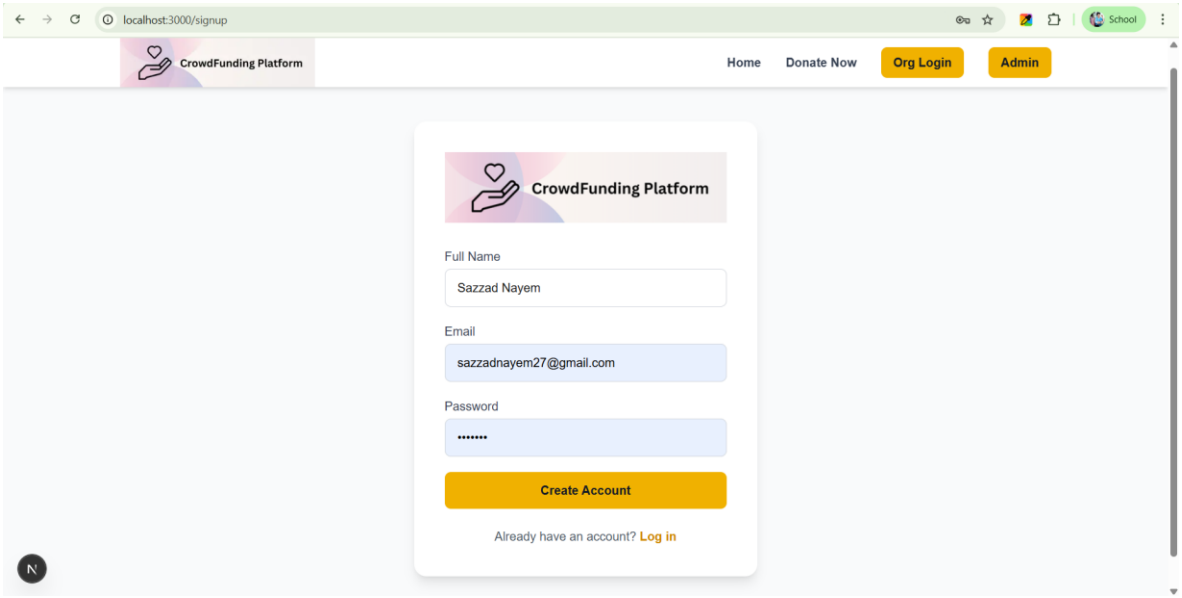


Figure 7.2: SignUp

5.2.3 Login for Campaign creates and Admin

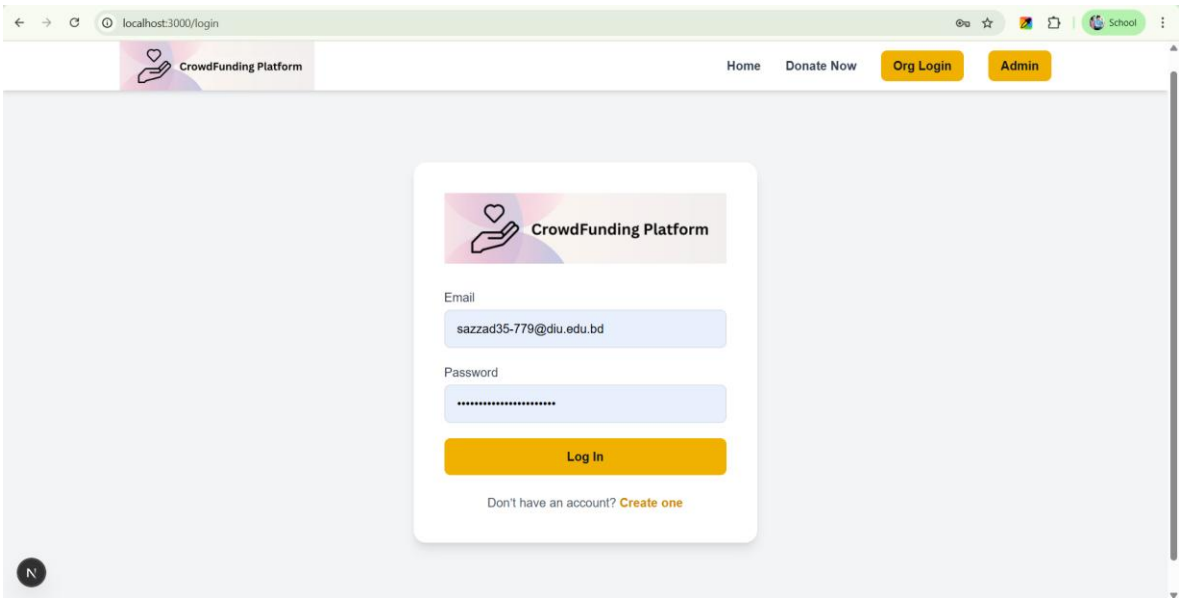


Figure 7.3: Login

5.2.4 Create a New Campaign

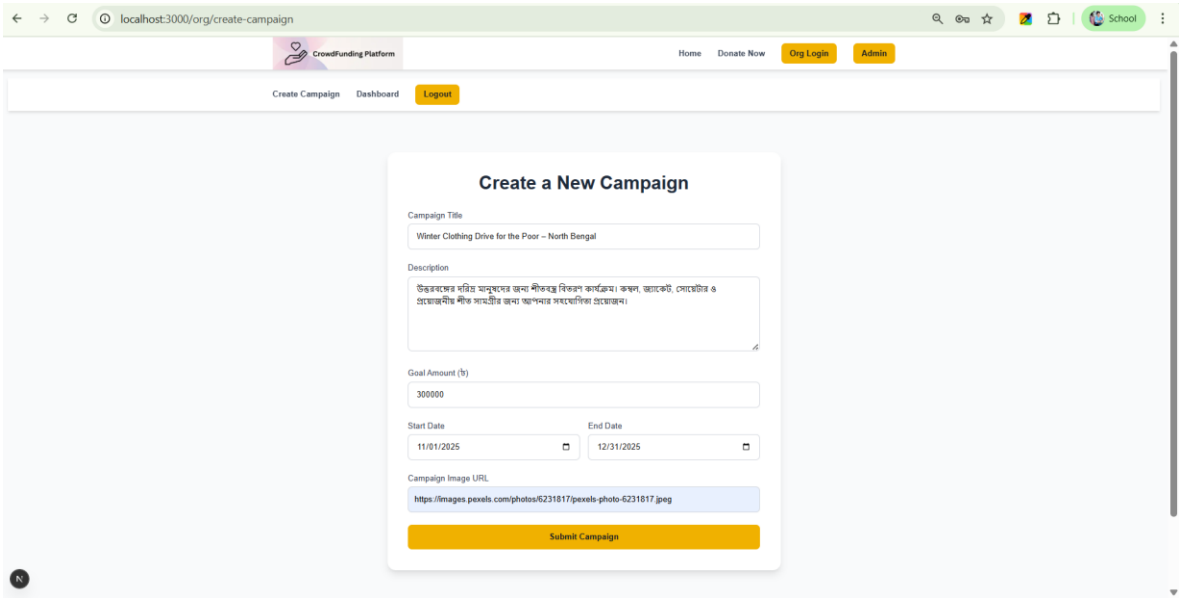


Figure 7.4: Create-campaign

5.2.5 Manage-campaign Dashboard

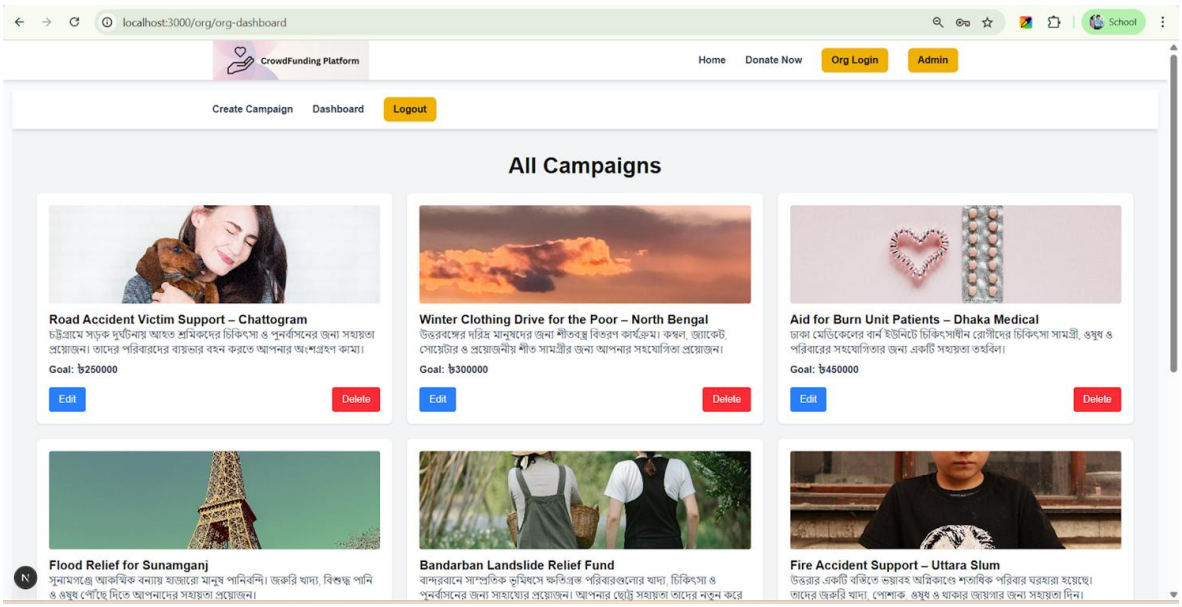


Figure 7.5: Manage-campaign Dashboard

5.2.6 Manage-campaign Update

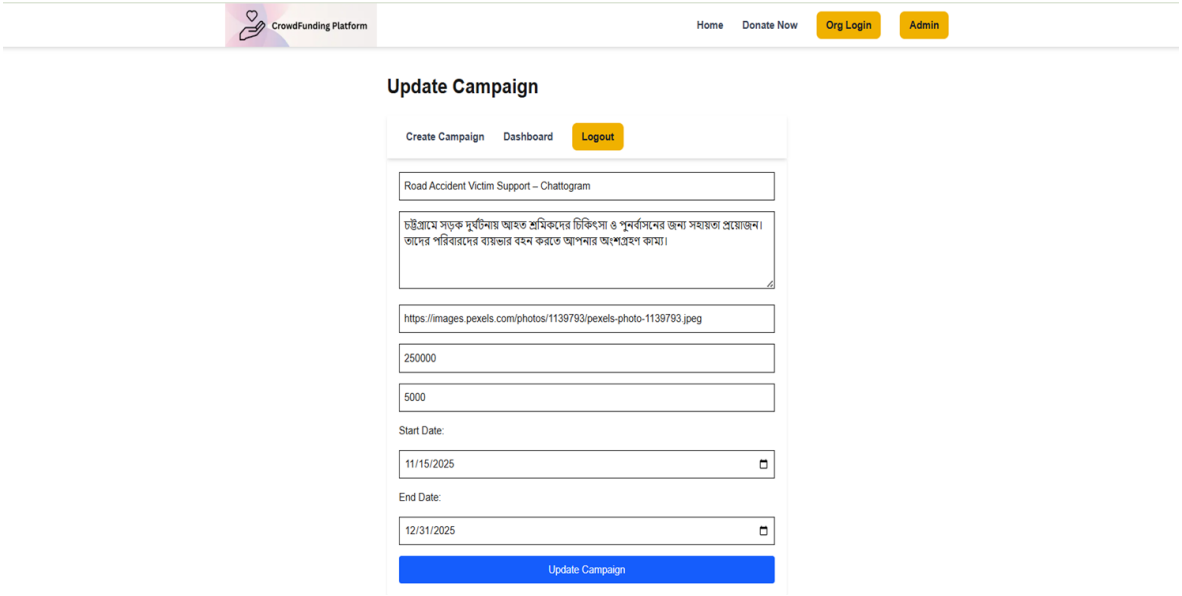


Figure 7.6: Manage-campaign Update

5.2.7 Admin Dashboard Pending Campaign Approvals

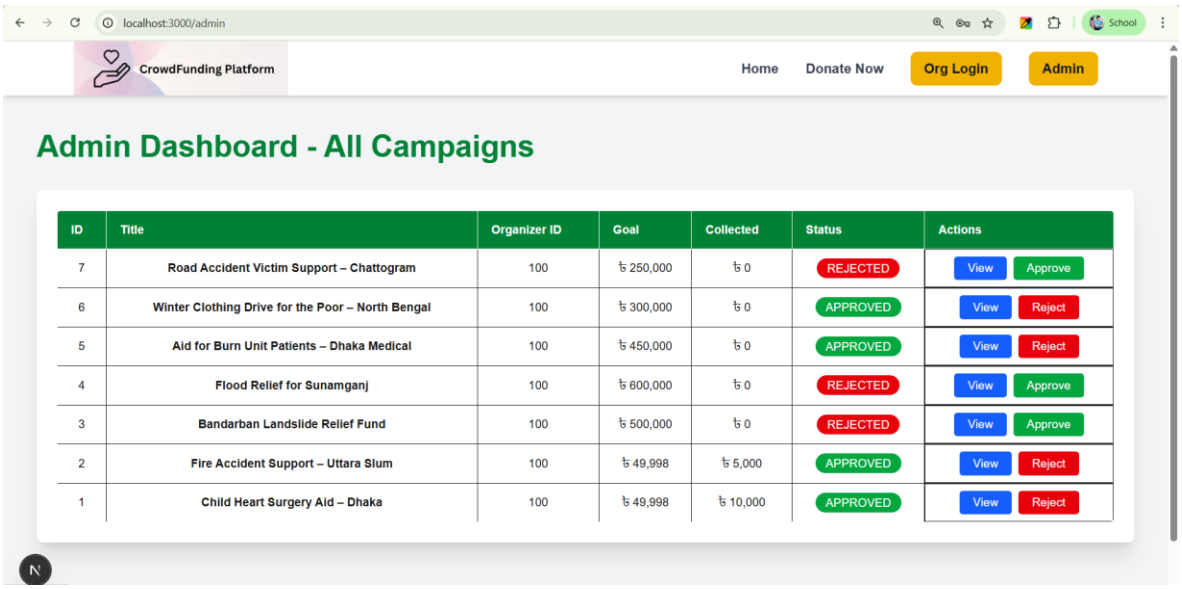


Figure 7.7: Admin-dashboard

5.2.8 Ongoing Donation Campaigns in Homepage

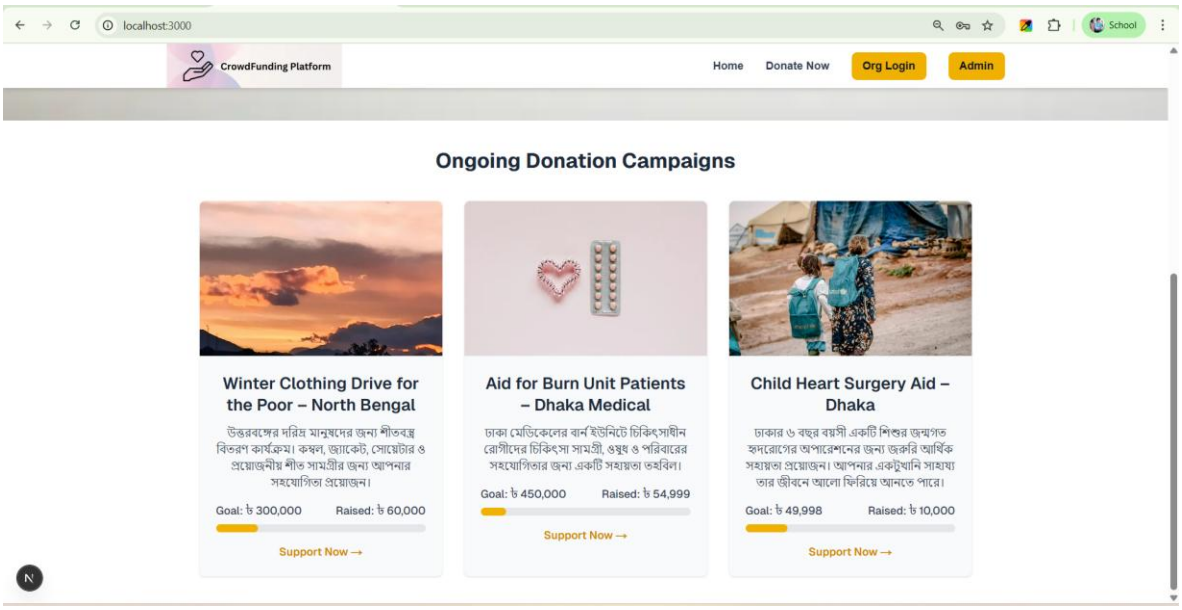


Figure 7.8: Homepage

5.2.9 Donate Now Ongoing Campaigns

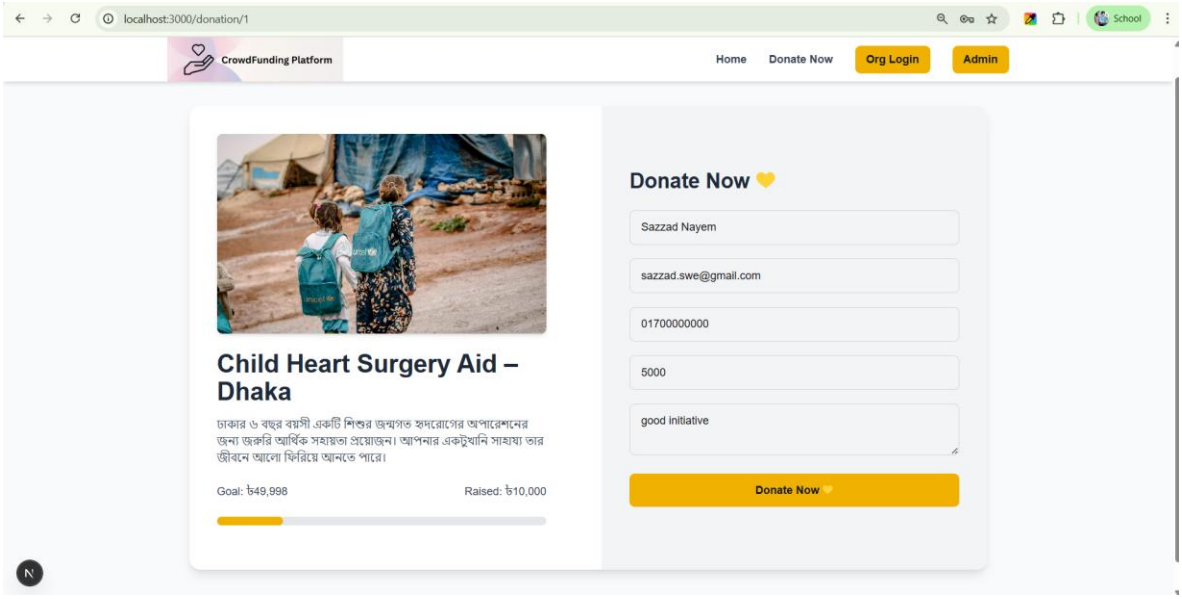


Figure 7.9: Donation

5.2.10 Donation (SSL Commerz)

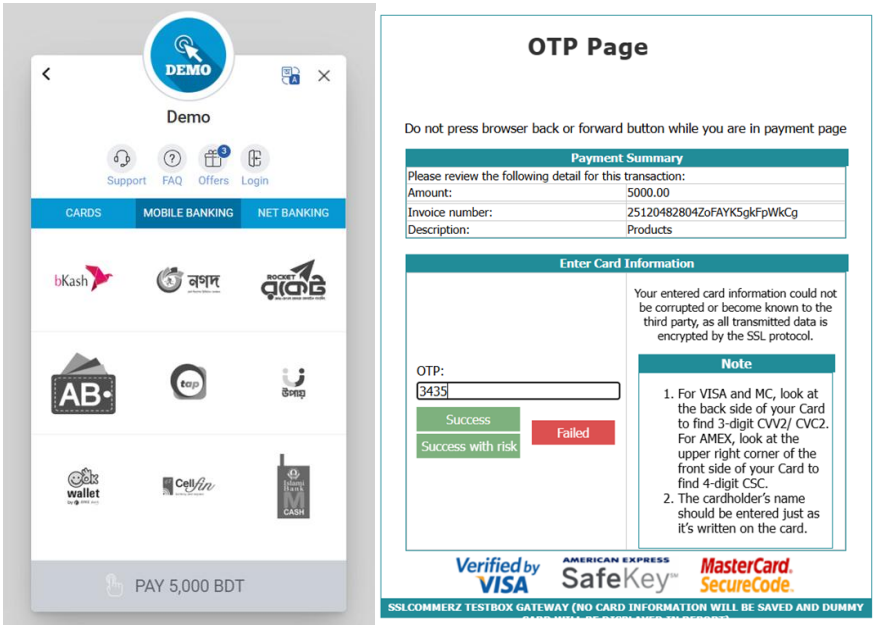


Figure 7.10: Donation (SSL Commerz)

5.2.11 Admin-dashboard View

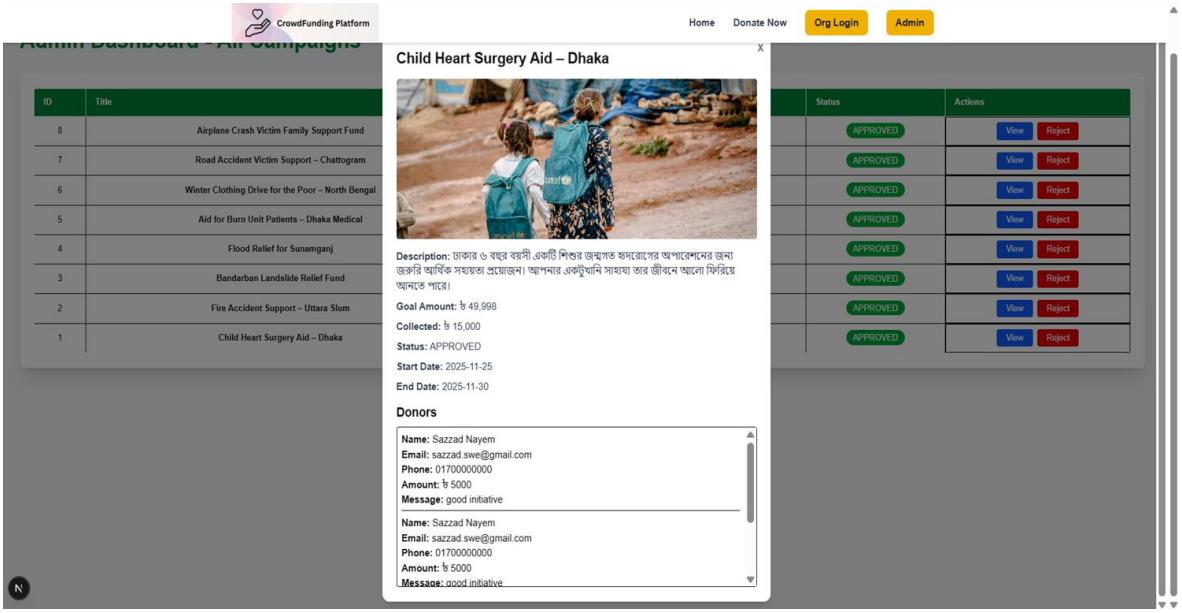


Figure 7.11: Admin-dashboard View

Figure 7: Project Functionalities

5.3 Summary

The instructions to the user necessary for using the Donation and Crowdfunding System were described in this chapter. It also provides instructions for Admins and typical users, such as information on login, managing campaigns, the process of making a donation and system navigation. The purpose of these directions is to make the use and interaction with the system be a relative one.

Chapter 6 Project Summary

6.1 Introduction

The chapter presents the complete work, its limitations and scopes for further enhancements. And it gives an end statement from the project process and results.

6.2 Project Limitation

Despite successfully implementing the core features, the project has the following limitations:

- **Time Constraints:**
  Not all advanced features (like AI-driven fraud detection, automated verification, etc.) could be completed.

- **Budget Constraints:**
  The system was developed without access to paid APIs or premium hosting services.

- **Technical Limitations:**
  - Payment integration supports only limited gateways (e.g., SSLCommerz).
  - No advanced analytics dashboard for campaigns.
  - Real-time notification system not fully implemented.

- **Incomplete Features:**
  - Email/SMS automation partially implemented.
  - Donor verification system limited.
  - Campaign creator identity verification not fully integrated

6.3 Scope

The scope of this project includes:

Included in the Project

- User Registration and Login
- Admin Dashboard
- Create, Approve and Manage Donation Campaigns
- Payment integration using SSLCommerz
- Donor Information Collection

- Campaign listing and details
- Admin controls for monitoring donations
- Secure authentication system

Excluded from the Project

- AI-based fraud detection
- Cross-country donation support
- Mobile Application
- Multi-language support
- Live chat between donors and campaign creators
- Blockchain-based transparent funding

6.4 Future Work

To enhance the system in future development phases, the following improvements can be considered:

- Implement mobile applications (Android/iOS).
- Add AI-based fraud campaign detection.
- Introduce campaign popularity analytics dashboard.
- Add Feedback & Review System
- Add Admin Dashboard & Reporting
- Add Payment Dashboard
- Add multi-language support.
- Use Blockchain to ensure transparent donation flow.
- Add real-time chat with campaign creators.
- Add more payment gateways (Visa, Mastercard, PayPal, Stripe).
- Automated donor certificate generation.

6.5 Conclusion

Donation and Crowdfunding Management System has effectively achieved all of the stated objectives as it serves as a platform where users can easily create campaigns, make donations in secure way and manage their fundraising activities. The platform provides ease of use, secure payment processing and great admin workflow.

Webxtrak is a project in which significant learning was acquired in web app development, paymentapi integration, database management and system design. Professional and scalable The platform is upgraded with new technologies, features can be added on request to provide a professional and scalable set of tools.

REFERENCES

1. Oporajoy. (2023). *Oporajoy: Crowdfunding platform for social causes in Bangladesh.* Retrieved from https://oporajoy.org

2. Cholo Shobai. (2023). *Online crowdfunding platform for humanitarian projects in Bangladesh.* Retrieved from https://choloshobai.com

3. GoFundMe. (2023). *Global donation-based crowdfunding platform.* Retrieved from https://www.gofundme.com

4. Kickstarter. (2023). *International platform for creative project funding.* Retrieved from https://www.kickstarter.com

5. Indiegogo. (2023). *Flexible crowdfunding platform supporting worldwide projects.* Retrieved from https://www.indiegogo.com

Accounts Clearance