MAKERERE UNIVERSITY

COLLEGE OF COMPUTING AND INFORMATICS SCIENCE

DEPARTMENT OF COMPUTER SCIENCE

Technical Report

Charify Crowdfunding Mobile App

Prepared by

Group Name: Group 26

RONALD ATUHAIRE	**22/U/3003/EVE**	**2200703003**
EDWARD KABOGGOZA	**22/U/20894/EVE**	**2200720894**
SSEMAGANDA GEORGE	**22/U/3847/EVE**	**2200703847**

Mentor: **Engineer Bainomugisha**

Course: **CSC 1304 Practical Skills Development**

Date: **19-August-2023**

Contents

Abstract

Charify" is a fictional term that we created by combining the words **"Charity"** and **"amplify."** The purpose of suggesting **"Charify"** as a startup name for a Charity digital platform is to evoke a sense of charitable action and the idea of amplifying the impact of charitable endeavors through technology.

The **Charify** app is a transformative digital platform designed to revolutionize charitable endeavors in Uganda. Fueled by the pressing need to address challenges faced by charitable organizations and individuals, **Charify** emerges as a comprehensive solution.

1 Introduction

In a country where the spirit of giving is deeply rooted, the Charify app emerges as a beacon of innovation. Charitable organizations and individuals have long encountered obstacles in effectively gathering support and creating impactful campaigns due to limited access to technology, cumbersome donation processes, and challenges in showcasing their endeavors. These limitations have hindered their potential to effect meaningful change within their communities.

This project report serves as a comprehensive documentation of the Charify app's journey from conceptualization to realization.

1.1 User Challenge

Charitable organizations and individuals in Uganda currently face several challenges in efficiently receiving donations and creating impactful campaigns due to limited access to technology, inadequate platforms, and difficulties in showcasing their locations. These challenges hinder their ability to reach a wider audience, gather support, and maximize the impact of their charitable endeavors.

Existing donation channels lack convenience and accessibility, leading to a need for a platform that integrates with major mobile network operators' APIs (MTN and Airtel) for seamless mobile money donations.

The Charify app addresses these user challenges by providing a comprehensive solution that integrates APIs, showcases locations through Google Maps, facilitates campaign creation, and ensures user-friendly design for inclusivity and accessibility. Through this, the app aims to empower charitable organizations and individuals, enabling them to overcome these challenges and make a more significant positive impact in their communities.

1.2 Project Goals

1. Visibility of Charitable Organizations and Needy Individuals:
- Utilize Google Maps integration to showcase the locations of organizations and individuals in need, fostering transparency and localized support.

2. Improved Donation Channels:
- Integrate both MTN API and Airtel API to facilitate seamless and convenient mobile money donations.

3. Campaign Creation and Management:
- Enable both organizations and individuals to create campaigns with detailed information, multimedia content, goals, and updates to engage donors effectively.

4. User Accessibility and Localization:
- Design a user-friendly app that supports local languages, optimized for low-bandwidth networks, and provides intuitive navigation for inclusivity.

1.3 Functional Requirements

User Authentication and Profiles:

- Users should be able to create accounts and log in using email, mobile number, or social media accounts.
- Users should have profile pages where they can view and update their personal information.
- Users can reset their passwords through a secure process.

Donation and Payment:

- Users should be able to donate funds to campaigns using MTN and Airtel mobile money transfers.
- The app should provide a seamless payment process with clear instructions.
- Users should receive confirmation of their donations via notifications and emails.
- Transaction history and donation details should be accessible to users in their profiles.

Campaign Creation and Management:

- Charitable organizations and individuals should be able to create campaigns with details such as campaign title, description, goals, and multimedia content.
- Campaign creators can set fundraising targets and track progress.
- Campaigns should display updates and stories about their impact.
- Creators should be able to edit, pause, or close their campaigns.

Google Maps Integration:

- Campaigns with location data should be displayed on a map using Google Maps API.
- Users should be able to view campaign locations and find causes near them.

Analytics and Reporting:

- Charify administrators should have access to analytics and reports about user engagement, donation trends, and campaign performance.
- Users should receive insights about their campaign's progress and donation history.

Localization and Accessibility:

- The app should support multiple languages, including local dialects.
- The user interface should be designed with accessibility features for users with disabilities.
- The app should be optimized for low-bandwidth networks to ensure usability in areas with limited connectivity.

Notification Services:

- Users should receive real-time notifications for campaign updates, interactions, and important app-related events.
- Users can manage their notification preferences.

Security and Privacy:

- Users' personal and payment information should be securely stored and transmitted.
- The app should comply with data protection regulations and allow users to control their data.

Admin Panel:

- Admins should have a dedicated panel to manage user accounts, campaigns, and overall app settings.

2 Project Results

2.1 Product Design

< TODO*: Provide a system architecture and design of your product. Diagrams and description are required*>

System Architecture:

The Charify system architecture consists of several key components that work together to create a seamless and effective Charify platform. Here's an overview of the architecture:

1. Payment API:
-This was implemented using Flutter wave endpoints by using a python/flask REST API that was hosted on render. (https://charify-api.onrender.com)

2. User Interface (UI):
- This includes screens for donation, campaign creation, browsing campaigns, user profiles, and settings.
- The UI is designed to be user-friendly, accessible, and responsive on various devices.

3. Client-Side Application:
- The client-side application handles the presentation logic and user interactions.
- It communicates with the server-side components to fetch and send data.
- The client-side also includes local storage for offline capabilities.

4. Server-Side Application:
- The server-side application manages core business logic, authentication, and database interactions.
- It validates user requests, processes payments, and manages campaign data.
- The server-side ensures data security and performs necessary API calls to external services.

5. API Integrations:
- Charify integrates with external APIs: MTN API for mobile money transfers, Airtel API for payments, and Google Maps API for location visualization.
- The APIs are used to securely transfer funds, process transactions, and display campaign locations on maps.

6. Database:
- The database stores user profiles, campaign details, donation history, and other relevant data.
- It ensures data integrity, consistency, and enables efficient data retrieval.

7. Authentication and Security:
- The system implements authentication mechanisms to ensure secure user access.
- Encryption is applied to sensitive data, such as payment information and user credentials.
- User roles and permissions control access to different app features.

8. Notification Services:

- Notification services send real-time notifications to users about campaign updates, donation confirmations, and important app-related events.

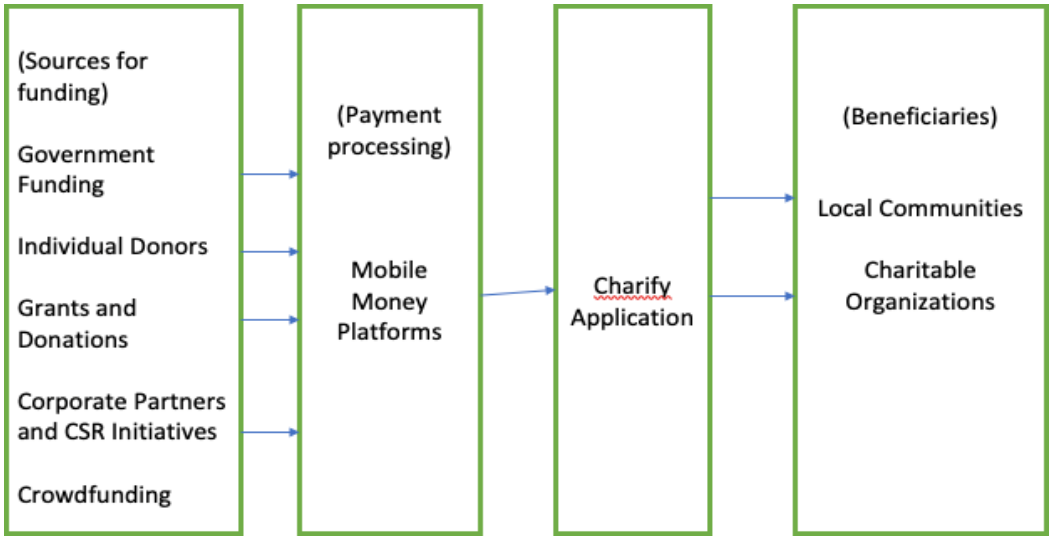
9. Analytics and Reporting:

- Analytics tools gather data on user interactions, donation trends, and campaign performance.
- Reports provide insights to users and administrators for decision-making.

System Components Interaction:

1. A user interacts with the UI/UX of the Charify app, which communicates with the client-side application.
2. The client-side application sends requests to the server-side application for data retrieval, user authentication, and transaction processing.
3. The server-side application processes user requests, interacts with the database for data storage/retrieval, and communicates with external APIs for payment processing and location data.
4. Google maps API helps to find locations data.
5. The external APIs (MTN, Airtel, Google Maps) are accessed securely through the server-side application for donation transactions and location display.
6. User authentication is managed by the server-side application, ensuring secure access to user-specific data.
7. Notifications are sent to users through the notification services based on campaign updates, donation confirmations, etc.
8. Analytics tools gather data from user interactions and campaign activities, providing insights through reporting features.

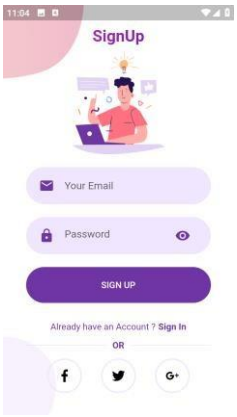
In summary, the Charify system architecture comprises user interfaces, client and server-side applications, external API integrations, a database, authentication mechanisms, notification services, and analytics tools. This design ensures a user-friendly, secure, and efficient Charify platform that fulfills the project's objectives.



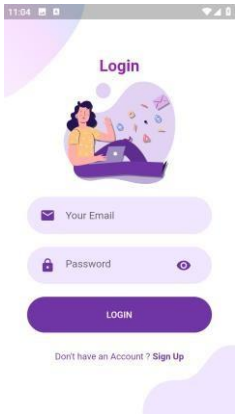
2.2 Product Functionality and Screenshots

Flash screen

Welcome Screen

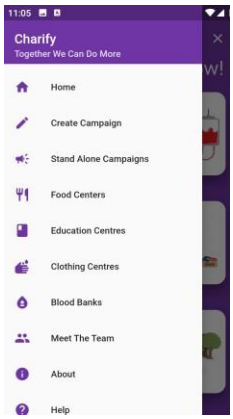


Sign Up Page

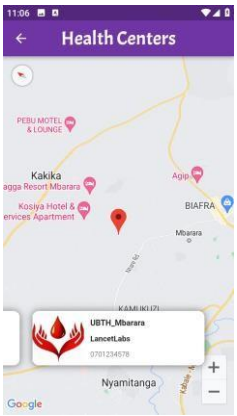


Login Page

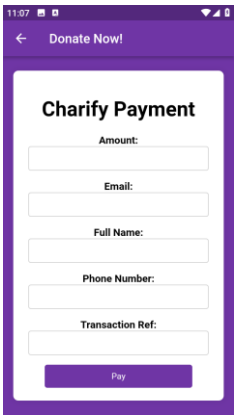
Home Page



Menu Drawer



Google Maps



Payment Page

2.3 Project website and repository

Project repository: https://github.com/Charify-Uganda/Charify

Project Portifolio website: https://charifyug.onrender.com/

Project Site: https://charify-web.onrender.com/

OTHER

Project Video:

https://www.youtube.com/watch?v=WeAm5XM7Ot4

Project Notion:
https://www.notion.so/3af223be6b5d483894320e3211291245?v=648b16a8c6a64faebd7793148547b777

REST API:

https://charify-api.onrender.com

3 Limitations and Next Steps

3.1 Limitations

Limitation 1: Technical Challenges with API Integration
Integrating multiple APIs (MTN, Airtel, Google Maps) can be complex and might lead to compatibility issues or unexpected errors.

Potential Solution:
- Thoroughly research and understand the APIs before integration.
- Develop a testing environment to simulate API interactions before implementing them in the app.
- Use proper error handling and logging mechanisms to identify and resolve issues quickly.
- Consider using third-party libraries or SDKs that simplify the integration process.

Limitation 2: Network Infrastructure
In some areas of Uganda, internet connectivity might be limited, leading to slow app performance or difficulties in using the app.

Potential Solution:
- Design the app to be responsive and lightweight, optimizing it for low-bandwidth connections.
- Implement offline capabilities where users can access certain functionalities even without an internet connection.
- Provide clear instructions on how to use the app with slow connections and offer tips for improving performance.

Limitation 3: Privacy and Security Concerns
Handling sensitive user data (like mobile money transactions) requires stringent security measures to protect user privacy.

Potential Solution:
- Implement strong encryption protocols for data transmission and storage.
- Comply with relevant data protection regulations and obtain necessary permissions from users.
- Regularly conduct security audits and vulnerability assessments to identify and address potential threats.

Addressing these limitations through careful planning, ongoing monitoring, and adaptive strategies can contribute to the success and sustainability of the Charify project.

3.2 Next Steps

- Expand Payment Methods: Continuously update and expand the platform's payment methods to accommodate emerging preferred payment options in Uganda.
- Regional Scaling: We consider scaling the platform to address challenges faced by charitable organizations and individuals in the neighboring countries.
- Blockchain for Transparency: Implement blockchain technology to enhance transparency and accountability in fund utilization and project progress.

- Localized Content and Languages: Expand multilingual support to include additional local languages.

- Machine Learning and AI Integration: Integrate machine learning algorithms to provide personalized recommendations to donors, suggest impactful campaign strategies and analyze trends in donor behavior.

- Ability to start a campaign by the users not only by the admins.

- Offline Capabilities: The app should have certain functionalities available even when users are offline, allowing them to interact with campaigns and update their profiles.

- Planning to use Mango DB

References

We Used APA Citation format!

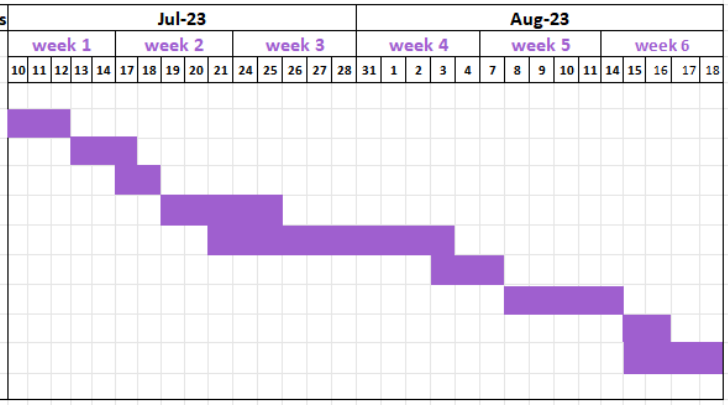
- ***GoFundMe*** *https://www.gofundme.com/en-gb*
- ***Flutter documentation*** *https://docs.flutter.dev/*
- ***Momo API*** *https://momodeveloper.mtn.com/api-documentation*
- ***REST API-hosting/PaaS*** *https:// render.com*
- ***Flutter wave documentation*** *https://developer.flutterwave.com/*
- ***Fire base documentation*** *https://firebase.google.com/docs*
- ***Dart documentation*** *https://dart.dev/guides*
- ***Fire store documentation*** *https://firebase.google.com/*

4 Appendix A – Project Work plan

<Provide project work plan that was followed. Gannt chart format is strongly recommended see https://en.wikipedia.org/wiki/Gantt_chart >

Gantt Chart for Charify Mobile App Development

ID	Task Name	Duration	Dependencies
1	Project Initiation	3 day	
2	Research and Requirements Gathe	3 days	Task 1
3	System Design and Architecture	2days	Task 2
4	UI/UX Design	5days	Task 3
5	Development	10 days	Task 3, 4
6	Testing and Quality Assurance	3 days	Task 5
7	User Testing and Feedback	5days	Task 6, 5
8	Deployment and Launch	2 days	Task 7
9	Post-Launch Monitoring and Maintenan	on-going	Task 8



Appendix B – Contribution by Team Members

No.	Team Member	Contribution
1.	Ronald Atuhaire	• Python/Flask REST API integration • Google maps API implementation • GitHub repo Admin • Flutterwave payment integration • Innovation lead
2.	Edward Kaboggoza	• Project management- Notion. • Research and development • Localisation specialist • Code reviewing and testing. • App load balancing
3	Ssemaganda George	• UI/UX designs with Figma. • Flutter screens and widgets • Project report • Project architecture and plan • User authentication