**JRM Sacco Management System**

***Best Enterprise Resource Planning (ERP) System***

***For Savings and Credit Cooperative Society (SACCO)***

## Team

|  |  |  |
| --- | --- | --- |
| **Names** | **Roles** | **Reason** |
| Julius Kinyua | Backend**,** APIs |  |
| Ruku Ubi | Backend**,** webStatic |  |
| Michael Otor | Backend, Database |  |

**Contact Details**

|  |  |  |  |
| --- | --- | --- | --- |
| **Names** | **WhatsApp No.** | **Email** | **Linked** |
| Julius Kinyua | +254703873717 | Juliusczar36@gmail.com | https://www.linkedin.com/in/julius-kinyua |
| Ruku Ubi | **+**2348129091921 |  |  |
| Michael Otor | **+**2348169953690 |  |  |

## Technologies

**Console & Backend -** Python

**Web Static** - HTML & CSS

**Database** - MySQL

**Web Framework & APIs** - Flask

**Alternative Technologies**

**Web Frameworks & APIs (Flask, Django)**

Flask is a lightweight and minimalistic web framework, offering greater flexibility and freedom to choose the preferred libraries and components, whereas Django provides a more comprehensive and structured approach, better suited for larger projects with built-in features.

The school project (JRM Sacco Management System) project requires a quick setup and simple requirements, Flask's simplicity and ease of use make it an excellent choice. In contrast, Django is more suitable for complex applications that benefit from its built-in admin panel, authentication system, and ORM.

Flask's microframework nature allows developers to have a leaner codebase and reduced overhead, making it ideal for small to medium-sized projects, while Django's batteries-included approach is preferable for enterprise-level applications with more extensive functionalities.

**Databases (MySQL, PostgreSQL)**

MySQL is favored in the Sacco Management system project for its simplicity and ease of use, which make it an attractive option for projects with straightforward requirements or when one prefers a more straightforward setup. MySQL's performance can excel in specific scenarios, particularly with read-heavy workloads such as the Sacco Management system, making it a preferred choice for applications where rapid data retrieval is crucial.

## Challenges

The Sacco management system aims to address the inefficiencies and challenges faced by Savings and Credit Cooperatives (Saccos). It intends to streamline operations, automate member registration, track savings and loans, generate reports, and enhance transparency and accountability.

However, the Portfolio Project will not solve broader economic issues or regulatory compliance challenges faced by Saccos. It focuses solely on internal management processes and does not address external market factors.

The Portfolio Project will benefit Sacco administrators, employees, and members by simplifying tasks, reducing errors, and providing real-time financial insights. Users will include administrators responsible for overseeing operations, employees managing day-to-day transactions, and members accessing their financial information.

While the project's core functionality is not tied to a specific locale, some aspects, such as regulatory compliance and reporting requirements, may vary based on the country or region where the Sacco operates. Thus, minor adaptations may be needed for different locales.

## Risks

**Technical Risks and Safeguards:**

1. **Data Security Breach:** A security breach could lead to unauthorized access to sensitive member information, compromising their privacy and potentially leading to identity theft or financial fraud. Safeguards include implementing robust data encryption, access controls, and regular security audits.
2. **System Downtime:** Unexpected system failures or downtime can disrupt Sacco operations and result in member dissatisfaction. Safeguards involve redundant servers, disaster recovery plans, and monitoring systems for early detection and prompt resolution of issues.
3. **Integration Challenges:** If the system needs to integrate with other financial institutions or platforms, compatibility issues may arise. Safeguards involve conducting thorough compatibility tests and ensuring API documentation is up-to-date.

**Non-Technical Risks and Strategies:**

1. **Regulatory Compliance:** Failure to comply with financial regulations and reporting requirements may lead to legal consequences and damage the Sacco's reputation. Strategies involve regular compliance checks, staying updated with regulations, and seeking legal counsel.
2. **Member Trust:** Poor customer service or mismanagement may erode member trust, leading to loss of business. Strategies include training staff in communication and ethical practices, implementing transparent policies, and addressing member feedback promptly.
3. **Economic Conditions:** Economic fluctuations could impact the ability of members to repay loans, affecting the Sacco's financial stability. Strategies include diversifying loan portfolios, conducting risk assessments, and offering financial counseling to members facing hardship.

## Infrastructure

**Branching and Merging Process**

Our team will follow the GitHub flow for branching and merging. We will create feature branches from the main branch (which will likely be named "master" or "main") for each new task or feature. After completing the task, a pull request will be submitted for review. The team will collaboratively review the code, ensuring code quality, functionality, and adherence to coding standards. Upon approval, the branch will be merged into the main branch.

**Deployment Strategy**

For deployment, we will use a continuous integration/continuous deployment (CI/CD) pipeline. Changes merged into the main branch will trigger an automated build and test process. Once all tests pass, the app will be automatically deployed to a staging environment for further testing and validation. After successful testing in the staging environment, the app will be deployed to the production environment, making the latest features available to users.

**Populating App with Data**

To populate the app with data, we will utilize various methods. For testing purposes, we will use mock data to simulate real-world scenarios. Additionally, during development, we will use seed data scripts to initialize the database with sample records. For production data, we will provide an interface for administrators to import member data or integrate with other systems using data migration tools or APIs.

**Testing Tools and Automation**

We will emphasize testing to ensure a robust and reliable app. For unit testing, we will use libraries like Unittest for Python. We will also conduct API testing. To automate the testing process, we will integrate testing into our CI/CD pipeline, triggering tests automatically on code changes. Continuous monitoring will help us identify and address issues promptly.

## Existing Solutions

1. **WebSacco:**
   * Similarities: [WebSacco](https://websacco.com/) provides features for member registration, savings tracking, and loan management, just like our intended system.
   * Differences: Our system aims to have a more intuitive user interface, seamless integration with financial institutions, and advanced reporting capabilities, which may differ from WebSacco.
2. **Sunafricgroup:**
   * Similarities: [Sunafricgroup software](https://sunafricgroup.co.ke/Microfinance.html) focuses on member savings and loan processing, similar to our system.
   * Differences: Our system plans to offer additional features, such as generating comprehensive financial statements and facilitating communication between administrators and members. Sunafricgroup does not have these functionalities.

**Reason for Reimplementation:**

While there are existing solutions available, we have chosen to reimplement the Savings and Credit Cooperative Society (SACCO) management system based on specific requirements unique to modern organizations. We aim to customize the system to align with Sacco's internal processes, member preferences, and regulatory obligations. Additionally, existing solutions may come with unwanted features or complexities that do not suit rising needs. Moreover, reimplementation allows us to integrate the latest technologies and development practices, enhancing the overall efficiency and user experience of the Sacco management system.