Software Requirements Specification (SRS) Document

Team Number : 10

Members:

- Gowlapalli Rohit
- Kadali Lakshmi Nirmala
- Losetti Mourya
- Vempati Siva Koti Reddy

Brief problem statement

The Airdrop Insurance initiative in the Gurue District of Mozambique aims to provide a social safety net for the financially disadvantaged through a voucher system. This system can be accessed through a mobile platform, allowing businesses & residents to purchase personal and crop insurance. The vouchers can be redeemed for the local digital currency or other goods. The program also leverages collective risk to gain stronger bargaining power in reinsurance negotiations. Our aim is to improvise the app that has been provided and to remove various privacy issues from it.Our Primary Goal is to resolve privacy issues and deploying it in google play store,.

Users profile

Members: Users who apply for insurance using digital tokens. They may not be familiar with computer or software usage. Eg: Farmers, Jobholders, etc.

Community Manager : Verifies official documents and audio recordings to grant User the access to the Airdrop Application.

Admin: Administrates the Airdrop Application.

Project Modules

User Management: It handles user accounts and information.

Reinsurance Embodiment : It is the process of carrying out and putting into action the use of reinsurance in a particular insurance program or system. It involves incorporating reinsurance as a risk management strategy. It aims to enhance the stability and sustainability of an insurance program.

Privacy Protection: It ensures that we only take the necessary permissions from the user , so that app can't violate google play store terms and conditions. Hence we can avoid problems during app deployment.

API Design for Monte Carlo Simulation: Here, we design an API for flutter for implementation of reinsurance using Monte Carlo Simulation.

App Deployment: Here, we try to find the bugs in the app and try to deploy it in the google play store.

Feature requirements (described using use cases):

- **User Management:**

  (1) User Login: This is a basic login page where users enter their details and sign in to the application.

  (2) User Registration: This is a basic registration page where users enter their details and sign up for the application.

  (3) User Verification: This is done using the coinbase account of the user.

  (4) Community managers can be elected by the other users through voting in polls. Once a sufficient number of votes are cast, a request is sent to the admin for approval to become a community manager.

  This will be done in R1.

- **Reinsurance Embodiment:**

  (1) Reinsurance Vaults: Bundled reinsurance vaults are used to sustain unprecedented demand in insurance claims.

  (2) Monte-Carlo Simulation: This simulation is utilized to model the distribution of tokens based on factors such as weather conditions, job layoff patterns and previously collected data.

  (3) Automated claim: Token distribution is automated and is dependent on many factors such as weather conditions, job layoffs etc.

This will be done in R2.

- **Privacy Protection:**

(1) Making sure all the privacy issues are resolved before deployment: Removed MANAGE_EXTERNAL_STORAGE permission from the required permissions.

This will be done in R1.

- **API Design for Monte Carlo Simulation:**

(1) Design: The API should be designed in a way that makes it easy for developers to integrate Monte Carlo simulations into their own applications and programs, while also providing robust and efficient algorithms for running the simulations.

(2) Purpose: A well-designed API for Monte Carlo simulation will provide clear documentation, easy-to-use interfaces, and flexible configuration options to ensure that it can be used in a wide variety of contexts.

This will be done in R2.

- **App Deployment:**

(1) Goes through alpha testing after the deployment. The purpose of alpha testing is to identify any major issues or bugs in the software, and to assess the software's overall functionality and usability.

(2) After this, the beta testing is done. It is a type of software testing that is performed after the alpha testing phase and before the final release of the software to the general public. The purpose of beta testing is to gather feedback from a wider group of users about the software's functionality, performance, and usability, and to identify any remaining issues or bugs that were not discovered during alpha testing.

This will be done in R1. (The Deployment)