WALLET AUTHENTICATION SYSTEM

W-A-S

Payment, Wallet and Data

TECHNICAL DOCUMENT

22nd, March 2021

Presented by
Joshua Israel.

Table of Contents

Version Control

	Name	Date	Version
Prepared by	Joshua Israel	22-03-2021	1.0

Overview

This is a formal document that aims to provide functional, technical, operational and user requirements for the wallet authentication system. The information in this document will guide the design, development and engineering, data models, integrations, risks, constraints, input and output analysis of the entire platform to ensure the platform provides value as covered in the scope.

Background

Wallet Authentication System

A digital wallet (or e-wallet) is a software-based system that securely stores users' payment information and passwords for numerous payment methods and websites. By using a digital wallet, users can complete purchases easily and quickly with near-field communications technology.

Goal

To design and develop a cloud-based API that that will:
- Provide the user to create their wallet for transactions purposes.

Assumptions and Constraints

Assumptions

The following assumptions were/are made:
- Availability of payment terminal service providers to partner and provide the foundation for the application to run on them,
- Availability of cloud services to provide hosting capabilities, security and performance standards.

Points of Contact

These are the names, titles and roles of all participants on the project.

Engineer
- Joshua Israel

Functional & Technical Requirements

These functional requirements describe the core functionality of the application. This section includes the architecture, data and functional process requirements.

Scope

The scope includes:

- The design and development of wallet authentication system.

Functional Specifications

W-A-S

W-A-S is a wallet authentication system. This system allows users to create an account, fund their account, transfer funds to another user's account and can withdraw funds from their account.

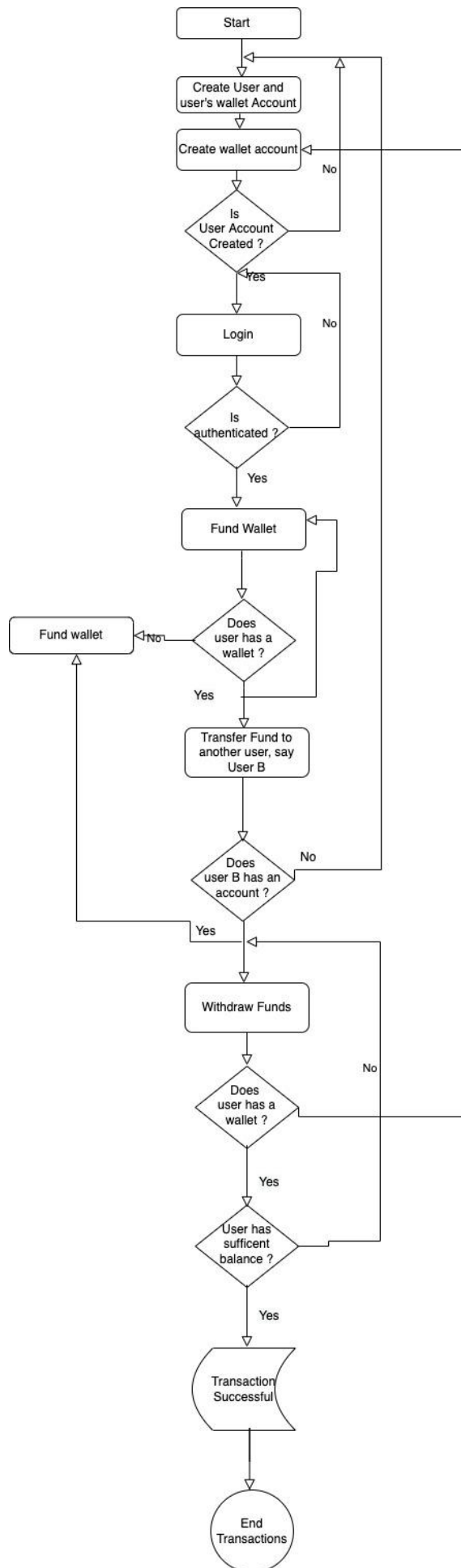| S/N	Description	Category
1	System should allow users to create their account	Functional
2	System should allow users to login to their account	Functional
3	System should allow users to fund their account	Functional
4	System should allow users to transfer funds to another user's account	Functional
5	System should allow users to withdraw funds from their account	Functional

Technologies / Programming Languagues used:
1. NodeJs
2. Express
3. Mocha
4. TypeORM
5. Docker
6. TypeScript
7. NPM

Technical Specifications
System Flow Chart

PRODUCT REQUIREMENT DOCUMENT
Functional | Technical | User Requirements