

**National University of singapore**

Architecting Scalable System

**PROJECT PROPOSAL**

(iJooz E-Wallet and QR Code Purchase)

VERSION 1.0

Kevin Chen Xiaojie (E0385016)

Zou Xuan (E0384799)

Kunal Jadhav (E0385000)

Nurdin Effendi (E0384876)

**PRIVATE & CONFIDENTIAL**

*Content of this document is provided as part of project stated above, and may not   
be used, reproduced or distributed except for the sole purpose of that project.*

Document History

|  |  |  |
| --- | --- | --- |
| **Version** | **Date** | **Comments** |
| 0.1 | 29 Mar 2019 | Draft |
| 0.2 | 6 Apr 2019 | Update |
| 0.3 | 7 Apr 2019 | Update |
| 0.4 | 9 Apr 2019 | Update |
| 1.0 | 9 Apr 2019 | Final |

Document Usage

This document is confidential to the client to whom it is presented, and is not intended for public use. It must not be reproduced or transmitted in any form or by any means, in whole or in part, without the express permission of the document owner.

The functional and design specifications contained herein summarise the features of the proposed system and the deliverables from document owner to the client under the system.

Table of Contents

[1 Overview 4](#_Toc5735767)

[1.1 About Our Sponsor (iJooz) 4](#_Toc5735768)

[1.2 Business Problems and Solution 4](#_Toc5735769)

[2 Project Proposal 6](#_Toc5735770)

[2.1 Logical Architecture 6](#_Toc5735771)

[2.2 System Context 8](#_Toc5735772)

[2.3 Logical Functional Model 9](#_Toc5735773)

[3 Project Scope and Milestones 18](#_Toc5735774)

[3.1 Scope of Work 18](#_Toc5735775)

[3.2 Milestones and Effort 19](#_Toc5735776)

# Overview

## About Our Sponsor (iJooz)

Fruits Vending Pte Ltd is a premium smart vending machines manufacturer and operator in Singapore. They specialize in providing freshly squeezed orange juice vending services to the customers with consistent, excellent quality at competitive prices.

As of March 2019, Fruits vending Pte Ltd has more than 500 “i.Jooz” smart machines island wide, and the company is planning to expand up to 1000 machines in a year and 3000 machines worldwide.

Their products and services possess strong presence in the market and their network of vending machines have expended across the island, contributing to a better world of wellness for everybody.

More information about iJooz can be found at <http://www.fruitsvending.com/about-us.html>

## Business Problems and Solution

This project will assist iJooz to establish a membership system in fullfilling the following requirements:

### Recognize iJooz Customers and Get Their Feedback

Currently iJooz unable to recognize their customers even though thousand of cups of orange juice sold daily through iJooz vending machine. They also unable to get the feedback from customers who bought the juice directly from iJooz vending machine.

With this platform, iJooz will be able to recognize their customers and improve the quality of their business based the customer data/feedback that they will get from this platform.

### Reduce Cash Payment and Promote “i.Jooz QR Code” Cashless Payment

Currently cash payment is the most desired payment mode by iJooz customers, even though the vending machine is also accepting other payment mode such as ez-link card, credit card, Alipay pay, WeChat pay, and their own “i.Jooz QR Code” payment.

“i.Jooz QR Code” payment is currently not opened for public and can only be purchased by corporate customers through offline payment mode.

With this platform, all kind of customers will be able to purchase “i.Jooz QR Code” and use it to get the juice from vending machine.

### Reward Royal Customers

Currently iJooz unable to send promotional information and able to identify their royal customers.

With this platform, iJooz will be able to identify their royal customers and give them rewards by sending them discount promotional code. All customers who registered as member in this platform will also never miss any promotional information from iJooz, because iJooz will be able to send promotional information to them easily through this platform.

### Scalability and Ability to Replicate the Business to Other Country’ Franchisers Easily

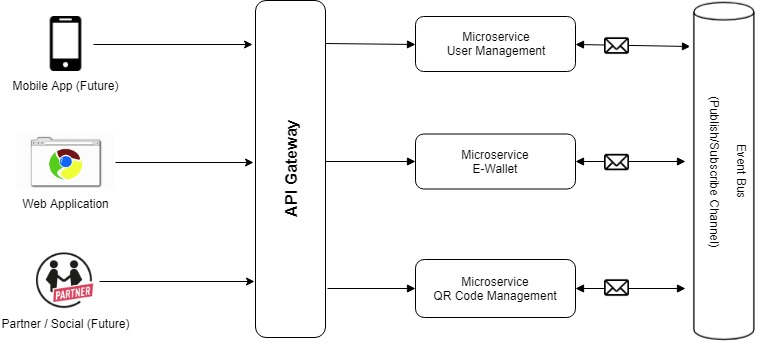
Since iJooz is planning to grow their business by adding more vending machines world wide (not only in Singapore), the platform must be able to support up to 500 thousand of members and 1000 vending machines.

In addition to that, cloning/replicating the platform to other countries should be effortless, so they can setup new business in other countries easily.

# Project Proposal

## Logical Architecture

Logical architecture of the application is described as below:



### Mobile App (Future)

We are designing the application to be accessible by mobile app in the future. However, this will not be part of the project

### Partner / Social (Future)

We are designing the application to be accessible by partner / social in the future. However, this will not be part of the project

### Web Application

Web application will be the entry point of customer to use the application in this project. Customer will be using browser such as Internet Explorer or Chrome of Firefox or any other browser to use the application

### API Gateway

This is the gateway for Web Application or Mobile App or Partner/Social to access the microservices.

### Microservice User Management

This is the microservice to manage user accounts. Details of activities of this microservice can be found at section 2.2

### Microservice E-Wallet

This is the microservice to manage the customer’s balance in E-Wallet. Activities here includes top-up and deduct E-Wallet balance. Details of activities of this microservice can be found at section 2.2

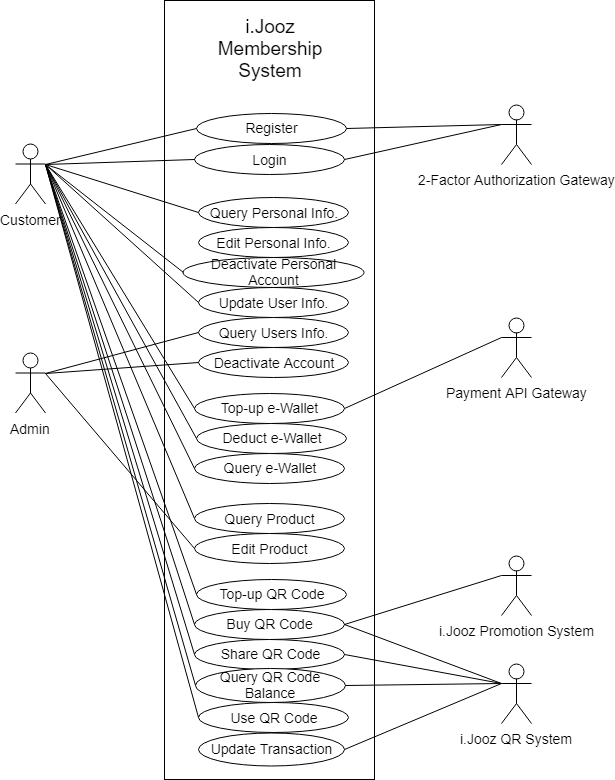
### Microservice QR Code Management

This is the microservice to manage customer's QR code. Activities here includes purchasing and using the QR code to purchase the orange juice from vending machine. Details of activities of this microservice can be found at section 2.2

### Event Bus

This is message queue to facilitate every microservice to send data to other microservices for any update that is done in each microservice.

## System Context



Application that we are going to build will have 3 major use cases and they are as follows:

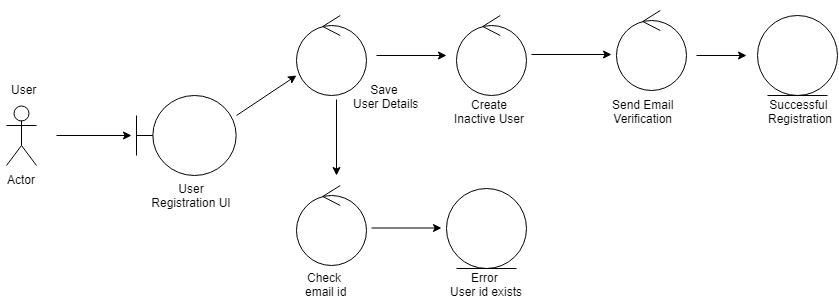
* Customer Management
  + Registration
  + Login
  + Query Personal Info
  + Update Personal Info
  + Deactivate Personal Account
  + Query User Info
  + Update User Info
  + Deactivate Account
* e-Wallet Management
  + Top-up e-Wallet
  + Deduct e-Wallet
  + Query e-Wallet
* QR Code Management
  + Query Product
  + Edit Product
  + Top-up QR Code
  + Buy QR Code
  + Share QR Code
  + Query QR Code Balance
  + Use QR Code
  + Update Transaction of QR Code

Detail of architectural of each use case will be explained in logical functional model.

## Logical Functional Model

### Customer Management

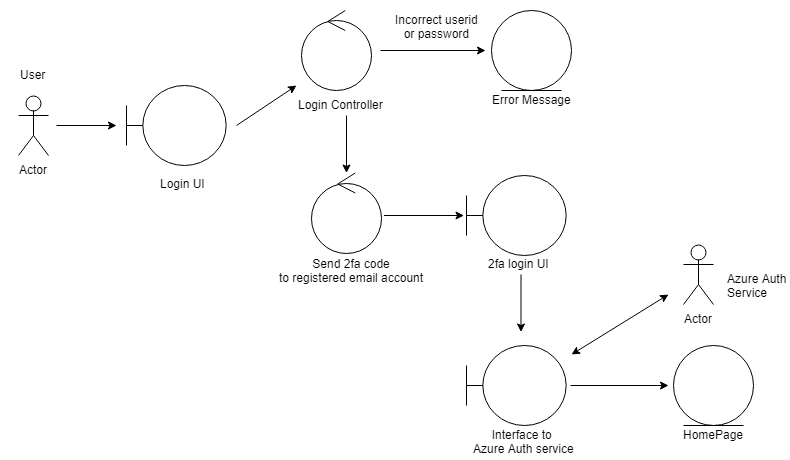
#### Registration



Process flow of event:

1. Customer go to user registration area and register themselves
2. System checks the eligibility of user registration
3. If successful, system creates user information and mark the registered user as inactive
4. System send email verification to the registered email address
5. Customer receives the email in their email platform and click activation link
6. If activation successful, system will activate the user

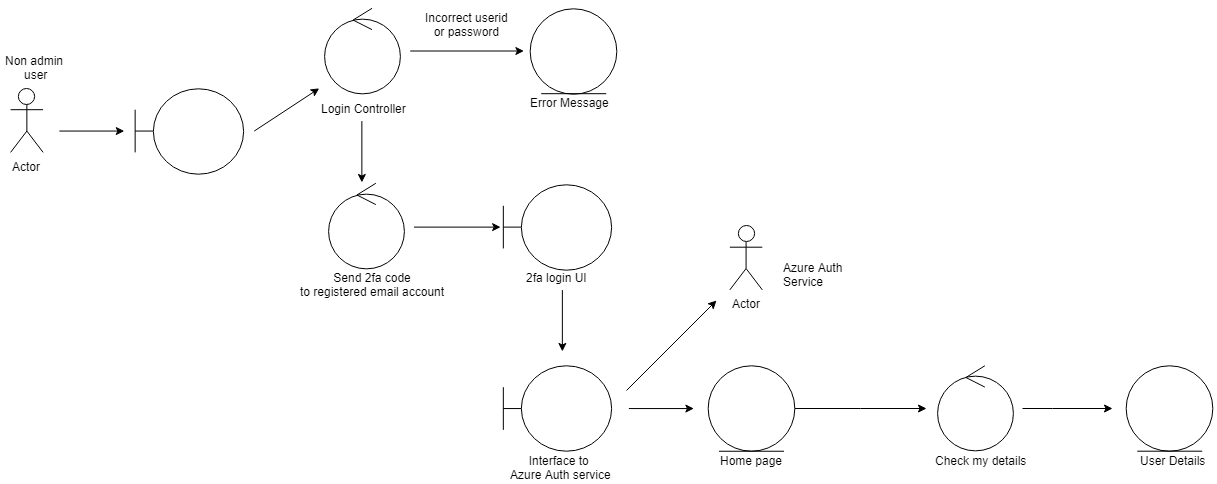
#### Login



Process flow of event:

1. Customer login through application login area
2. System validate the login and password
3. If successful, system will send 2 factor authentications through registered email account. If not, system will prompt error message
4. Customer key-in 2nd authentication
5. System validate 2nd authentication
6. If successful, customer will be redirected to application home page

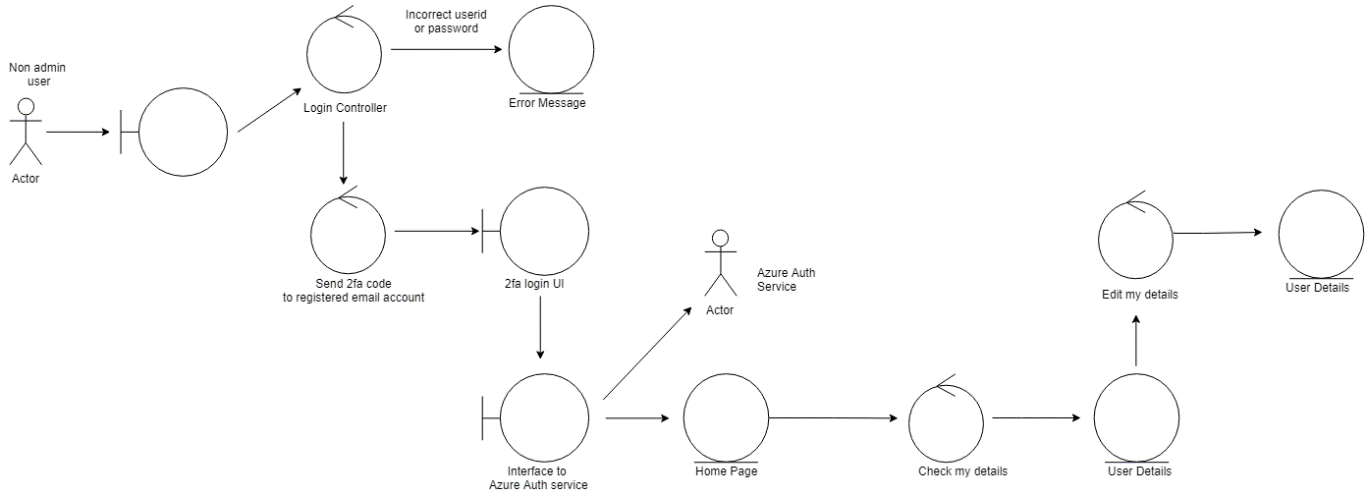
#### Query Personal Info



Process flow of event:

1. After customer successfully logged in the system, they go to personal information area
2. System display their personal information

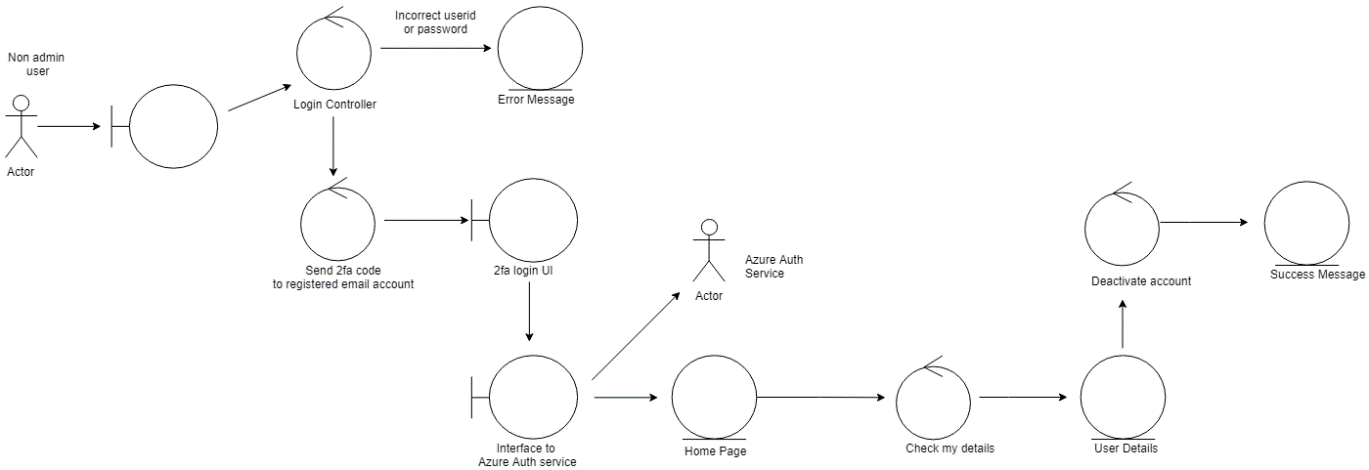
#### Update Personal Info



Process flow of event:

1. After customer successfully logged in the system, they go to personal information area
2. Customer modify and save their personal info
3. System save the modified personal info

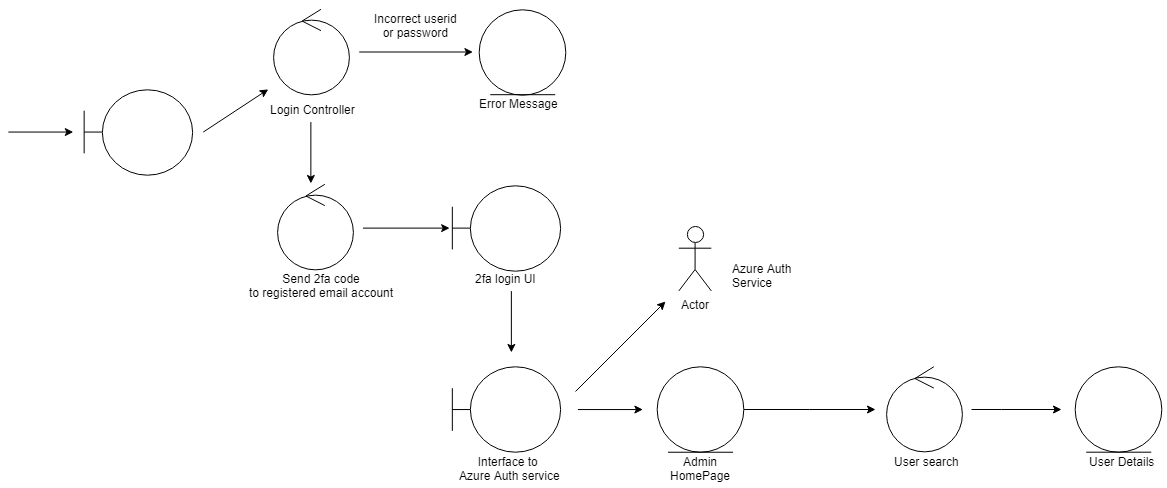
#### Deactivate Personal Account



Process flow of event:

1. After customer successfully logged in the system, they go to personal information area
2. Customer submit a request to deactivate the account
3. System checks the eligibility of the deactivation
4. If successful, customer account is deactivated

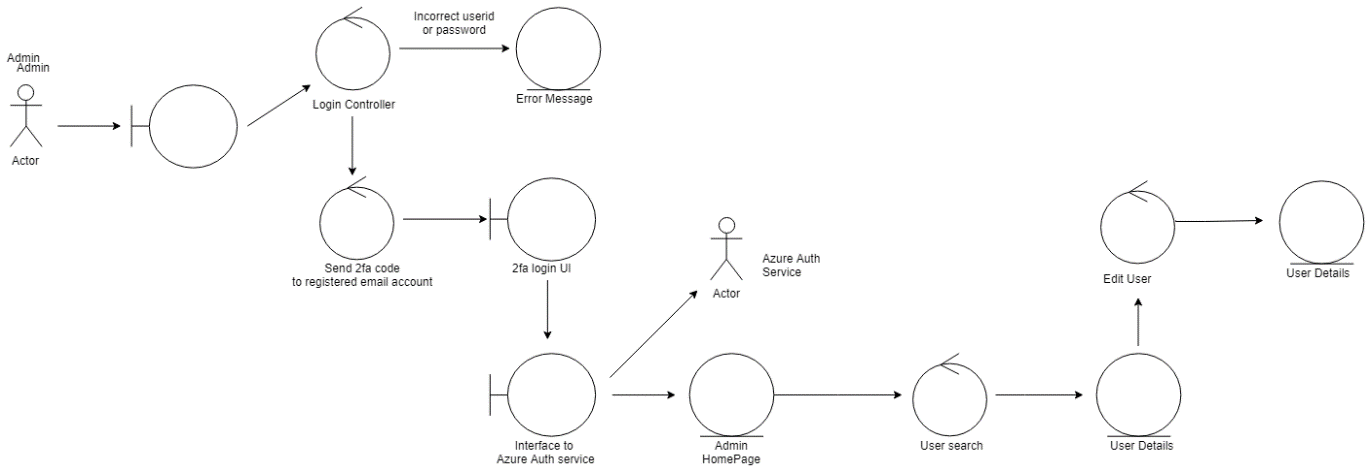
#### Query User Info



Process flow of event:

1. After admin successfully logged in the system, they go to customer search area
2. Admin search customer
3. System display the result of the search

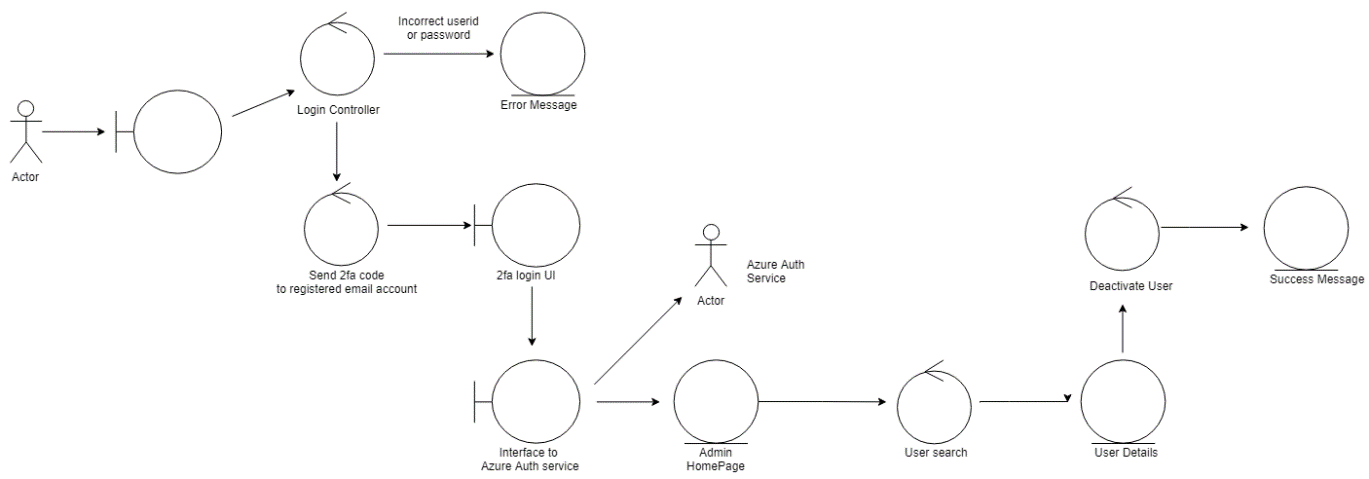
#### Update User Info



Process flow of event:

1. After admin successfully logged in the system, they go to customer modification area
2. Admin select a customer that they want to modify
3. Admin make modification with the selected customer and save it
4. System save the modified information of selected customer

#### Deactivate Account

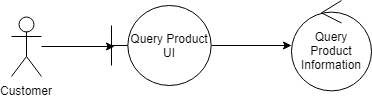


Process flow of event:

1. After admin successfully logged in the system, they go to customer deactivation area
2. Admin select a customer that they want to deactivate
3. Admin deactivate the selected customer
4. System deactivate the selected customer

### e-Wallet Management

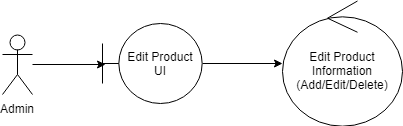
#### Query Product



Process flow of event:

1. Customer make an API request through application to query the product
2. System authenticate the request
3. If successful, system will send the product information

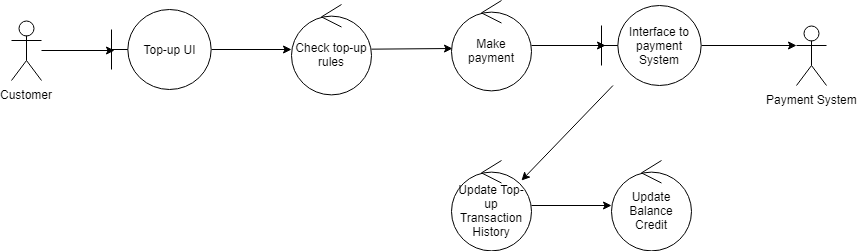
#### Edit Product



Process flow of event

1. Admin make an API request to create or edit the product through application
2. System authenticate the request
3. If successful, system will create or update the product information

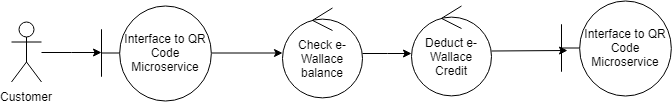
#### Top-up e-Wallet



Process flow of events:

1. Customer go to e-Wallet top up area
2. System will check the top-up rules including the eligibility
3. Customer make payment for the amount that they want to top-up in their e-Wallet
4. Customer is redirected to third party payment gateway
5. If successful, system will update customer’s e-Wallet balance and create transaction history

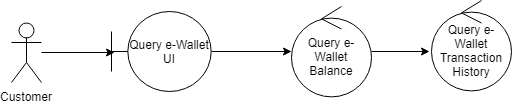
#### Deduct e-Wallet



Process flow of events:

1. Customer make a transaction and triggering an API request to system to deduct their e-Wallet balance
2. System check the eligibility of deduction
3. If successful, system will deduct customer’s e-Wallet balance and send a successful update back to customer

#### Query e-Wallet

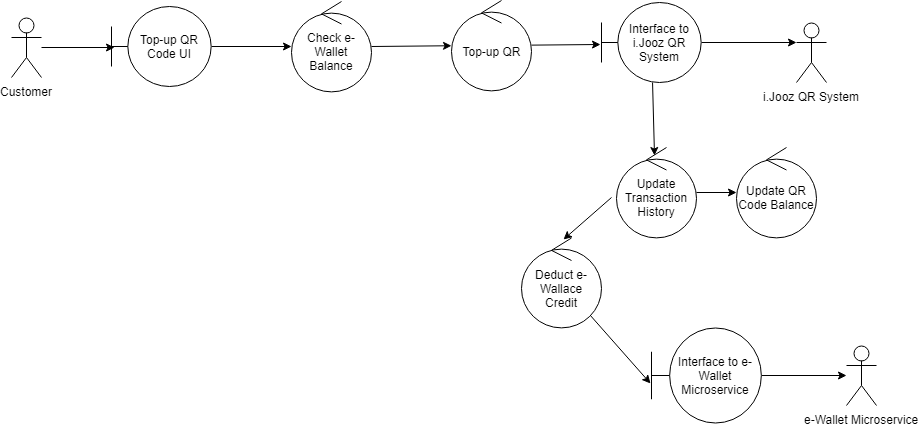


Process flow of event:

1. Customer go to application and query for their e-Wallet
2. System check the eligibility of the request
3. If successful, system will send the detail of customer e-Wallet

### QR Code Management

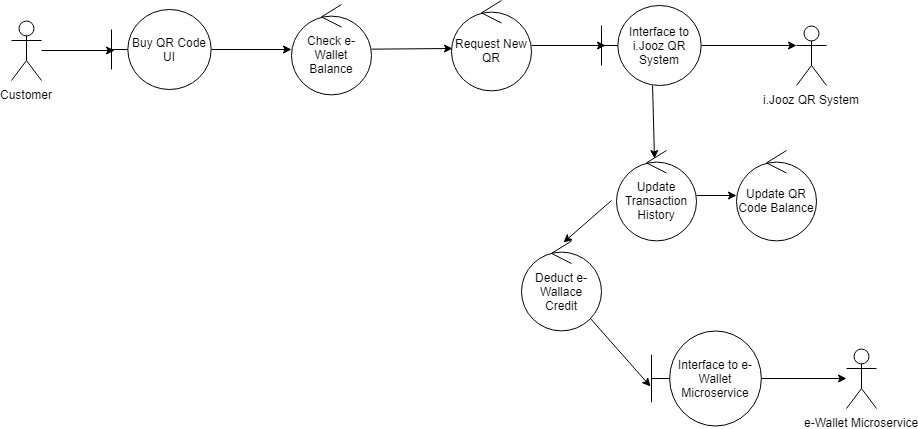
#### Top-up QR Code



Process flow of events:

1. Customer go to QR code top-up area
2. System will check the eligibility of QR code top-up by checking the customer’s balance in their e-Wallet
3. If eligible, system will proceed the QR code top-up and send update to i.Jooz system through their API interface
4. If successful, system will do the following in sequence
   1. Update customer’s QR code balance and create transaction history
   2. Deduct customer’s e-Wallet balance through e-Wallet API interface

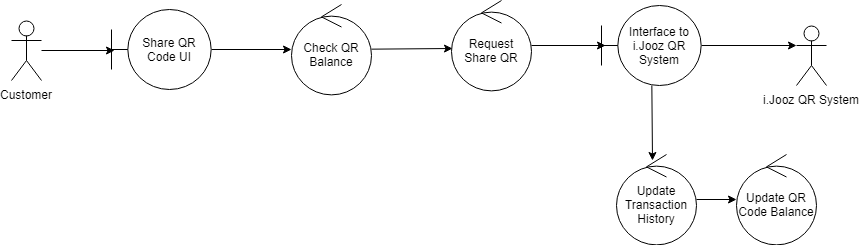
#### Buy QR Code



Process flow of events:

1. Customer go to QR code purchase area and purchase the QR code
2. System will check the eligibility of purchase by checking the balance in customer’s e-Wallet
3. If eligible, system will request new QR code from i.Jooz system through their API interface
4. If successful, system will do as follows in sequence:
   1. Retrieve QR code and create its balance
   2. Update transaction history
   3. Deduct customer’s e-Wallet balance through e-Wallet API interface

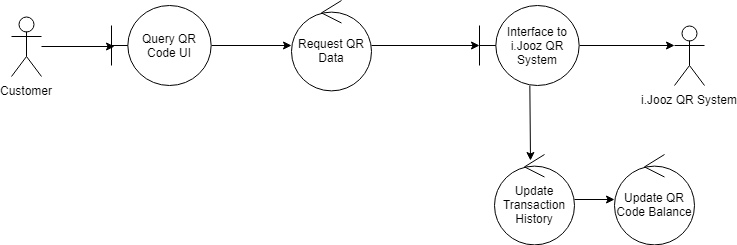
#### Share QR Code



Process flow of events:

1. Customer go to QR code sharing area and define recipient and QR code amount that they want to share
2. System will check the eligibility of sharing by checking QR code balance
3. If eligible, system will proceed to submit QR code sharing request from customer
4. If successful, system will do as follows in sequence
   1. System will share the QR code balance that customer want to share to recipient
   2. Update QR code balance of customer
   3. Update transaction history

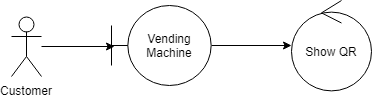
#### Query QR Code Balance



Process flow of events:

1. Customer request for QR code balance
2. System send the request to i.Jooz API interface
3. If successful, system will create transaction history and update customer’s QR code balance

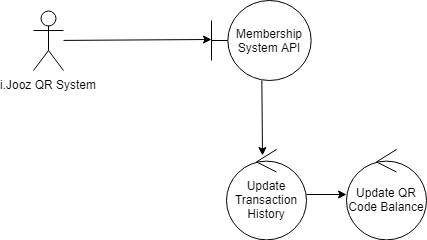
#### Use QR Code



Process flow of events:

1. Customer go to i.Jooz vending machine
2. Customer select on of available QR code
3. If successful, i.Jooz vending machine will accept the QR code and deduct the balance of customer’s QR code through system API interface

#### Update Transaction of QR Code



Process flow of events:

1. i.Jooz API make a request to the system to update transaction history and as well as QR code balance
2. System API receives the request and make update to transaction history and as well as QR code balance

# Project Scope and Milestones

## Scope of Work

### Use Cases

Use cases that we are going to develop are all use cases that are mentioned in section 2.3

### DevOps

Our plan to have automated DevOps is to have the following platforms:

#### Jenkins

This platform will be used for automated continuous integration (CI) and continuous delivery (CD)

#### GitHub

This platform will be used for source code

### Cloud Native and Scalability

Our plan to implement cloud native and scalability is to have the following platform:

#### .Net Core Microservice

This platform will be used to develop all microservices (API)

#### Azure SQL database

This platform will be used as database service in each microservices

#### Azure Event Bus

This platform will facilitate the message queue sent from each microservice

#### Azure API gateway

This platform will handle API request routing from web UI to microservice

### Platform Engineering

Engineering platforms that we are going to use are as follows:

* .Net Core
* Azure SQL Server Database

## Milestones and Effort

|  |  |  |
| --- | --- | --- |
|  | Activities | Estimated Duration |
| 1 | Planning and Create Project Proposal Document | 23 Mar 2019 - 8 Apr 2019 |
| 2 | Exploration, POC, Installation  - DevOps  - API Gateway  - Event Bus | 7 Apr 2019 - 12 Apr 2019 |
| 3 | Develop Web UI | 13 Apr 2019 - 11 May 2019 |
| 4 | Develop User Management | 13 Apr 2019 - 11 May 2019 |
| 5 | Develop QR Code Management | 13 Apr 2019 - 11 May 2019 |
| 6 | Develop E-Wallet Management | 13 Apr 2019 - 11 May 2019 |
| 7 | Integration Testing and Deployment | 11 May 2019 - 18 May 2019 |
| 8 | Final Report Preparation | 18 May 2019 - 20 May 2019 |