Team project – 953202 Introduction to SE
2-2021

Chaorai Prompt

Introduction to Software Engineer 95303 Project

Team name: Untitled

Kan Katpark	Project manager, Software Architecture & Analyst
Jakkaphat Polylake	Software Designer & Tester
Dusit Chunviset	Software Designer & Tester
Tinnapat Rattanawilaikul	Software Designer & Analyst
Sarin Patanompee	Software Tester & Architecture

Project detail

Ideas

A software project (Mobile Web Application) that aims to create a centralized market for farmers to list agricultural products for free. Which farmer can sell their products at prices as high as they want them to be. The company will take a percentage cut from the listed price of the sold products.

User research

Background

- **Problem**
    - Middleman exploits farmers in Thailand.
    - Farmers can't find the market to sell their products.

- **Purpose**
    - Aim to be a place where Thai farmers can freely sell their products.

- **Scope**
    - Thai farmer

Project vision statement

For Thai farmers

WHO want to sell their products at a fair price

THE Chaorai Prompt is an E-Commerce mobile web application

THAT aims to be a new open market of agricultural products

UNLIKE other e-commerce because we focus on agricultural products.

Perspectives

- User
- Farmer
- Admin

Feature:

Everyone:

- Chat
- Payment
- Authentication
- Authentication with third-party
- Register
- Share product
- Update self-information
- 2 Factor Authentication
- Report product

User:

- Buy product
- Check order status
- Confirm receive order
- Comment and Rate the product
- Open store

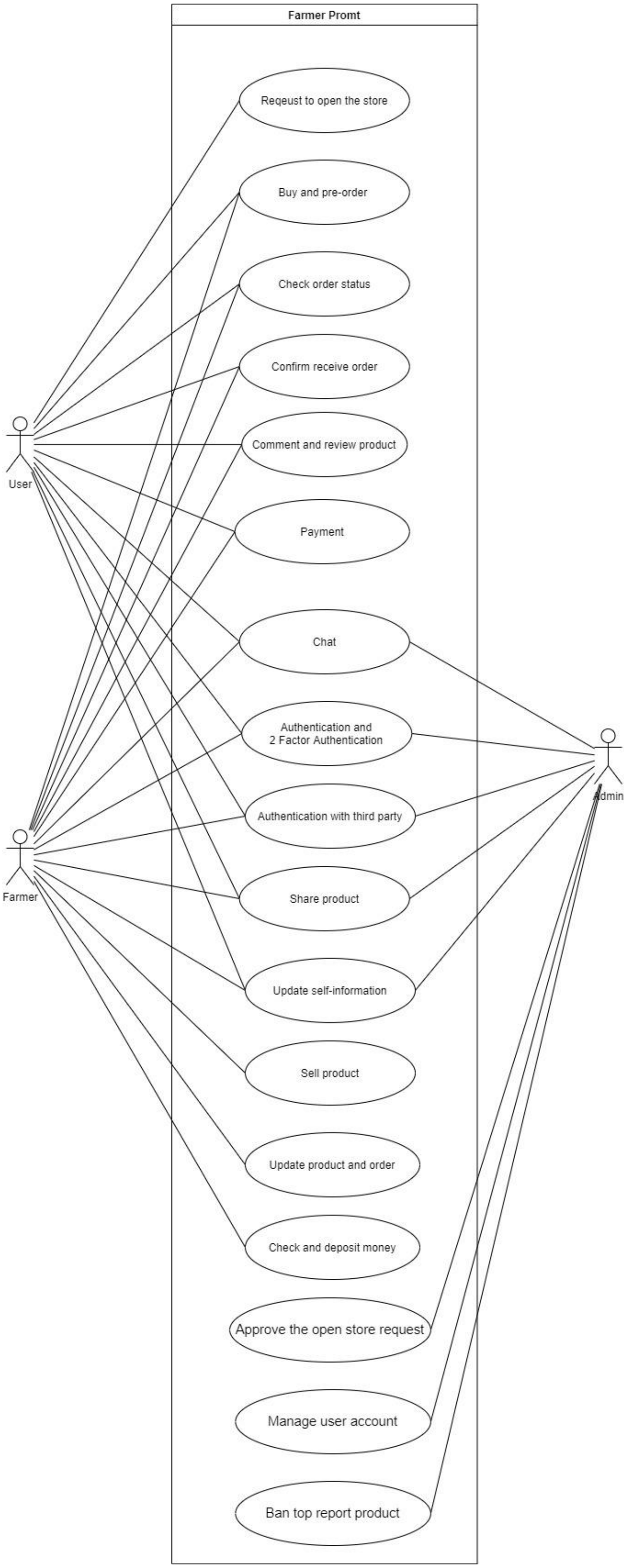
Farmer:

- Add and remove their products
- Update product and order status
- Check and withdraw money
- Can use user’s feature

Admin:

- Approve the open store request
- Ban most report product
- Manage user account

Use Case diagram



Use case description

Use case Name	Register
Description	1. Enters email and password 2. Hit sign up
Actor	Everyone

Use case Name	Register with third-party
Description	1. Hit register with third-party
Actor	Everyone

Use case Name	Login
Description	1. Enters username and password into the login form 2. Hit login button 3. 2FA if the user had added telephone number
Actor	Everyone

Use case Name	Login with third-party
Description	1. Hit login with third-party 2. 2FA if the user had added telephone number
Actor	Everyone

Use case Name	Update self-information
Description	1. Hit edit profile button 2. Edit 3. Hit update button
Actor	Everyone

Use case Name	Add 2-factor authentication
Description	1. User hit adds phone number button 2. User enters their phone number 3. User hit send OTP button 4. Enter OTP code 5. Submit
Actor	Everyone

Use case Name	Report Product
Description	1. Hit report button 2. Select report topic 3. Hit submit
Actor	Everyone

Use case Name	Share product
Description	1. Hit share button 2. Select platform to share
Actor	Everyone

Use case Name	Send a message
Description	1. Hit the "Chat" button 2. Select other users 3. Text a message 4. Hit "send" button
Actor	Everyone

Use case Name	User adds the product to their cart
Description	1. Select product 2. Hit buy/preorder button 3. Product is added to the cart
Actor	User

Use case Name	User checkout and payment
Description	1. Hit checkout button 2. Confirm product list 3. Select payment method 4. Confirm address 5. Checkout
Actor	User

Use case Name	User check their order status
Description	1. Hit my order button 2. Select order which they want to check
Actor	User

Use case Name	User confirm receive product
Description	1. Hit my order button 2. Select order which they want to confirm 3. Hit confirm receive button
Actor	User

Use case Name	User review product
Description	1. Give star rate 2. Write comment 3. Hit submit button
Actor	User

Use case Name	User asks to open the store
Description	1. Hit “sale your product” button 2. Enter their information. 3. Hit submit button
Actor	User

Use case Name	Farmers edit their information
Description	1. Hit “my store” button 2. Hit edit store 3. Update their information 4. Hit submit
Actor	Farmer

Use case Name	Farmers check their product
Description	1. Hit “my product” button 2. Check their product status
Actor	Farmer

Use case Name	Farmers add their product
Description	1. Hit “my product” button 2. Hit add product 3. Fill in the new product information 4 Hit submit
Actor	Farmer

Use case Name	Farmers update their product
Description	1. Hit edit product button 2. Update the information 3. Hit submit
Actor	Farmer

Use case Name	Farmers delete their product
Description	1. Hit delete product button 2. Hit confirm
Actor	Farmer

Use case Name	Farmers check their sale order
Description	1. Hit “sale order” button
Actor	Farmer

Use case Name	Farmers update their sale order
Description	1. Hit “sale order” button 2. Hit “On delivery” button
Actor	Farmer

Use case Name	Farmers deposit money
Description	1. Hit “My wallet button” button 2. Select bank account to receive money 3. Hit submit 4. Do OTP verification
Actor	Farmer

Use case Name	Farmers check their money
Description	1. Hit “My wallet button” button 2. Select “History” menu
Actor	Farmer

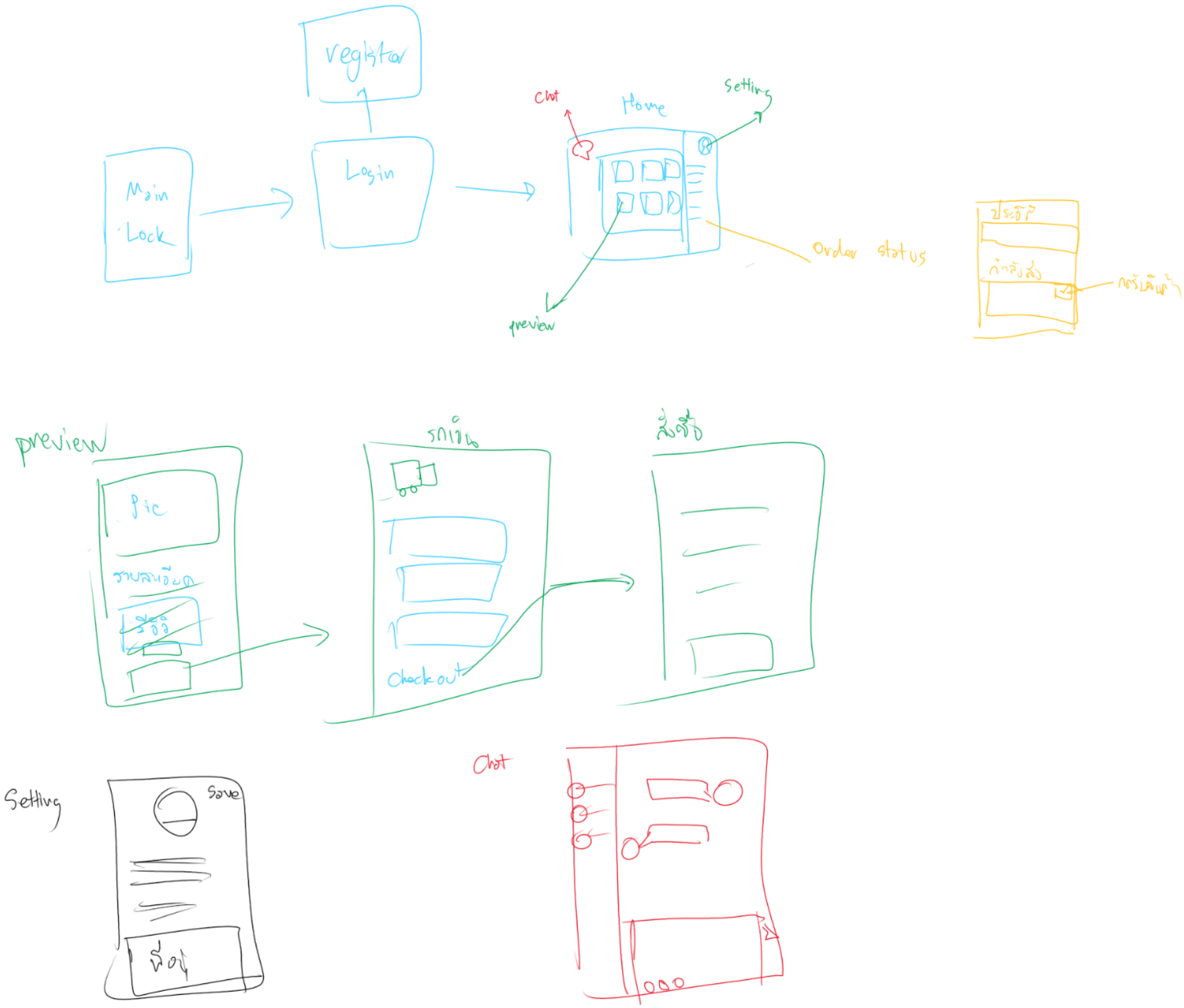
Use case Name	Admin approve store
Description	1. Hit manage Store button 2. Select the request store in list 3. Hit approve or cancel
Actor	Admin

Use case Name	Admin ban top report product
Description	1. Hit manage product button 2. Read the information 3. Hit Ban button 4. The auto message will be sent to the product owner
Actor	Admin

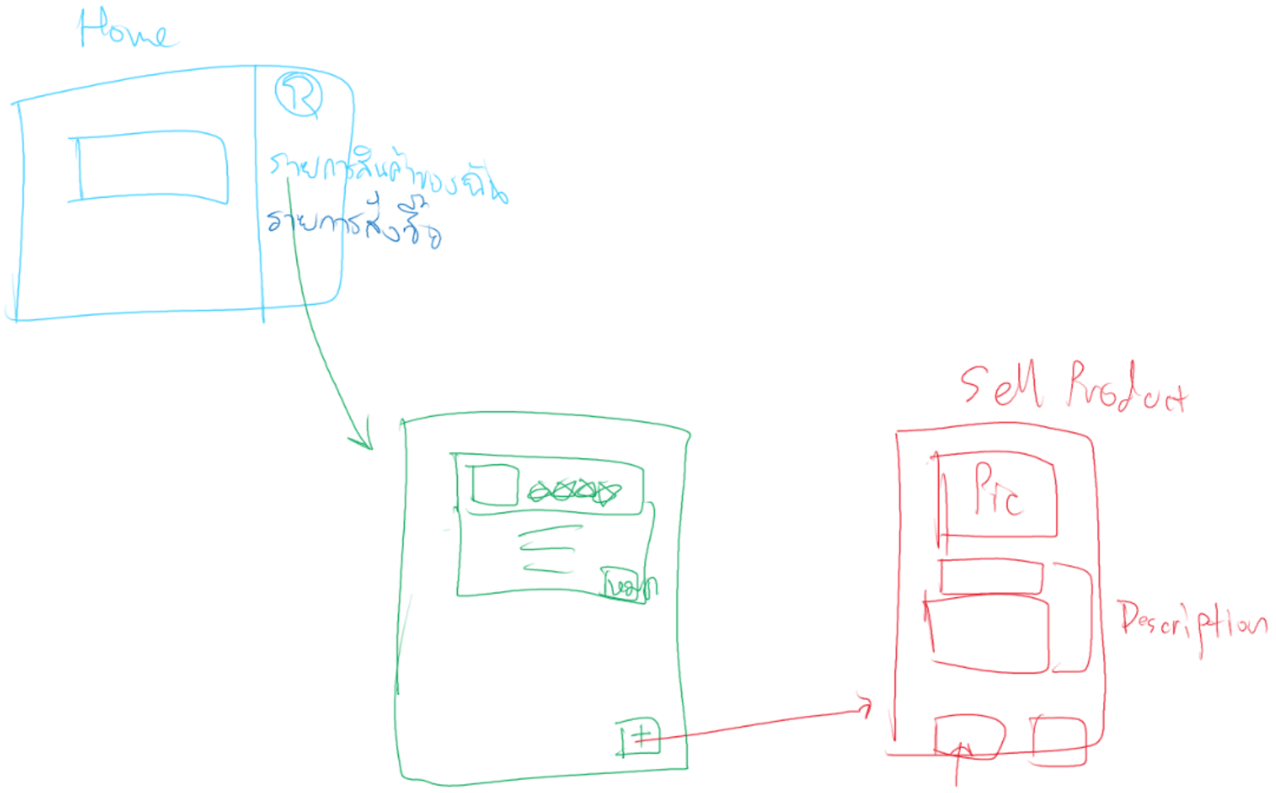
Use case Name	Admin manage user
Description	1. Hit manage user
Actor	Admin

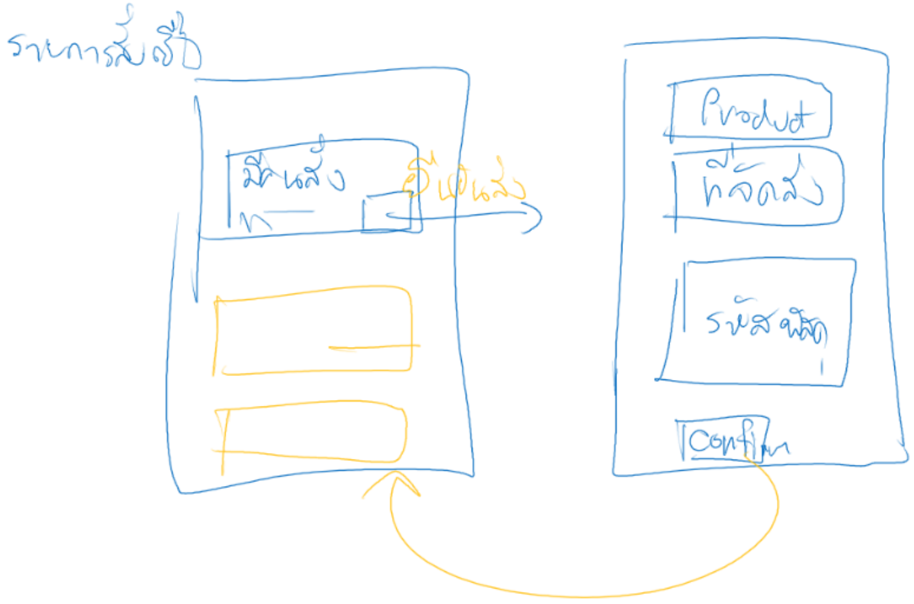
Sketch Design

User

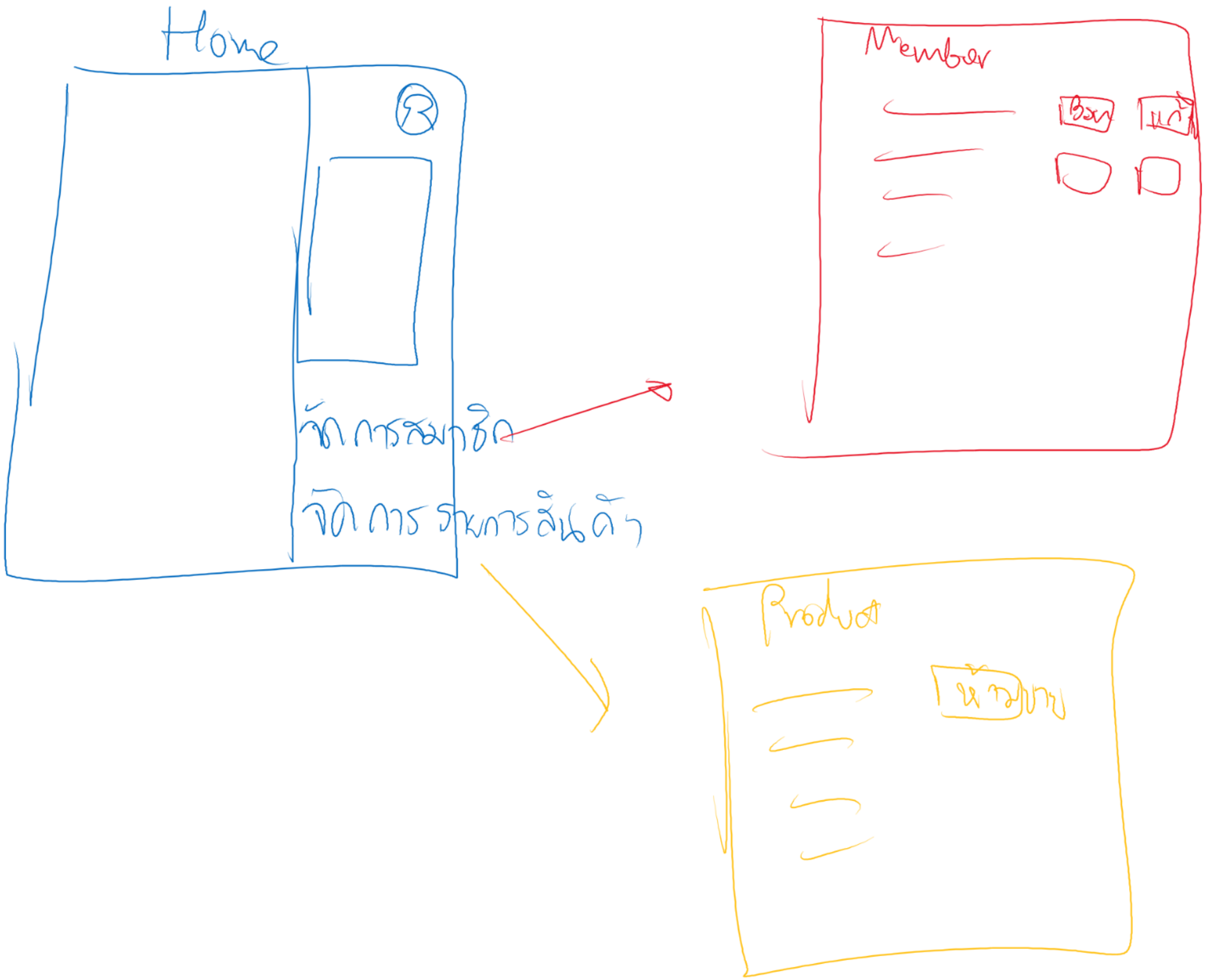


Former


Home
R
รายการสินค้าของฉัน
รายการสั่งซื้อ
Sell Product
Pic
Description
+


รายการสั่งซื้อ
มีคนสั่ง
ส่งแล้ว
Product
ที่จัดส่ง
รหัสพัสดุ
Confirm

Admin


Home
B
จัดการสมาชิก
จัดการรายการสินค้า
Member
ลบ
แก้
Product
ห้ามขาย

Architecture

RESTFUL web and mobile web application that is deployed with docker on the cloud server.

Technology

Frontend: NextJS

NextJS is a Javascript framework that is an upgraded version of ReactJS. NextJS can work in SEO easier than ReactJS and NextJS also can do SSG, ISR, CSR, and SSR that is the reason why we chose NextJS.

Backend: TypeScript

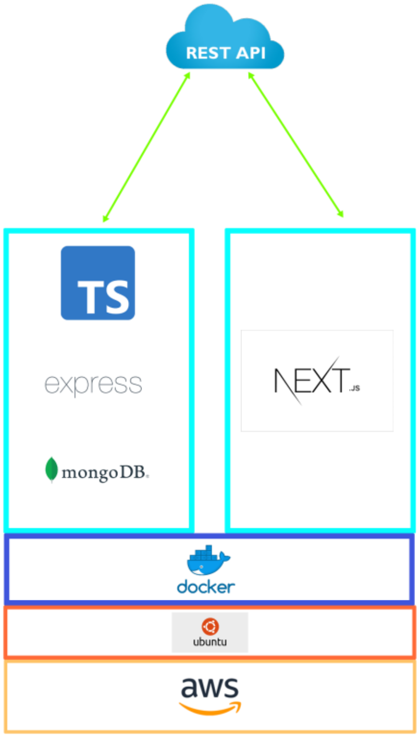
Typescript is chosen to use on the server-side to receive the request and send the response and also work with authentication. The libraries that we will use are Express and JSONWebtoken.

Database: MongoDB

MongoDB is a document-based database and it is horizontally scalable.

Tool

- Git
- Docker
- Discord
- Adobe XD
- Amazon AWS



Software process model

Extreme Programming

"Extreme Programming is a lightweight methodology for small to medium-sized teams developing software in the face of vague or rapidly changing requirements"

Kent Beck

From the definition, XP is suitable for our team that have 5 peoples and the requirements or feature can be changed or added any time. And our team members can work separately for 24 hours by leaving a message or today's progression in discord and other members can continuously work on it.

Project Timeline

Task name	Duration (day)	Start date (d/m)	End date (d/m)	Responsibility
Project launch	1	27/01	27/01	Everyone
Select topic and scope of software project	2	28/01	29/01	Everyone
Use research and product vision	1	30/01	30/01	Everyone
List of features	1	31/01	31/01	Everyone
Select software architecture and technology	1	01/02	01/02	Kan Sarin
Create a use case diagram	18	02/02	20/02	Jakkapat Tinnapat
Create a use case description	18	02/02	20/02	Tinnapat Dusit
Create sketch design	1	02/02	02/02	Everyone
Implement software prototype #1	6	03/02	09/02	Tinnapat Jakkapat Dusit Kan
Hybrid meeting	1	09/02	09/02	Everyone
Fix list of feature and requirement	1	09/02	09/02	Everyone
Implement software prototype #2	7	09/02	16/02	Tinnapat Jakkapat Dusit Kan

Test #1 User feature	3	14/02	17/02	Sarin
Fix test #1	4	17/02	21/02	Tinnapat Jakkapat Dusit
Test #2 Farmer feature	3	19/02	22/02	Sarin
Fix test #2	1	22/02	22/02	Tinnapat Jakkapat Dusit
Test #3 Admin feature	1	22/02	23/02	Sarin
Fix test #3	1	23/02	23/02	Tinnapat Jakkapat Dusit

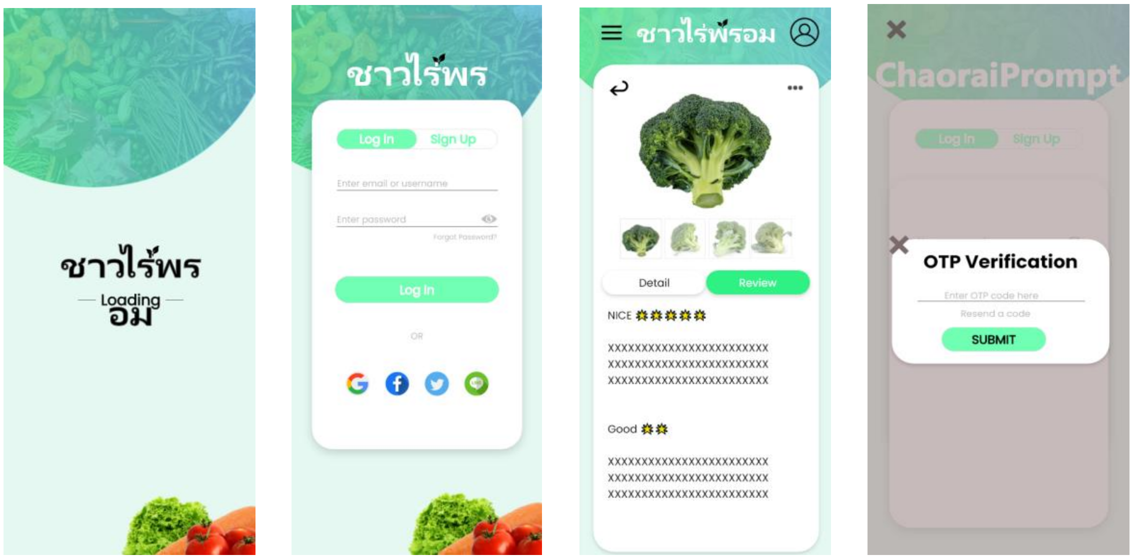
Usability testing

Prototype name: Chaorai Prompt 0.1

Feature list: Whole feature

Number of participants: 2 people

Before test UI



Context (User, task, environment):

User: 2 Students

Task: Test the basic functions for

#1 Users,

#2 Farmers,

#3 Administrators,

and point out design flaws.

Environment:

#1 User

#2 Farmer

#3 Administrators

Techniques (Think aloud/Observation/Interview): Think Aloud

Results

User feedbacks

User Part:

In general. The UI lacks conciseness. Clarity could be better; more information needs to be presented. The PromptPay function is too hard to use.

Texts need to have more contrast with the backgrounds, text cut-off. Ease of use could be better on the home screen; 2 products per screen is too low. Sidebars come up when you don't want them to. The star in the star rating is straight up an emoji, and needs to be replaced. Inconsistent back/return button; sometimes an arrow, sometimes an x.

Farmer Part:

There are a lot of misaligned icons and UI elements. The UI doesn't have proper margin/padding. The Sale Order descriptions don't give enough information at a glance and go against basic color signals. Icons can be misinterpreted and need to be replaced with better ones.

Transaction history needs to give more information, better color coding, clear symbols.

A lot of buttons could be repositioned into many pages, such as profile, to avoid cluttering the sidebar.

Page for financials needs to be redone to display relevant information and not a huge cash icon.

Administrator Part:

Sidebar icon needs a notification icon.

Images here don't have rounded edges despite most of the UI having rounded edges. Inconsistent color scheme.

Text cut-off, Lack of color variety.

Lack of search function in User Page; There could be like thousands of users, possibly, in that page.

Term of Uses pop-up needs to have a smaller font size for texts.

The Chat UI overall looks okay, but after calling up the chat list everything starts getting too close for comfort.

Images inside icons are still misaligned.

Failure points:

Some words are used incorrectly, thereby misleading the user.

Analysis of the user feedback and failure points (What UI of prototype should be fixed):

Color Palette has to be altered and enforced in many objects with foreground and background color properties.

We have to stay consistent with the choice of the icons.

Lots of icons have to be recentered based on weight.

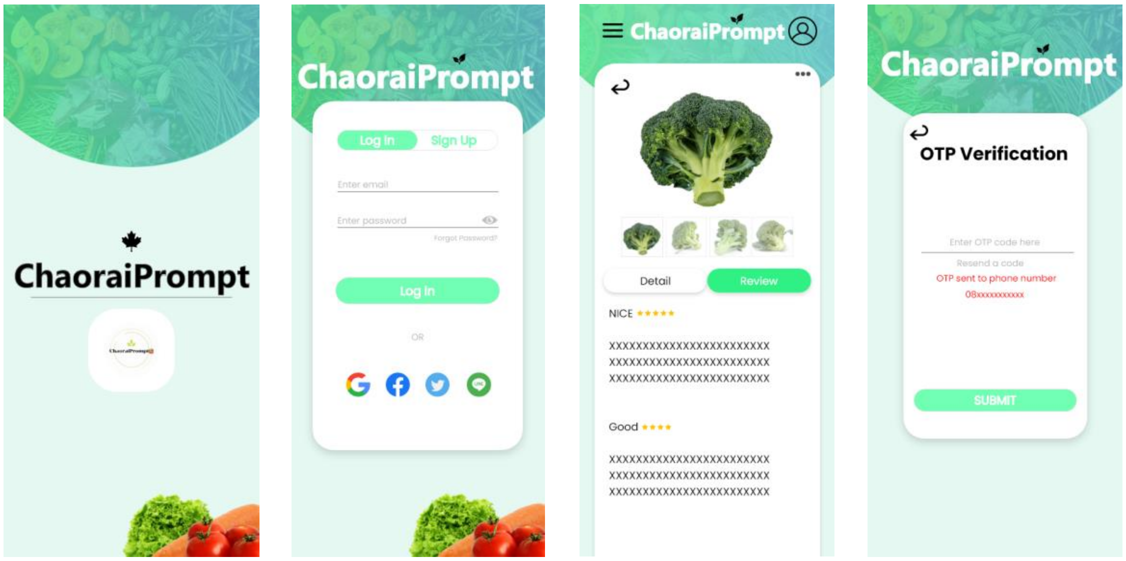
Sidebar access needs to be refined.

Font size needs to be changed in a lot of places.

The UI has to give more information at a glance.

The improved version of the prototype after refining the previous version of the prototype

After test UI



Software Limitation

- Cancel orders and refund product features
- Farmer store reliability features (Store reliability, review store from buyer)
- Admin revoke a farmer store license