

**Campus Meal Ordering System**

**Use case model**

**By *Team Foodie***

**Lab Group: TS3**

**Date: August 2020**

**Instructors:**

**Bo An**

**Leong Yin Yoke Junie**

**Qiu Wei**

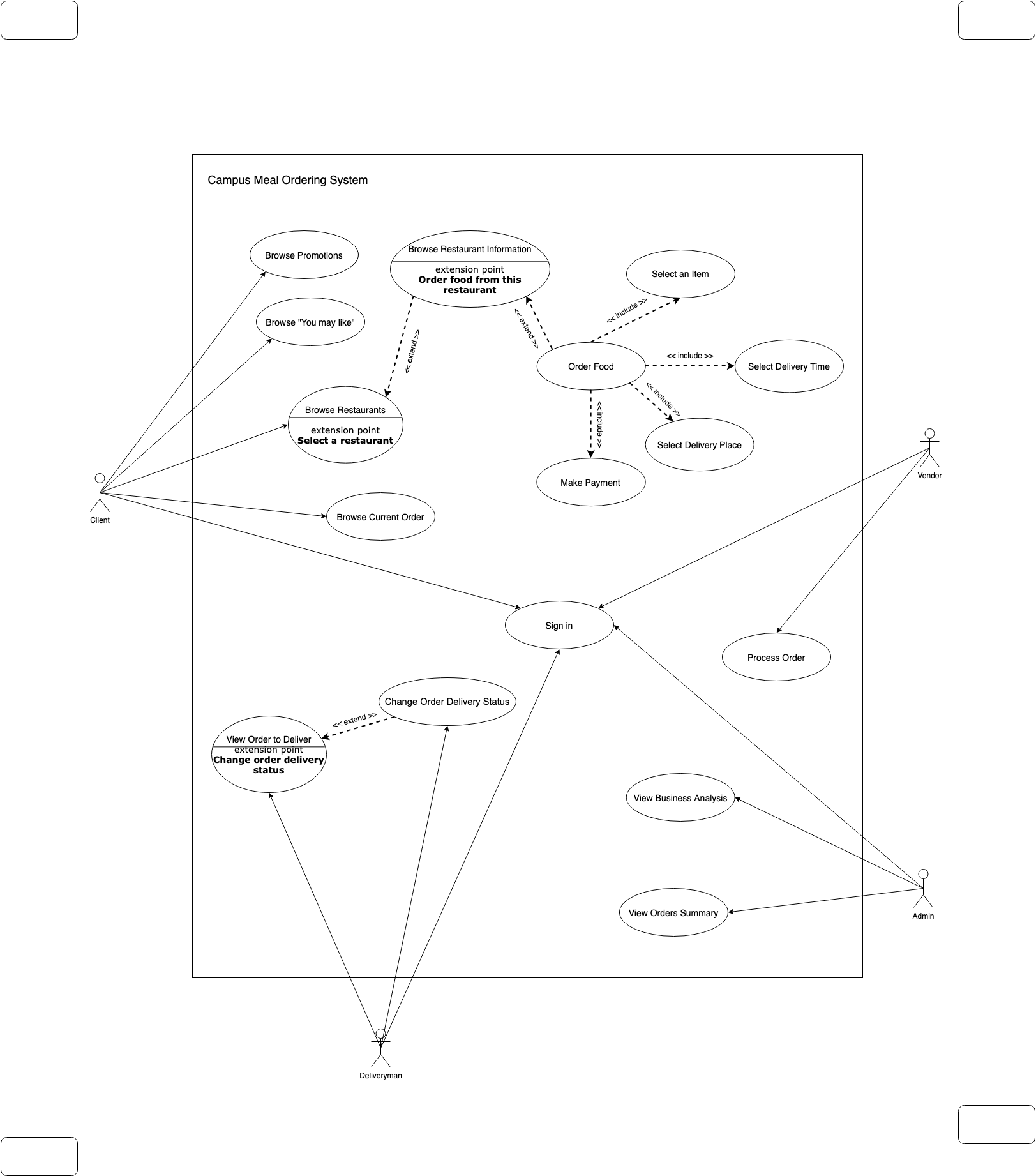
**Wang Rundong**

# **SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**

**NANYANG TECHNOLOGICAL UNIVERSITY**

This document is the use case model of “Campus Meal Ordering System”. The use case model including a use case diagram, list of use cases and use case descriptions for each use case.

**Use case diagram**



**Use cases list**

There are four actors in the system: client, vendor, deliveryman, and admin. Each of these actors has several use cases.

**Client:**

* UCC-001: Sign in
* UCC-002: Browse Promotions
* UCC-003: Browse “You may like”
* UCC-004: Browse Restaurants
* UCC-005: Browse Restaurant Information
* UCC-006: Order Food
* UCC-007: Select an Item
* UCC-008: Select Delivery Time
* UCC-009: Select Delivery Place
* UCC-010: Make payment
* UCC-011: Browse Current Order

**Vendor:**

* UCV-001: Sign in
* UCV-002: Process Order

**Deliveryman:**

* UCD-001: Sign in
* UCD-002: Process order
* UCD-003: Change Order Delivery Status

**Admin:**

* UCA-001: Sign in
* UCA-002: View Orders Summary
* UCA-003: View Business Analysis

**Use case descriptions**

## **Client**

| Use Case ID | UCC-001 |
| --- | --- |
| Use Case Name | Sign in |
| Use Case Description | Sign in to the application |
| Actor | Initiating actor: Client |
| Entry Conditions | 1. The user has a valid google account and password. |
| Exit Conditions | 1. The user successfully signs in, or 2. The user fails to sign in. |
| Flow of Events | | Actor Steps | System Steps | | --- | --- | | 1.The user inputs his/her own google account and password | 2.Google API verifies credentials | |  | 3. System check if the user is a valid client | |  | 4. loads and returns client homepage | |
| Alternative Flows | AFS2 if the google account and password are not valid   1. Systems returns message “invalid account and password, please try again” 2. The system returns to Step 1   AFS3 if the user identity is not matched   1. Systems returns message “invalid user identity, please try again” 2. The system returns to Step 1 |
| Exceptions | NA |
| Includes | NA |

| Use Case ID | UCC-002 |
| --- | --- |
| Use Case Name | Browse Promotions |
| Use Case Description | Allows Client to view and navigate to promotional items |
| Actor | Initiating actor: Client |
| Entry Conditions | 1. Client clicks on Browse Promotions page |
| Exit Conditions | 1. Client presses on the back button 2. A specific promotion is selected |
| Flow of Events | | Actor Steps | System Steps | | --- | --- | | 1.Client selects Browse Promotions page | 2.System returns list of Promotions | | 3.Specific promotion is selected | 4.System redirects user to restaurant with selected promotions | |
| Alternative Flows | AF-S3: If the user selects back button  1. System will display the previous page  AF-S4: If user is not logged in when selecting promotion  1. System redirects user to Sign-In page and prompts the user to log in |
| Exceptions | EX1: Specific promotion does not exist   1. Error will be thrown and list of promotions are displayed |
| Includes | NA |

| Use Case ID | UCC-003 |
| --- | --- |
| Use Case Name | Browse “You may like” |
| Use Case Description | Allows Client to view and navigate to recommended items |
| Actor | Initiating actor: Client |
| Entry Conditions | 1. Client clicks on “You may like” page |
| Exit Conditions | 1. Back button is pressed 2. Recommended item is selected |
| Flow of Events | | Actor Steps | System Steps | | --- | --- | | 1.Client selects Browse “You may like” page | 2.System returns list of recommendations | | 3.Specific restaurant on recommendation list is selected | 4.System redirects user to restaurant | |
| Alternative Flows | AF-S3: If the user selects back button   1. The system will display the previous page   AF-S4:If user is not logged in upon selection of Browse “You may like” page   1. The system will redirect userSign-In page and prompts the user to log in |
| Exceptions | EX1: Specific item does not exist   1. Error will be thrown and list of recommended items are displayed |
| Includes | NA |

| Use Case ID | UCC-004 |
| --- | --- |
| Use Case Name | Browse Restaurants |
| Use Case Description | Allows Client to view and navigate list of restaurants |
| Actor | Initiating actor: Client |
| Entry Conditions | 1. Client clicks on Browse Restaurants |
| Exit Conditions | 1. Back button is pressed 2. Restaurant is selected |
| Flow of Events | | Actor Steps | System Steps | | --- | --- | | 1.Client selects Browse Restaurants | 2.System returns list of restaurants | | 3.Specific restaurant is selected | 4.System redirects user to the selected restaurant page | |
| Alternative Flows | AF-S3: If the user selects back button   1. The system will display the previous page   AF-S4: If user is not logged in upon selection of restaurant   1. The system will redirect user to Sign-In page and the user will be prompted to log in |
| Exceptions | EX1: Specific restaurant does not exist   1. Error will be thrown and list of restaurants are displayed |
| Includes | NA |

| Use Case ID | UCC-005 |
| --- | --- |
| Use Case Name | Browse Restaurant Information |
| Use Case Description | Client views a particular restaurant and its relevant information |
| Actor | Initiating actor: Client |
| Entry Conditions | 1. Client has clicked on Browse Restaurants 2. Then, presses on a restaurant of his/her choice |
| Exit Conditions | Back button is pressed or restaurant is selected |
| Flow of Events | | Actor Steps | System Steps | | --- | --- | | 1.Client selects Browse Restaurants | 2.System returns list of restaurants | | 3.Specific restaurant is selected | 4.System redirects user to the selected restaurant page | |
| Alternative Flows | AF-S3: If the user selects back button   1. The system will display the previous page   AF-S4: If user is not logged in upon selection of restaurant   1. The system will redirect user to Sign-In page and prompts the user to log in |
| Exceptions | EX1: Specific restaurant does not exist   1. Error will be thrown and user will be redirected to list of restaurants |
| Includes | NA |

| Use Case ID | UCC-006 |
| --- | --- |
| Use Case Name | Order Food |
| Use Case Description | A client orders food and set the order details |
| Actor | Initiating actor: Client |
| Entry Conditions | Client is logged in, selected a restaurant and in the restaurant menu page. |
| Exit Conditions | Back button is pressed or a new order is placed successfully. |
| Flow of Events | | Actor Steps | System Steps | | --- | --- | | 1. Client selects an item in the menu to the cart.  2. Client proceeds to the cart for checking out. | 3. System displays the cart page with delivery options including delivery place and delivery time. | | 4. Client selects delivery time and delivery place. | 5. Systems indicates user to make payment | | 6. Client makes payment | 7. System shows order successfully placed | |
| Alternative Flows | AF-S1: If the user wants to add another item   1. The system takes the user back to the menu of this restaurant. 2. The user returns to Step 1.   AF-S6: If the use wants to edit the delivery time and delivery place   1. The user returns to Step 4. |
| Exceptions | NA |
| Includes | UCC-007, UCC-008, UCC-009, UCC-010 |

| Use Case ID | UCC-007 |
| --- | --- |
| Use Case Name | Select an Item |
| Use Case Description | A client selects items in a restaurant to place an order. This use case is included in the “Order Food” use case. |
| Actor | Initiating actor: Client |
| Entry Conditions | Client is logged in, selected a restaurant and in the restaurant menu page. |
| Exit Conditions | An item is chosen successfully. |
| Flow of Events | | Actor Steps | System Steps | | --- | --- | | 1. Client selects an item from the restaurant menu. | 2. System shows the popup describing the current item with a “Add Item” button. | | 3. Client chooses to add this item. | 4. System adds the item to the client’ cart | |
| Alternative Flows | AF-S3: If the user dismisses the item popup   1. The system takes the user back to the menu of this restaurant. |
| Exceptions | NA |
| Includes | NA |

| Use Case ID | UCC-008 |
| --- | --- |
| Use Case Name | Select Delivery Time |
| Use Case Description | A client selects delivery time of the order |
| Actor | Initiating actor: Client |
| Entry Conditions | Client has logged in, selected a restaurant and added an item to cart. Client plans to check out the cart. |
| Exit Conditions | Back button is pressed or delivery time is set successfully. |
| Flow of Events | | Actor Steps | System Steps | | --- | --- | | 1. Client selects “set delivery time” in the cart checkout page. | 2. System shows a group of choices of available delivery time. | | 3. Client chooses a desired delivery time. | 4. System sets the delivery time for the order. | |
| Alternative Flows | AF-S3: If the users wants to go back   1. The systems takes user back to the cart checkout page 2. The user returns to Step 1. |
| Exceptions | NA |
| Includes | NA |

| Use Case ID | UCC-009 |
| --- | --- |
| Use Case Name | Select Delivery Place |
| Use Case Description | A client selects delivery place of the order |
| Actor | Initiating actor: Client |
| Entry Conditions | Client has logged in, selected a restaurant and added an item to cart. Client plans to check out the cart. |
| Exit Conditions | Back button is pressed or delivery time is set successfully. |
| Flow of Events | | Actor Steps | System Steps | | --- | --- | | 1. Client selects “set delivery place” in the cart checkout page. | 2. System shows a group of choices of available delivery places. | | 3. Client chooses a desired delivery place. | 4. System sets the delivery place for the order. | |
| Alternative Flows | AF-S3: If the users wants to go back   1. The systems takes user back to the cart checkout page 2. The user returns to Step 1. |
| Exceptions | NA |
| Includes | NA |

| Use Case ID | UCC-0010 |
| --- | --- |
| Use Case Name | Make payment |
| Use Case Description | A client make payment to the order |
| Actor | Initiating actor: Client |
| Entry Conditions | Client has logged in, selected a restaurant, added an item to cart and set the delivery information. Client plans to make payment of the order. |
| Exit Conditions | Back button is pressed or payment is made. |
| Flow of Events | | Actor Steps | System Steps | | --- | --- | | 1. Client selects “make payment ” in the cart checkout page. | 2. System shows the popup displaying the amount the client should pay. | | 3. Client chooses “pay”. | 4. System receives the payments and marks the order as placed. | |
| Alternative Flows | AF-S3: If the users wants to go back   1. The systems takes user back to the cart checkout page 2. The user returns to Step 1. |
| Exceptions | EX1: Order does not exist   1. Error will be thrown and user will be redirected to checkout cart page   EX2: Payment cannot be processed   1. Error will be thrown and user will be redirected to checkout page |
| Includes | NA |

| Use Case ID | UCC-011 |
| --- | --- |
| Use Case Name | Browse Current Order |
| Use Case Description | Allows Client to view the order |
| Actor | Initiating actor: Client |
| Entry Conditions | Order is completed by client (UCC-006 is done) |
| Exit Conditions | The user closes the page |
| Flow of Events | | Actor Steps | System Steps | | --- | --- | | 1.Client selects Browse Current Order page | 2.System returns Current Order details | |
| Alternative Flows | NA |
| Exceptions | NA |
| Includes | NA |

## 

## **Vendor**

| Use Case ID | UCV-001 |
| --- | --- |
| Use Case Name | Sign in |
| Use Case Description | Sign in to the application |
| Actor | Initiating actor: vendor |
| Entry Conditions | 1. The user has a valid google account and password, and 2. The user is a valid vendor registered |
| Exit Conditions | 1. The user successfully signs in, or 2. The user fails to sign in. |
| Flow of Events | | Actor Steps | System Steps | | --- | --- | | 1.The user inputs his/her own google account and password | 2.Google API verifies credentials | |  | 3. System check if the user is a valid vendor | |  | 4. loads and returns vendor homepage | |
| Alternative Flows | AFS2 if the google account and password are not valid   1. Systems returns message “invalid account and password, please try again” 2. The system returns to Step 1   AFS3 if the user identity is not matched   1. Systems returns message “invalid user identity, please try again” 2. The system returns to Step 1 |
| Exceptions | NA |
| Includes | NA |

| Use Case ID | UCV-002 |
| --- | --- |
| Use Case Name | Process Order |
| Use Case Description | Process the order from client |
| Actor | Initiating actor: vendor  Participating actor: Client, Delivery man |
| Entry Conditions | 1. The user is verified as a vendor 2. A specified order is assigned to vendor (UCC-006) |
| Exit Conditions | 1. The vendor accepts the order, or 2. The vendor rejects the order |
| Flow of Events | | Actor Steps | System Steps | | --- | --- | | 2. Vendor views the order | 1. System prompt order from clients to vendor | | 3. Vendor accepts the order | 4. System updates the result | | 5. Vendor gets feedback that operation is successful | 6. System updates order status for delivery man | |  | 7. System updates order status for client | |
| Alternative Flows | AFS3 if vendor rejects the order:   1. System updates the result and notify vendor that operation is successful 2. The system goes to Step 7 |
| Exceptions | NA |
| Includes | NA |

## 

## **Deliveryman**

| Use Case ID | UCD-001 |
| --- | --- |
| Use Case Name | Sign in |
| Use Case Description | Sign in to the application |
| Actor | Initiating actor: Delivery man |
| Entry Conditions | 1. The user has a valid google account and password, and 2. The user is a valid delivery man registered |
| Exit Conditions | 1. The user successfully signs in, or 2. The user fails to sign in. |
| Flow of Events | | Actor Steps | System Steps | | --- | --- | | 1.The user inputs his/her own google account and password | 2.Google API verifies credentials | |  | 3. System checks if the user is a valid delivery man | |  | 4. loads and returns delivery man homepage | |
| Alternative Flows | AFS2 if the google account and password are not valid   1. Systems returns message “invalid account and password, please try again” 2. The system returns to Step 1   AFS3 if the user identity is not matched   1. Systems returns message “invalid user identity, please try again” 2. The system returns to Step 1 |
| Exceptions | NA |
| Includes | NA |

| Use Case ID | UCD-002 |
| --- | --- |
| Use Case Name | View Orders to Deliver |
| Use Case Description | Allows delivery man to view orders to deliver |
| Actor | Initiating actor: Delivery Man |
| Entry Conditions | The user is verified as a Delivery Man |
| Exit Conditions | The user closes this page |
| Flow of Events | | Actor Steps | System Steps | | --- | --- | | 1.Delivery Man selects View Orders page | 2.System returns list of orders that are assigned to Delivery Man at that moment | |
| Alternative Flows | NA |
| Exceptions | NA |
| Includes | NA |

## 

| Use Case ID | UCD-003 |
| --- | --- |
| Use Case Name | Change Order Delivery Status |
| Use Case Description | Allows delivery man to set order delivery status: picked up or delivered |
| Actor | Initiating actor: Delivery Man  Participating actor: Client |
| Entry Conditions | Choose an order from UCD-002 |
| Exit Conditions | The user close this page |
| Flow of Events | | Actor Steps | System Steps | | --- | --- | | 1.Delivery man selects an order | 2.System returns the details of the order | | 3. Delivery man set the status to “picked up” or “delivered” | 4.System updates the order status | |  | 5.System updates the new order status for client | |
| Alternative Flows | NA |
| Exceptions | NA |
| Includes | NA |

## 

## 

## **Admin**

| Use Case ID | UCA-001 |
| --- | --- |
| Use Case Name | Sign in |
| Use Case Description | Sign in to the application |
| Actor | Initiating actor: Admin |
| Entry Conditions | 1. The user has a valid google account and password, and 2. The user is a valid admin |
| Exit Conditions | 1. The user successfully signs in, or 2. The user fails to sign in. |
| Flow of Events | | Actor Steps | System Steps | | --- | --- | | 1.The user inputs his/her own google account and password | 2.Google API verifies credentials | |  | 3. System check if the user is a valid admin | |  | 4. loads and returns admin homepage | |
| Alternative Flows | AFS2 if the google account and password are not valid   1. Systems returns message “invalid account and password, please try again” 2. The system returns to Step 1   AFS3 if the user identity is not matched   1. Systems returns message “invalid user identity, please try again” 2. The system returns to Step 1 |
| Exceptions | NA |
| Includes | NA |

| Use Case ID | UCA-002 |
| --- | --- |
| Use Case Name | View Orders Summary |
| Use Case Description | Returns a summary of all orders |
| Actor | Initiating actor: Admin |
| Entry Conditions | The user is verified as Admin. |
| Exit Conditions | Data is returned |
| Flow of Events | | Actor Steps | System Steps | | --- | --- | | 1.Admin enters View Orders Summary page | 2.System returns the Orders Summary | |
| Alternative Flows | NA |
| Exceptions | NA |
| Includes | NA |

| Use Case ID | UCA-003 |
| --- | --- |
| Use Case Name | View Business Analysis |
| Use Case Description | Returns an analysis of all businesses |
| Actor | Initiating actor: Admin |
| Entry Conditions | The user is verified as Admin. |
| Exit Conditions | Data is returned |
| Flow of Events | | Actor Steps | System Steps | | --- | --- | | 1.Admin enters View Business Analysis | 2.System computes and returns the Business Analysis | |
| Alternative Flows | NA |
| Exceptions | NA |
| Includes | NA |