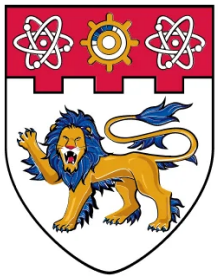


NANYANG TECHNOLOGICAL UNIVERSITY
SINGAPORE

SC2006 – Software Engineering

Lab 4 Deliverables

Lab Group	SCSX
Team	FeedItForward
Members	Toh Jing Qiang (U2121442H)
	Toh Jing Hua (U21210232L)
	Tommy Wee Chung Kiat (U2120448F)
	Tay Jia Ying, Denise (U2122458K)
	Twu Pin Yang (U2121072C)
	Teh Min Ze (U2111370H)

1. Black Box Testing

I. AuthController

Control class to test - **AuthController**
The AuthController class manages user authentication within the application, covering both user registration (signup) and user login.

The AuthController is a crucial component responsible for managing **user authentication** within the application. When a user attempts to log in, the controller collects their email and password. Subsequently, it validates the input data for correctness, ensuring the integrity and accuracy of the provided information. To authenticate the user, the controller retrieves the hashed password associated with the provided email from the database. The retrieved hashed password is then compared with the user's provided password. If the comparison results in a match, signifying successful authentication, the user will be directed to the home screen. Information managed by the AuthController in the login process includes email and password.

Apart from user authentication, the AuthController is also responsible for **user registration**. When a new user registers for the application, the controller will ensure essential information is collected, validating the input data ensuring correctness and completeness. Prior to storing the user's password in the database, the password will undergo a secure hashing process. The AuthController will also generate a user ID and assign it to the newly registered user. This unique identifier becomes a key element in distinguishing and managing user data. Additionally, if the role of the user selected is Hawker or Driver, the system will require additional inputs such as Vehicle Number, License Number for Driver and Business Name, Operating Hours, Food Type, Postal Code for Hawker. Information managed by AuthController in the sign up process includes Name, Email, Address, Contact Number, Role, Password, Confirm Password and Profile Picture.

II. Equivalence Class and Boundary Value Testing

Equivalence Class Testing

A basic black-box test design technique in which test cases are designed to execute representatives from equivalence partitions. Equivalence classes (partitions) are portions of an input or output domain. The behavior of a component or system is assumed to be the same for every member of a partition class. Test cases are designed to cover each partition at least once. The operation of equivalence partitioning is performed by splitting a set (domain) into two or more disjoint sets where all the members of each subset share some trait in common ie.

- Produces the same output
- If one catches a bug, the others probably will too
- If one value does not catch a bug, the others probably will not either

This trait is not shared with the members of other subsets.
Valid equivalence classes describe valid situations and the system should handle them normally.
Invalid equivalence classes describe invalid situations and the system should reject them.

Boundary Value Testing

Boundary Value Testing is an extension of equivalence partitioning and applies only when the members of an equivalence class are ordered. It focuses on testing the boundaries of the input domain. It checks how the system behaves at the edges or limits of acceptable input values.

1. Login Function

Valid Equivalence Class: Usernames and passwords with correct formats.
Invalid equivalence Class: Usernames and passwords with incorrect formats or missing information.

2. Sign up function

Valid Equivalence Class: Name, Email, Address, Contact Number, Role, Password, Confirm Password and Profile Picture with correct formats
Invalid Equivalence Class: Name, Email, Address, Contact Number, Role, Password, Confirm Password and Profile Picture with incorrect formats or missing information

- If role selected is Hawker:

Valid Equivalence Class: Business Name, Operating Hours, Food Type and Postal Code with correct formats.
Invalid equivalence Class: Business Name, Operating Hours, Food Type and Postal Code with incorrect formats or missing information.

- If role selected is Driver:

Valid Equivalence Class: Vehicle number, License number with correct formats.
Invalid equivalence Class: Vehicle number, License number with incorrect formats or missing information.

III. Test Cases and Results

a) Login

Input parameters: Email and Password

No.	Test Input	Expected Output	Actual Output	Pass?
1	**(Valid)** Email: “admin1@gmail.com” **(Valid)** Password: “Password”	Successful login	Successful login	Yes
2	**(Invalid)** Email: “” **(Valid)** Password: “Password”	Login failed, system notify “Please enter your email and password!”	Login failed, system notify “Please enter your email and password!”	Yes
3	**(Invalid)** Email: “test123” **(Valid)** Password: “Password”	Login failed, system notify “Please include an @ in your email.”	Login failed, system notify “Please include an @ in your email.”	Yes
4	**(Valid)** Email: “admin1@gmail.com” **(Invalid)** Password: “”	Login failed, system notify “Please enter your email and password!”	Login failed, system notify “Please enter your email and password!”	Yes
5	**(Valid)** Email: “admin1@gmail.com” **(Invalid)** Password: “Pass123”	Login failed, system notify “Password length must be at least 8 characters long!”	Login failed, system notify “Password length must be at least 8 characters long!”	Yes

b) Sign up

Input Parameters: Name, Email, Address, Contact Number, Role, Password, Confirm Password and Profile Picture

No.	Test Input	Expected Output	Actual Output	Pass?
1	**(All Valid Inputs)** Name: “Admin Jane” Email: “admin1@gmail.com” Address: “160 Paya Lebar Road 07-08 Orion Industrial Building” Contact Number: “98553641” Password: “Password” Confirm Password: “Password”	Successful signup	Successful signup	Yes
2	**(All Valid Inputs except Name)** Name: “”	System notify “Please input all fields.”	System notify “Please input all fields.”	Yes
3	**(All Valid Inputs except Email)** Email: “”	System notify “Please input all fields.”	System notify “Please input all fields.”	Yes
4	**(All Valid Inputs except Address)** Address: “”	System notify “Please input all fields.”	System notify “Please input all fields.”	Yes
5	**(All Valid Inputs except Contact Number)** Contact Number: “”	System notify “Please input all fields.”	System notify “Please input all fields.”	Yes
6	**(All Valid Inputs except Password)** Password: “”	System notify “Please input all fields.”	System notify “Please input all fields.”	Yes
7	**(All Valid Inputs except Password)** Password: “pass123”	System notify “Password must be at least 8 characters.”	System notify “Password must be at least 8 characters.”	Yes
8	**(All Valid Inputs except Confirm Password)** Confirm Password: “”	System notify “Please input all fields.”	System notify “Please input all fields.”	Yes
9	**(All Valid Inputs except Confirm Password)** Password: “Password” Confirm Password: “Pass”	System notify “Password and Confirm Password must match.”	System notify “Password and Confirm Password must match.”	Yes

10	**(All Valid Inputs except Profile Picture)** No picture/file uploaded	System notify “Please upload a photo.”	System notify “Please upload a photo.”	Yes
11	**(All Valid Inputs except Profile Picture)** Incorrect file type uploaded	System notify “Failed to upload photo!”	System notify “Failed to upload photo!”	Yes

c) Signup - if role selected is Driver

Additional input parameters: Vehicle number, License number

No.	Test Input	Expected Output	Actual Output	Pass?
1	**(Valid)** Vehicle Number: “1GKLVKED8AJ155580” **(Valid)** License Number: “C006-340-56-864-0”	Successful signup	Successful signup	Yes
2	**(Invalid)** Vehicle Number: “” **(Valid)** License Number: “C006-340-56-864-0”	System notify “Please input all fields.”	System notify “Please input all fields.”	Yes
3	**(Valid)** Vehicle Number: “1GKLVKED8AJ155580” **(Invalid)** License Number: “”	System notify “Please input all fields.”	System notify “Please input all fields.”	Yes

d) Signup - if role selected is Hawker

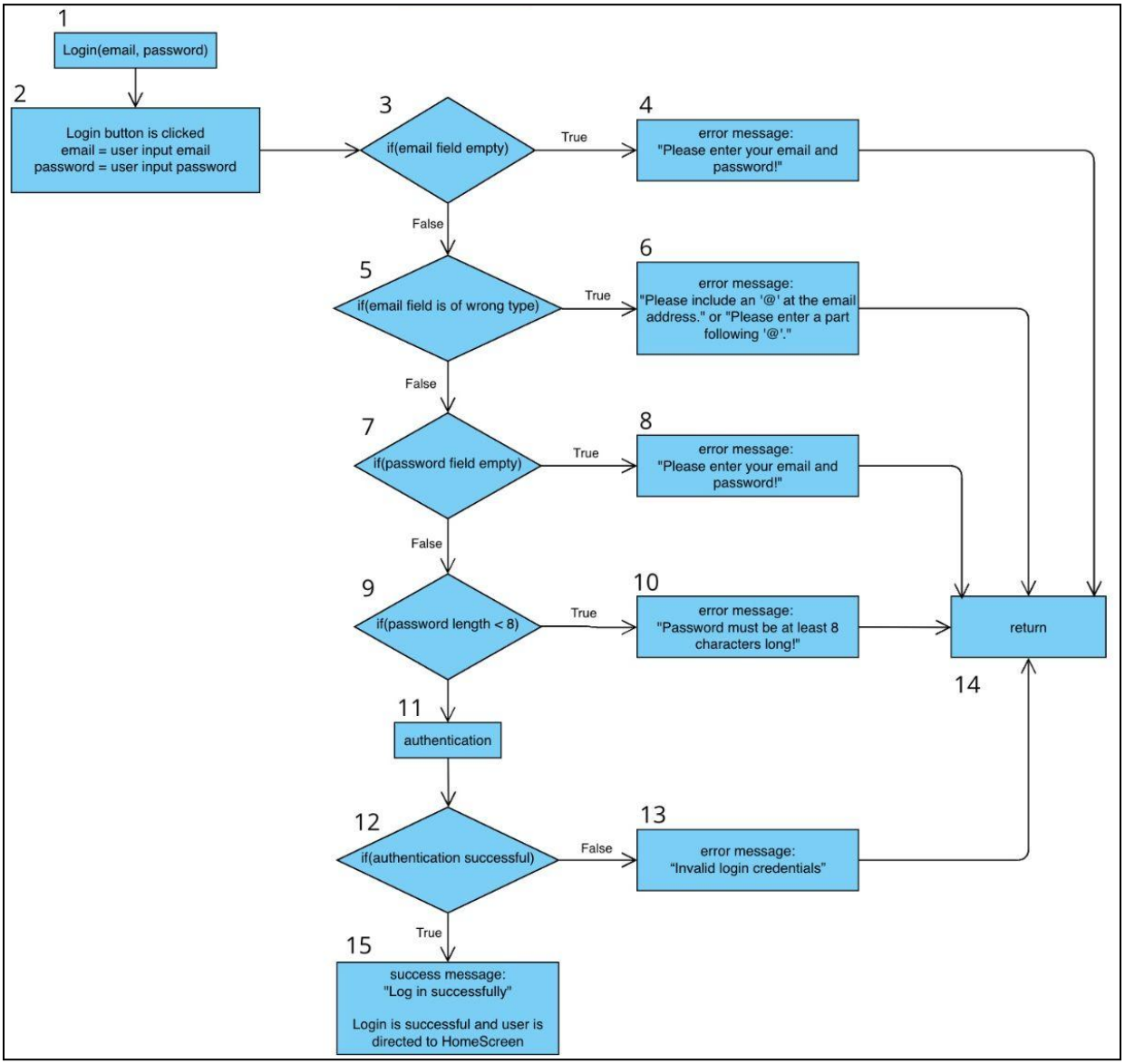
Additional input parameters: Business Name, Operating Hours, Food Type, Postal Code

No.	Test Input	Expected Output	Actual Output	Pass?
1	**(Valid)** Business Name: “A Hot Hideout (NTU)” **(Valid)** Operating Hours: “9am - 5pm” **(Valid)** Food Type: “Mala” **(Valid)** Postal Code: “636957”	Successful signup	Successful signup	Yes
2	**(Invalid)** Business Name: “” **(Valid)** Operating Hours: “9am - 5pm” **(Valid)** Food Type: “Mala” **(Valid)** Postal Code: “636957”	System notify “Please input all fields.”	System notify “Please input all fields.”	Yes
3	**(Valid)** Business Name: “A Hot Hideout (NTU)” **(Invalid)** Operating Hours: “” **(Valid)** Food Type: “Mala” **(Valid)** Postal Code: “636957”	System notify “Please input all fields.”	System notify “Please input all fields.”	Yes
4	**(Valid)** Business Name: “A Hot Hideout (NTU)” **(Valid)** Operating Hours: “9am - 5pm” **(Invalid)** Food Type: “” **(Valid)** Postal Code: “636957”	System notify “Please input all fields.”	System notify “Please input all fields.”	Yes
5	**(Valid)** Business Name: “A Hot Hideout (NTU)” **(Valid)** Operating Hours: “9am - 5pm” **(Valid)** Food Type: “Mala” **(Invalid)** Postal Code: “”	System notify “Please input all fields.”	System notify “Please input all fields.”	Yes
6	**(Valid)** Business Name: “A Hot Hideout (NTU)” **(Valid)** Operating Hours: “9am - 5pm” **(Valid)** Food Type: “Mala” **(Invalid)** Postal Code: “123456”	System notify “Invalid Postal Code!”	System notify “Invalid Postal Code!”	Yes

2. White Box Testing

I. Login

I.I Control Flow Graph



If the image is unclear, please refer to the raw image file that is uploaded together with this document.

I.II Basic Path Testing

Cyclomatic Complexity = | decision points | + 1 = 5 + 1 = 6

Basis Paths

1. Baseline path: 1, 2, 3, 5, 7, 9, 11, 12, 15
2. Basis path 2: 1, 2, 3, 4, 14
3. Basis path 3: 1, 2, 3, 5, 6, 14
4. Basis path 4: 1, 2, 3, 5, 7, 8, 14
5. Basis path 5: 1, 2, 3, 5, 7, 9, 10, 14
6. Basis path 6: 1, 2, 3, 5, 7, 9, 11, 12, 13, 14

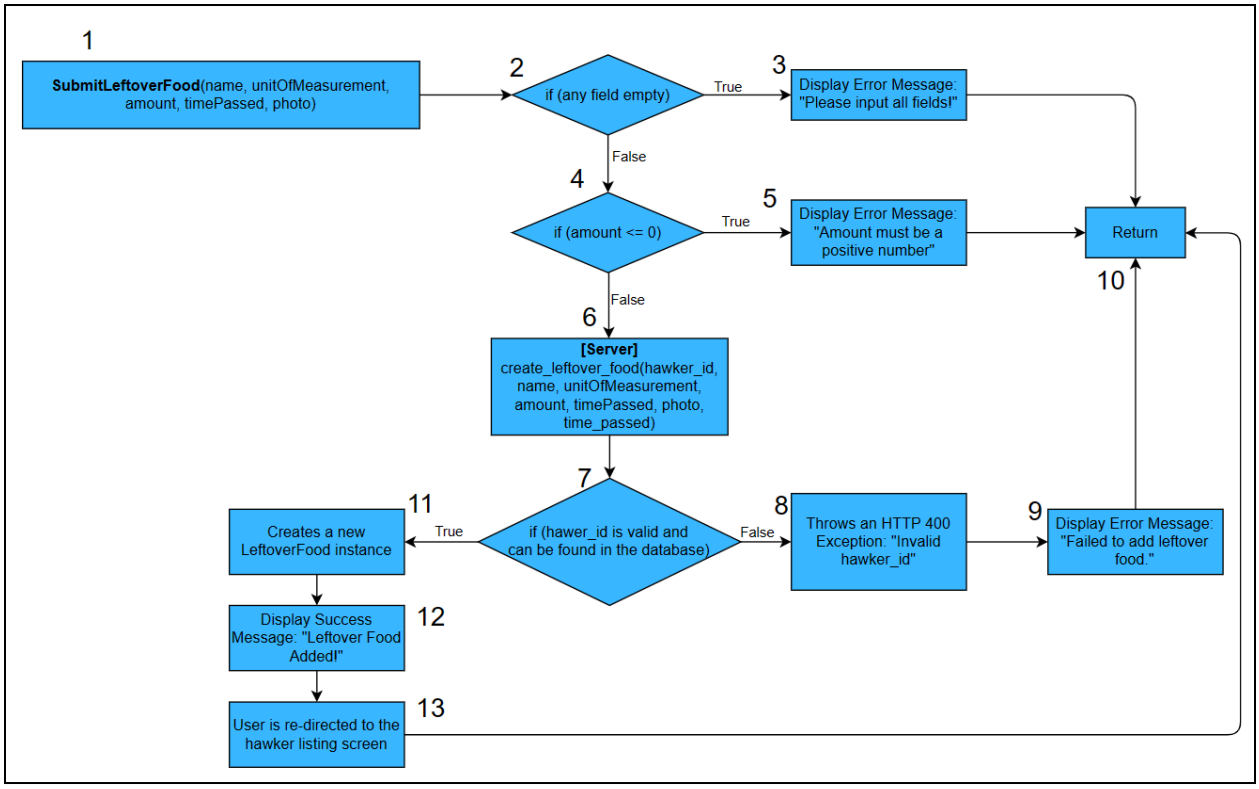
I.III Test Cases and Results

Login(email, password)

No.	Test Input	Expected Output	Actual Output	Pass?
1	email = “admin1@gmail.com” password = “123123123”	"Log in successfully"	"Log in successfully"	Yes
2	email = “” password = “123123123”	"Please enter your email and password!"	“Please enter your email and password!"	Yes
3	email = “admin1” password = “123123123”	"Please include an '@' at the email address."	"Please include an '@' at the email address."	Yes
4	email = “admin1@” password = “123123123”	“Please enter a part following '@'.”	“Please enter a part following '@'.”	Yes
5	email = “admin1@gmail.com” password = “”	"Please enter your email and password!"	“Please enter your email and password!"	Yes
6	email = “admin1@gmail.com” password = “1234567”	"Password must be at least 8 characters long!"	"Password must be at least 8 characters long!"	Yes
7	email = “pinyang@gmail.com” password = “123123123”	“Invalid login credentials”	“Invalid login credentials”	Yes

II. SubmitLeftoverFood

II.I Control Flow Graph



If the image is unclear, please refer to the raw image file that is uploaded together with this document.

II.II Basic Path Testing

Cyclomatic Complexity = | decision points | + 1 = 3 + 1 = 4

Basic Paths

1. Baseline path: 1, 2, 4, 6, 7, 11, 12, 13, 10
2. Basis path 2: 1, 3, 10
3. Basis path 3: 1, 5, 10
4. Basis path 4: 1, 2, 4, 5, 7, 8, 9, 10

I.III Test Cases and Results

SubmitLeftoverFood(name, unitOfMeasurement, amount, timePassed, photo)

- **Inputs:** name, unitOfMeasurement, amount, timePassed, photo
- **Implied Input:** hawker_id (passed automatically by inferring from the current logged in Hawker User)

No.	Test Input	Expected Output	Actual Output	Pass?
1	name = “Nasi Lemak” unitOfMeasurement = “packets” amount = 5 timePassed = “1 hour” photo = “nasilemak.jpg” hawker_id = 5	Display success message: “Leftover food added!”	Display success message: “Leftover food added!”	Yes
2	name = “” unitOfMeasurement = “packets” amount = 5 timePassed = “1 hour” photo = “nasilemak.jpg” hawker_id = 5	Display error message: “Please input all fields!”	Display error message: “Please input all fields!”	Yes
3	name = “Nasi Lemak” unitOfMeasurement = “packets” amount = -1 timePassed = “1 hour” photo = “nasilemak.jpg” hawker_id = 5	Display error message: “Amount must be a positive number.”	Display error message: “Amount must be a positive number.”	Yes
4	name = “Nasi Lemak” unitOfMeasurement = “packets” amount = “5” timePassed = “1 hour” photos = nasilemak.jpg hawker_id = -1 (invalid and not found in database)	HTTP 400 Exception thrown in the server with message “Invalid hawker_id”. Display error message: “Failed to add leftover food.”	HTTP 400 Exception thrown in the server with message “Invalid hawker_id”. Display error message: “Failed to add leftover food.”	Yes