

**NANYANG  
TECHNOLOGICAL  
UNIVERSITY**  

---

**SINGAPORE**

<b>Lab Group</b>	SCED
<b>Team</b>	Open Circuit
<b>Members</b>	Leck Kye-Cin (U2423244A)
	Lee Jun De, Kavan (U2423202E)
	Lee Yong Liang (U2422741K)
	Lee Zi An (U2323563A)
	Lim Seow Kiat (U2420309G)
	Lim Xiao Xuan (U2423602D)

## Table of Contents

<b>1. Black Box Testing</b>	<b>3</b>
I. ReviewController	3
II. Equivalence Class & Boundary Value Testing	4
Equivalence Class Testing	4
Boundary Value Testing	4
1. Create Review Function	4
2. Delete Review Function	4
3. Retrieve Reviews Function	4
III. Test Cases & Results	5
A. Create Review (Consumer)	5
B. Delete Reviews (Consumer)	7
C. Retrieve Reviews (Consumer)	7
D. Retrieve Reviews (Business)	8
<b>2. White Box Testing</b>	<b>9</b>
I. Login	9
I.I Control Flow Graph	9
I.II Basic Path Testing	10
I.III Test Cases & Results	10
II. AddMenuItem	11
I.I Control Flow Graph	11
I.II Basic Path Testing	12
I.III Test Cases & Results	12

# 1. Black Box Testing

## I. ReviewController

### Control class to test — ReviewController

The **ReviewController** manages all review-related operations within the HawkerSG application. Each stall can receive reviews and average ratings, which are retrievable by both consumers and business owners through the provided endpoints.

The **ReviewController** is a crucial component responsible for handling **user-submitted reviews** on hawker stalls. When a consumer submits a review, the controller first validates that all required inputs such as the target type, target id, and star rating are provided in the correct format. It then performs additional logic such as:

- Checking if the consumer exists in the system.
- Running content moderation on the review text using `guard_review_text()` to ensure that the submitted feedback is appropriate, through the help of LLMs to identify foul and sensitive reviews.
- Verifying that the consumer has not already reviewed the same target (to prevent duplicates).

Once the review passes validation, the controller stores it in the database with attributes such as the star rating, description, images, and ownership details. Successfully created reviews are then displayed to other users through our frontend.

Apart from creating reviews, the ReviewController also supports **review retrieval** and **deletion**.

- When **retrieving** reviews, it returns all reviews associated with a specific hawker stall or consumer.
- When **deleting** a review, the controller verifies ownership to ensure that only the original consumer can remove their review.  
This will ensure data integrity and prevent unauthorised actions.

Information managed by the ReviewController includes:

- **Consumer ID**
- **Target Type**
- **Target ID**
- **Star Rating**
- **Description**
- **Images**
- **Created At** and **Updated At**

## II. Equivalence Class & Boundary Value Testing

### Equivalence Class Testing

Equivalence class testing divides the input domain into valid and invalid partitions to verify that each produces consistent outcomes.

**Valid Equivalence Classes** represent acceptable inputs (e.g., properly formatted reviews).

**Invalid Equivalence Classes** represent disallowed inputs (e.g., missing or incorrect data, duplicate submissions, or inappropriate

### Boundary Value Testing

#### 1. Create Review Function

- **Valid Equivalence Class:**  
Review input fields (target\_type, target\_id, star\_rating, description) provided in the correct format and within boundaries (star\_rating 1–5, description ≤ 250 chars).
- **Invalid Equivalence Class:**  
Missing or invalid inputs, duplicate reviews, or descriptions violating moderation rules.

#### 2. Delete Review Function

- **Valid Equivalence Class:**  
Review exists and belongs to the authenticated consumer.
- **Invalid Equivalence Class:**  
Review ID does not exist or does not belong to the consumer.

#### 3. Retrieve Reviews Function

- **Valid Equivalence Class:**  
Valid target\_id and target\_type provided, with existing reviews.
- **Invalid Equivalence Class:**  
Invalid or non-existent target\_id or target\_type.

### III. Test Cases & Results

#### A. Create Review (Consumer)

No.	Test Input	Test Output	Expected Output	Pass?
1	<b>(Valid)</b> Rating: 5 stars <b>(Valid)</b> Review: "Amazing food, super crispy!"	Review successfully created and displayed under selected stall	Review successfully created and displayed under selected stall	Yes
2	<b>(Invalid)</b> Missing rating <b>(Valid)</b> Review: "Crispy tofu, will order again!"	Error message: "Please provide a rating."  Review not submitted	Error message: "Please provide a rating."  Review not submitted	Yes
3	<b>(Valid)</b> Rating: 5 stars <b>(Valid)</b> Empty review text	Review successfully created and displayed under selected stall	Review successfully created and displayed under selected stall	Yes
4	<b>(Valid)</b> Rating: 5 stars <b>(Invalid)</b> Review exceeds 250 characters	Consumer is not allowed to type further. Review successfully created and displayed under selected stall	Consumer is not allowed to type further. Review successfully created and displayed under selected stall	Yes
5	<b>(Valid)</b> Rating: 4 <b>(Invalid)</b> Review: "This stall is	Review blocked by moderation: "Inappropriate	Review blocked by moderation: "Inappropriate	Yes

	sh*t, don't buy here"	review content."	review content."	
6	<b>(Invalid)</b> Duplicate review by same consumer for same stall	Error: "You've already reviewed this stall."	Error: "You've already reviewed this stall."	Yes
7	<b>(Valid)</b> Rating: 5 <b>(Valid)</b> Review: "Loved it!" <b>(Valid)</b> 3 uploaded photos	Review successfully created with uploaded images displayed under selected stall	Review successfully created with uploaded images displayed under selected stall	Yes
8	<b>(Valid)</b> Rating: 5 <b>(Valid)</b> Review: "Great food!" <b>(Invalid)</b> 6 uploaded photos	Error message: "You can only upload up to 5 photos."	Error message: "You can only upload up to 5 photos."	Yes

## B. Delete Reviews (Consumer)

No.	Test Input	Test Output	Expected Output	Pass?
1	<b>(Valid)</b> Consumer ID: 12  <b>(Valid)</b> Review ID: 45 (belongs to the consumer)	Review successfully deleted.  Confirmation message displayed: "Review deleted."	Review successfully deleted.  Confirmation message displayed: "Review deleted."	Yes
2	<b>(Valid)</b> Consumer ID: 12  <b>(Invalid)</b> Review ID: 9999 (does not exist)	Error message: "Review not found."  Review not deleted.	Error message: "Review not found."  Review not deleted.	Yes
3	<b>(Valid)</b> Consumer ID: 12  <b>(Invalid)</b> Review ID: 56 (belongs to another consumer)	Error message: "Review not found or not owned by consumer."  Review not deleted.	Error message: "Review not found or not owned by consumer."  Review not deleted.	Yes

## C. Retrieve Reviews (Consumer)

No.	Test Input	Test Output	Expected Output	Pass?
1	<b>(Valid)</b> Consumer ID: 12 (consumer with 3 reviews)	List of all reviews written by this consumer displayed successfully.	List of all reviews written by this consumer displayed successfully.	Yes
2	<b>(Valid)</b> Consumer ID: 12 (consumer with no reviews)	Empty list displayed with message: "No reviews yet."	Empty list displayed with message: "No reviews yet."	Yes
3	<b>(Invalid)</b> Consumer ID: 9999 (non-existent consumer)	Error message: "Consumer not found."	Error message: "Consumer not found."	Yes
4	<b>(Valid)</b> Consumer ID: 12 (reviews include images)	All reviews displayed correctly, including associated images.	All reviews displayed correctly, including associated images.	Yes

#### D. Retrieve Reviews (Business)

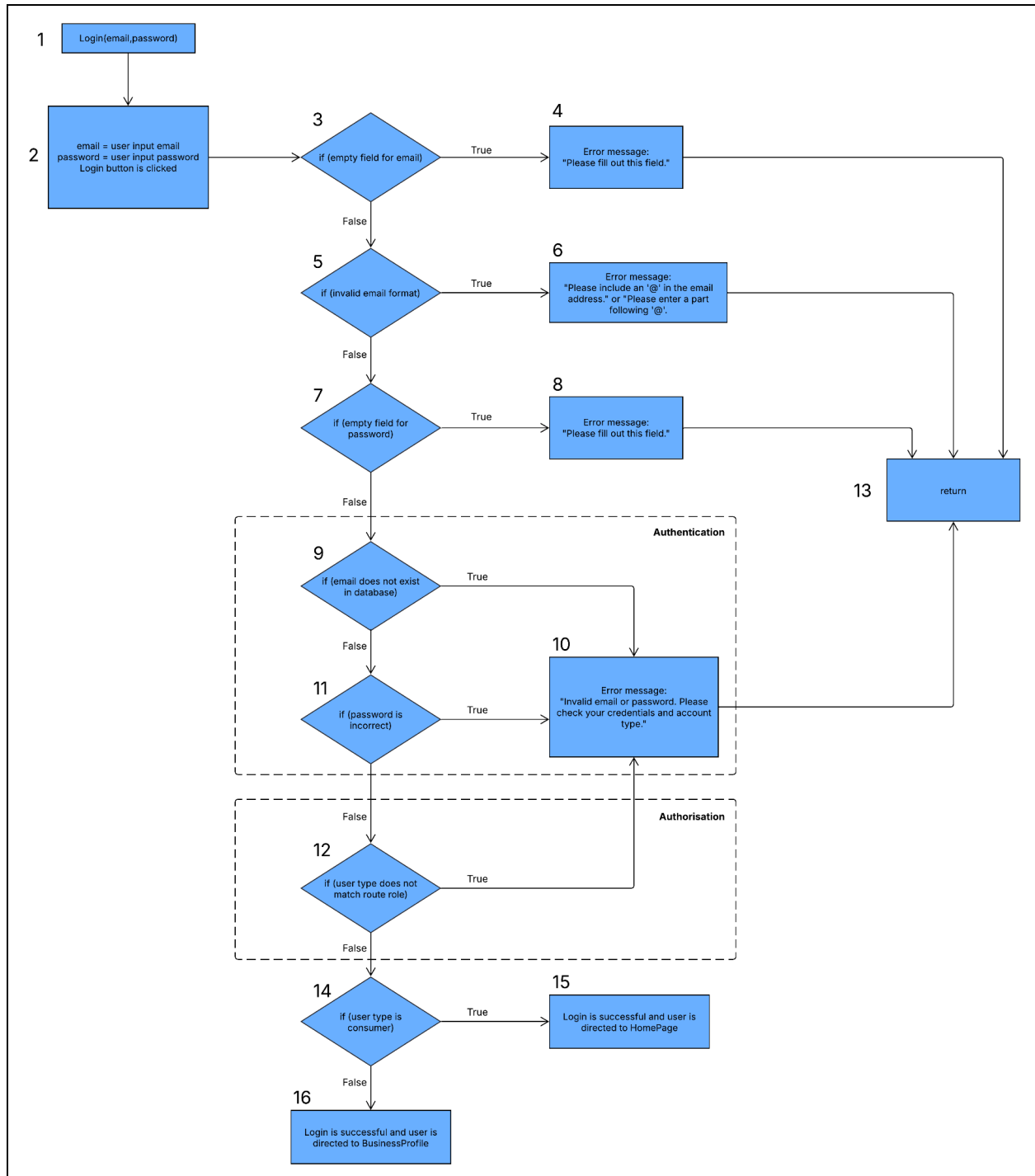
No.	Test Input	Test Output	Expected Output	Pass?
1	<b>(Valid)</b> Target Type: "business" <b>(Valid)</b> Target ID: 21 (stall exists with 5 reviews)	All 5 reviews displayed successfully under this business.	All 5 reviews displayed successfully under this business.	Yes
2	<b>(Valid)</b> Target Type: "business" <b>(Invalid)</b> Target ID: 9999 (non-existent stall)	Message displayed: "No reviews found."	Message displayed: "No reviews found."	Yes
3	<b>(Invalid)</b> Target Type: "invalid_type" <b>(Valid)</b> Target ID: 21	Error message: "Invalid target type."	Error message: "Invalid target type."	Yes
4	<b>(Valid)</b> Target Type: "business" <b>(Valid)</b> Target ID: 21 (multiple reviews with different timestamps)	Reviews displayed in descending order by creation date (newest first).	Reviews displayed in descending order by creation date (newest first).	Yes
5	<b>(Valid)</b> Target Type: "business" <b>(Valid)</b> Target ID: 21 (reviews include photos)	All reviews displayed correctly with corresponding photos.	All reviews displayed correctly with corresponding photos.	Yes



## 2. White Box Testing

### I. Login

#### I.1 Control Flow Graph



## I.II Basic Path Testing

**Cyclomatic Complexity** = |decision points| + 1 = 7 + 1 = 8

### Basis Paths

1. Baseline path: 1, 2, 3, 5, 7, 9, 11, 12, 14, 16
2. Basis path: 1, 2, 3, 5, 7, 9, 11, 12, 14, 15
3. Basis path: 1, 2, 3, 4, 13
4. Basis path: 1, 2, 3, 5, 6, 13
5. Basis path: 1, 2, 3, 5, 7, 8, 13
6. Basis path: 1, 2, 3, 5, 7, 9, 10, 13
7. Basis path: 1, 2, 3, 5, 7, 9, 11, 10, 13
8. Basis path: 1, 2, 3, 5, 7, 9, 11, 12, 10, 13

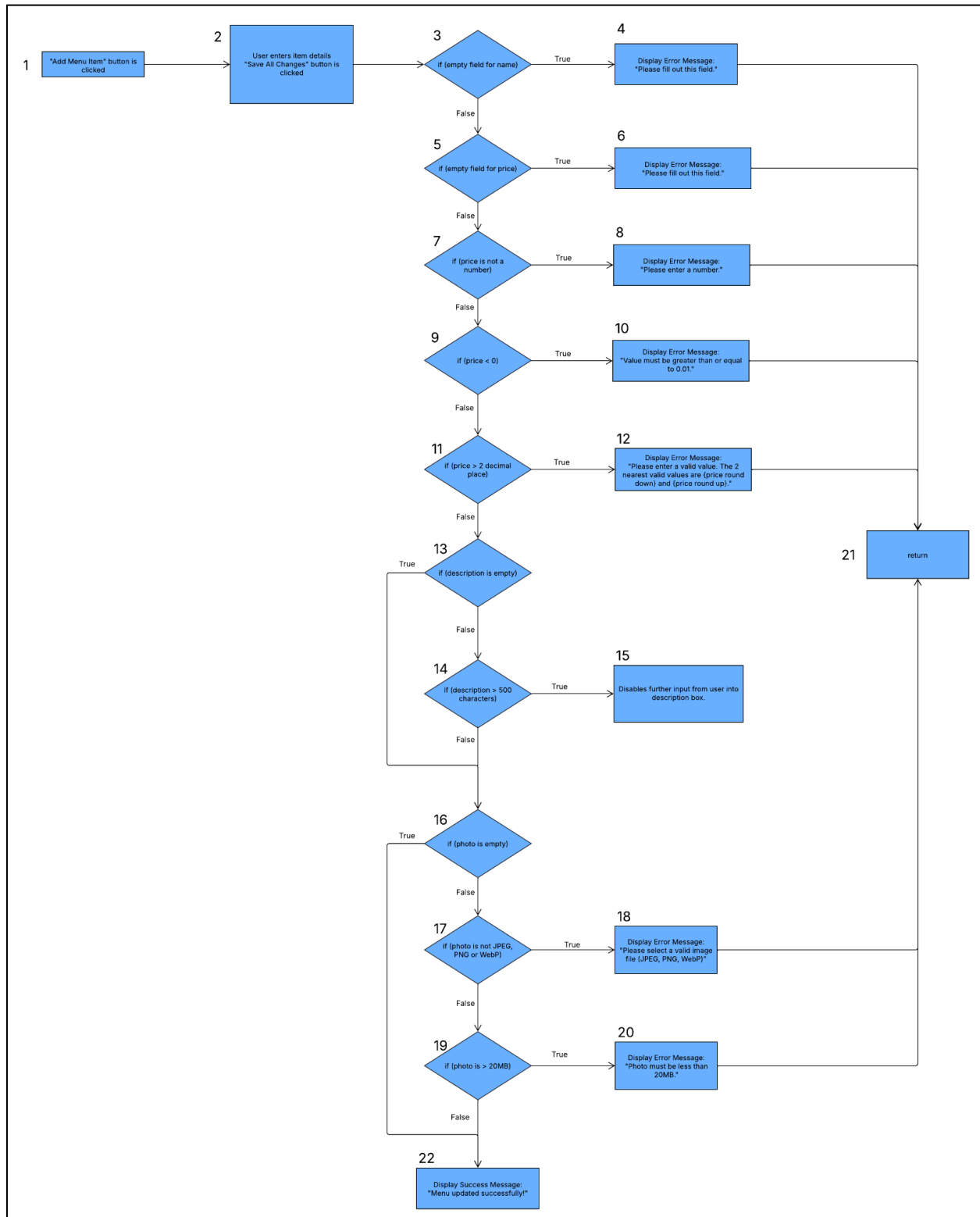
## I.III Test Cases & Results

Login(Email, Password, User Type)

No.	Test Input			Expected Output	Test Output	Pass?
	Email	Password	User Type			
1	business@gmail.com	Password@123	Business	Redirected to BusinessProfile	Redirected to BusinessProfile	Yes
2	consumer@gmail.com	Password@123	Consumer	Redirected to HomePage	Redirected to HomePage	Yes
3		Password@123	Consumer	Please fill out this field	Please fill out this field	Yes
4a	consumer	Password@123	Consumer	Please include an '@' in the email address	Please include an '@' in the email address	Yes
4b	consumer@	Password@123	Consumer	Please enter a part following '@'	Please enter a part following '@'	Yes
5	consumer@gmail.com		Consumer	Please fill out this field	Please fill out this field	Yes
6	FakeUser@gmail.com	Password@123	Consumer	Invalid email or password	Invalid email or password	Yes
7	user@gmail.com	WrongPass@123	Consumer	Invalid email or password	Invalid email or password	Yes
8	consumer@gmail.com	Password@123	Business	Invalid email or password	Invalid email or password	Yes

## II. AddMenuItem

### I.I Control Flow Graph



## I.II Basic Path Testing

**Cyclomatic Complexity** = |decision points| + 1 = 10 + 1 = 11


### Basis Paths

1. Baseline Path 1: 1, 2, 3, 5, 7, 9, 11, 13, 14, 16, 17, 19, 22
2. Baseline Path 2: 1, 2, 3, 5, 7, 9, 11, 13, 16, 22
3. Basis Path: 1, 2, 3, 4, 21
4. Basis Path: 1, 2, 3, 5, 6, 21
5. Basis Path: 1, 2, 3, 5, 7, 8, 21
6. Basis Path: 1, 2, 3, 5, 7, 9, 10, 21
7. Basis Path: 1, 2, 3, 5, 7, 9, 11, 12, 21
8. Basis Path: 1, 2, 3, 5, 7, 9, 11, 13, 14, 15
9. Basis Path: 1, 2, 3, 5, 7, 9, 11, 13, 16, 17, 19, 22
10. Basis Path: 1, 2, 3, 5, 7, 9, 11, 13, 14, 16, 17, 18, 21
11. Basis Path: 1, 2, 3, 5, 7, 9, 11, 13, 14, 16, 17, 19, 20, 21

## I.III Test Cases & Results

AddMenuItem(name, price, description, photo)

- Inputs: name, price, description, photo
- Implied input: license\_number (retrieved from the business account that is logged in).

No.	Test Input	Test Output	Expected Output	Pass?
1	Name = "Fishball Noodle" Price = 5.50 Description = "Delicious!" Photo = "fishball_noodle.jpg" (5 MB)  license_number = "SFA-123"	Display success banner. "Menu updated successfully!"	Display success banner. "Menu updated successfully!"	Yes
2	Name = "Fishball Noodle" Price = 5.50 Description = "" Photo = ""  license_number = "SFA-123"	Display success banner. "Menu updated successfully!"	Display success banner. "Menu updated successfully!"	Yes
3	Name =  Price = 5.50 Description = "Delicious!" Photo = "fishball_noodle.jpg" (5 MB)  license_number = "SFA-123"	Display error. "Please fill out this field."	Display error. "Please fill out this field."	Yes

4	Name = "Fishball Noodle" Price = "" Description = "Delicious!" Photo = "fishball_noodle.jpg" (5 MB)  license_number = "SFA-123"	Display error. "Please fill out this field."	Display error. "Please fill out this field."	Yes
5	Name = "Fishball Noodle" Price = ! Description = "Delicious!" Photo = "fishball_noodle.jpg" (5 MB)  license_number = "SFA-123"	Display error. "Please enter a number."	Display error. "Please enter a number."	Yes
6	Name = "Fishball Noodle" Price = -1 Description = "Delicious!" Photo = "fishball_noodle.jpg" (5 MB)  license_number = "SFA-123"	Display error. "Value must be greater than or equal to 0.01."	Display error. "Value must be greater than or equal to 0.01."	Yes
7	Name = "Fishball Noodle" Price = 1.55555 Description = "Delicious!" Photo = "fishball_noodle.jpg" (5 MB)  license_number = "SFA-123"	Display error. "Please enter a valid value. The 2 nearest valid values are 1.55 and 1.56."	Display error. "Please enter a valid value. The 2 nearest valid values are 1.55 and 1.56."	Yes
8	Name = "Fishball Noodle" Price = 5.50 Description = "Lorem ipsum dolor sit amet consectetur adipiscing elit. Quisque faucibus ex sapien vitae pellentesque sem placerat. In id cursus mi pretium tellus dui convallis. Tempus leo eu aenean sed diam urna tempor. Pulvinar vivamus fringilla lacus nec metus bibendum egestas. laculis massa nisl malesuada lacinia integer nunc posuere. Ut hendrerit semper vel class aptent taciti sociosqu. Ad litora torquent per conubia	User input into the Description box is disabled.	User input into the Description box is disabled.	Yes

	nostra inceptos himenaeos. Lorem ipsum dolor sit amet consectetur adipiscing elit. Quisqu" Photo = "fishball_noodle.jpg" (5 MB)  license_number = "SFA-123"			
9	Name = "Fishball Noodle" Price = 5.50 Description = "" Photo = "fishball_noodle.jpg" (5 MB)  license_number = "SFA-123"	Display success banner. "Menu updated successfully!"	Display success banner. "Menu updated successfully!"	Yes
10	Name = "Fishball Noodle" Price = 5.50 Description = "Delicious!" Photo = "fishball_noodle.pdf" (5 MB)  license_number = "SFA-123"	Display error. "Please select a valid image file (JPEG, PNG, WebP)"	Display error. "Please select a valid image file (JPEG, PNG, WebP)"	Yes
11	Name = "Fishball Noodle" Price = 5.50 Description = "Delicious!" Photo = "fishball_noodle.jpg" (22 MB)  license_number = "SFA-123"	Display error. "Photo must be less than 20MB."	Display error. "Photo must be less than 20MB."	Yes