

# EECS 2311 Take Home Assignment

Team: Salon AI

Section: [SectionName]

Student: Ali

March 24, 2025

## End-to-End Manual Testing

### User Story 1: Customer: Register and log in

#### Test Case 1: Successful Registration

1. Navigate to the homepage
2. Click on "Register" in the navigation bar
3. Fill in all required fields:
  - First Name: "Test"
  - Last Name: "User"
  - Email: "testuser@example.com"
  - Password: "SecurePass123!"
  - Confirm Password: "SecurePass123!"
  - Phone Number: "416-555-1234"
4. Click "Register" button
5. Verify successful registration message
6. Verify redirection to the login page

**Expected Result:** User is successfully registered, shown a confirmation message, and redirected to login.

#### Test Case 2: Registration with Existing Email

1. Navigate to the homepage
2. Click on "Register"
3. Fill in all required fields with a previously registered email
4. Submit the form

**Expected Result:** System should display an error message indicating the email is already registered.

**Test Case 3: Successful Login**

1. Navigate to the homepage
2. Click on "Login"
3. Enter valid credentials (email and password)
4. Click "Login" button
5. Verify successful login
6. Verify redirection to customer dashboard

**Expected Result:** User is successfully logged in and redirected to their dashboard.

**Test Case 4: Login with Invalid Credentials**

1. Navigate to the homepage
2. Click on "Login"
3. Enter invalid email and/or password
4. Click "Login" button

**Expected Result:** System should display an error message about invalid credentials.

**Test Case 5: Password Reset**

1. Navigate to the login page
2. Click on "Forgot Password"
3. Enter registered email address
4. Click "Reset Password"
5. Check for confirmation message

**Expected Result:** System should display a message confirming that reset instructions have been sent to the email.

# Bug and Issue Reports

## Bug 1: Registration Form Validation Issue

**Summary:** Password validation accepts weak passwords despite stated requirements.

**Steps to Reproduce:**

1. Navigate to registration page
2. Fill in all fields with valid information
3. Enter a simple password like "password123"
4. Submit the form

**Expected Behavior:** System should reject the password for not meeting security requirements (lacking uppercase, special characters).

**Actual Behavior:** Registration proceeds with the weak password.

**Severity:** High

## Bug 2: Login Session Persistence

**Summary:** User session is lost when refreshing the page after login.

**Steps to Reproduce:**

1. Login with valid credentials
2. Refresh the browser page

**Expected Behavior:** User should remain logged in after page refresh.

**Actual Behavior:** User is redirected to the login page.

**Severity:** High

# Code Review

## Code Smells Checked

- Long methods
- Duplicated code
- Large classes
- Comments (excessive or missing)
- Dead code
- Security vulnerabilities
- Error handling

## Code Smells Detected

### 1. Insufficient Error Handling in Authentication Flow

The login and registration components don't properly handle all potential API error responses.

**Recommendation:** Implement comprehensive error handling with user-friendly messages.

### 2. Insecure Password Storage

Password validation happens client-side only, with no enforcement server-side.

**Recommendation:** Implement proper server-side validation and secure password hashing.

### 3. Duplicated Validation Logic

Form validation logic is duplicated across registration and profile update components.

**Recommendation:** Extract validation to a shared validation service.