Candy Atlas Inventory Management System – Documentation Contributions

This document presents my individual contributions to the project's documentation, including the project introduction, persona, user stories, use case specifications, and a test case.

Report Introduction

Purpose of the Original Project Report

This section provides context from the group's original project documentation. The report outlined the planning and development of an Inventory Management System, covering the system's users, functionality, design artefacts, and testing approach.

Project Context

The project involved developing a robust Inventory Management web application for a fictional candy company. The system was designed to enable inventory tracking and management by multiple user roles, including stock updates, report generation, and invoice handling. The goal was to streamline internal inventory processes through a role-based system interface.

Software Engineering Principles Applied

Agile methodology, such as Scrum, was applied for iterative development. A total of two sprints were conducted. The principle was applied to ensure the project was broken into smaller, manageable tasks, allowing features to be developed incrementally. A clear timeline for task completion was also provided, ensuring the project remained on track.

Persona

Name	Donny Baker
Details	- Age: 44 - Occupation: Candy Atlas Administrator - Role: Admin - Disability: None
Background	Donny is an administrator who works at Candy Atlas after leaving his previous job at a small cybersecurity company which went bankrupt. His daily tasks include managing user information and creating new user accounts. His work contains dealing with private information, so requires strong security for his account. Donny has been an employee of Candy Atlas for many years, so is experienced and confident with the systems.
Goals	- Easily view user information that is clearly formatted. - Manage user information, including updating, adding and deleting in a quick and efficient manner without having to waste time. - Ensure no private information is leaked
Frustrations	- Lack of confirmation after completing an action, such as updating, deleting or adding, making it difficult to know whether an action was successful or not. - Information that is formatted in an unclear way - Websites that don't give a clear and direct way to update, delete or add information
Technology	**Devices:** Laptop, mobile phone
Scenario	Donny logs into his admin account and starts his daily tasks of managing users. He checks if any new users need to be added, and proceeds to create accounts for them. If needed, he deletes or updates user information, and then receives a confirmation if the action was successful or not. Donny also maintains the system on a daily basis, handles problems and enquiries from other employees by phone call or email and accesses database history of transactions.

User Stories

Title: User Account Creation	**Priority:** Urgent	**Estimate:** 3
As an admin, **I want to** create an user account (includes email, password, name, role etc.), **so that** employees can access the system.		

Acceptance Criteria
 	
Given I am logged into my admin account,
When a new user account needs to be created,
Then the system should:
  ● Save new account information to database
  ● Display confirmation message indicating successful account creation		

Stage of Implementation: Complete		

Title: User Account Deletion	**Priority:** Urgent	**Estimate:** 3
As an admin, **I want to** delete a user account, **so that** the user cannot access the system anymore.		

Acceptance Criteria
 	
Given I am logged into my admin account,
When a user account needs to be deleted,
Then the system should:
  ● Delete user's details from database
  ● Display confirmation message indicating successful account deletion		

Stage of Implementation: Complete		

Tabular Use Case Specifications

Use Case 1

UC-001	Login
Description	The user logs in to their account using their username and password.
Actor/s	All users.
Data	User ID, Username, Password
Trigger	The user enters their username and password and clicks login.
Pre-Condition	User must have an account.
Post-Condition	User is logged in.
Primary Scenario	1. User enters username. 2. User enters password. 3. The system validates the username and password. 4. The user logs in and is directed to their dashboard.
Alternative Scenario	3.1. The login details are incorrect 3.2. The system displays an error message and prompts the user to try again.

Use Case 2

UC-002	Log Sale
Description	The sales clerk inputs details of a sale into the system.
Actor/s	Sales Clerk
Data	Sale ID, Item Name, Item Quantity, Cost
Trigger	The sales clerk clicks Add New Sale.
Pre-Condition	Sales clerk must be logged in.
Post-Condition	The sale is recorded.
Primary Scenario	1. Sales clerk selects Add New Sale. 2. Sales clerk enters sale details. 3. The system records the sale. 4. The sales clerk receives confirmation.
Alternative Scenario	2.1. Sales clerk does not fill in all required fields. 2.2. The system displays an error message and prompts the sales clerk to fill in all required fields.

Test Case

Test Case ID	TC001
Test Description	Verify that the admin can successfully create a user account
Pre-Conditions	1. The admin is logged in 2. The create user form is accessible and properly displayed
Steps to Execute	1. Navigate to Add User page 2. Enter valid full name (e.g. John Doe) 3. Enter valid email (e.g. john@gmail.com) 4. Select role (e.g. Admin) 5. Enter valid password (e.g. password1232) 6. Select branch (e.g. Candy Atlas Sheffield) 7. Click submit button
Expected Result	● The user account should be successfully created ● A confirmation message indicating successful account creation is displayed ● The data is saved correctly in the database
Status (Pass/Fail)	Pass