

Software Test Document

for

Teipon Gadget System

Version [1.0]

Prepared by

MOHAMAD HAZIK HAIKAL BIN RAZAK (2024779495)

EIMAN DAMIEN BIN ROHMAN (2024779447)

WAN MUHAMMAD NAQIB BIN WAN MOHD NAZRI (2024926951)

MUHAMMAD TAUFIQ BIN SHOEID (2024542603)

MUHAMMAD LUQMAN MU`IZZ BIN ANUAR (2024974353)

MOHAMAD SYAZMIR B MOHD SAIDI (2024556469)

MUHAMMAD MAHDI BIN MOHD NORIZAL (2024947555)

SYARIFUL HUSAINI BIN SHARI FUDDIN (2024568373)

24 January 2025

Table of Contents

1. Introduction	1
1.1 Purpose.....	1
1.2 Definitions, Acronyms and Abbreviation.....	1
1.3 Intended Audience and Reading Suggestions	2
1.3.1 Part 1 (Introduction)	2
1.3.2 Part 2 (Test Overview)	2
1.3.3 Part 3 (Test Preparation)	2
1.3.4 Part 4 (Test Summary)	3
1.3.5 Part 5 (Test Results)	3
1.4 Product Scope	3
1.5 References.....	4
2. Test Overview.....	5
2.1 Strategy	5
2.2 Result Classifications.....	5
3. Test Preparation.....	6
3.1 Hardware.....	7
3.2 Software	8
3.3 Other	8
3.4 Safety, Security And Privacy Precautions	9
4. Test Summary	10
5. Test Results.....	34
5.1 Test Instances.....	34
5.1.1 Resolved Test Incidents	34
5.1.2 Unresolved Test Incidents	34
5.2 Recommendations	34
Appendix A: Approval.....	36
Appendix B: Glossary	37
Appendix C: Activity Diagram	38
Appendix D: Use Case Diagram	39
Appendix E: Class Diagram	40

Revision History

Name	Date	Reason For Changes	Version
MOHAMAD HAZIK HAIKAL BIN RAZAK	23/1/2024	TSD	1.0

1. Introduction

1.1 Purpose

The purpose of this Test Summary Document is to provide a comprehensive overview of the testing activities and results for the Teipon Gadget System. This system is primarily designed to facilitate the sale of gadgets to users, ensuring a seamless, secure, and user-friendly purchasing experience.

This release introduces new features aimed at enhancing the system's functionality and usability, specifically Administrators. This Test Summary Document (TSD) documents the testing process carried out to validate the accuracy, performance and reliability of the newly implemented features, and at the same time ensure the continued functionality of existing system components.

1.2 Definitions, Acronyms and Abbreviation

Acronym	Definition
TMS	Transaction Management System
TSD	Test Summary Document
DBMS	Database Management System
PHP	Hypertext Preprocessor
CSS	Cascading Style Sheets
HTML	Hypertext Markup Language
JavaScript	A programming language for web development
IDE	Integrated Development Environment
Laragon	Cross-platform web server solution stack
MySQL	An open-source relational database management system
Dialogflow	A tool by Google for building conversational interfaces like chatbots.
Ngrok	A tunneling service to expose a local server to the internet.

Webhook	A way for one application to send real-time data to another application events.based on specific
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
SQL	Structured Query Language
MVC	Model-View-Controller

1.3 Intended Audience and Reading Suggestions

The Test Summary Report (TSR) is intended for individuals that are directly involved in the architecture and design of the Teipon Gadget System. This includes the project manager, software developers, software designers and software tester.

1.3.1 Part 1 (Introduction)

This section provides a summary of the Teipon Gadget System project, including its purpose, objectives, product scope, general system details, and major constraints associated with the platform.

1.3.2 Part 2 (Test Overview)

This section outlines the Teipon Gadget System test strategy, providing an overview of the testing process and the classification of results.

1.3.3 Part 3 (Test Preparation)

This section details tasks and recommendations to be addressed before executing the tests. It includes the hardware and software to be used, along with other specific preparations required before testing. This section also highlights safety, security, and privacy precautions.

1.3.4 Part 4 (Test Summary)

Readers are interested in understanding what was tested, and the outcomes should refer to this section. It includes details about unit testing, component testing, and system testing conducted for Teipon Gadget System.

1.3.5 Part 5 (Test Results)

This section documents unexpected results, problems, or defects encountered during testing. It also provides recommendations for improving the design, operation, or future testing of Teipon Gadget System based on the reported findings.

1.4 Product Scope

Primarily, Teipon Gadget System is implemented to streamline the sale of gadgets to users to ensure a smooth, secure, and user-friendly purchasing experience. The scope of Teipon Gadget System is to provide an intuitive e-commerce platform designed to meet the needs of both gadget enthusiasts and system administrators. The system aims to facilitate seamless transactions, manage inventory efficiently, and enhance the overall user experience.

This Test Summary Report (TSR) is aimed at specifying the functionalities of the software to be developed and can also assist users and administrators in using the system effectively. It focuses on the system's operational workflow, the users, and the database that supports the platform.

The system subsystem handles all the activities between the users and the database. Registered users can browse gadgets, view product details, place orders, and track their purchases. The system also includes secure payment processing and order confirmation mechanisms, ensuring a reliable transaction process.

Users will be able to access features such as browsing gadgets, adding items to the cart, and tracking order progress once registered and logged in to Teipon Gadget System. The system's objective is to offer a highly functional, reliable, and user-friendly platform that caters to gadget purchasing needs.

As for the database, it is designed to store essential data such as user credentials, product information, order details, and sales analytics. It authenticates users at login and ensures the integrity and security of sensitive data.

Lastly, administrators can log in to the system to manage its functionalities. They are empowered to perform tasks such as adding, updating, or deleting product information, managing inventory levels, and monitoring system performance. Administrators also have access to sales insights and other metrics, helping them optimize the platform for better business outcomes.

1.5 References

- Affordable Second-Hand Smartphones, Tablets & Laptops | CompAsia MY. (2022). CompAsia E-Store. <https://compasia.my/>
- DirectD. (n.d.). DirectD: Retail & wholesale of mobile devices and accessories. <https://www.directd.com.my/>
- Halomobile. (n.d.). Halomobile: Buy, trade-in & repair used Apple devices. <https://www.halomobile.com.my/>
- Mobile shop Billing software | POS Software for Mobile shop. (n.d.). <https://www.gofrugal.com/retail/electronics-pos/mobile-shop-billing-software.html>
- SRO, E. (n.d.). Example Software Requirements Specification (SRS) | REQView Documentation. <https://www.reqview.com/doc/iso-iec-ieee-29148-srs-example/>
- [SRS] [SRS.docx](#)
- [SDD] [SDD.docx](#)

2. Test Overview

2.1 Strategy

The Teipon Gadget System version was tested on the platform located on the website. The users need to go to <http://localhost/TeiponGadgetSystem/> to access the system and the system has already been tested for 2 days from 18th January 2025 to 19th January 2025.

1. Unit testing (18th January 2025 – 19th January 2025)
2. Component testing (18th January 2025 – 19th January 2025)
3. System testing (18th January 2025 – 19th January 2025)

2.2 Result Classifications

The software testing results are defined according to the following rules:

1. **Pass:** All steps meet the expected results.
2. **Fail:** One or more steps do not meet the expected results or are incomplete.

3. Test Preparation

Laragon/XAMPP is required to turn on the Teipon Gadget System. These software packages enable its local host to use its database to connect to our system where it will recognize the information taken from the credentials of the user to match with the credentials inside the database. Any input from the website will be stored in the database. Teipon Gadget System database and system stores the following information:

- **Product Information**

The product information includes details about phone gadgets, such as product names, product variants descriptions, prices, stock availability etc. This data allows the system to display accurate and up-to-date information to users browsing the store.

- **Customer Information**

Customer details such as email and password are securely stored in the database. These credentials are used for validating user logins and managing personalized features like order history and preferences.

- **Orders and Purchases**

All information regarding customer orders is stored in the database. This includes order details such as product quantities, prices, and transaction dates, enabling customers to view their purchase history.

3.1 Hardware

Teipon Gadget System will work on any computer running Windows 7 or above. It is compatible with both laptops and desktops, provided they meet the minimum or recommended requirements for optimal performance. Below are the hardware specifications:

Component	Minimum Requirement	Recommended Requirement
Processor (CPU)	Intel Core i3 (6th gen or newer)	Intel Core i5 (6th gen or newer)
RAM	2-4GB RAM	6GB RAM
Disk Storage	HDD with 40 GB or more free space	SSD with 40 GB or more free space
Graphics Card (GPU)	512MB	1GB
Operating System (OS)	Windows 7	Windows 10

The Teipon Gadget System relies on an active and stable internet connection for seamless operation, particularly to ensure that all elements imported from the web are properly loaded and function as intended. Some of the examples are the stylesheets (CSS), JavaScript Libraries, and third-party APIs.

3.2 Software

The Teipon Gadget System relies on MySQL as its database management system to store, retrieve, and manage data effectively. To facilitate this, the system uses development environments like Laragon or XAMPP, which bundle essential tools such as Apache, PHP, and MySQL, providing a reliable local server setup for the application.

3.3 Other

Teipon Gadget System integrates an AI Chatbot to enhance the user experience by providing personalized product recommendations. The chatbot leverages Artificial Intelligence to analyse user preferences, browsing behaviour, and previous purchases, ensuring relevant suggestions tailored to each user.

Purpose of the AI Chatbot:

- The AI chatbot is designed to assist users in finding gadgets based on their specific needs and preferences.
- It interacts with users through natural language, making the experience intuitive and user-friendly.

How It Works:

- **Data Input:** The chatbot gathers input through user interactions, such as inquiries about product categories, budgets, or features.
- **Recommendation Engine:** Using algorithms like content-based filtering or collaborative filtering, the chatbot compares user input against the database of gadgets stored in MySQL.

Example User Interaction:

- A user asks: *"Are there phones with 64gb storage?"*
- The AI chatbot queries the database for products under the specified price range and suggests gadgets based on specifications.

3.4 Safety, Security And Privacy Precautions

The Teipon Gadget System is designed with safety, security, and privacy in mind to ensure the protection of user data, prevent unauthorized access, and maintain the integrity of the system. Below are the key precautions implemented:

User Data Security

- All user credentials (email and password) are stored in the MySQL database with encryption techniques such as Bcrypt for passwords to prevent unauthorized access in case of a data breach.

AI Chatbot Privacy Measures

- The AI Chatbot does not collect or store unnecessary personal information beyond what is required to deliver product recommendations. Conversations are anonymized and encrypted to protect user identity.

Error Message Handling

- Clear and informative error messages are provided to ensure users understand when something goes wrong. This helps in preventing frustration while maintaining security by avoiding information leakage.
- For example: "User not logged in", "Incorrect Password" and so on.

4. Test Summary

Customer Registration

Requirement Id: R1	Requirement Description: Make a registration test.
	Test Scenario: Verify that customers can register with valid details and handle invalid inputs correctly.
Pre-Condition	<ol style="list-style-type: none"> 1. The registration form is accessible on the website. 2. All fields which are email, name, phone number, state, city, postal code, address, username, password, confirm password are visible and required.
Post-Condition	<ol style="list-style-type: none"> 1. Customer accounts are created and stored in the database. 2. Customer receives a confirmation message upon successful registration.
Test Steps	<ol style="list-style-type: none"> 1. Go to website: http://localhost/TeiponGadgetSystem/register/register.php 2. Fill in all required fields which are email, name, phone number, state, city, postal code, address, username, password, confirm password are visible and required. 3. Press the register button.

Test Case Id	Test Case Descriptions	Sample Input/Test Data	Expected Output	Results	Pass/Fail (Remarks)
TR 1.1.1	Verify successful registration with valid inputs.	Email: abu@gmail.com Name: abu phone number: 123456789	Registration success with a successful message stating "Registration	Users receive successful messages and are redirected to the login page.	Pass

		state: Selangor city: Shah Alam postal code: 40000 address: No 10 Jalan Kebun username: abu password: 123 confirm password: 123	Successful! Your account has been created successfully!" message and moves to the login page		
TR 1.1.2	Verify registration fails when passwords do not match.	Email: abu@gmail.com Name: abu phone number: 123456789 state: Selangor city: Shah Alam postal code: 40000 address: No 11 Jalan Kebun username: abu password: 123 confirm password: 123456	Customer cannot register and message Stating "Password do not match"	Registration was unsuccessful, and an error message about an invalid matching password was shown.	Pass

TR 1.1.3	Verify registration fails with an invalid email address.	Email: abugmail.com Name: abu phone number: 123456789 state: Selangor city: Shah Alam postal code: 40000 address: No 11 Jalan Kebun username: abu password: 123 confirm password: 123	Users cannot register and are prompted to enter a valid email address.	Users cannot register and are prompted to enter a valid email address.	Pass
TR 1.1.4	Verify registration fails when any required field is empty.	Email: abu@gmail.com Name: abu phone number: 123456789 state: Selangor city: Shah Alam postal code: 40000 address: ""	Users cannot register and are notified that all fields are mandatory.	Registration was unsuccessful, and an error message about empty fields was shown.	Pass

		username: abu password: 123 confirm password: 123			
TR 1.1.5	Verify registration fails with an invalid phone number.	Email: abu@gmail.com Name: abu phone number: 12345 state: Selangor city: Shah Alam postal code: 40000 address: No 11 Jalan Kebun username: abu password: 123 confirm password: 123	User cannot register and message stating, "Phone number must have 9 or 10 digits."	Registration was unsuccessful, and an error message about an invalid phone number was shown.	Pass

TR 1.1.6	Verify registration fails with an already existing username.	Email: abu@gmail.com Name: abu phone number: 0123456789 state: Selangor city: Shah Alam postal code: 40000 address: No 11 Jalan Kebun username: customer password: 123 confirm password: 123	User cannot register and is shown an error indicates that the username is already taken.	Registration was unsuccessful, and an error message about the username already being in use was shown.	Pass
-------------	--	---	--	--	------

Login

Requirement Id: R2	Requirement Description: Make Login testing
	Test Scenario: Customer/Staff/Admin wants to log in to the Teipon Gadget system.
Pre-Condition	Customer/Staff/Admin needs to fill out the login form, which includes the username and password before entering the system.
Post-Condition	Customer/Staff/Admin will enter the system after successfully logging in.

Test Steps	<ol style="list-style-type: none"> 1. Go to website: http://localhost/TeiponGadgetSystem/login/login.php 2. Choose user as Customer/Staff 3. Insert username. 4. Insert password. 5. Press the Login button.
------------	---

Test Case Id	Test Case Descriptions	Sample Input/Test Data	Expected Output	Results	Pass/Fail (Remarks)
TR 1.2.1	Verify successful login with valid credentials for admin.	Username: admin Password: 123 User: staff	Admin should be able to login into the system.	Admin successfully logs in and is redirected to the dashboard.	Pass
TR 1.2.2	Verify unsuccessful login with invalid credentials for admin.	Username: admin Password: 123456 user: customer	Receives an error message stating "Invalid username or password."	Admin is not able to log in into the system and receive error messages.	Pass
TR 1.2.3	Verify successful login with valid credentials for admin.	Username: staff Password: 123 User: staff	Staff should be able to login into the system.	Staff successfully logs in and is redirected to the dashboard.	Pass

TR 1.2.4	Verify unsuccessful login with invalid credentials for staff.	Username: staff Password: 123456 user: staff	Receives an error message stating "Invalid username or password."	Staff are not able to log in into the system and receive error messages.	Pass
TR 1.2.5	Verify successful login with valid credentials for customer.	Username: amin Password: 123 User: customer	Customers should be able to login into the system.	Customer successfully logs in and is redirected to the dashboard.	Pass
TR 1.2.6	Verify unsuccessful login with invalid credentials for Customer.	Username: amin Password: 123456 user: customer	Receives an error message stating "Invalid username or password."	Customer is not able to log in into the system and receive error messages.	Pass
TR 1.2.7	Verify login attempt with an empty username.	Username: "" Password: ""	Receives an error message stating "Username cannot be empty."	Admin/Staff/Customer is not able to log in into the system and receive error messages.	Pass

Customer cart

Requirement Id: R3	Requirement Description: Verify the functionality of the customer's cart, including adding and removing items, and ensuring items are retained after logout.
	Test Scenario: 1. Add Items to Cart 2. Removed Items from Cart 3. Cart Persistence after Logout
Pre-Condition	Customers are logged into the system.
Post-Condition	1. Cart functionality operates as expected. 2. Cart data persisted as expected.
Test Steps	1. Go to website: http://localhost/TeiponGadgetSystem/customer/customer_home.php 2. Choose an item then click button 'buy'. After that choose the phone variant and quantity. 3. On the cart button, click button remove one time. 4. On customer name button, click it then click 'logout' 5. Login back.

Test Case Id	Test Case Descriptions	Sample Input/Test Data	Expected Output	Results	Pass/Fail (Remarks)
TR 1.3.1	Verify cart increases after customers add two items.	Item: Apple iPhone 15 Variant: Blue 128gb Storage 4gb RAM Quantity: 2	Cart quantity increases by 2.	Cart quantity increases by 2.	Pass

TR 1.3.2	Verify successful removal of one item from cart.	Item: Apple iPhone 15 Remove click: one time	Cart quantity decreases by 1.	Cart quantity decreases by 1.	Pass
TR 1.3.3	Verify cart persistence after logout		The Cart contains the previously added iPhone 15.	The Cart contains the previously added iPhone 15.	Pass

Customer Checkout

Requirement Id: R4	Requirement Description: Verify that customers can successfully proceed to the payment page after adding items to their cart.
	Test Scenario: Customers ensure the items they want to purchase.
Pre-Condition	<ol style="list-style-type: none"> 1. Customer has added at least one item to the cart. 2. Customer is logged in to the system.
Post-Condition	Customer is successfully redirected to the payment page.
Test Steps	<ol style="list-style-type: none"> 1. Click on the "Cart" icon on navbar 2. Verify that the cart displays the correct items and quantities. 3. Click the "Checkout" button. 4. Verify that the customer is redirected to the payment page.

Test Case Id	Test Case Descriptions	Sample Input/Test Data	Expected Output	Results	Pass/Fail (Remarks)
TR 1.4.1	Verify customers can access checkout page		Redirect to checkout page	Can see checkout page	Pass
TR 1.4.2	Verify customer can click button 'complete purchase'		Redirect to payment page	Can see checkout page	Pass

Customer Payment

Requirement Id: R5	Requirement Description: Verify that customers can successfully upload the file in pdf/img format.
	Test Scenario: Customers make payment by uploading the file in pdf/img format.
Pre-Condition	<ol style="list-style-type: none"> 1. Customer has successfully completed the checkout process. 2. Customer is logged in to the system.
Post-Condition	Customers successfully upload the file in pdf/img format into system.
Test Steps	<ol style="list-style-type: none"> 1. Access payment page. 2. Scan QR. 3. Upload receipt into the system. 4. Click button submit.

Test Case Id	Test Case Descriptions	Sample Input/Test Data	Expected Output	Results	Pass/Fail (Remarks)
TR 1.5.1	Verify customer can upload pdf/img file less than 5mb	Upload pdf/img file less than 5mb	Customer successfully made the payment	Successful alert popup said, "Payment receipt uploaded successfully."	Pass
TR 1.5.2	Verify customer cannot upload file beside pdf/img	Upload txt file beside pdf file	Customers unsuccessfully uploaded it into system	Error alert popup said "Invalid file type. Please upload a PDF or image."	Pass
TR 1.5.3	Verify customer cannot upload file more than 5mb	Upload pdf/img file with more than 5mb	Customers unsuccessfully uploaded it into system	Error alert popup said "File Too Large"	pass
TR 1.5.4	Verify system behavior with empty file upload	Attempt to upload without selecting any file.	Customers unsuccessfully uploaded it into system	The error alert popup said, "Please select a file to upload."	pass

Customer View Purchases

Requirement Id: R6	Requirement Description: Verify that customers can successfully cancel orders and track the status of their orders.
	Test Scenario: Customers cancel their order or check their order status.
Pre-Condition	<ol style="list-style-type: none"> 1. Customers have successfully completed the checkout process for an item. 2. Customers are logged in to the system. 3. The order is currently in the "Pending Payment" status.
Post-Condition	<ol style="list-style-type: none"> 1. The order is successfully canceled by the customer. 2. The customer can view the status of their orders, including cancelled orders.
Test Steps	<ol style="list-style-type: none"> 1. Access to view purchase page. 2. Go to column 'pending payment' 3. Click button 'cancel order' 4. Track Order Status

Test Case Id	Test Case Descriptions	Sample Input/Test Data	Expected Output	Results	Pass/Fail (Remarks)
TR 1.6.1	Verify customer can cancel order		Item with status 'pending payment' was no longer in pending payment column	The order moves to column 'Order Cancelled'	Pass

TR 1.6.2	Verify customers can track their previous order		Customers will know their orders status	Customers can see their previous order status	Pass
-------------	---	--	---	---	------

Customer Profile Setting

Requirement Id: R7	Requirement Description: Customer can update their profile information
	Test Scenario: Customers update their profile information.
Pre-Condition	Customers are logged into the system.
Post-Condition	Customer's profile information is updated successfully.
Test Steps	<ol style="list-style-type: none"> 1. Go to profile setting 2. Identify Fields for Update 3. Enter New Information 4. Click submit button

Test Case Id	Test Case Descriptions	Sample Input/Test Data	Expected Output	Results	Pass/Fail (Remarks)
TR 1.7.1	Verify customer can update their details	Change: Email: abu12@gmail.com Name: abu baida phone number: 98765432	Customers successfully update their information details	Successful alert popup said, "Details updated successfully."	Pass

		state: Kelantan city: Tumpat postal code: 16200 address: No 10 Jalan Wangi username: abu12 new password: 456			
TR 1.7.2	Verify customers cannot use existing username	Username: customer	Customers unsuccessfully update their profile username	Error alert popup said, "The username is already taken."	Pass
TR 1.7.3	Verify customers cannot use existing email	Email: cust@yopmail.com	Customers unsuccessfully update their profile email	Error alert popup said, "The email address is already taken"	Pass
TR 1.7.4	Verify customer cannot update with null value	Full Name: " "	Customers unsuccessfully update their profile.	Error alert popup said, "Please fill out the field"	Pass

TR 1.7.5	Verify user must fill the email with email format	Email: abu12.com	Customers unsuccessfully update their profile email	Error alert popup said, "Please include an '@' in email address"	Pass
-------------	---	------------------	---	--	------

Admin Register Staff Account

Requirement Id: R8	Requirement Description: New staff and admin accounts be created by Admin.
	Test Scenario: Create new staff account or admin account
Pre-Condition	Admin is logged in the system.
Post-Condition	<ol style="list-style-type: none"> 1. The staff or admin account has been created. 2. New staff or admin information is added into the database.
Test Steps	<ol style="list-style-type: none"> 1. Go to website: http://localhost/TeiponGadgetSystem/admin_dashboard/admin_dashboard.php 2. Go to Register Staff 3. In actions column, choose Admin type user to create an admin account and choose Staff type user to create staff account. 4. After done fill the form with correct format and data, click the Register Admin for admin section or click Register Staff for staff section to create new account.

Test Case Id	Test Case Descriptions	Sample Input/Test Data	Expected Output	Results	Pass/Fail (Remarks)
TR 1.8.1	Verify successful registration with valid inputs.	Full Name: Abu bin Ali Username: Abu Email: abu@gmail.com Password: 123 Confirm password: 123	Registration success with a successful message stating “Success! Staff member has been registered successfully. Redirecting to the admin dashboard...”	Admins receive successful messages and are redirected to the admin dashboard.	Pass
TR 1.8.2	Verify registration fails when passwords do not match.	Full Name: Abu bin Ali Username: Abu Email: abu@gmail.com Password: 123 Confirm password: 12345	Admin cannot register and message Stating “Password do not match”	Registration was unsuccessful, and an error message about an invalid matching password was shown.	Pass
TR 1.8.3	Verify registration fails with an	Full Name: Abu bin Ali Username: Abu	Admin cannot register and are prompted	Users cannot register and are prompted	Pass

	invalid email address.	Email: abugmail.com Password: 123 Confirm password: 123	to enter a valid email address.	to enter a valid email address.	
TR 1.8.4	Verify registration fails when any required field is empty.	Full Name: Username: Abu Email: abu@gmail.com Password: 123 Confirm password: 123	Admin cannot register and are notified that all fields are mandatory.	Registration was unsuccessful, and an error message about empty fields was shown.	Pass

Admin Manage Staff Account

Requirement Id: R9	Requirement Description: Edit staff details and delete staff account by admin
	Test Scenario: Admin wants to edit staff information and delete staff account
Pre-Condition	Admins need to have a session to access the admin dashboard.
Post-Condition	1. Staff information successfully update 2. Staff information is updated in the database.
Test Steps	1. Go to website: http://localhost/TeiponGadgetSystem/admin_dashboard/admin_dashboard.php 2. Go to Manage Staff.

	<p>3. In actions column, click on the edit or delete button.</p> <p>4. After making some changes to the staff information, click the update button</p> <p>5. To make the account disable, click on delete.</p>
--	--

Test Case Id	Test Case Descriptions	Sample Input/Test Data	Expected Output	Results	Pass/Fail (Remarks)
TR 1.9.1	Verify admin can edit staff information	Change any staff details.	Staff details were updated successfully.	Successful alert popup said, "Customer details updated successfully".	Pass
TR 1.9.2	Verify admin can delete staff account	Click on the delete button.	Confirming on disabling staff account.	Alert popup said, "Customer disabled successfully"	Pass

Staff/Admin Manage Customer Account

Requirement Id: R10	Requirement Description: Edit customer's details and disable customer's account by staff/admin
	Test Scenario: Staff/admin wants to edit customer's information and disable customer account
Pre-Condition	Staff/admin need to have a session to access the Staff/Admin dashboard.
Post-Condition	<p>1. Customer successfully registers</p> <p>2. Customer information is added to the database and is displayed on the customer management page.</p>

Test Steps	<ol style="list-style-type: none"> 1. Go to website: http://localhost/TeiponGadgetSystem/manage_customer/manage_customer.php 2. Go to Manage Customer. 3. In actions column, click on the edit or disable button. 4. After making some changes to the customer's information, click the update button 5. To make the account disable, click on disable.
------------	--

Test Case Id	Test Case Descriptions	Sample Input/Test Data	Expected Output	Results	Pass/Fail (Remarks)
TR 1.10.1	Verify staff/admin can edit customer's information	Change any customer's details.	Customer details updated successfully.	Successful alert popup said, "Customer details updated successfully".	Pass
TR 1.10.2	Verify staff/admin can disable customer account	Click on the disabled button	Confirming on disabling customer account.	Alert popup said, "Customer disabled successfully"	Pass
TR 1.10.3	Verify staff/admin can enable account	Click on the enable button	Confirming on enabling back customer account.	Alert popup said, "Customer enabled successfully".	Pass

Manage products by Staff/Admin

Requirement Id: R11	Requirement Description: Staff/Admin add new / edit / delete product
	Test Scenario: Staff/admin wants to add new product
Pre-Condition	<ol style="list-style-type: none"> 1. Staff/Admin members must be logged in with appropriate permission to manage products. 2. Orders exist in the database with valid data, including statuses and associated receipts.
Post-Condition	<ol style="list-style-type: none"> 1. The product list reflects the changes (new, updated, or deleted products) 2. All changes are saved and synchronized in the database
Test Steps	<ol style="list-style-type: none"> 1. Go to website: http://localhost/TeiponGadgetSystem/product/manage_product.php 2. Click on "Add New Product" button. 3. Enter Product details. 4. To edit the product, click on the edit on the specific product. 5. To delete the product, click on the delete button.

Test Case Id	Test Case Descriptions	Sample Input/Test Data	Expected Output	Results	Pass/Fail (Remark)
TR 1.11.1	Add a new product with all field filled	Add any product in all details	Product added successfully and appears in the list	Alert popup said, "Product Added successfully"	Pass
TR 1.11.2	Add a product with missing	Not enter product name	Scrolled to the unfilled data	The page is scrolled to the unfilled box	Pass

	mandatory data				
TR 1.11.3	Change product details	Change any details of the product	The product is successfully updated	A message popup said "Product and variants updated successfully"	Pass
TR 1.11.4	Delete product	Click delete button	The product is deleted successfully	Popup message said "Product deleted successfully"	Pass

Manage Orders by Staff/Admin

Requirement Id: R12	Requirement Description: Staff/Admin change order status / view receipt / view order / delete order
	Test Scenario: Staff/Admin wants to add new product
Pre-Condition	<ol style="list-style-type: none"> 1. The user is logged in as a Staff/Admin member with the required permissions. 2. Orders exist in the database with valid data, including statuses and associated receipts.
Post-Condition	<ol style="list-style-type: none"> 1. The order status is updated in the database and displayed correctly on the interface. 2. Receipts are accessible and viewable without errors. 3. Deleted orders are permanently removed from the database. 4. View the details of the order of the customer.

Test Steps	<ol style="list-style-type: none"> 1. Go to website: http://localhost/TeiponGadgetSystem/order/manage_order.php 2. Update the Status of Order by clicking on Order Status column. 3. Click on Receipt to view the receipt 4. Click the view button to find out the details of the order 5. Click on the delete button to delete order
------------	--

Test Case Id	Test Case Descriptions	Sample Input/Test Data	Expected Output	Results	Pass/Fail (Remark)
TR 1.12.1	Verify the changes order status	Update the status from Pending Payment to Completed	Order status updated successfully	Alert popup said, "Order status updated successfully".	Pass
TR 1.12.2	View receipt	Click the "View Receipt" button for an order	The receipt is displayed correctly	Showed the receipt of the order	Pass
TR 1.12.3	View Order Details	Click on the View in the Action column.	The order details like customer, order date, total amount, order status and ordered products	Show the details of the order	Pass

TR 1.12.4	Verification after deleting order	Clicked on the delete button on the specific order	Order is successfully deleted	Popup message said that “Order deleted successfully”	Pass
--------------	---	--	-------------------------------------	--	------

Update Staff/Admin Details

Requirement Id: R13	Requirement Description: Change Staff/Admin Information				
	Test Scenario: Staff/Admin wants to change his/her information				
Pre-Condition	Staff/Admin members must be logged in with appropriate permissions to update.				
Post-Condition	1. The details have changed in the database. 2. All changes are saved and synchronized in the database				
Test Steps	1. Go to website: localhost/TeiponGadgetSystem/settings_page/staff_settings.php or admin_settings.php. 2. Change details of the staff/admin 3. Enter “Update Details” button.				

Test Case Id	Test Case Descriptions	Sample Input/Test Data	Expected Output	Results	Pass/Fail (Remark)
TR 1.13.1	Verify changes for staff/admin details	Change any details of staff/admin	Staff information updated successfully	Alert popup said, “Details updated successfully”	Pass
TR 1.13.2	Validation for required fields	Leave mandatory fields blank	The system prevents submission	Alert popup said “Please fill	Pass

				out this field” on the blank field	
TR 1.13.3	Handle invalid input formats	Enter invalid data and attempt to save the changes	The system prevents submission	Alert popup said “Please include an “something” in this field” on the selected field	Pass

5. Test Results

Problems often arise and are detected by the developer after system testing. The developer is responsible for any issues or defects that occur in the system while it is being developed. Because of source code or incorrect code usage, human error is the primary cause of most computer defects. Test the system to determine what issues end users might have after we have finished a few tests for each system feature. Therefore, every test passes, and everything functions as it should. There are no expected outcomes, including any problems or defects that may arise with this technology.

5.1 Test Instances

No problems, flaws, or unexpected outcomes were found throughout testing. The system appropriately blocked access when a user submitted the wrong password and asked them to change it. The system operated as planned and the login functionality was carefully tested, guaranteeing safe user access and appropriate handling of authentication problems. The system successfully satisfied all predetermined requirements, and all test data was gathered and presented as anticipated.

5.1.1 Resolved Test Incidents

Not applicable

5.1.2 Unresolved Test Incidents

Not applicable

5.2 Recommendations

The customer section is overall excellent, with numerous features added to create a user-friendly environment, allowing customers to easily navigate and access the features that meet their needs. Similarly, the staff and manager pages are well-designed, enabling efficient management of data such as customer information, product catalogs, and staff or admin accounts.




To ensure the system maintains high quality, improvements should focus on enhancing service quality and interface navigation. This can be achieved primarily through

valuable user feedback, which plays a critical role in optimizing system performance. Additionally, the system's theme and design should be regularly updated to keep it fresh and engaging.

For the management section, it is important to consider improvements in business report management by implementing enhanced features and a more intuitive GUI. This will make it easier for management to handle the system efficiently.

Appendix A: Approval

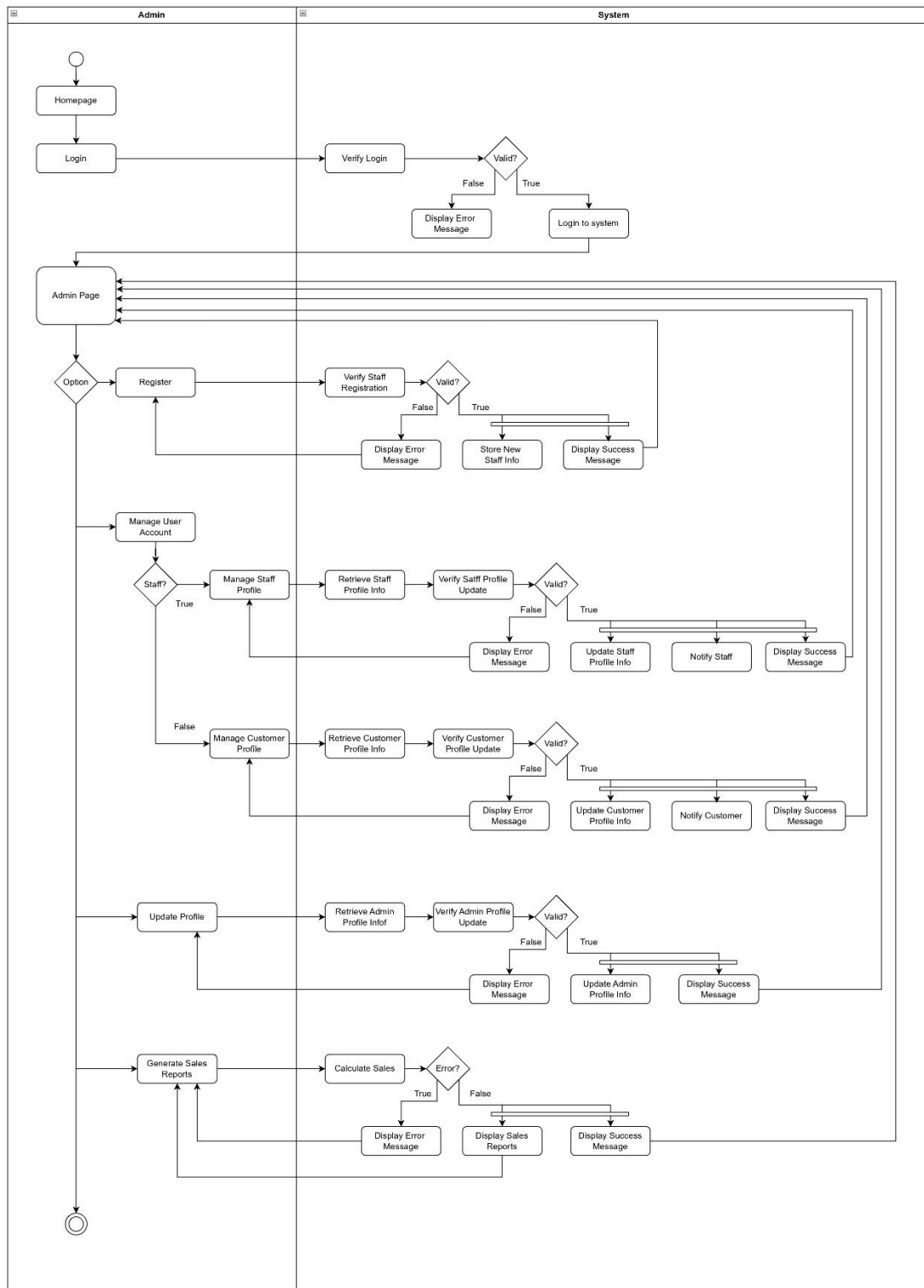
The undersigned acknowledge they have reviewed the Teipon Gadget System Software Design Description and agree with the approach it presents. Changes to this Software Design Description will be coordinated with and approved by the undersigned or their designated representatives.

1. Signature :  Date: 22/1/2025
Name : Mohamad Hazik Haikal bin Razak
Student Id: 2024779495
Role : Project Manager
2. Signature :  Date: 22/1/2025
Name : Muhamaad Luqman Mu'izz bin Anuar
Student Id: 2024974353
Role : Software Tester
3. Signature :  Date: 22/1/2025
Name : Wan Muhammad Naqib bin Wan Mohd Nazri
Student Id: 2024926951
Role : Software Requirement Analyst

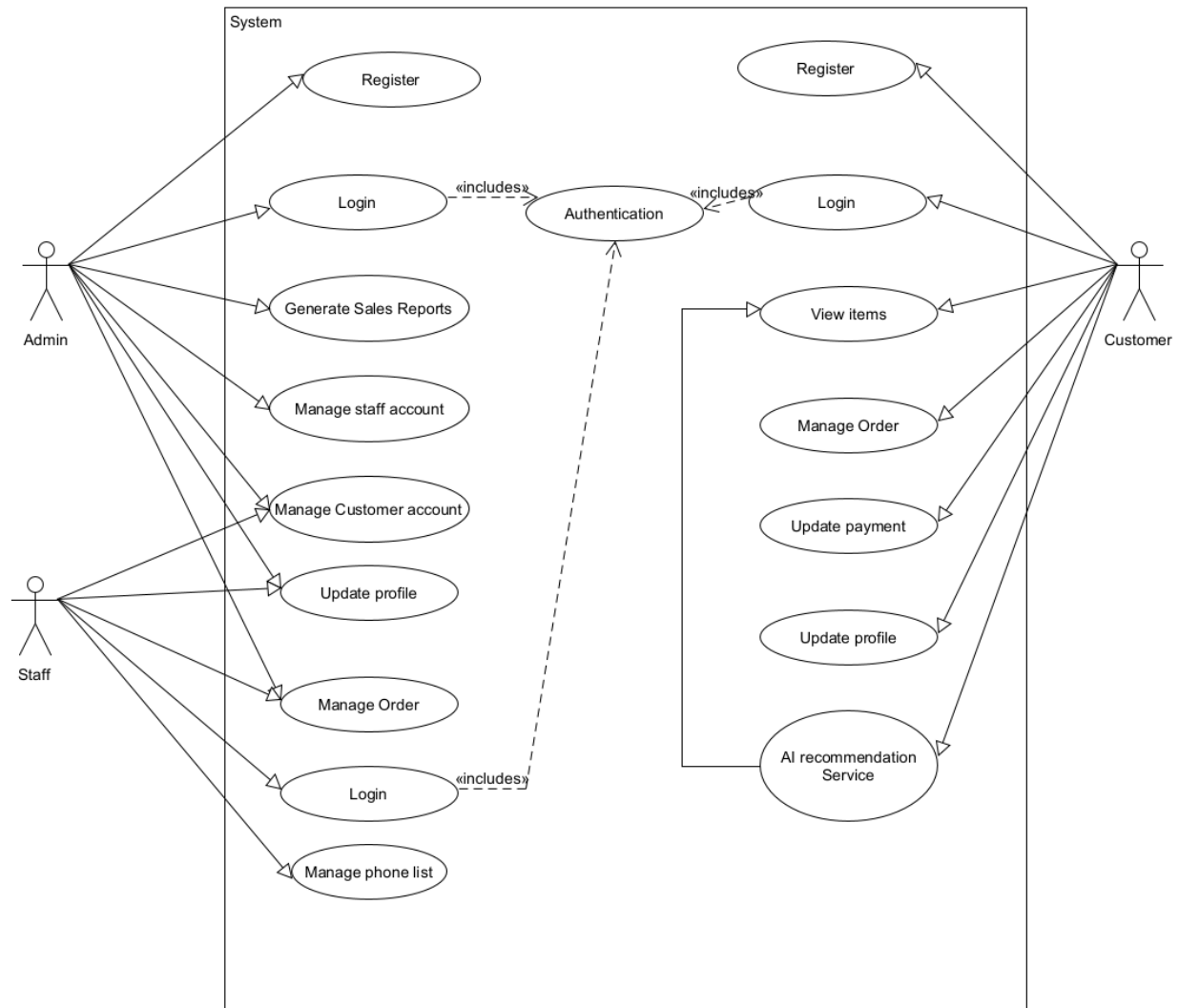
Appendix B: Glossary

SRS	A document that describes the system's requirements and specifications.
DBMS	Software that helps manage and organize databases.
PHP	A scripting language used for creating dynamic web pages.
CSS	A style sheet language for designing how web pages look.
HTML	The standard markup language for creating web pages.
JavaScript	A programming language to make web pages interactive.
IDE	Software used by developers to write, test, and debug code.
Laragon	A tool for setting up a local web server environment.
MySQL	A system for managing relational databases.
Dialogflow	A tool by Google for building conversational interfaces like chatbots.
Ngrok	A tunneling service to expose a local server to the internet.
Webhook	A way for one application to send real-time data to another application events.based on specific
HTTP	A protocol is used to transfer data over the web.
HTTPS	A secure version of HTTP with data encryption.
SQL	A language used to manage and query databases.
MVC	Model-View-Controller

Appendix C: Activity Diagram



Appendix D: Use Case Diagram



Appendix E: Class Diagram

