Software Requirements Specification

Version 3.0

23rd October 2023

Laundry Management System

Arpit Sharma

UMD Email ID – [arpit777@umd.edu](mailto:arpit777@umd.edu)

UMD Directory ID – arpit777

Submitted in partial fulfillment

Of the requirements of

ENPM809W - Introduction to Secure Coding for Software Engineering

# Table of Contents

[Table of Contents i](#_Toc148990952)

[List of Figures ii](#_Toc148990953)

[1.0. Introduction 1](#_Toc148990954)

[1.1. Purpose 1](#_Toc148990955)

[1.2. Scope of Project 1](#_Toc148990956)

[1.3. Glossary 1](#_Toc148990957)

[1.4. References 2](#_Toc148990958)

[2.0. Overall Description 2](#_Toc148990959)

[2.1 System Environment 2](#_Toc148990960)

[2.2 Functional Requirements Specification 3](#_Toc148990961)

[2.2.1 **User Use Cases** 3](#_Toc148990962)

[Use case 1: Check washer availability 4](#_Toc148990963)

[Use case 2: Check dryer availability 4](#_Toc148990964)

[Use case 3: Update User information 5](#_Toc148990965)

[Use case 4: Initiate washing process 5](#_Toc148990966)

[Use case 5: Initiate drying process 6](#_Toc148990967)

[Use case 6: Process completion notification. 6](#_Toc148990968)

[Use case 7: Update E-wallet 7](#_Toc148990969)

[2.2.2 **Admin User Use Cases** 7](#_Toc148990970)

[Use case 7: Read/Write/Delete Washer information 8](#_Toc148990971)

[Use case 8: Read/Write/Delete Dryer information 8](#_Toc148990972)

[Use case 9: Read/Write/Delete User information 9](#_Toc148990973)

[2.3 User Characteristics 9](#_Toc148990974)

[2.4 Non-Functional Requirements 9](#_Toc148990975)

[**2.4.1 Security Requirements** 10](#_Toc148990976)

[3.0 Requirements Specifications 11](#_Toc148990977)

[3.1 External Interface Requirements 11](#_Toc148990978)

[3.2 Functional Requirements 12](#_Toc148990979)

[3.2.1 Check Washer Availability 12](#_Toc148990980)

[3.2.2 Check Dryer Availability 12](#_Toc148990981)

[3.2.3 Update User Information 13](#_Toc148990982)

[3.2.4 Initiate Washing Process 13](#_Toc148990983)

[3.2.5 Initiate Drying Process 14](#_Toc148990984)

[3.2.6 Process Completion Notification 15](#_Toc148990985)

[3.2.7 Update E-wallet 15](#_Toc148990986)

[3.2.8 Read/Write/Delete Washer information. 16](#_Toc148990987)

[3.2.9 Read/Write/Delete Dryer information. 16](#_Toc148990988)

[3.2.10 Read/Write/Delete User information 17](#_Toc148990989)

[3.3 Misuse Cases 18](#_Toc148990990)

[3.3.1 Brute force on user login 19](#_Toc148990991)

[3.3.2 Man in the middle attack 19](#_Toc148990992)

[3.3.3 SQL Injection on login page 20](#_Toc148990993)

[3.3.4 Weak password hashing 21](#_Toc148990994)

[3.3.5 Unsecure Session management 22](#_Toc148990995)

[3.3.6 Sensitive information disclosure 23](#_Toc148990996)

[3.4 Threat model 23](#_Toc148990997)

# List of Figures

[Figure 1 - System Environment 2](#_Toc144920569)

[Figure 2 - User Use Cases 3](#_Toc144920570)

[Figure 3 - Admin User Use Cases 7](#_Toc144920571)

# 1.0. Introduction

## 1.1. Purpose

The purpose of this document is to present a detailed description of the Laundry Management System. The purpose of this project is to track the availability of laundry services like washing and drying and notify the customers about the process completion. The user can also use the e-wallet to pay for the cost of the services. This document contains an explanation of the interfaces and the features of the system.

## 1.2. Scope of Project

This software system will be a Web Laundry Management System for a local laundry room in an apartment building. This project will alert the users in real time about the laundry services completion and will provide availability of washers and dryers in the laundry room. This will enhance the time management and centralize the process by enabling the user to initiate the laundry process through the application itself. The application will provide an easy to use interface for the user to look for available machine, start the laundry by paying through the e-wallet on the payment portal and then get alert when the laundry is completed.

## 1.3. Glossary

|  |  |
| --- | --- |
| **Term** | **Definition** |
| Washer | The machine used to wash clothes |
| Dryer | The machine used to dry the washed clothes |
| User | The person who will use the system to perform laundry services |
| Admin User | User with admin privileges, who can create, update, delete anything in the system |
| Payment Details | The credit/debit card details of the user using the laundry management system |
| E-Wallet | A database containing the money which user has added using the payment details |
| Availability | It shows the machines which are idle, i.e. available to use. |
| Cost | The amount deducted from the user account to use the laundry service like washing or drying. |
| PII | Personally Identifiable Information of the user like email id, phone number, name, date of birth etc. |

## 1.4. References

IEEE. *IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications.* IEEE Computer Society, 1998.

# 2.0. Overall Description

## 2.1 System Environment

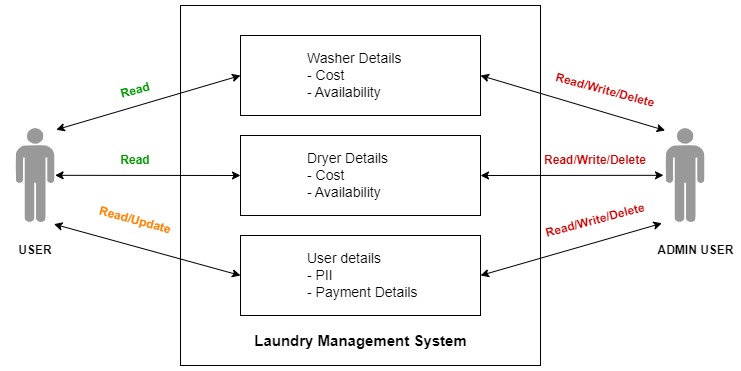


Figure - System Environment

The Laundry Management System has two active actors and one cooperating system. The User and Admin User accesses the Resident portal through the Internet and interact with the application.

The user can view washer and dryer information and availability. The user can also read/update the user’s PII and payment details. The user can add money to the e-wallet using the payment details.

The admin user on the other hand can Read/Write/Delete everything in the application like washer information, dryer information and any user information.

## 2.2 Functional Requirements Specification

This section outlines the use cases for each of the active readers separately. The user and the admin user.

### 2.2.1 **User Use Cases**

The User has the following sets of use cases:

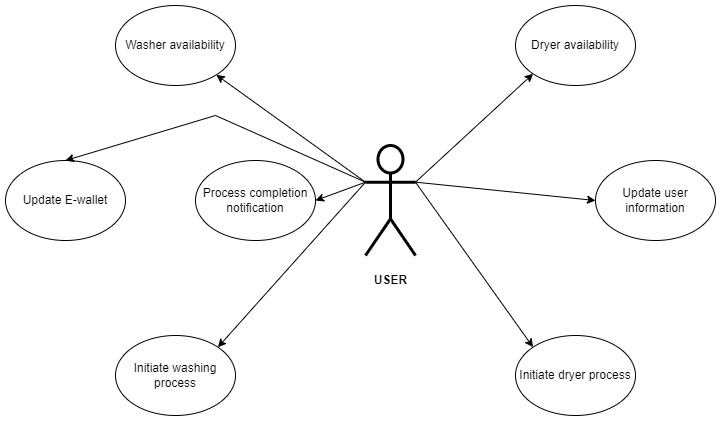


Figure - User Use Cases

#### Use case 1: Check washer availability

**Diagram:**

User

Washer Availability

**Brief Description**

The User signs in into the Resident Portal on Laundry management system website and then checks the availability of the washer.

**Initial Step-By-Step Description**

The user must sign up for the application and provide the details of the laundry room and payment information as well.

1. The user signs into the resident portal on laundry management system application
2. The user can see the availability of the washers in the laundry room as provided by the application.

#### Use case 2: Check dryer availability

**Diagram:**

User

Dryer Availability

**Brief Description**

The User signs in into the Resident Portal on Laundry management system website and then checks the availability of the dryer.

**Initial Step-By-Step Description**

The user must sign up for the application and provide the details of the laundry room and payment information as well.

1. The user signs into the resident portal on laundry management system application
2. The user can see the availability of the dryers in the laundry room as provided by the application.

#### Use case 3: Update User information

**Diagram:**

User

User Information

-PII and Payment

**Brief Description**

The User can provide the PII and payment details at the time of sign up.

**Initial Step-By-Step Description**

The user must sign up for the application.

1. The user signs into the resident portal on laundry management system application
2. The user can goto the profile section to update the PII and Payment information.
3. User saves the updated information.

#### Use case 4: Initiate washing process

**Diagram:**

User

Initiate washing process

**Brief Description**

The User initiates the washing process by selecting the available washing machine and then paying using the payment information stored in the application.

**Initial Step-By-Step Description**

The user must sign up for the application.

1. The user signs into the resident portal on laundry management system application
2. The user can select the available washing machine.
3. The user physically goes to the laundry room and puts all the laundry in the available machine along with the detergent.
4. The user then go to the application and go to payment and pay for the washing process.
5. The application will start the washing machine.

#### Use case 5: Initiate drying process

**Diagram:**

User

Initiate drying process

**Brief Description**

The User initiates the drying process by selecting the available dryer machine and then paying using the payment information stored in the application.

**Initial Step-By-Step Description**

The user must sign up for the application.

1. The user signs into the resident portal on laundry management system application
2. The user can select the available dryer machine.
3. The user physically goes to the laundry room and puts all the laundry in the available machine.
4. The user then goto the application and goto payment and pay for the drying process.
5. The application will start the dryer machine.

#### Use case 6: Process completion notification.

**Diagram:**

User

Process completion notification

**Brief Description**

The User gets the notification once the washing/drying process has been completed.

**Initial Step-By-Step Description**

The user must sign up for the application.

1. The user signs in to the resident portal on laundry management system application
2. The user will get notification in the application once the washing/drying has been finished.

#### Use case 7: Update E-wallet

**Diagram:**

Update E-wallet

User

Process completion notification

**Brief Description**

The User can add and update money in the e-wallet of the Laundry Management System.

**Initial Step-By-Step Description**

The user must sign up for the application.

1. The user signs into the resident portal on laundry management system application
2. The user goes to the profile section of the application.
3. The user can select e-wallet and click add money.

### 2.2.2 **Admin User Use Cases**

The Admin User has the following sets of use cases:

**Admin User**

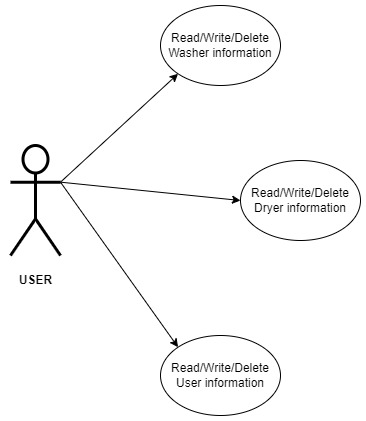


Figure - Admin User Use Cases

#### Use case 7: Read/Write/Delete Washer information

**Diagram:**

Admin user

Read/Write/Delete Washer information

**Brief Description**

The Admin User can Read/Write/Delete Washer information from the application

**Initial Step-By-Step Description**

The admin user must use the admin account to sign into the application.

1. The admin user signs in to the resident portal on laundry management system application
2. The admin user can perform any action on the washer information.

#### Use case 8: Read/Write/Delete Dryer information

**Diagram:**

Admin user

Read/Write/Delete Dryer information

**Brief Description**

The Admin User can Read/Write/Delete Dryer information from the application

**Initial Step-By-Step Description**

The admin user must use the admin account to sign into the application.

1. The admin user signs in to the resident portal on laundry management system application
2. The admin user can perform any action on the dryer information.

#### Use case 9: Read/Write/Delete User information

**Diagram:**

Admin user

Read/Write/Delete User information

**Brief Description**

The Admin User can Read/Write/Delete any User information from the application

**Initial Step-By-Step Description**

The admin user must use the admin account to sign into the application.

1. The admin user signs in to the resident portal on laundry management system application
2. The admin user can perform any action on the user information

## 2.3 User Characteristics

The user is expected to use the internet and search engine to navigate to the web application. The homepage of the Laundry Management System web application will provide the user with a login page. The user will sign up/login using the credentials and will land on the dashboard page.

The admin user is expected to use the internet and search engine to navigate to the web application. The admin user can only use the admin credentials to login the admin role which has the admin privileges.

## 2.4 Non-Functional Requirements

The Laundry Management System will be hosted on a server with high availability and low latency. The washer and dryer machines will be decided by the laundry management company.

This project focuses on providing the user real-time access to the availability of washers and dryers in the laundry room and the functionality to use the e-wallet to perform transactions for the cost of washing and drying.

### **2.4.1 Security Requirements**

**2.4.1.1 Scenario 1** – The application does not have account lockout functionality for repeated failed login attempts.

**Security Requirement 1** – The application must implement an account lockout feature which locks the user account for 1 hour after 3 unsuccessful login attempts. This will eliminate the brute force attack vulnerability.

**2.4.1.2 Scenario 2** – The application does not have transport layer security on the communication channel.

**Security Requirement 2** – The application must implement transport layer security and use TLS v1.2 or higher to protect the channel between the user and the application server. This will encrypt the channel and the attacker cannot get their hands on the data.

**2.4.1.3 Scenario 3** – The application uses SQL queries without sanitization of user input.

**Security Requirement 3** – The application must use parameterized queries and sanitize user input before executing any SQL query to send or retrieve data from the database.

**2.4.1.4 Scenario 4** – The application uses a weak password hashing algorithm to save password hashes.

**Security Requirement 4** – The application must use a strong password hashing algorithm like BeCrypt or Argon2 to hash the passwords and store the secure password hash in the database. This will secure the user’s passwords.

**2.4.1.5 Scenario 5** – The application does not have any session expiration functionality.

**Security Requirement 5** – The application must automatically logout the user that is expiring the user session after 20 minutes of signing in. This will prevent any unauthorized access to the user session.

**2.4.1.6 Scenario 6** – The application does not store credit card details of the users securely.

**Security Requirement 6 –** The application must store the credit card information of the users securely by encrypting the data and then storing it in the database. This will prevent sensitive information disclosure to an extent.

# 3.0 Requirements Specifications

## 3.1 External Interface Requirements

There will be an external interface which will link the user’s e-wallet of the Laundry Management System with the bank details database. This will be used to add money to the e-wallet.

The user bank details database contains the payment information (credit card/debit card details) tagged with the current balance of the user in the bank account.

E-Wallet will be responsible for sending and receiving data through this interface and will update the wallet balance. Once the wallet balance gets updated, the new balance will be updated in the bank details database. E-wallet will also contain the payment details of the user which will be used to connect to the selected user’s bank details.

## 3.2 Functional Requirements

### 3.2.1 Check Washer Availability

|  |  |
| --- | --- |
| **Use Case Name** | Check washer availability |
| **XRef** | Section 2.2.1, Check washer availability |
| **Trigger** | The user logs in the application to check washer availability in the laundry room |
| **Precondition** | The user is on the login page of the Laundry Management System and is authenticated. |
| **Basic Path** | 1. The user is authenticated and logged in. 2. The dashboard page contains the Washer Information field which shows the available washing machines. |
| **Alternative paths** | None |
| **Postcondition** | The user gets to know about the washing machines which are available for usage. |
| **Exception paths** | The user might abandon the checking of washing machine. |
| **Other** | The user will not get logged in if provides invalid credentials. The system shows availability of 4 machines in the laundry room. |

### 3.2.2 Check Dryer Availability

|  |  |
| --- | --- |
| **Use Case Name** | Check dryer availability |
| **XRef** | Section 2.2.1, Check dryer availability |
| **Trigger** | The user logs in the application to check dryer availability in the laundry room |
| **Precondition** | The user is on the login page of the Laundry Management System and is authenticated. |
| **Basic Path** | 1. The user is authenticated and logged in. 2. The dashboard page contains the Dryer Information field which shows the available dryer machines. |
| **Alternative paths** | None |
| **Postcondition** | The user gets to know about the dryer machines which are available for usage. |
| **Exception paths** | The user might abandon the checking of dryer machine. |
| **Other** | The user will not get logged in if provides invalid credentials. The system shows availability of 4 machines in the laundry room. |

### 3.2.3 Update User Information

|  |  |
| --- | --- |
| **Use Case Name** | Update User Information |
| **XRef** | Section 2.2.1, Update User Information |
| **Trigger** | The user selects the edit information field to edit the user details |
| **Precondition** | The user must access the Profile section after logging in the application using their credentials |
| **Basic Path** | 1. The user is authenticated. 2. The user selects the Profile field and gets to the profile page. 3. The user selects Edit information field to edit the user details. 4. The user selects save field after entering the data and the system saves the information |
| **Alternative paths** | During sign up, the user must provide the user information. |
| **Postcondition** | The updated user information is saved. |
| **Exception paths** | In Step 3, if there are any empty fields, the system won’t save the information. The user might abandon updating the user information at any time. |
| **Other** | The user information includes Name of the user, email address, phone number, mailing address. |

### 3.2.4 Initiate Washing Process

|  |  |
| --- | --- |
| **Use Case Name** | Initiate Washing Process |
| **XRef** | Section 2.2.1, Initiate Washing Process |
| **Trigger** | Select initiate washing |
| **Precondition** | The user must access the Dashboard page after logging in the application using their credentials |
| **Basic Path** | 1. The user is authenticated. 2. The user selects the available washing machine from the Washer information field. 3. The system shows the available machine. 4. The user selects the Initiate washing field to start the washing machine. |
| **Alternative paths** | None |
| **Postcondition** | The selected washing machine starts. |
| **Exception paths** | If the user’s e-wallet doesn’t have enough money to pay the washing cost, the system will decline the process and send a pop up on the webpage. |
| **Other** | The selected washing machine starts and updates as unavailable on the dashboard |

### 3.2.5 Initiate Drying Process

|  |  |
| --- | --- |
| **Use Case Name** | Initiate Drying Process |
| **XRef** | Section 2.2.1, Initiate Drying Process |
| **Trigger** | Select initiate dryer |
| **Precondition** | The user must access the Dashboard page after logging in the application using their credentials |
| **Basic Path** | 1. The user is authenticated. 2. The user selects the available dryer machine from the Dryer information field. 3. The system shows the available machine. 4. The user selects the Initiate dryer field to start the dryer machine |
| **Alternative paths** | None |
| **Postcondition** | The selected washing machine starts. |
| **Exception paths** | If the user’s e-wallet doesn’t have enough money to pay the washing cost, the system will decline the process and send a pop up on the webpage. |
| **Other** | The selected dryer machine starts and updates as unavailable on the dashboard |

### 3.2.6 Process Completion Notification

|  |  |
| --- | --- |
| **Use Case Name** | Process Completion Notification |
| **XRef** | Section 2.2.1, Process Completion Notification |
| **Trigger** | Completion of the washing/dryer task |
| **Precondition** | The washing/dryer machine must be running selected by the authenticated user. |
| **Basic Path** | 1. The user is authenticated. 2. The user initiates washing/drying process. 3. After the completion of the process, the system automatically sends a notification and a pop up on the webpage regarding the completion. |
| **Alternative paths** | None |
| **Postcondition** | The user gets notified about the task completion |
| **Exception paths** | None |
| **Other** | After the completion of task, the unavailable machine state is changed to available on the dashboard. |

### 3.2.7 Update E-wallet

|  |  |
| --- | --- |
| **Use Case Name** | Update E-wallet |
| **XRef** | Section 2.2.1, Update E-wallet |
| **Trigger** | The user selects add money field |
| **Precondition** | The user must access the Profile section after logging in the application using their credentials. |
| **Basic Path** | 1. The user is authenticated. 2. The user selects the Profile field and gets to the profile page. 3. The user selects E-wallet field to open the e-wallet details. 4. The user enters the amount in the input box and selects add money field. |
| **Alternative paths** | None |
| **Postcondition** | The money is added in the e-wallet |
| **Exception paths** | 1. If the user has entered wrong payment information in the e-wallet section, the transaction will fail, and the system will send a pop up on the screen. 2. If the user’s bank account doesn’t have the money entered in the input text, the transaction will fail, and the system will send a pop up on the screen. |
| **Other** | The user must have provided the payment information by the time of sign up and the details must be legit. |

### 3.2.8 Read/Write/Delete Washer information.

|  |  |
| --- | --- |
| **Use Case Name** | Read/Write/Delete Washer information |
| **XRef** | Section 2.2.2, Read/Write/Delete Washer information |
| **Trigger** | The admin user selects the washing machine field from the dashboard page which contains the washing Information field. |
| **Precondition** | The admin user is on the login page of the Laundry Management System and is authenticated using admin credentials. |
| **Basic Path** | 1. The admin user is authenticated. 2. The admin user selects the washing machine field to view/change/delete the selected machine’s information from the list of washing machines. 3. The admin user selects Edit information field to edit the machine name and availability. 4. The admin user selects save field after entering the data and the system saves the information |
| **Alternative paths** | In step 4, the admin user can select delete field to delete the washing machine details from the system. |
| **Postcondition** | The washing machine information is modified/deleted. |
| **Exception paths** | None |
| **Other** | The admin user can modify the machine’s details in case of any technical glitch. |

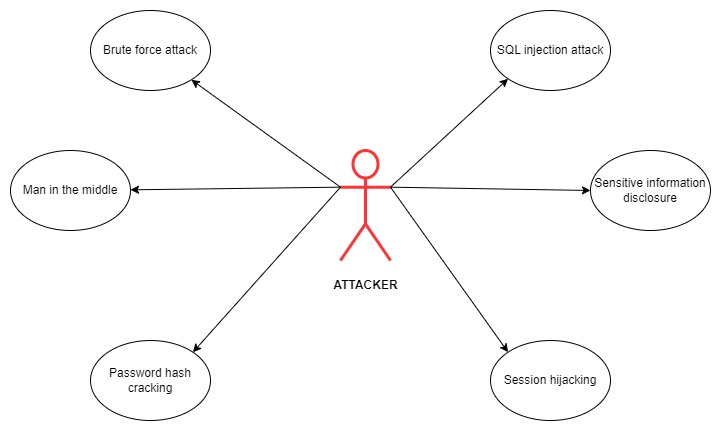
### 3.2.9 Read/Write/Delete Dryer information.

|  |  |
| --- | --- |
| **Use Case Name** | Read/Write/Delete Dryer information |
| **XRef** | Section 2.2.2, Read/Write/Delete Dryer information |
| **Trigger** | The admin user selects the dryer machine field from the dashboard page which contains the dryer Information field. |
| **Precondition** | The admin user is on the login page of the Laundry Management System and is authenticated using admin credentials. |
| **Basic Path** | 1. The admin user is authenticated. 2. The admin user selects the dryer machine field to view/change/delete the selected dryer’s information from the list of washing machines. 3. The admin user selects Edit information field to edit the machine name and availability. 4. The admin user selects save field after entering the data and the system saves the information |
| **Alternative paths** | In step 4, the admin user can select delete field to delete the dryer machine details from the system. |
| **Postcondition** | The dryer machine information is modified/deleted. |
| **Exception paths** | None |
| **Other** | The admin user can modify the machine’s details in case of any technical glitch. |

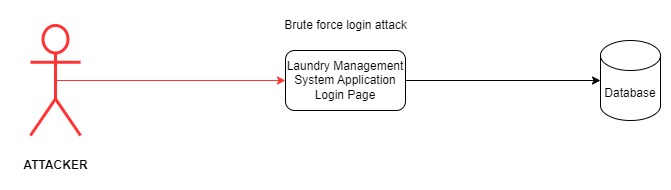
### 3.2.10 Read/Write/Delete User information

|  |  |
| --- | --- |
| **Use Case Name** | Read/Write/Delete User information |
| **XRef** | Section 2.2.2, Read/Write/Delete User information |
| **Trigger** | The admin user selects the user from the user list in profile section. |
| **Precondition** | The admin user must access the Profile section after logging in the application using their admin credentials |
| **Basic Path** | 1. The admin user is authenticated. 2. The admin user selects the Profile field and gets to the profile page. 3. The admin user selects the user to view/change/delete the user information from the list of users. 4. The admin user selects Edit information field to edit the user details. 5. The admin user selects save field after entering the data and the system saves the information |
| **Alternative paths** | In step 4, the admin user can select delete field to delete the user details from the system. |
| **Postcondition** | The user information is modified/deleted. |
| **Exception paths** | None |
| **Other** | The admin user have access to all the registered users details and payment information. |

## 3.3 Misuse Cases

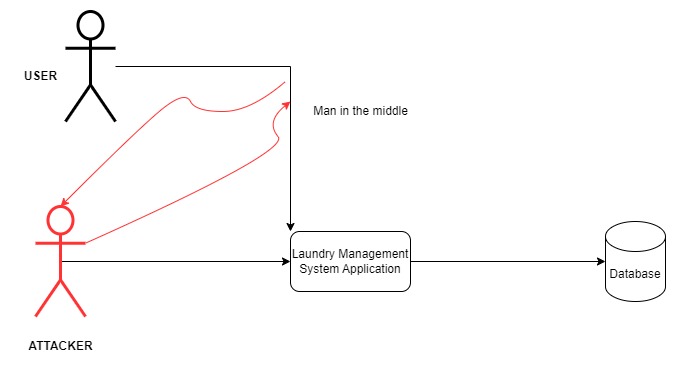


### 3.3.1 Brute force on user login



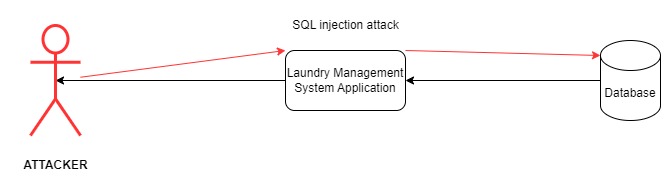
|  |  |
| --- | --- |
| **Use Case Name** | Brute force on user login |
| **Trigger** | The attacker can perform brute force password cracking attack on the login page. |
| **Precondition** | The attacker must be able to access the URL of the application. |
| **Basic Path** | 1. The attacker intercepts the login request. 2. Dictionary or brute force attack can be used to crack the password of any user. |
| **Postcondition** | The user credentials are available to the attacker. |
| **Exception paths** | None |
| **Reason** | There is no account lockout feature on failed login attempts. |

### 3.3.2 Man in the middle attack



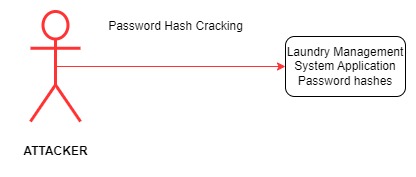
|  |  |
| --- | --- |
| **Use Case Name** | Man in the middle attack |
| **Trigger** | The attacker can perform man in the middle attack and intercept the channel between any user and application server. |
| **Precondition** | The attacker must be able to access the URL of the application. |
| **Basic Path** | 1. The attacker intercepts the traffic between any user and the application server. 2. The Data is in plaintext hence, confidentiality has been breached. |
| **Postcondition** | The attacker can understand all the traffic. |
| **Exception paths** | None |
| **Reason** | There is no transport layer security applied on the communication channel between the user and the application server. |

### 3.3.3 SQL Injection on login page



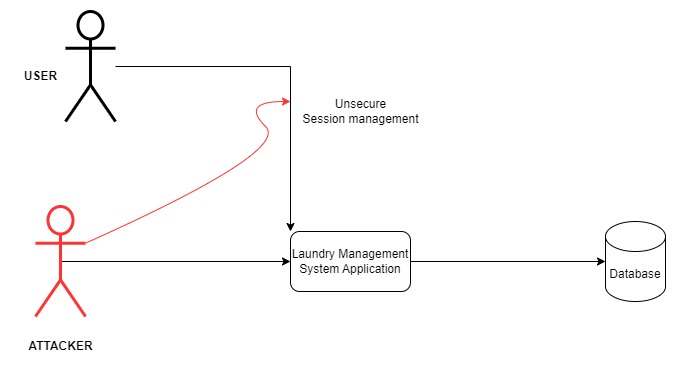
|  |  |
| --- | --- |
| **Use Case Name** | SQL Injection on login page |
| **Trigger** | The attacker can perform a SQL injection attack on the login page. |
| **Precondition** | The attacker must be able to access the URL of the application. |
| **Basic Path** | 1. The attacker intercepts the traffic between their machine and the application server. 2. Then, they try different SQL injection payloads to access the data in the database. |
| **Postcondition** | The attacker can access the data present in the database of the application leading to information disclosure. The attacker can access the payment details of the users. |
| **Exception paths** | None |
| **Reason** | There is no transport layer security applied on the communication channel between the user and the application server. |

### 3.3.4 Weak password hashing



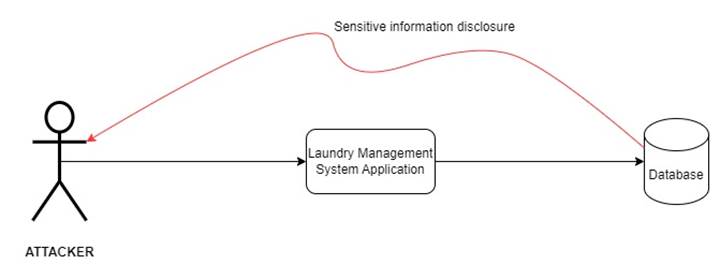
|  |  |
| --- | --- |
| **Use Case Name** | Weak password hashing |
| **Trigger** | The attacker can crack the password hashes stored in the database. |
| **Precondition** | The attacker must have the password hashes of the users. |
| **Basic Path** | 1. The attacker gets access to the database. 2. They steal the password hashes stored in the database. 3. They perform password hash cracking attacks to crack the hashes. |
| **Postcondition** | The attacker has the original passwords of the users. |
| **Exception paths** | None |
| **Reason** | The application is using weak password hashing algorithm to hash the passwords and save it in the database. |

### 3.3.5 Unsecure Session management



|  |  |
| --- | --- |
| **Use Case Name** | Unsecure Session Management |
| **Trigger** | The attacker can access the user’s account if the user has logged in and forgot to logout. |
| **Precondition** | The attacker must trick user to login into their machine. |
| **Basic Path** | 1. The attacker gets access to the user’s machine or tricks user to login the application into their machine. |
| **Postcondition** | The attacker can use the user’s account for their benefit. |
| **Exception paths** | None |
| **Reason** | There is no session timeout feature implemented which allows the session to expire after a certain time preventing any unauthorized use of the user’s account. |

### 3.3.6 Sensitive information disclosure



|  |  |
| --- | --- |
| **Use Case Name** | Sensitive information disclosure |
| **Trigger** | The attacker can steal the user’s credit card information. |
| **Precondition** | The attacker must have access to the data stored in the database. |
| **Basic Path** | 1. The attacker gets access to the database. 2. They access the E-Wallet table. 3. They can steal the user’s card number from it. |
| **Postcondition** | The attacker has he card numbers of the users. |
| **Exception paths** | None |
| **Reason** | The application is storing user’s card details in plaintext inside the database. |

## 3.4 Threat model

Below is the threat model of the laundry management system along with the report which contains the potential threats in the application architecture. These threats need to be validated and then mitigated to make the application secure.

I used STRIDE as the threat modeling methodology and used the Microsoft Threat modeling tool to prepare the model. Please view the full report attached below.



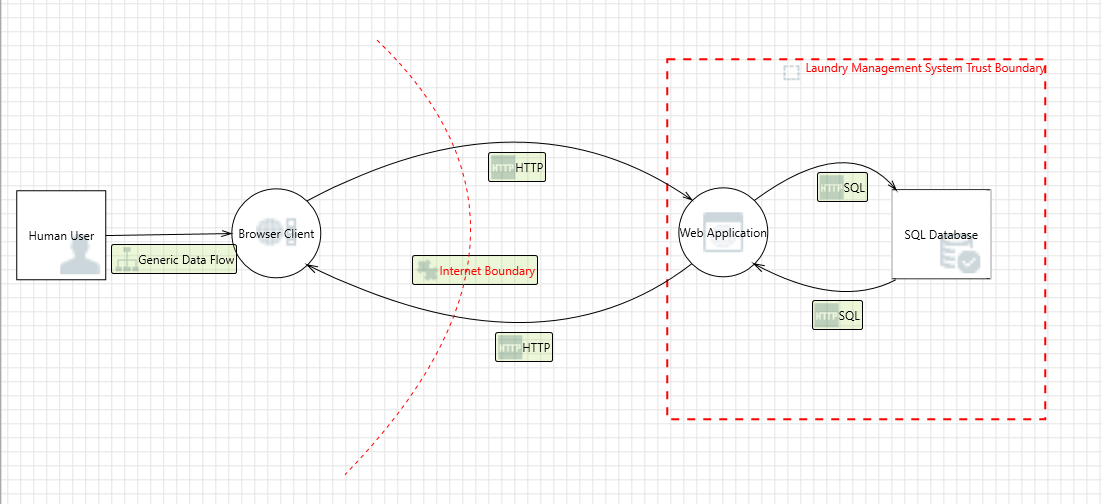
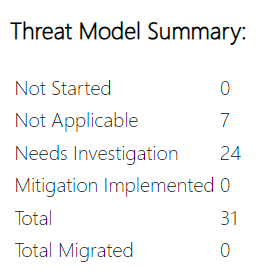


Figure - Threat Model of Laundry Management System