



Software Requirements Specification

Version 1.0

Thrift Fashion Web Application

Category: Web Application Development



Table of Contents

1.1

Background and Need for the Web Application

1.2

Proposed Solution

1.3

Purpose of this Document

1.4

Scope of Project

1.5

Functional Requirements

1.6

Non-Functional Requirements

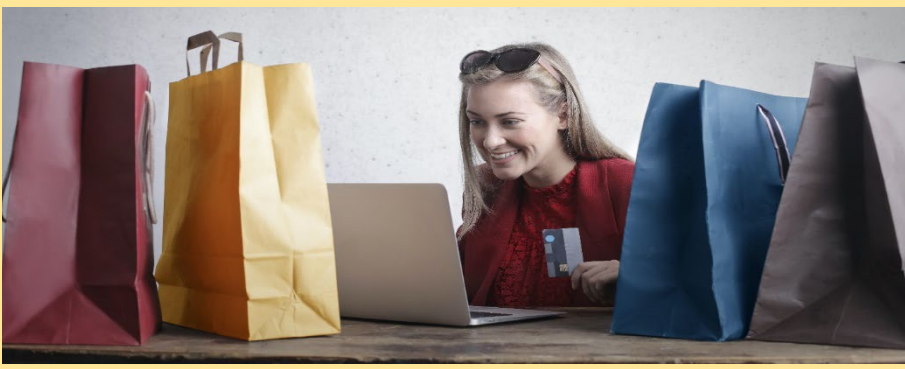
1.7

Interface Requirements

- Hardware
- Software
- Database Design

1.8

Project Deliverables



1.1

Background and Need for the Web Application

People often discard clothing items when they are no longer in use. They buy clothing from malls, online retail stores, or regular neighborhood stores and later, when these clothes outgrow their utility, they throw them away or dump them in bins. This has even happened in the case of high cost clothing. People who can afford expensive fashion typically do not like to repeat their outfits. They move on to buy newer items barely after using current items on two or three occasions. The older ones are then just discarded. This causes burden on the planet because as the number of such discarded fashion (which is still in good condition, nevertheless) increases, the effort to dispose them also increases.

These days, recycling, sustainability, and reuse are not just buzzwords, but actionable behaviors to support a better planet. Instead of discarding clothes that are still in good condition, but no longer required by the original buyers, these clothes can be sold as thrift fashion or pre-used clothing.

In simple terms, shopping for thrift fashion is second hand purchases for fashionable clothing.

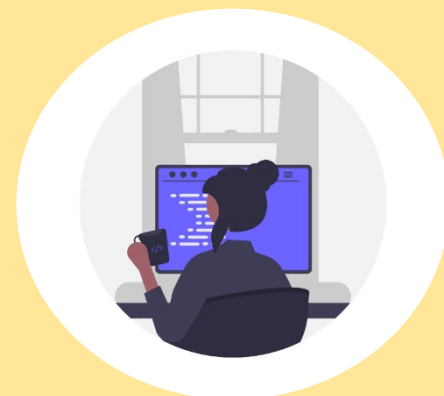
This supports the theme of recycling and sustainability and is also affordable by many who could not purchase new fashionable clothing.

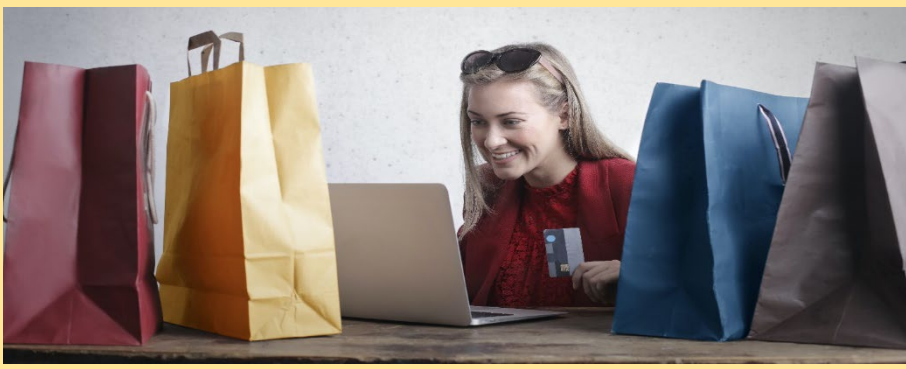
1.2

Proposed Solution

The proposed solution is a Web application titled **ThriftFashion**. It should help individuals with online shopping for pre-used fashionable clothing and provide better access to such products. Users can browse for products, search, filter, and so on. submit their feedback regarding the application using a feedback form.

You have been given a contract to design and build this e-commerce based Web application.





1.3

Purpose of this Document

The purpose of this document is to present a detailed description of the Web application titled **ThriftFashion**.

ThriftFashion provides an easy shopping facility to the customers, wherein they can sit at one place and shop for pre-used fashionable clothing online anytime, anywhere.

This document explains the purpose and features of the Web application, its interfaces, what the Web application will do, and the constraints under which it must operate. This document is intended for both stakeholders and developers of the Web application.

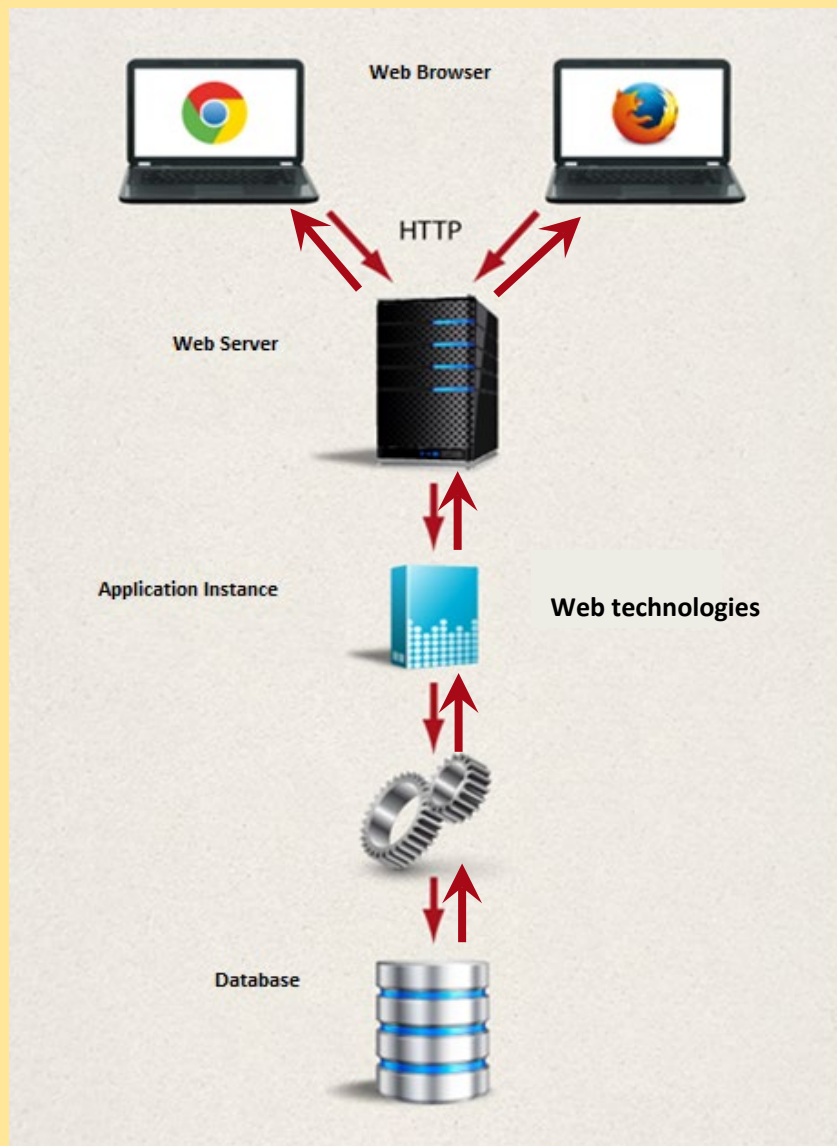
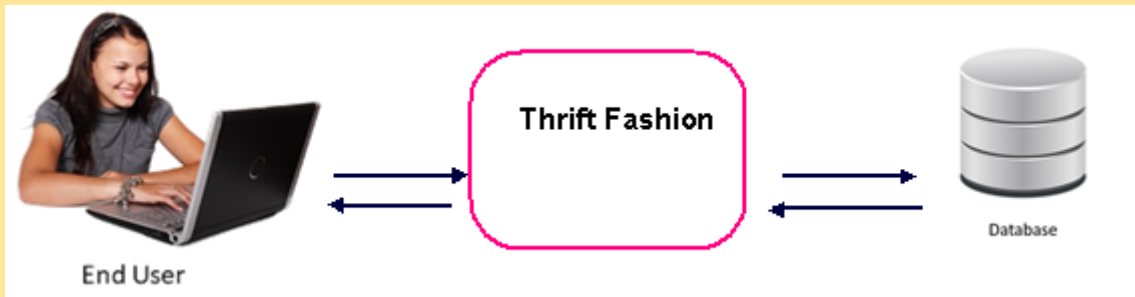
1.4

Scope of the Project

This Web portal will be a responsive and visually appealing Web application to be used by individuals. This portal will be designed to aid users in online shopping. Users can browse through various products, see product description, price, and so on, they can search for a specific product, check out deals for a specific day, purchase products, and check the total amount they are expected to pay. Besides these, there will be a number of other functionalities related to online shopping.



However, the application will not have any feature/functionality for implementing or authenticating payment and delivery. These actions are beyond the scope of this application.



Broad View of the System



1.5

Functional Requirements

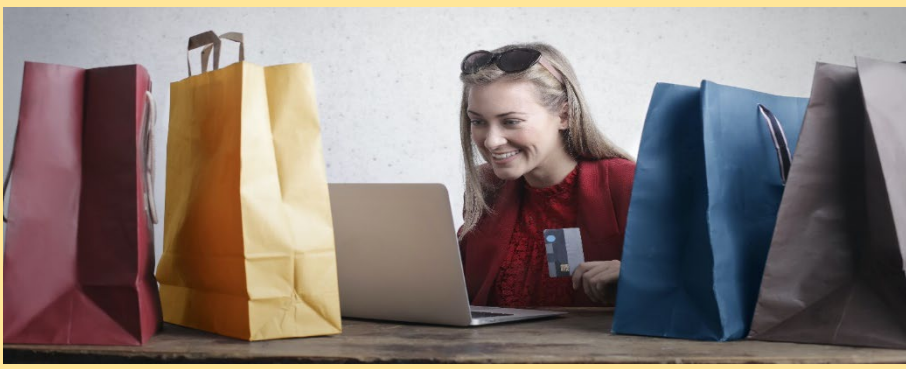
Application will be designed with a set of forms/pages with menus representing choice of activities to be performed.

Following are the functional requirements of the system (note, 'product' refers to clothing item products):

- **Home Page:** It will display general purpose information pages through menus such as Login, Registration, Settings, Products, Today's Deals, Find Product, Checkout, About Us, Contact Us, Feedback, and so on. A few featured products should be displayed on the home page. These will be displayed as images with brief captions.
- **Register:** It will enable new customers (henceforth called as users) to register themselves with ThriftFashion. At the time of registration, users must provide Name, Email ID, Contact Number, and Username, and then, select their Password.

Appropriate error-checking must be done on the fields of the form to ensure correct data. For example, email id can be checked to see if it is of appropriate format. (Hint: Use client-side validation).

- **Login:** It will allow successfully registered users to login to the ThriftFashion application and access various features of the application through menus or other options such as Settings (Create, Update, and Delete Profile will be the sub options). A profile can include detailed information about the user, such as address for dispatch of products, saving of credit card information, and so on.
- **Settings:** Users will be able to manage their accounts by using sub options such as Create, Update, and Delete Profile. Users can also add addresses using this option.
- **Products:** This option will showcase the products one can order online via ThriftFashion. Products can be displayed through an image gallery and can have product information such as availability, stock, price, discount (if any), product description, and so on.
- **Find Product:** Using this option, users can search for a specific product.



- **Sort Products:** User can view a sorted list of products based on price, brand name, or launch date of product.
- **Filter Products:** User can narrow down the list of products to get a filtered list based on price, brand, or launch date of product.
- **Checkout:** This should show the shopping cart contents and total bill based on product price and quantity of products purchased. You do not have to implement payment functionality, just list the cart contents and compute the bill.
- **Cancel Order:** User can cancel an already placed order through this option.
- **Contact Us:** This menu option should display Email id, address, and contact number of the organization who is developing the system.
- **Submit Feedback:** Users can submit their feedback regarding the application using a feedback form.

Note: Boilerplate or readymade HTML template can be used, provided it is only for design aspect and not for implementing application functionality.





1.6

Non-Functional Requirements

There are several non-functional requirements that should be fulfilled by the system.

The system should be:



- **Safe to use:** The system should not result in any malicious downloads or unnecessary file downloads.
- **Accessible:** The system should have clear and legible fonts, user-interface elements, and navigation elements.
- **User-friendly:** The system should be easy to navigate with clear menus and other elements and easy to understand.
- **Operability:** The system should operate in a reliably efficient manner.
- **Performance:** The system should demonstrate high value of performance through speed and throughput. In simple terms, the system should be fast to load and page redirection should be smooth.
- **Security:** The system should implement adequate security measures such as authentication. For example, only registered users can access certain features.
- **Capacity:** The system should support large number of users.
- **Availability:** The system should be available 24/7 with minimum downtime.
- **Compatibility:** The system should be compatible with latest browsers.



These are the bare minimum expectations from the project. Once the functional and non-functional requirements are fulfilled, you can use your own creativity and imagination to add more features.



1.7 Interface Requirements

Hardware

- Intel Core i5 Processor or higher
- 8 GB RAM or above
- Color SVGA
- 500 GB Hard Disk space
- Mouse
- Keyboard

Software

Frontend: HTML5, CSS3, Bootstrap, AngularJs/Angular, JavaScript, jQuery, AJAX, XML

Client & Server:

Java 8 or higher, Java EE 7 or higher/Jakarta EE
OR

C# 7.2 or higher with ASP.NET MVC 5 and ASP.NET MVC Core (optional)
OR

PHP 7.0 or higher version with Laravel Framework
OR

Python 3.0 or higher with Flask

Database: MySQL 5.7 or higher/SQL Server 2016 or higher



Database Design

Data Dictionary: User, UserProfile, Products, Deals, Billing, and so on

Based on the given specifications, you will define suitable entities, attributes for these entities, and identify relationships between the entities.

For example, some entities along with their attributes can be identified as follows:

User:
1. User ID
2. Name
3. Email ID
4. Contact Phone
5. Username
6. Password

Address:
1. User ID
2. AddressType
3. Address Line1
4. Address Line2
5. City
6. State
7. ZipCode

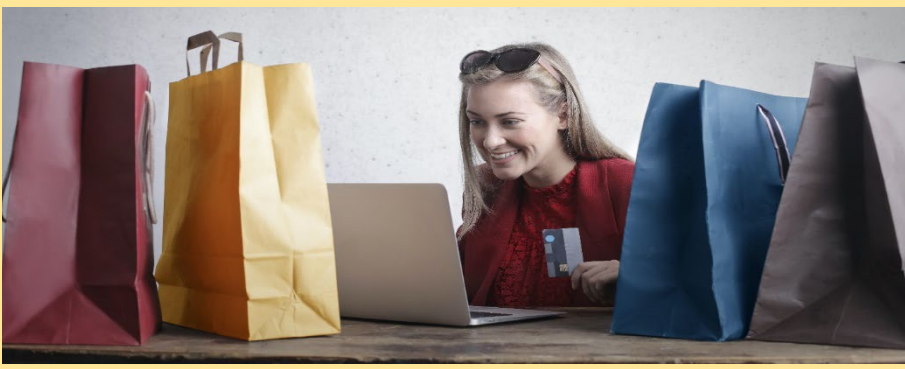
Product:
1. Product ID
2. Product Name
3. Description
4. Price
5. Stock
6. Launch Date

Billing:
1. User ID
2. Product ID
3. PurchaseDate
4. Quantity
5. Discount
6. Bill Amount

It is assumed
that all amounts
will be in \$

Similarly, you can define other entities and also relationships between entities and methods representing activities on the entities.

Note: These are just examples, you do not have to adhere to these structures and can design your own table structure with more or less columns.



1.8

Project Deliverables

You will design and build the project and submit it along with a complete project report that includes:

- Problem Definition
- Design specifications
- Diagrams such as flowcharts for various activities, Data Flow Diagrams, and so on
- Database Design
- Source Code
- Test Data Used in the Project
- Project Installation Instructions (if any)

Documentation is considered as a very important part of the project. Ensure that documentation is complete and comprehensive. The consolidated project will be submitted as a zip file with a ReadMe.doc file listing assumptions (if any) made at your end and SQL scripts files (.sql) containing database and table definitions. If local servers such as XAMPP are used to test the project, include screenshots of working pages. Include port settings and other details in ReadMe file.

Over and above the given specifications, you can apply your creativity and logic to improve the system.

Sitemap: To understand the flow of ThriftFashion Web application, you will have to create a Sitemap and add it to the home page of your application.

~~~ End of Document ~~~