**SRM INSTITUTE OF SCIENCE AND TECHNOLOGY**

Software Requirements Specifications

*for*

**Online fashion styling website**

***(BEYOND)***

This is the software requirements specifications (SRS) document for the online fashion styling system (BEYOND). In particular, the document details the specifications of Checkout and recommending process of the system. The intended audience of the SRS are users, system administrators and as per request the customer. Also, any future maintenance staff should be aware of the information contained herein. However, foreknowledge of certain computer systems and basic knowledge of computers is an asset to understanding the SRS.

Preface 2

Table of Contents 3

1. [Introduction](#_TOC_250009) 4
   1. Purpose 4
   2. Scope 4
   3. Definitions, Acronyms and Abbreviations 4
   4. References 5
   5. Overview 5
2. [Overall Description](#_TOC_250008) 6
   1. Product perspective 6
   2. Product functions 6
   3. User characteristics 7
   4. Constraints 7
   5. Assumptions and dependencies 8
   6. Specific Requirements 9
   7. External interface requirements 9
      1. [User interfaces](#_TOC_250007) 9
      2. [Hardware interfaces](#_TOC_250006) 9
      3. [Software interfaces](#_TOC_250005) 9
      4. [Communications interfaces](#_TOC_250004) 10
   8. Functional requirements 10
      1. [Use case 1](#_TOC_250003) 10
      2. [Use case 2 20](#_TOC_250002)
   9. Performance Requirements 15
   10. Design constraints 15

# Introduction

***1.1. Purpose***

The purpose of the document is to collect and analyze all assorted ideas that have come up to define the system, its requirements with respect to consumers. In short, the purpose of this SRS document is to provide a detailed overview of our software product, its parameters and goals. This document describes the project's target audience and its user interface, hardware and software requirements. It defines how our client, team and audience see the product and its functionality.

***1.2. Scope***

Online fashion styling system (BEYOND) is designed to provide quality service to the customers and to allow influencers and users to search for the suitable outfits and to add new outfits to the website so that people can improve their fashion sense and can access the outfits. The system requires Internet connection and has to be designed with a database capable of maintaining inventory details up-to-date. All outfits and accessories can be added to the favorites and chat bot can be used for other queries/info.

It needs to have 3 key features:

* + User friendly for computer illegitimate.
  + Easy to maintain by administrators.
  + Only accessible by registered users.

***1.3. Definitions, Acronyms and Abbreviations***

Modem - modulator-demodulator; a device that modulates an analog carrier signal to en code digital information, and also demodulates such a carrier signal to decode the

transmitted information.

Ethernet – a family of frame – based computer networking technologies for Local Area Networks.

MySQL – is a RDBMS based on SQL, which is used for adding, removing, and modifying information in the database.

RDBMS – Relational Database Management System SQL – Structured Query Language

HTML – Hypertext Markup Language PHP – Hypertext Preprocessor XAMPP -

CSS – Cascading Style Sheet HTTP – Hypertext Transfer Protocol

***1.4. References***

### --WeConnect

<https://www.weconnectfashion.com/articles/section/trends>

***1.5. Overview***

The remaining sections of this document provide a general description, including characteristics of the users of this project, the product's hardware, and the functional and data requirements of the product. : Section 2 gives an overall description of the software. It gives what level of proficiency is expected of the user, some general constraints while making the software and some assumptions and dependencies that are assumed. Section 3 gives specific requirements which the software is expected to deliver. Functional requirements are given by various use cases. Some performance requirements and design constraints are also given.

# Overall Description

***2.1. Product perspective***

The product is a web-based system that allows the customer to view recommended outfits using the Internet and allows customers to search the items of their desire from the site. It is aimed towards the people who don’t know what suits them and what not, how to put outfits together, what to buy, and unaware of what to wear even if they have a closet full of clothes.Therefore, here we come up with a fashion styling website which solves all these mentioned problems. It should be user-friendly, ‘quick to learn’ and reliable application for the above purpose.

***2.2. Product functions***

It consists of two modules:

Customer module Administrator module

* A customer should have a user account for saving and customizing the outfits.
* Administrator provides the customer with an account following proper registration procedures to prevent malpractices in the usage of the website.
* Any visitor is allowed to browse through the product list, their prices and bidding procedures.

Account Management (AM): AM allows users to create, edit, and view accounts information. It

also allows the user to log-in/out of the system.

Search Engine (SE): SE is the tool that assists the user in finding a specific item in the database. It can receive search criteria, find search criteria, and return the results of the search.

Product Management (PM): PM allows administrator to manage the product line shown on

the web site.

Favorites: it is temporary storage for customers browsing on the web. Items from the favorites can be viewed on the site at any point of time.

***2.3. User characteristics***

There are following types of users in the Online fashion styling system (BEYOND).

**Administrators:** The administrators have complete control over all the activities that can be performed.

* They verify the user after he/she registers.
* They must provide rules for the usage of the website.
* They must maintain the website and update the same making necessary changes at times.
* They must take care of the security issues involved in the transactions.
* They must inform the users about their transaction status and keep them updated about the progress through emails.
* Any failures in the system have to be detected and repaired.

**Customers:** They are the registered members of the online fashion styling website. They can view the list of products.

***2.4. Constraints***

* **Cold start**- A new user or item added based problem is difficult to handle as it is impossible to obtain a similar user without knowing previous interest or preferences.
* **Sparsity-** It happens many times when most of the users do not give ratings or reviews to the items they purchased and hence the rating model becomes very sparse which could lead to data sparsity problems, it decreases the possibilities of finding a set of users with similar ratings or interest.
* **Synonymy-** Synonymy arises when a single item is represented with two or more different names or listings of items having similar meanings, in such condition, the recommendation system can’t recognize whether the terms shows various items or the same item.
* **Privacy-** The recommendation system is bound to have the personal information of users and use it to the fullest in order to provide personalized recommendation services. To deal with this issue, the recommendation systems must ensure trust among their users.
* **Scalability-** Recommendation systems interpret results on large datasets inefficiently, some advanced large-scaled methods are required for this issue.
* **Latency-** We observe many products are added more frequently to the database of recommendation systems, only already existing products are recommended to users as newly added products are not rated yet. So an issue of Latency arises. The collaborative filtering method and category-based approach in combination with user-item interaction can be used to deal with this issue.

***2.5. Assumptions and dependencies***

- Administrator is created in the system already.

**3. Specific Requirements**

***3.1. External interface requirements***

### User interfaces

The user interface shall be web-based, allowing users to remotely access the system via several applications. Users will be able to use the software through applications such as Microsoft Internet Explorer, Mozilla, et al. Each part of the user interface intends to be as user friendly as possible. The fonts and buttons used will be intended to be very fast and easy to load on web pages. The pages will be kept light in space so that it won’t take a long time for the page to load. The starting page will be a home page regarding the products and a number of buttons with the specific category of the products catalog/favorites. The customer will be given an option to search a product or a category through the search engine provided in the website. In adding product to his/her favorite, the customer will only have to drag the desired product to be purchased and drop it to the cart. While in removing the product, he/she will select the item in the favorites. There will be buttons added for the managing of accounts of the customer and the administrator.

### Hardware interfaces

Since the application must run over the internet, all the hardware shall require to connect internet will be hardware interface for the system. As for e.g. Modem, WAN – LAN, Ethernet Cross-Cable.

### Software interfaces

During our system development, we have to design both static and dynamic website interfaces, create website functions and a database system, edit photos and pictures, and print out reports, so its has a set of software requirements.

The following are needed requirements.

Web browser - Microsoft Internet PHP

XAMPP application MySQL Database server

### Communications interfaces

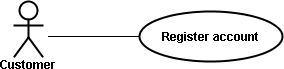
The system shall use the HTTP protocol for communication over the internet.

***3.2. Functional requirements***

### Use case 1

**Customer**

**Use Case: Register account**

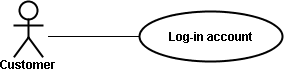
Diagram:

Description: Register a new customer account with the system. Steps:

1. Customer first clicks on the button or link to initiate registration process.
2. System prompts the customer to fill out his/her first name, last name, email address, and their password.
3. Customer enters fields.
4. System validates the customer's information.
5. System creates a new account for the Customer.
6. System displays an account home page to Customer.

## Use Case: Log-in account

Diagram:

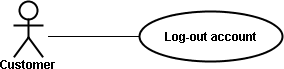


Description: Login to a customer account with the system. Steps:

1. Customer clicks on the button or link to initiate the login process.
2. System prompts the customer for his/her email and password.
3. System verifies the information.
4. System displays account home page to the Customer.

## Use Case: Log-out account

Diagram:

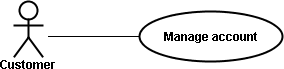


Description: Logout the customer account on the system. Steps:

1. Customer clicks the button or link in order to initiate logout process.
2. System displays the home page.

## Use Case: Manage account

Diagram:

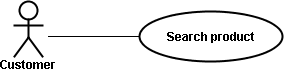


Description: Edit the customer account details. Steps:

1. Customer clicks the button or link to initiate the process to edit the account.
2. System displays the account home page to the Customer.
3. Customer clicks the button or link in order to edit the account details.
4. System verifies the changes.
5. System stores new account information.

## Use Case: Search Product

Diagram:

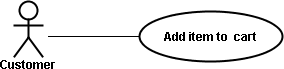


Description: Customer searches the desired product. Steps:

1. Customer enters a category or name of the product in the search engine.
2. System searches the matching product.
3. System displays the results.

## Use Case: Add item to Favorite

Diagram:

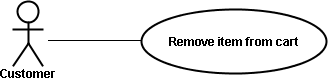


Description: Customer adds item(s) in the favorites. Steps:

1. Customer specifies the keywords of the requirements for a good outfit.
2. System checks the availability of the product.
3. Customer drags the desired item and drops it to the favorites.
4. System adds the item(s) to the cart.
5. Customer confirms that the item is added to favorites.
6. Customer returns to favorites.

## Use Case: Remove item from Favorite

Diagram:

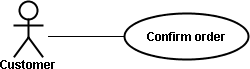


Description: Customer removes item(s) in the cart Steps:

* 1. Customer selects the item in the cart and clicks the button to initiate the deletion of item(s).
  2. System removes the item(s) from the cart.

## Use Case: Create Collection

Diagram:



Description: Customer creates a collection out of his/her favorite outfits and accessories

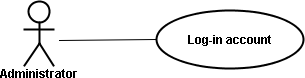
. Steps:

1. Customer selects an outfit or an accessory from product catalog/favorite
2. Customer clicks on the menu of that particular item and selects the option “add to collection”.
3. System adds the selected item to the collection and it will be displayed in the collection.

## Administrator

**Use Case: Log-in account**

Diagram:

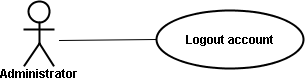


Description: Login to an Administrator account with the system. Steps:

1. Administrator clicks button or link to initiate login process.
2. System prompts the Administrator for email and password.
3. System verifies information.
4. System displays account home page to Administrator.

## Use Case: Logout account

Diagram:



Description: Logout the Administrator account on the system. Steps:

* 1. Administrator clicks button or link to initiate logout process.
  2. System displays home page.

### Use case 2

***3.3. Performance Requirements***

Maximum possible quick response to the orders is required, also should provide fast updation of records. The changes if any made should be reflected automatically in the next screens.

***3.4. Design constraints***

The design constraints are that the browser at each place may not follow similar screen resolutions, browsers etc. This can lead to the website not having the impact it is planned to have.

***3.5. Software system attributes***

The necessary qualities of software products are

* + 1. Security:

The application is password protected and also any updation of new product entries is done by only privileged users.

* + 1. Maintainability:

The application is to be designed so that it is easily maintained. Also it should allow incorporating new requirements in any module of system.

* + 1. Portability

The application will be easily portable on any window based system.