Software Requirements Specification

For Credible Seller’s presenting

in-Credibles

**Version 1.0 approved**

**Prepared by Group No. 3**

**5215**

**14/10/2021**

**Table of Contents**

[**Introduction**](#_3znysh7) **1**

[**1.1 Purpose**](#_2et92p0) **1**

[**1.2 Document Conventions**](#_tyjcwt) **1**

[**1.3 Intended Audience and Reading Suggestions**](#_3dy6vkm) **1**

[**1.4 Product Scope**](#_1t3h5sf) **2**

[**1.5 References**](#_4d34og8) **2**

[**2. Overall Description**](#_2s8eyo1) **3**

[**2.1 Product Perspective**](#_17dp8vu) **3**

[**2.2 Product Functions**](#_3rdcrjn) **3**

[**2.3 User Classes and Characteristics**](#_lawvc8exgtui) **4**

[**2.4 Operating Environment**](#_lnxbz9) **4**

[**2.5 Design and Implementation Constraints**](#_n5elz47cyhyj) **4**

[**2.6 User Documentation**](#_1ksv4uv) **5**

[**2.7 Assumptions and Dependencies**](#_w93hwvqmk7bn) **5**

[**3. External Interface Requirements**](#_1ybjwgjf4v9h) **6**

[**3.1 User Interfaces**](#_lrw7vo4agrfx) **6**

[**3.2 Hardware Interfaces**](#_9gcnqzrhittd) **6**

[**3.3 Software Interfaces**](#_hdrdaao2uk1l) **6**

[**3.4 Communications Interfaces**](#_q9f3iosqmmnj) **6**

[**4. System Features**](#_2jxsxqh) **7**

[**4.1 Account Management**](#_4huzz74p2tj4) **8**

[**4.2. Add / Delete sellers from user’s list.**](#_vturrxhygruq) **13**

[**4.3. Searching/Browsing Feature**](#_qijz8iyjzs7j) **16**

[**4.4. Sellers Private List**](#_3j2qqm3) **20**

[**4.5. Sellers Public Credibility**](#_1y810tw) **22**

[**4.6. Rating System**](#_3f5ehzct36k5) **24**

[**4.7. Sell Product**](#_4i7ojhp) **27**

[**4.8. Shopping cart & Transaction Management**](#_huvn4qvq9lx) **30**

[**5. Other Nonfunctional Requirements**](#_2xcytpi) **35**

[**5.1. Performance Requirements**](#_1ci93xb) **35**

[**5.2. Safety Requirements**](#_3whwml4) **35**

[**5.3. Security Requirements**](#_2bn6wsx) **36**

[**5.6. Software Quality Attributes**](#_qsh70q) **36**

[**5.5. Business Rules**](#_3as4poj) **37**

[**6. Other Requirements**](#_1pxezwc) **37**

**Revision History**

| **Name** | **Date** | **Reason For Changes** | **Version** |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |

# Introduction

## 1.1 Purpose

This document is meant to delineate the features of in-Credibles, to serve as a guide to the developers on one hand and a software validation document for the prospective client on the other.

in-credibles - The Credibility Seller’s List for Online Sellers, a mobile and desktop-based application is intended to provide complete solutions for vendors and retailers through a single gateway using our application. It will enable clients to set up their network connections with ease, they will be able to set up the connection they want only with the clients that meet their benchmark of credibility score with just a click away and without having to visit and meet the clients physically.

## 1.2 Document Conventions

The document is divided into 4 major sections that include the Introduction, Overall Description, Product Features, and the Non-functional requirements. Each section is further divided into subsections to provide in-depth detail for the readers of this document.

To highlight some important ideas, the words are written in bold.

## 1.3 Intended Audience and Reading Suggestions

This document is intended for the various members that aid in developing and designing software. Such members include the software developer, marketing staff, users, and testers.

Section 1: This section talks about the product itself that we have to offer and what its purpose is. Moreover, it also tells us who this product is mainly targeted for, therefore, it can provide the marketing staff with an idea of which kind of customers they should be targeting when making marketing plans.

Section 2: This section briefs about the general functions the product has to offer and what need arose for the development of this software. This section also gives an overview of the constraints regarding design and implementation and also what dependencies the software might have and what assumptions we’re making.

Section 3: A detailed description of the functional requirements that the system has to offer is given. Every functional requirement is explained thoroughly along with providing the use cases. This section can be vital for the system developer as it will help them identify each attribute of the functional requirement and work on it.

Section 4: Non - functional requirements of the system are highlighted and can help the developers as well as the testing team to understand what approach to use to make the software efficient, effective, portable, maintainable, and usable.

## 1.4 Product Scope

* The purpose of in-Credibles is to let the buyers know the credibility of their sellers to provide them a safe and secure shopping experience.
* Nowadays people find online shopping very convenient so in-Credibles will ease their shopping experience even more with the seller's credibility in front of them.
* in-Credibles has a responsive user interface that lists categories of the products hence people from all business or professional domains can sell their products.
* This software aims to provide buyers with a safe space to purchase their desired products.
* The credibility list will ensure that the buying/selling process is done legitimately without any scams.
* No malicious sellers/buyers also called “black hat” hackers could attack the online business.
* The system registers the seller’s region to avoid problems.
* The initial version of the software is limited to working in Pakistan but can be expanded to foreign countries in the future.

## 1.5 References

* Buy and sell for free anywhere in Pakistan with OLX Online Classifieds. OLX. (n.d.). Retrieved October 9, 2021, from <https://www.olx.com.pk/>
* Daraz Group is a multinational technology company that focuses on e-commerce, logistics, payment infrastructure and financial services.
* Retrieved from Alibaba.com: <https://www.alibaba.com/>
* HOW TO SELL: The OLX selling and shipping process. (n.d.). Retrieved October 14, 2021, from eurosender.com: <https://www.eurosender.com/en/selling-guide/olx>
* Bandakkanavar, R. (2018, july 4). *Software Requirements Specification document with example*.Retrieved,October-9-2021,from: <https://krazytech.com/projects/sample-software-requirements-specificationsrs-report-airline-database>
* [Internet Marketing Ethics and Web Ethical Issues | Professional Web Services (pwebs.net)](https://pwebs.net/i/internet-ethics/)
* [D1201022936.pdf (iosrjournals.org)](https://iosrjournals.org/iosr-jece/papers/Vol.%2012%20Issue%201/Version-2/D1201022936.pdf)

# 2. Overall Description

## 2.1 Product Perspective

The in-Credibles is an open-source mobile and web-based application implementing a client-server model. It implements intermediary centric B2B e-commerce model. It facilitates matching and expanding the network of credible buyers and sellers of goods and services.

It is an extension of an already existing e-commerce system however with the additional component of credibility. These targeted audiences are busy people who don’t have time to shop or who simply might not be interested in visiting stores and dealing with a lot of formalities.

## 2.2 Product Functions

1. The system will allow new users to create an account.
2. The system will allow users to log in.
3. The system will allow users to log out of the system.
4. Users can browse for product items.
5. Users can view product categories.
6. Users can search for sellers.
7. Users can add a seller to their public seller’s list.
8. Users can remove a seller from their public seller’s list.
9. Users can add or update the credibility score of sellers.
10. Users can access their seller's list.
11. Users can view mutual sellers and credibility scores in their public seller’s list.
12. Users can live chat with the sellers. (TBD7)
13. Users can opt for online payment or cash on delivery. The system will generate a receipt of their transaction.
14. Users can add multiple products to their shopping cart.
15. Users can view and edit their profile information.
16. The system keeps track of each user’s transaction history.
17. Admin has the right to modify/cancel orders.
18. Admin can view the transaction history of every seller.
19. Users can add products that they want to sell.
20. Users can delete their account.

## 

## 2.3 User Classes and Characteristics

There are 3 user classes in this software

1. Administrator: The administrator of the whole in-Credibles system. They will be involved only when there is an issue in the transaction process. Responsible for overseeing the daily transaction process and resolving any issues related to the transaction process.
2. Seller: They will register and add their products for selling purposes. Sellers will be frequent users of this product and will partake in many of the product functions stated above. They will also be the only users who can give credibility scores to other sellers. Sellers will be expected to sell products and give credibility scores to other sellers responsibly.
3. Buyer: They will buy products from sellers. They will also be frequent users depending on their needs of getting an item. Buyers will be expected to buy products from sellers based on their credibility rating. Buyers can also take the role of a seller.

## 2.4 Operating Environment

Hardware requirements for insurance on the internet will be the same for all the sellers which are as follows:

1. Processor: Dual Core
2. RAM: 2 GB is required.
3. Hard Disk: 20 GB should be available.
4. Android version: 5.0 and up.
5. NIC( Network interface card): For each Seller
6. Internet connection should be available.
7. The components of the software shall work with the Windows operating system environment and Android operating system.
8. Database to be used = TBD6

## 

## 2.5 Design and Implementation Constraints

Items and issues that may limit the options available to the software developers are legal and ethical constraints regarding various e-business legal and Internet marketing ethical issues of both B2B and B2C business practices online and also customer’s organization’s corporate and regulatory policies.

It is anticipated that all related governing directives both social and governmental regulations will be adhered to;

Copyright infringement is strictly not allowed as the content of the website should be obeying the copyright law reference: Copyright Ordinance, 1962.

The expansion of global domain name registrars from all countries across the world, most of which don't even share the same social/ethical value standards, makes protecting company trademarks, copyrights, and proprietary rights even more challenging and can cause it difficult to protect corporate identities on the Internet.

The increasing growth of e-commerce has caused the number of online transactions to rise which involves the exchange of sensitive information such as credit card details and personal information which brings a security and privacy concern of being exposed to hacking and fraud. Providing best security features using encrypted network communication and should have a certification from a certifying authority for meeting certain security standards. Employing encryption for keeping user information secure may impose a processing overhead that might hinder timing requirements.

In case of an influx of users then the system might not be able to handle all the users and there might be a possibility of crashing due to heavy load. In such a case, no testing features might be available.

The nature of the project requires keeping a track of the source code between developers and hence we plan to implement a control system using Github by using pull/push code commits to/from Github.

The system is compatible with Microsoft Windows operating systems and Android operating systems.

## 2.6 User Documentation

User Documentation in a printable format (PDF) to be provided to the client to be easily accessed and shared. it will include a user manual for the sellers, technicians and a copy of documentation and link to the current source for future contributors.

The system will also include a “Help” tab which will redirect the users to our Help Centre which will include a FAQ section, links to video tutorials for guidance and a link to the Support Portal and Community where the users can share questions concerning any issues and other users may answer them.

Meanwhile, for more information, users may contact our online 24/7 help support at [in-Credibles@softengr.com](mailto:csl@softengr.com).

## 2.7 Assumptions and Dependencies

* The user has access to a computer/mobile phone.
* The IPv4 link has been established.

# 3. External Interface Requirements

## 3.1 User Interfaces

<Describe the logical characteristics of each interface between the software product and the users. This may include sample screen images, any GUI standards or product family style guides that are to be followed, screen layout constraints, standard buttons and functions (e.g., help) that will appear on every screen, keyboard shortcuts, error message display standards, and so on. Define the software components for which a user interface is needed. Details of the user interface design should be documented in a separate user interface specification.>

## 3.2 Hardware Interfaces

<Describe the logical and physical characteristics of each interface between the software product and the hardware components of the system. This may include the supported device types, the nature of the data and control interactions between the software and the hardware, and communication protocols to be used.>

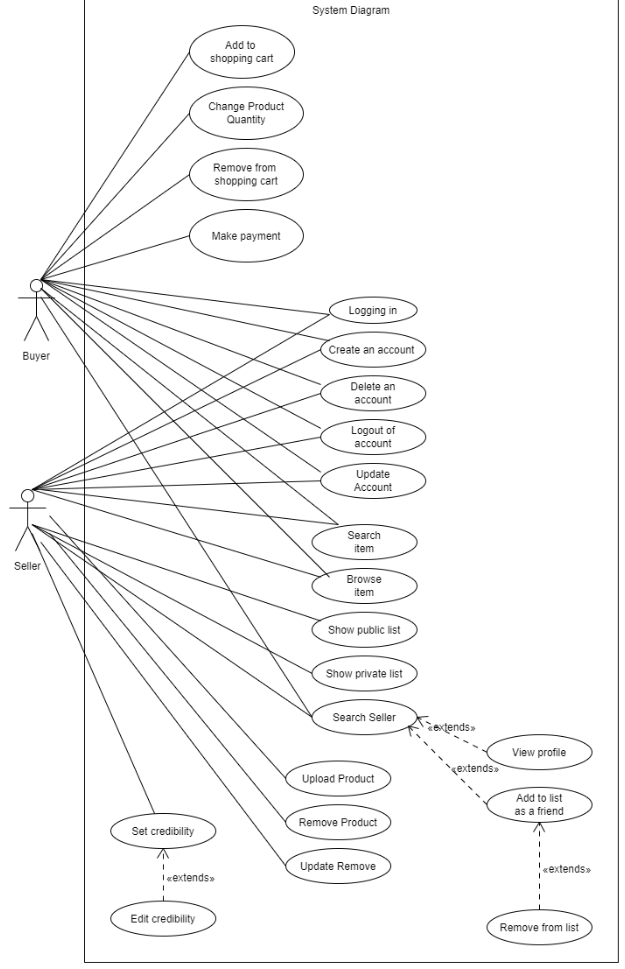
## 3.3 Software Interfaces

<Describe the connections between this product and other specific software components (name and version), including databases, operating systems, tools, libraries, and integrated commercial components. Identify the data items or messages coming into the system and going out and describe the purpose of each. Describe the services needed and the nature of communications. Refer to documents that describe detailed application programming interface protocols. Identify data that will be shared across software components. If the data sharing mechanism must be implemented in a specific way (for example, use of a global data area in a multitasking operating system), specify this as an implementation constraint.>

## 3.4 Communications Interfaces

<Describe the requirements associated with any communications functions required by this product, including e-mail, web browser, network server communications protocols, electronic forms, and so on. Define any pertinent message formatting. Identify any communication standards that will be used, such as FTP or HTTP. Specify any communication security or encryption issues, data transfer rates, and synchronization mechanisms.>

# 4. System Feature



## 4.1 Account Management

**4.1.1 Description and Priority**

Account management is a high priority feature where the system should be able to facilitate account management for multiple users. System should have a provision for users to create, delete and log into an account.

**4.1.2 Stimulus/Response Sequences**

* Login Case: User opens the website via supported browser/android app. System displays a pop-up box with ‘login’ and ‘signup’ buttons. User clicks on the login button and enters its credentials i.e. name and password which is then cross verified by the system. System grants access to the seller's page upon successful login attempts.

* Signup Case: User opens the website via supported browser/android app. System displays a pop-up box with ‘login’ and ‘signup’ buttons. User clicks on the signup button which prompts the system to display a registration form. User fills out the form and the system saves the information of the user in the database and creates a unique user-name for the user. System grants access to sellers’ page upon successful signup attempt.

* Reset Credentials Case: User clicks on the login button and enters invalid credentials i.e. name and password which is then cross verified by the system. System displays an error message and reset credential button. System checks for the invalid credentials and displays the option to reset it. User enters new credentials along with proof of identity. System updates the credentials of users in the database.

* Logout Case: User navigates to settings and clicks on the logout button. System approves the request. User successfully logs out.
* Delete Case: User navigates to settings and clicks on delete account button. System prompts a verification box. System deletes the account upon successful attempt.

**4.1.3 Functional Requirements**

*REQ-AM1:* The System should display a popup box with ‘login’ and ‘signup buttons when a user clicks on the website/app.

*REQ-AM2:* Users will have the option to enter their credentials at the login screen.

*REQ-AM3:* System should cross-validate the UserID and password. If correct credentials are provided the system will grant access to the seller's page otherwise an error message will be displayed.

*REQ-AM4:* System will lock the user account on 3 failed login attempts.

*REQ-AM5:* Users will have the option to register an account. System should display a registration form where users will fill out details such as their name, email, phone, marriage status, organization, address, age, and CNIC. System should generate a unique UserId for users.

*REQ-AM6:* System should display a confirmation message upon successful registration attempt. System should email users with a link to confirm the provided email by the user.

*REQ-AM7*: Users will have the option to upload their pictures.

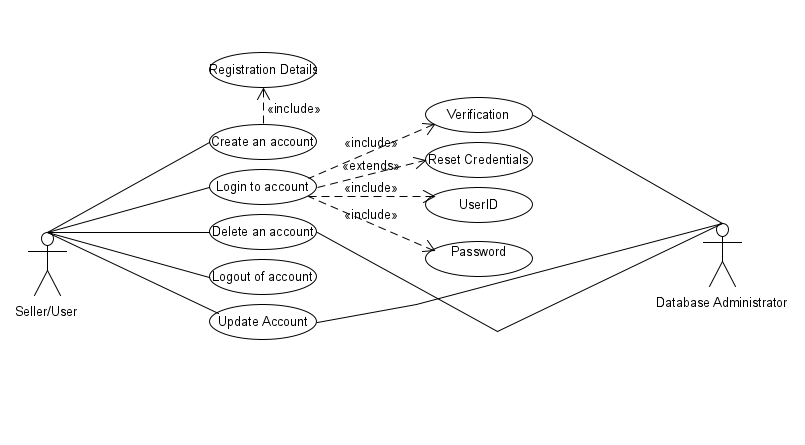
*REQ-AM8*: Users will have the option to reset their passwords.

*REQ-AM9:* System should display a reset password option upon an unsuccessful login attempt.

*REQ-AM10:* System should email the user with an encrypted link. Users will have the option to enter a new password. System should update the user’s new password in the database.

*REQ-AM11:* System should validate the type of input user provides while registering for an account. System should display an error message in case the user enters invalid inputs in registration fields.

*REQ-AM12:* System should make all fields in registration form mandatory i.e system should prompt an error message if the user doesn’t fill out any of the fields.



| **Use Case Name** | **Create a new seller account** |
| --- | --- |
| Related Requirements | AM1 , AM2 , AM5 , AM6 , AM11 |
| Goal in Context | A new user asks to sign up on in-Credibles |
| Preconditions | New user should have necessary information for proof of identity |
| Successful End Conditions | An account is created for the new user |
| Failed End Conditions | An account isn’t created for the new user |
| Primary Actors | User |
| Secondary Actors | None |
| Triggers | User clicks on the signup button on in-Credibles website/app |
| Included Cases | N/A |
| Main Flow | | Step | Action | | --- | --- | | 1 | New user clicks on signup button to create an account | | 2 | The system displays a registration form | | 3 | The new user fills out the registration form | | 4 | System validates the details and saves it in the database or displays an error message incase of invalid input or empty fields in form. | | 5 | New in-Credibles account is created | | 6 | A confirmation email is sent to the user's email address. | |
|  |
|  |

| **Use Case Name** | **Login account** |
| --- | --- |
| Related Requirements | AM1 , AM2 , AM3 , AM4 , AM11 |
| Goal in Context | An existing user requests to login to in-Credibles |
| Preconditions | User should have an existing account with valid credentials |
| Successful End Conditions | User is logged into in-Credibles account |
| Failed End Conditions | in-Credibles denies access to user’s account |
| Primary Actors | User |
| Secondary Actors | Database Administrator |
| Triggers | User clicks on the login button on in-Credibles website/app |
| Included Cases | Reset credentials |
| Main Flow | | Step | Action | | --- | --- | | 1 | User clicks on login button to access their account | | 2 | The system displays login screen | | 3 | The user enters their credentials | | 4 | Database Administrator cross verifies the credentials in database | | 5  Include : reset credentials | System displays successful login message upon successful login attempt or displays error message and reset credentials option | | 6 | System grants access to users account | |
|  |
|  |

| **Use Case Name** | **Reset credentials** |
| --- | --- |
| Related Requirements | AM3 , AM9 , AM10 , AM11 |
| Goal in Context | An existing user requests to reset credentials of in-Credibles account |
| Preconditions | User should have an existing account with necessary identity of proof |
| Successful End Conditions | User is able to reset their credentials |
| Failed End Conditions | in-Credibles denied the privilege to reset credentials |
| Primary Actors | User |
| Secondary Actors | Database Administrator |
| Triggers | User enters invalid credentials |
| Included Cases | N/A |
| Main Flow | | Step | Action | | --- | --- | | 1 | User clicks on login button to access their account | | 2 | The system displays login screen | | 3 | The user enters invalid credentials | | 4 | Database Administrator cross verifies the credentials in database | | 5 | System displays error message and resets credentials option | | 6 | Database Administrator checks the invalid credential and displays the option to reset it. | | 7 | User enters new credentials and proof of identity | | 8 | System validates the new credentials and updates it in the database | | 9 | System grants access to user’s account | |
|  |
|  |

## 4.2. Add / Delete sellers from user’s list.

**4.2.1 Description and Priority**

This is a medium priority feature. With this feature, the user will be able to search for sellers, and if interested they can add sellers to their public lists. Every user of the application will have their own public and private lists. Users will be able to see the direct connections and products of sellers who will be in their public lists, but won't be able to see products and connections of sellers who will be in their private lists.

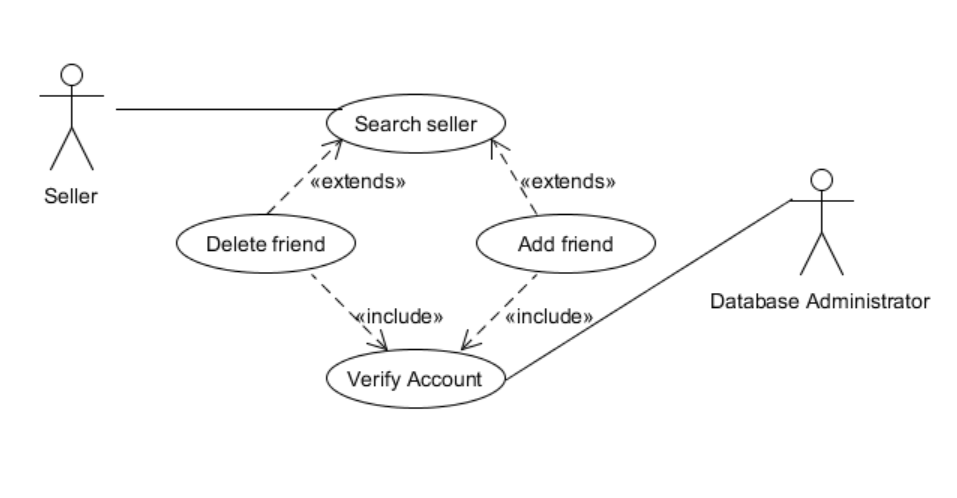
**4.2.2 Stimulus/Response Sequences**

* Users will click ‘Add Seller’ to add sellers to their public list.
* Users will click ‘Delete Seller’ to delete a seller from their private list, and the database will be updated too.

**4.2.3 Functional Requirements**

*REQ-AD1:* User will click on ‘Add seller’ through a responsive user interface to add a seller to his public list and an update will be made in the database, adding the seller to the buyer's list. The database will be checked for the seller and if the seller is already in the user's list, the buyer will be prompted that the seller already exists, and the user could click on the seller's profile to view items on sale.

*REQ-AD2:* User will click on ‘Delete seller’ through a responsive user interface. The database will be checked for the seller and if the seller is already not in the user's list, the user will be prompted that seller does not exist in his public list. However, if the seller exists in the user's public list, he will simply be deleted and an update will be made in the database, adding the seller to the buyer's list.



| **Use case name** | **Adding / Deleting a seller from the seller's list.** | |
| --- | --- | --- |
| Related Requirements | AD-1, AD-2 | |
| Goal in Context | Successfully add / delete a seller from the seller’s list. | |
| Preconditions | Must have an existing account on in-Credibles. | |
| Successful End Conditions | Seller added to the buyer’s list (in case of adding a seller)  Seller deleted from the buyer’s list (in case of deleting a seller) | |
| Failed End Condition | Failed to add / delete a seller from the buyer's list. | |
| Primary Actors | Buyer/Seller | |
| Secondary Actors | None | |
| Trigger | User initiates the process to add a seller to his list / delete seller from his list. | |
| Included Cases | **None** | |
| Main flow | Step | Action |
|  | 1 | User searches for a seller. |
|  | 2 | If the user is interested in that seller, he may add the seller to his public list. If a user wishes to cut connection with that seller, he may delete him from his public list. |
|  | 3  **Include::Verify account** | The account’s existence will be verified from the database. |

## 4.3. Searching/Browsing Feature

***4.3.1 Description and Priority***

The searching/browsing feature includes searching/browsing for items, people and also being aware of the trending searches. This is a high priority feature as with the help of this feature an individual will be able to search for their friends on the platform and add them and also be able to find any item they are looking for. Moreover, this feature will enable an individual to know of all the items that are being merchandised on the platform. On a scale of 1-10, this feature will be rated a 7-8 overall. The search seller component however, is a high priority requirement as without being able to search for a seller, users will be unable to add them as friends. This feature will help an individual successfully grow their business and build strong connections as well.

***4.3.2 Stimulus/Response Sequences***

* When the user clicks on any category of any particular item, the person will have the option to browse through the items and look through trending items and trending sellers for any particular items.
* When the user uses the search bar, they will have the option to search for any certain seller or item.

***4.3.3 Functional Requirements***

*REQ SB1:* Users should be able to browse the items available

*REQ SB2*: Users should be able to view the trending items

*REQ SB3:* Users should be able to view the trending sellers for particular item category

*REQ SB4:* Users should be able to search up sellers by their names

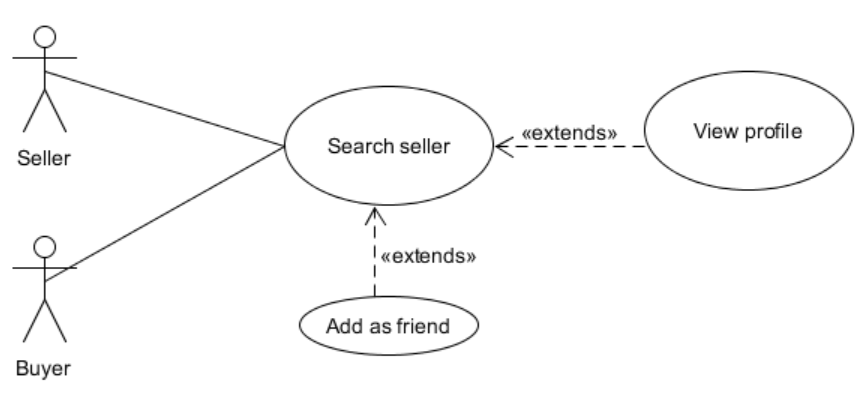
*REQ SB5:* Users should be able to search up particular items

*REQ SB6:* Users will be aware of the variety of products available on the platform.

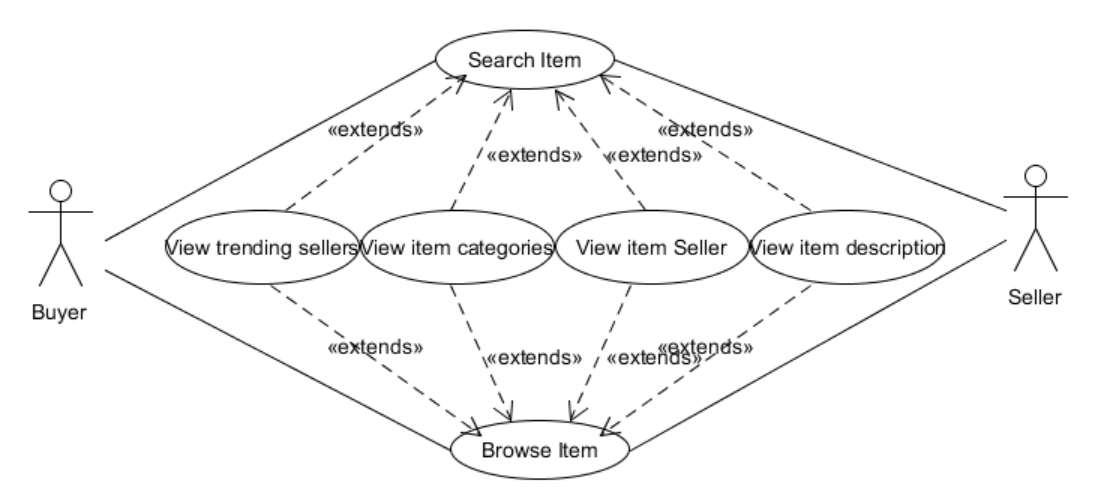
*REQ SB7:* Sellers can search for other sellers and then be able to add them

*REQ SB8*: In case an item that is not available on the platform is searched for, the user should be made aware of the unavailability.

*REQ SB9*: In case the user adds in the name of a person who is not on the platform, the user should be made aware of the non-existence of the person on the platform.



| **Use case name** | **Searching Seller** | |
| --- | --- | --- |
| Related Requirements | REQ SB4, REQ SB7, REQ SB9 | |
| Goal in Context | To search for any individual | |
| Preconditions | User should have an account and be logged on | |
| Successful End Conditions | Successfully be able to search for any seller | |
| Failed End Condition | Searching gives errors and no result is shown. | |
| Primary Actors | Seller, Buyer | |
| Secondary Actors | None | |
| Trigger | When an individual goes on the search bar. | |
| Included Cases | None | |
| Main flow | Step | Action |
|  | 1 | The user clicks on the search bar |
|  | 2 | The user types in the name of the person they want search |
|  | 3 | The user can now view all people with the name they have searched |
|  | 4 | The user can add the person as their friend |
|  | 5 | The user can view the items the person sells |



| **Use case name** | **Browsing & Searching Items** | |
| --- | --- | --- |
| Related Requirements | REQ SB1, REQ SB2, REQ SB3, REQ SB5, REQ SB6, REQ SB8 | |
| Goal in Context | To browse or search for any sort of item | |
| Preconditions | User should have an account and be logged on | |
| Successful End Conditions | Successfully be able to browse or search for any item | |
| Failed End Condition | Browsing or searching gives errors and no result is shown. | |
| Primary Actors | Seller, Buyer | |
| Secondary Actors | None | |
| Trigger | When an individual goes on the search bar or is surfing through the home page or any item category. | |
| Included Cases | None | |
| Main flow | Step | Action |
|  | 1 | The user clicks on any item category. |
|  | 2 | The user browses through the items in the list |
|  | 3 | The user can view the product details and also the respective seller’s for that product. |
|  | 4 | The user can also view the trending sellers for the certain item that they have searched for |
|  | OR | |
|  | 1 | The user clicks on the search bar |
|  | 2 | The user types in the name of any item they want to search |
|  | 3 | A list of items show up in accordance with the search |
|  | 4 | The user can view the product details and also the respective seller’s for that product. |
|  | 5 | The user can also view the trending sellers for the certain item that they have searched for |

## 4.4. Sellers Private List

**4.4.1 Description and Priority**

Private listof all the individual Seller’s to keep track of all the transactions. This feature is of **Medium Priority.**

**4.4.2 Stimulus/Response Sequences**

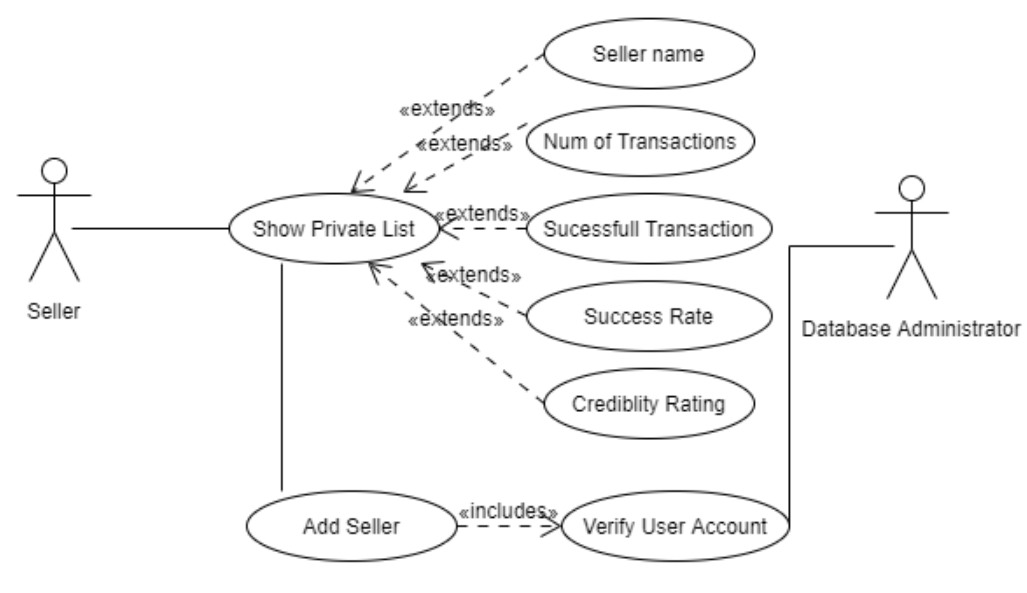
Users will first be prompted to Login and after that there will be a navigation button which will open and list down all his Private Transactions from each seller he/she has traded with.

**4.4.3 Functional Requirements**

*REQ-SPL1:* Private list of transactions of each seller will be not available to view by anyone and would only be restricted to the user who is logging in.

*REQ-SPL2:* This list will show all the previous sellers and their transactions with their success rate.

*REQ-SPL3*: There will be a button for each seller from this list to add them as their connection, in that case that seller will be added in the public seller list and the user will be asked to provide the credibility for the new connection.



| **Use Case Name** | **Sellers Private List** |
| --- | --- |
| Related Requirements | REQ-AD1, REQ SB9 |
| Goal in Context | To manage the transactions that took place |
| Preconditions | User should have an account signed up |
| Successful End Conditions | User will be displayed all the information of his sellers |
| Failed End Conditions | Unable to load the Seller information |
| Primary Actors | User |
| Secondary Actors | Database Administrator |
| Triggers | User clicks on show sellers |
| Included Cases | TBD-2 |
| Main Flow | | Step | Action | | --- | --- | | 1 | Users will be shown the button to view their private list. | | 2 | Users will be shown the list and details of all Sellers. | | 3 | Users will also be shown button to add sellers to public credibility list for each seller from private list | |
|  |
|  |

## 4.5. Sellers Public Credibility

**4.5.1 Description and Priority**

Public Credibility/Ratingof all the individual Seller’s for others to see and expand their network. This feature is of **High Priority.**

**4.5.2 Stimulus/Response Sequences**

* Users will first be prompted to Login and after that there will be a Label which will show his Public Credibility which other sellers have given him/her.
* There will be a navigation button to show the credibility of connections as well.

**4.5.3 Functional Requirements**

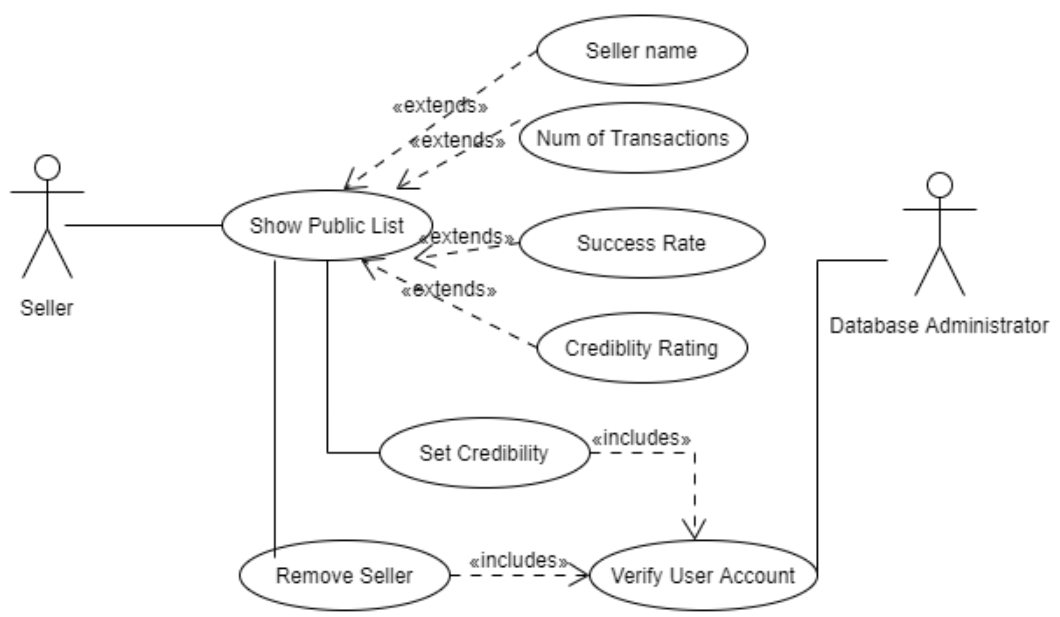
*REQ-SPC1*: After a seller has been added as a connection, users will be prompted to provide a rating for that particular connection. The credibility Rating and the Public Seller List can be viewed by anyone on the network of this user.

*REQ-SPC2*: The direct connections can view your Public seller list and can add any mutual connection if they wish to, these will be added as an **indirect connection** and be assigned the credibility automatically.

*REQ-SPC3*: Public list page will have a button for each seller to be removed.

*REQ-SPC4*: Sellers will also have the option to change the credibility rating for each user.

REQ-SPC5: A unique id that identifies each seller’s connection list will be generated.



| **Use Case Name** | **Sellers Public List** |
| --- | --- |
| Related Requirements | REQ-AD2, REQ SB9 |
| Goal in Context | To manage the transactions that took place |
| Preconditions | User should have an account signed up |
| Successful End Conditions | User will be displayed all the information of his sellers |
| Failed End Conditions | Unable to load the Seller information |
| Primary Actors | Seller |
| Secondary Actors | Database Administrator |
| Triggers | User clicks on show credible sellers list |
| Included Cases | TBD-3 |
| Main Flow | | Step | Action | | --- | --- | | 1 | Users will be shown the button to view their public list. | | 2 | Users will be shown the list and details of all Sellers. | | 3 | User will also be shown button to remove sellers from public credibility list | | 4 | User can also change the credibility for that user for others to see | |
|  |
|  |

## 4.6. Rating System

***4.6.1 Description and Priority***

Rating System is a high priority feature where the sellers, based on experience and exposure with other sellers, give credibility to those sellers on this platform. The system implements an algorithm that assures a fair and smooth rating process. If the seller wishes to add another seller in their public seller list then the system prompts the seller to add their credibility as well.

***4.6.2 Stimulus/Response Sequences***

* User logs into his account. User searches for a particular seller. User wishes to add the seller to his public sellers list. User clicks on the “add as connection” button. System approves the request and prompts a popup box where the user will give a credibility rating to this seller.
* The system will approve this action and will implement its algorithm and accordingly update the rating of the seller on the platform and in the database.
* The rating will be visible to all users.

***4.6.3 Functional Requirements***

*REQ-RF1:* Users should be able to set credibility ratings of sellers when adding them as a connection.

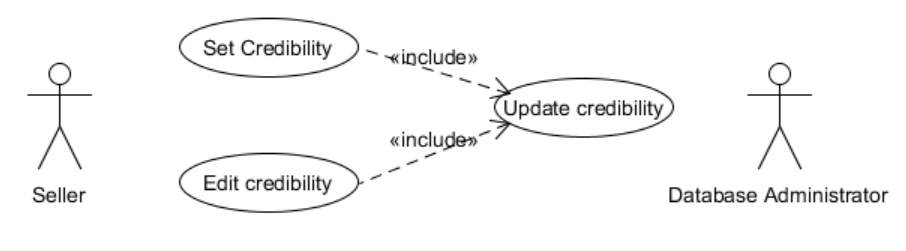
*REQ-RF2*: Users will also have the option to update the credibility ratings of sellers in their public sellers list.

*REQ-RF3:* Users should not be able to rate any seller they haven’t added as a connection.

*REQ-RF4:* The credibility rating of sellers who aren’t immediate family will be deducted by the algorithm accordingly.

*REQ-RF5:* The system shall not allow any sort of randomized rating. The system shall not let sellers exceed the daily limit of rating other sellers. The system should keep track of the ratings and reset the daily limit every day.

*REQ-RF6*: The system will not allow out of bound rating because it offers a slider (0-10).



| **Use case name** | **Giving Credibility Rating** | |
| --- | --- | --- |
| Related Requirements | *REQ-RF1, REQ-RF2, REQ-RF3, REQ-RF4, REQ-RF5, REQ-RF6* | |
| Goal in Context | To give the credibility rating to sellers | |
| Preconditions | Must have added other sellers | |
| Successful End Conditions | Successfully be able to give credibility rating to other sellers show credibility rating of sellers | |
| Failed End Condition | Unable to load the Seller information and give credibility rating | |
| Primary Actors | Seller | |
| Secondary Actors | None | |
| Trigger | User selects add credibility rating option | |
| Included Cases | TBD-4 | |
| Main flow | Step | Action |
|  | 1 | Seller adds someone |
|  | 2 | Seller gives credibility rating according to their own preference |

## 4.7. Sell Product

**4.7.1 Description and Priority**

This feature allows the user to upload and advertise their product in the system. Their product, once uploaded, will be visible to all the users. It is of high priority.

**4.7.2 Stimulus/Response Sequences**

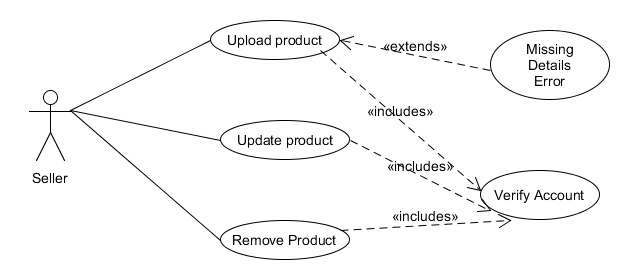
* User creates an account.
* User clicks on 'Sell Product’
* System opens a new page for the user to upload pictures and include details regarding the product
* User uploads a picture.
* System provides fields such as category, description, cost, preferred method of payment, contact details etc.
* User inputs the necessary details
* User clicks on advertise
* The system displays the product ad on the seller's profile.

**4.7.3 Functional Requirements**

REQ SP1: The system should allow users to upload the product for sale, provided the user is logged in and the product is legally available. A sell product option will be available. The user will be able to add details of the product.

REQ SP2: The system should allow the user to update the products' details, provided they are logged in

REQ SP3: The system should allow the user to remove the product, provided they are logged in.

**

| **Use case name** | **Upload Product** | |
| --- | --- | --- |
| Related Requirements | REQ SP1 | |
| Goal In Context | An existing user uploads a product for sale | |
| Preconditions | The user is logged in. | |
| Successful End Condition | The product is successfully uploaded on the site. An error generated based on missing information | |
| Failed End Condition | The product is not uploaded on the site | |
| Primary Actors | Seller | |
| Secondary Actors | None | |
| Trigger | The user clicks the 'sell product' option | |
| Included Cases | **Verify Account** | |
| Main Flow | **Step** | **Action** |
|  | 1 | The user clicks the 'sell product' option |
|  | 2  **include :: Verify Account** | The system checks whether the user is logged in. |
|  | 3 | The system provides interface to provide product details |
|  | 4 | User uploads product picture and adds product details |
|  | 5 | User chooses to advertise product |
|  | 6 | Product is displayed on user's profile and is visible to their friends |

| **Use case name** | **Update Product** | |
| --- | --- | --- |
| Related Requirements | REQ SP2 | |
| Goal In Context | The product is displayed with its updated information | |
| Preconditions | The user is logged in and the product does not violate any copyright laws | |
| Successful End Condition | The product is successfully updated | |
| Failed End Condition | The product is not updated on the site | |
| Primary Actors | Seller | |
| Secondary Actors | None | |
| Trigger | The user requests the system to edit product information (Edit button) | |
| Included Cases | **Verify Account** | |
| Main Flow | **Step** | **Action** |
|  | 1 | The user clicks the edit option |
|  | **2**  **include :: Verify Account** | The system checks whether the user is logged in. |
|  | 3 | The system opens product details |
|  | 4 | User adds new information regarding the product |
|  | 5 | User saves the new information |
|  | 6 | The system updates the new information into the database |
|  | 7 | Product is displayed on the user's profile and inventory along with the updated information |

| **Use case name** | **Remove Product** | |
| --- | --- | --- |
| Related Requirements | REQ SP3 | |
| Goal In Context | The product is removed from the user's inventory | |
| Preconditions | The user is logged in | |
| Successful End Condition | The product is successfully deleted | |
| Failed End Condition | The product is not removed from the site | |
| Primary Actors | Seller | |
| Secondary Actors | None | |
| Trigger | The user requests the system to delete product(s) from the inventory database | |
| Included Cases | **Verify Account** | |
| Main Flow | **Step** | **Action** |
|  | 1 | The user clicks the edit option |
|  | **2**  **include :: Verify Account** | The system checks whether the user is logged in. |
|  | 3 | The system opens product details |
|  | 4 | User adds new information regarding the product |
|  | 5 | User saves the new information |
|  | 6 | The system deletes the product from the database |
|  | 7 | System shows confirmation message for product deletion and the product is removed from the inventory |

## 4.8. Shopping cart & Transaction Management

**4.8.1 Description and Priority**

Shopping cart feature allows the user to store the products they wish to purchase inside a cart and increase or decrease the quantity of the product. Transaction management features include payment process after the user decides to purchase a product(s) from a seller(s). This feature is of low priority.

**4.8.2 Stimulus/Response Sequences**

* The user selects the ‘add to shopping cart’ option on a product profile.
* The system generates a shopping cart and adds the product in the cart. if a cart already exists with at least one product, the system adds the product details in the cart.
* The user changes the quantity.
* The user chooses to buy the products in the cart.
* The system provides the option to choose a method of payment.
* If the user chooses online payment, the system asks for credit card information.
* After choosing the method of payment, the user selects ‘purchase’.
* The system places the order and notifies the seller from whom the buyer is purchasing the product.
* The system generates an e-receipt of the transaction.

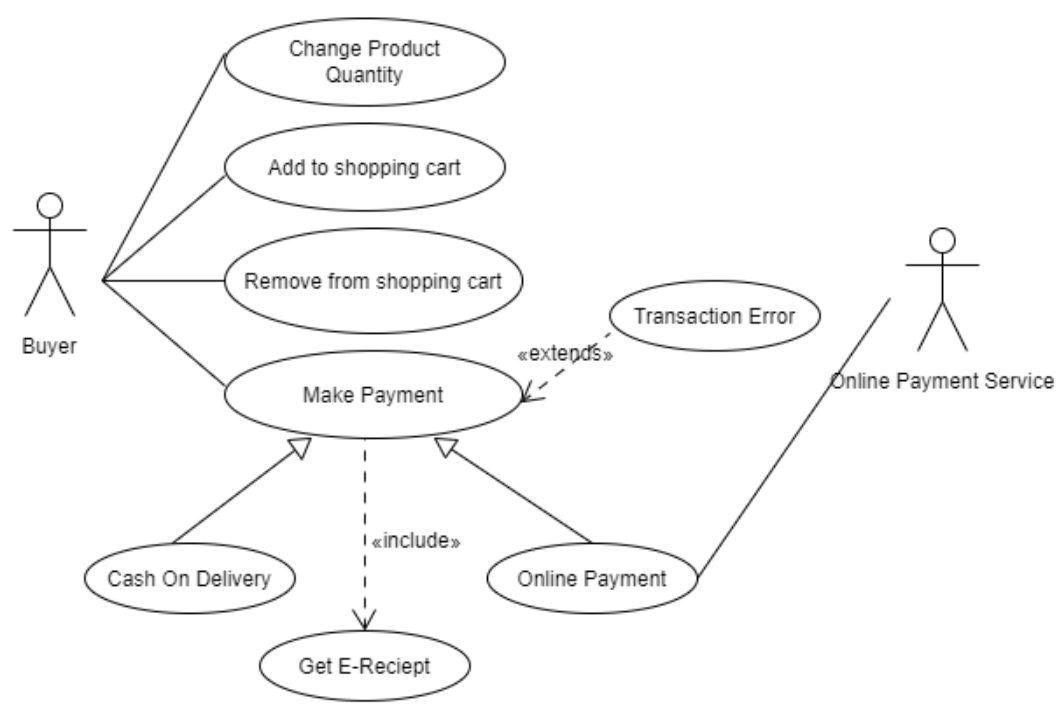
**4.8.3 Functional Requirements**

*REQ-SCT1:* The system should allow the user to add the products they wish to purchase in a shopping cart. An 'add to shopping cart' option would be available next to the product they are interested in. Upon clicking it, a shopping cart will be generated and the product instance will be visible within the shopping card. A user can add more than one product in their shopping cart. If the user goes to add another product, it should be included in the shopping cart previously created and so on.

REQ-SCT2: There would be an option to change quantities of the same product in a shopping cart, provided the user is logged in and there is at least one instance of the product in the cart.

REQ-SCT3: The user will also have the option to remove the product from the shopping cart, provided the user is logged in and there is at least one instance of the product in the shopping cart..

REQ-SCT4*:* The system should allow the user to make payments for the products they purchased. The user should be logged in and there should be at least one product in the shopping cart with the purchase option. There should be an option for buying where the user can add the address of delivery. Furthermore, during checkout, the user receives the option to make online payment or cash on delivery. In case of online payment, the user can fill all the necessary credit card details before placing an order. An error should be generated in case of invalid details or missing address. An e-receipt should be generated within the system and emailed to the user.



| **Use case name** | **Add to Shopping cart** | |
| --- | --- | --- |
| Related Requirements | REQ-SCT1 | |
| Goal In Context | User can add the products they wish to purchase in the shopping cart | |
| Preconditions | The user is logged in and can view the available products | |
| Successful End Condition | Product(s) are successfully added in the cart | |
| Failed End Condition | Product(s) are not added in the cart | |
| Primary Actors | Buyer | |
| Secondary Actors | None | |
| Trigger | The user asks the system to include the product in the shopping cart | |
| Included Cases | None | |
| Main Flow | **Step** | **Action** |
|  | 1 | The user asks the system to include the product in the shopping cart |
|  | 2 | The system creates shopping cart |
|  | 3 | The system adds the product in the shopping cart |
|  | 4 | The product is available in the shopping cart, along with the option to purchase item |

| **Use case name** | **Remove from Shopping cart** | |
| --- | --- | --- |
| Related Requirements | REQ-SCT3 | |
| Goal In Context | User can remove the products they wish to purchase in the shopping cart | |
| Preconditions | The user is logged in and has at least one instance of the product they wish to remove from the shopping cart | |
| Successful End Condition | Product(s) are successfully removed from the cart | |
| Failed End Condition | Product(s) remain in the cart | |
| Primary Actors | Buyer | |
| Secondary Actors | None | |
| Trigger | The user asks the system to remove the product from the shopping cart | |
| Included Cases | None | |
| Main Flow | **Step** | **Action** |
|  | 1 | The system presents the button to remove all instances of a product(s) |
|  | 2 | The user clicks the button |
|  | 3 | The system removes the product from the shopping cart |

| **Use case name** | **Change Product Quantity** | |
| --- | --- | --- |
| Related Requirements | REQ-SCT2 | |
| Goal In Context | User can change the quantity of the product(s) in their shopping cart | |
| Preconditions | The user is logged in and has at least one instance of the product in the shopping cart | |
| Successful End Condition | Product(s) quantity are successfully updated | |
| Failed End Condition | Product(s) quantity does not change | |
| Primary Actors | Buyer | |
| Secondary Actors | None | |
| Trigger | The user asks the system to change the product quantity in the shopping cart (clicking the + or - button) | |
| Included Cases | None | |
| Main Flow | **Step** | **Action** |
|  | 1 | The user asks the system to change the product quantity in the shopping cart (clicking the + or - button) |
|  | 2 | The user increases or decreases the quantity |
|  | 3 | The system displays the numeric quantity selected by the user |

| **Use case name** | **Make Payment** | |
| --- | --- | --- |
| Related Requirements | REQ-SCT4 | |
| Goal In Context | An existing user places order on the items in the shopping cart | |
| Preconditions | The user is logged in and there is at least one product in the shopping cart | |
| Successful End Condition | The payment is made successfully and e-receipt for the transaction is generated | |
| Failed End Condition | The payment is not made successfully. A transaction error is generated, in case the credit card details are wrong or the address is missing | |
| Primary Actors | Buyer | |
| Secondary Actors | Online Payment Service | |
| Trigger | The user asks the system to make payment | |
| Included Cases | **Get E-Receipt** | |
| Main Flow | **Step** | **Action** |
|  | 1 | The user asks the system to make payment |
|  | 2 | The system gives the option of cash on delivery or online payment |
|  | 3  **Online Payment** | The user enters credit card details |
|  | 4 | Product payment is made successfully |
|  | 5  **Include::Get E-receipt** | An e-receipt is generated in the system |
|  | 6 | A copy of e-receipt is emailed to the user |

# 5. Other Nonfunctional Requirements

## 5.1. Performance Requirements

To ensure an efficient and reliable working of our product, it is essential that some performance requirements are fulfilled.

* High throughput rate: System should be able to manage several transactions simultaneously without any bottlenecks.
* Response time: When a user is browsing through the seller’s list or any item category, the results should be delivered immediately and there should be minimum lag. Minimal to no lag should be the aim when it comes to processing any queries. Regardless of the traffic on the system, the response time should always be high. Response time should NOT be greater than 1 second.
* Usability: Each feature of the system should be organized in a manner that is easy for the user to navigate through and is not troublesome when it comes to finding out how to operate the application. Users should easily be able to search for sellers and give credibility scores.
* Reliability: There should be very few errors and in case of a new found error, it should be dealt with immediately and the backlog should be updated with this error and how it was resolved so in case the same error appears, it can be dealt with quickly. Furthermore, errors should be minimized to ensure a smooth experience for the users. Testing should be done every few weeks as part of maintenance to ensure reliability.
* Scalability: As this system is a new product, in case of unexpected load of users, the system should be able to cater to the additional workload. This can be taken care of by testing the scalability. The system should be designed in a way that in future new features can be added and improvements can be made.

## 5.2. Safety Requirements

* In case of an error with the system such as system crash or server issues, the system should not be down for more than 20 minutes.
* For minor inconveniences where some feature of the product is temporarily disabled or has crashed, an error message should be popped up to alert the user.
* The system should not allow users to create multiple accounts simultaneously.
* The system should not allow randomized ratings of sellers. Less than 5 sellers can be rated daily.
* The system should verify with the user before purchase of the product.
* In case of a significant transaction the system should transfer the transaction in an escrow account until verified by both seller and buyer.

## 5.3. Security Requirements

* The system should make sure that the data the user provides, once registering or later when the user edits, is protected and not sold to unknown sites or third parties without user- discretion.
* The transactions and payments a user does are secure and are not prone to any sort of hacking attempts or theft.
* The live chat with the customer/seller is only viewed by the parties in contact and not all users.
* Privacy of a user should be maintained as only those whom a user has added as their friend can view their profiles.
* Only the user and the administrator can view the user’s dealings’ history.
* Authenticity: The products on the system should not violate any copyright rules and there shouldn’t be any kind of trademark infringement.
* The images of the products the seller uploads should not be copyrighted.
* Whenever any product is purchased, the e-receipt is delivered to provide the user with proof.
* Access permissions for particular system information, i.e. transaction details, may only be changed by the system’s administrator.

## 5.6. Software Quality Attributes

For any software to be preferred and be worthwhile it needs to have desirable quality attributes.

Hence, for our system some desirable quality attributes are as follows:

* Usability: This consists of a wide variety of factors such as ease of operability, ease of learning, accessibility, system aesthetics, and user error protection.
* Ease of operability: The system should be easy to use and control for the user. It should enable the user to be able to successfully navigate through the system and get their work done by effectively carrying out dealings with other sellers or making purchases.
* Ease of learning: The system should be designed and developed such that the user can learn how to operate the system very quickly and that it is not a challenging experience for them. Ease of learning and operability is closely interrelated as if the system is easy to learn it will also be easy to use, consequently both have equal priority.
* Accessibility: Our system should not be procured for just those users who are able-bodied but even those with disabilities should be able to make use of the system easily.
* System aesthetics: The software should be pleasing to look for the end user and not be repulsive for them. All the features being offered by the system should be clearly visible and not be hidden.
* User error protection: In case the user is making any error while using the system, an error should be popped up alerting what is wrong with their selection. For example, if a seller repeatedly tries to open a profile of someone they are not friends with, a message should appear telling them they cannot access an individual’s profile if they are not friends with them.
* Adaptability: In case of any new requirements, the system should not just be able to implement those changes but also remain compatible with the standard. Being up to date with the demand and continuous changes in the real world is also very significant.
* Maintainability & Flexibility: The system should be designed and developed in such a way that is easy to maintain and testing is an easy job. The system should also be modifiable. If in the future any component needs to be added, for example, if we decide to expand our product’s availability to other foreign countries or if any unavoidable change arises.
* Portability: The system will be easily incorporated into any Windows-based system and Android smartphone.
* Reliability: The system should be able to handle a huge amount of users at the same time and the credible rating of the sellers should give a correct estimate of the trustworthiness of the seller.

## 5.5. Business Rules

* There will also be the role of a seller who will have their own credibility which other sellers have given them and also, the person can be the one to give rating/credibility to other sellers who are his direct connections.
* In case of an error with the system such as system crash or server issues, the system should not be down for more than 20 minutes.
* For minor inconveniences where some feature of the product is temporarily disabled or has crashed, an error message should be popped up to alert the user.
* The user should not be able to contact the seller outside of business hours, instead, a chatbot will be there to assist the buyer.
* The user should be able to carry out transactions only after he/she has created an account.
* The user should not be able to view sellers with no mutual friends.
* The user should not be able to view sellers with credibility scores below a specified threshold.
* The user should not be able to view sellers outside of their region.

# 6. Other Requirements

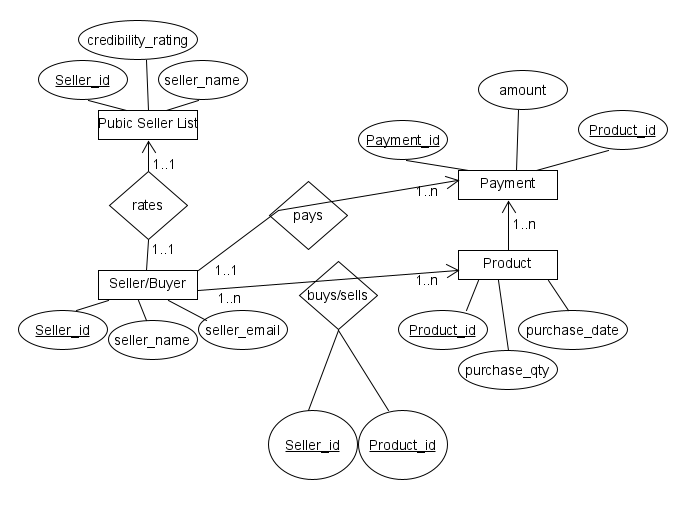
* Free, open-source software.
* The software is available for use within Pakistan and will primarily be available in English.
* Copyright laws and license agreements must be abided by by third parties.
* Under the license agreement, this software cannot be distributed without the official consent of the Institute of Business Administration.
* Any user found indulging in copying products, harassing sellers, catfishing buyers, and selling inauthentic products will be banned.
* The community guidelines must be complied with by all users.

**Appendix A: Glossary**

1. Intermediary B2B Model: The e-commerce business model tends to act as the intermediary, helping connect the businesses with other businesses.
2. Intermediary B2C Model: An e-commerce business model that focuses on the process of selling products directly between business and consumers.
3. Copyright infringement: also known as piracy is the use of works protected by copyright law without approval for a purpose for which consent is required, thereby infringing on the copyright holder's exclusive rights, such as the right to procreate, allocate, showcase, or perform the copyrighted material, as well as the right to create derivative works.
4. in-Credibles - Credibility Sellers List (for sellers who wants to expand their network)
5. SRS - Software Requirement Specification
6. GUI - Graphical User Interface
7. Stakeholder- The person who will participate in the system Ex. Clients, Administrator, Seller, etc.

**Appendix B: Analysis Models**

The following diagram is an ERD depiction.



**Appendix C: To Be Determined List**

1. TBD-1 New Seller Account.
2. TBD-2 Still unsure about the included cases for sellers private list.
3. TBD-3 Still unsure about the included cases for sellers' public list.
4. TBD-4 Still unsure about the included cases for giving credibility rating.
5. TBD-5 Unsure about product listing feature.
6. TBD-6 The database that will be used is not decided up till now. MongoDB and MS Access are preferred.
7. TBD-7 Still unsure about the implementation of live chat.