Software Requirements Specification

for

Fantastic 6 Online Store

**Version 1.1 approved**

**Prepared by Wangmo Tenzing, Fulya Kocaman, Vamshi Katipally, Abram Flores, Anthony Hernandez, Manthan Vasani**

**CPSC 541**

**5//2022**

**Table of Contents**

Contents

[1. Introduction 3](#_Toc102570740)

[1.1 Purpose 3](#_Toc102570741)

[1.2 Document Conventions 3](#_Toc102570742)

[1.3 Project Scope 3](#_Toc102570743)

[1.4 References 4](#_Toc102570744)

[2. Overall Description 4](#_Toc102570745)

[2.1 Product Perspective 4](#_Toc102570746)

[2.2 User Classes and Characteristics 4](#_Toc102570747)

[2.3 Operating Environment 5](#_Toc102570748)

[2.4 Design and Implementation Constraints 6](#_Toc102570749)

[2.5 Assumptions and Dependencies 6](#_Toc102570750)

[3. System Features 6](#_Toc102570751)

[3.1 Search Bar 6](#_Toc102570752)

[3.2 Purchase Item 7](#_Toc102570756)

[3.3 Login to Website 8](#_Toc102570759)

[3.4 Update Inventory 8](#_Toc102570762)

[4. Data Requirements 9](#_Toc102570765)

[4.1 Logical Data Model 9](#_Toc102570766)

[4.2 Data Dictionary 9](#_Toc102570767)

[4.3 Reports 9](#_Toc102570768)

[4.4 Data Acquisition, Integrity, Retention, and Disposal 10](#_Toc102570769)

[5. External Interface Requirements 10](#_Toc102570770)

[5.1 User Interfaces 10](#_Toc102570771)

[5.2 Software Interfaces 10](#_Toc102570772)

[5.3 Hardware Interfaces 11](#_Toc102570773)

[5.4 Communications Interfaces 11](#_Toc102570774)

[6. Quality Attributes 11](#_Toc102570775)

[6.1 Usability 11](#_Toc102570776)

[6.2 Performance 12](#_Toc102570777)

[6.3 Security 12](#_Toc102570778)

[6.4 Safety 12](#_Toc102570779)

[7. Internationalization and Localization Requirements 12](#_Toc102570780)

[8. Other Requirements 13](#_Toc102570781)

[Appendix A: Glossary 13](#_Toc102570781)

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Wangmo Tenzing | 4/11/2022 | Created | 1.0 |
| Fulya Kocaman | 5/4/2022 | Finalized Formatting | 1.1 |

# Introduction

## Purpose

The purpose of this document is to describe the functional and non-functional requirements of the first release of the F6 Online Direct Store, an e-commerce platform. More generally, this document will seek to provide the foundation for the forthcoming planning, design, and coding of the mentioned product. So to that end, the intended audience of this software requirements specification is broad. The development team working on this project -- as well as the testers -- should be aware of the contents of this document since they need to be fully aware of what they are building and what they should be looking out for during tests. Additionally, this software requirements specification (SRS) is intended for business personnel, like project managers and executives, as it will help with developing timelines and allocating resources. Furthermore, this document should be read by ABC’s marketing/branding department so that they can have a better idea of how to promote the platform. In addition, this SRS should be read by F6’s potential customers/vendors so that they can be aware of what the software is capable of; and also so that they can ensure that their expectations are met.

## Document Conventions

References to external writings - like the Vision and Scope document, for example - are to be made using italics. Definitions for terms or concepts defined within the context of this document are also to be denoted with italics, while references to sections contained within this document itself are to be capitalized and bolded. Additionally, the written sections within this SRS are to exclusively use 12-point Times New Roman font, while section headers - like **Document Conventions** - are to use bolded 14-point Times New Roman font. Size and font exceptions are to be made for any text contained within diagrams and models, though they should generally use Inconsolata font. Headings and subheadings should have numerical values, and they should use points to indicate subsections - for instance, 1.2.1 would be considered a subsection of 1.2, which is a subsection of 1.

## Project Scope

The F6 Online Direct Store is an e-commerce platform that is to host an expansive product selection from a large and varied quantity of sectors. Development of this software is centered around the business strategy of expanding reach and communication with existing and potential customers, as well as expanding the company’s overall catalog and income by opening up the possibility for 3rd-party vendors to sell on the platform (*refer to Section 1.2, Business Opportunity, of Vision & Scope Document).* Business objectives that ABC wishes to fulfill with the development of this software include: increasing gross profit (BO-1), increasing customer lifetime value (BO-3), and decreasing overhead ratio (BO-4), among others (*refer to section 1.3, Business Objectives, of Vision & Scope Document*). The major features that this release is to contain revolve around providing a simple and fluid shopping experience for the customers of ABC. For instance, users should be able to navigate through the company’s full product catalog by applying filters and by using search bars; additionally, when they click on any item, they should be taken to a separate page containing relevant information, like the quantity available, descriptions, shipping estimates, and other information along that vein. Other major features include the ability to register an account/login and also the ability to add items to a shopping cart from where users will have the ability to pay with numerous options.

## References

1. Wiegers, Karl and Betty, Joy. *Software Requirements, 3rd edition*. Microsoft Press, 2013.
2. Fakhroutdinov, Kirill. *UML Use Case Diagrams*. <https://www.uml-diagrams.org/use-case-diagrams.html>
3. Artifacts, Project 1: [Artificats - Backlogs, Tasks, Burndown Charts](https://docs.google.com/spreadsheets/d/15pVvwJdnghe83GMD1LpjSaShyV-cg6kwTxXcvufEfYc/edit?usp=sharing)
4. Vision and Scope Document, Project 1: [Vision\_and\_Scope](https://docs.google.com/document/d/1KqIzZReD2fURvFW0wOP2_wEjLioUEE1pxabgnh7zwYo/edit?usp=sharing)

# Overall Description

## Product Perspective

The F6 Online Direct Store is an entirely new product that has no prior precedent for ABC Inc apart from its basis in the company’s physical, in-person locations. It will potentially be the first increment of a 3-part release, where the first release will encompass the most significant operations of the software, like the ability for customers to login, browse, and purchase items, while the second phase of the release will revolve around allowing 3rd-party vendors to sell on the platform; additionally, the second phase will look to address any features that have been overlooked throughout the development cycle up to that point. And, finally, the third release will seek to create a mobile application for the platform.

https://www.edrawmax.com/context-diagram/

## User Classes and Characteristics

|  |  |
| --- | --- |
| *User Class* | *Characteristics* |
| *1. Customer (favored class)* | *CUS-01: Selects items for placement in a virtual shopping cart.*  *CUS-02: Executes purchases by selecting from a variety of payment options.*  *CUS-03: Browses and filters through product catalog.*  *CUS-04: Receives and applies discounts to purchases.*  *CUS-05: Creates login account for ease and convenience of shopping experience.*  *CUS-06: Tracks the status of orders* |
| *2. Developer* | *DEV-01: Modifies the layout/content of the platform*  *DEV-02: Renders items to be available for display and purchase.*  *DEV-03: Fixes bugs and ensures server stability.*  *DEV-04: Manages databases and the functionality of the various components of the platform, like the customer service system* |
| *3. Vendor* | *VEN-01: Checks sales metrics and analytics.*  *VEN-02: Requests items to be available for sale on ABC’s platform.*  *VEN-03: Creates and maintains a public profile account.*  *VEN-04: Helps update/maintain the status of sold orders.* |
| *4. Admin* | *AD-01: Directs the content and ads on the F6 platform.*  *AD-02: Approves and selects product requests from vendors.*  *AD-03: Manages and provides oversight of Vendor accounts.* |

## Operating Environment

Since the software is web-based, it is expected to operate well on most common operating systems, like Windows, MacOS, and Linux. Also, to the extent that the user has internet connectivity, he or she should be able to access the F6 Online Direct Store through any of the common hardware platforms of their choice, like Apple, Macintosh, Dell, etcetera. However, it should be noted that because this release is intended for desktop browsing, the best user experience will be realized with either a desktop or a laptop.

The predicted geographical location of the first users of the software is North America, specifically the United States. ABC has exclusively catered to the American Market thus far because of inadequate resources and uncertainty about the market abroad. However, with the development of this product, it will become easier and more cost-effective to reach new countries than it was previously. So, to that end, we anticipate that the geographical locations of users will widen over time, but they will always be predominantly American and concentrated in larger urban centers.

The locations of servers and databases are anticipated to be exclusively in the United States for the time being and are to be hosted on Microsoft’s Azure cloud platform.

## Design and Implementation Constraints

CO-01: Time and cost pose important constraints on the development of this product since the developers will have to work within the bounds of the duration and budget allocated by ABC.

CO-02: Since the Online Direct Store will be web-based, development is to be done predominantly with JavaScript and JavaScript-related frameworks due to the language’s widespread use in this domain.

CO-03: Since the company will be dealing with predominantly transactional data, the database type should be relational; PostgreSQL, MySQL, and Oracle are prime options to fulfill this need.

CO-04: Authentication for users should be handled through 3rd-party software, namely OAuth 2.0, which will allow users flexibility in that they will have the option to log in through an existing account on other services, like Google or Facebook.

CO-05: Additionally, the design and implementation of the Direct Store will partially be bound by state and national business laws concerning e-commerce. And since the intention is to eventually bring this product to a global audience, implementation will additionally begin considering international legislation that concerns this area.

## Assumptions and Dependencies

*AS-1:* One of the key assumptions made about the F6 Online Direct Store is that the majority of the product’s users will come from the United States, at least during the project’s first iteration. This assumption is critical because predictions about the store’s potential audience are what will shape the initial design and architecture of ABC’s e-platform.

AS-2: A second assumption made about the Direct Store is that Users will understand that the optimal medium for viewing and overall experience will be either a desktop, laptop, or other monitor of sufficient dimensions (approximately greater than or equal to 9 x 9). As such, it is expected that they will not attempt to make purchases through mobile devices if other options are available.

DEP-1: This product will rely on 3rd-party platforms to provide goods to fill its catalogs and listings, of which users can then browse, purchase, and enjoy.

DEP-2: Additionally, another critical dependency of this product is that it will rely on 3rd-party platforms to deliver its goods, like USPS, FedEx, and UPS, among others.

# System Features

## Search Bar

### Description

This feature allows users to search for any product they may want. It is pertinent that they have options. Some users may just want to browse products, but for those looking for a specific item, this feature comes in handy. It is a high priority. It also adds organization to the store. The developers on the backend will categorize the products into sections. Clothing goes under one category; there will be categories for shoes, accessories, stationery items, and food. Users will find what they want by either typing it in the search bar or searching it by category.

### Stimulus/Response Sequences

Putting a product name in the search bar and pressing enter will bring up all the products in that category.

Clicking on the item results from the search bar will bring up the information related to that product.

### Functional Requirements

System should be able to perform searches for different products when a user searches them.

System should be able to use the keywords user types in the search bar and display results that are valid and relevant to the search.

System should filter results and display exact matches of the search bar at the top of the page.

System should refresh the page if the search feature is frozen or crashes.

## Purchase Item

**Description**

Users can purchase any item, given they have the means to pay for it. This feature is a high priority since this is the key way the company will make money. Within this feature, there will be many subfeatures like choosing various payment methods, shipping options, and tracking information as well. This feature also opens up the option for returns/exchanges. “Purchase item” is akin to an umbrella feature, allowing customers access to many other features that may be very useful to their shopping experience. Similar to grouping products into categories, the features can also be grouped in similar categories as well.

### Stimulus/Response Sequences

Hitting the “Checkout Now” button will take the user to the checkout page, where they can enter their information and purchase the item.

Hitting the “Checkout Now” button will prompt users to choose the payment method they want to use to complete the purchase.

### Functional Requirements

System should accept multiple methods of payment and prompt users to proceed with the option they choose.

System should send an invoice to the user once their checkout process is complete.

System should allow users to request a refund/return for the items they bought.

## Login to Website

**Description**

Logging into the website is an important feature that allows users to create an account where all of their transaction histories will remain. They can also have things stored in their carts, but the company wants to encourage users to fulfill their checkout to avoid the cart abandonment issue that plagues e-commerce sites. All users will need an account to complete their purchases. This also is one of the primary ways the company will track user memberships. This is another high-priority feature.

### Stimulus/Response Sequences

Hitting the “Login” button will take the user to the login page where they can sign in to their accounts.

Hitting the “Create Account” button will take the user to the account page, where they can sign up and become a member of the store.

### Functional Requirements

System should notify the user that they’ve logged into the website with a banner message.

System should allow users to login into their account using their email and password.

System should allow new customers to sign up and make an account using their email.

## Update Inventory

**Description**

One of the goals of the company is to frequently update inventory and keep all products well-stocked. In order to do this, there will be an alert system that notifies the management team once a certain item has reached a certain number of stock. The management team will then get in contact with suppliers and place an order to bring the item back. This will allow the store to have almost all products available and ready to be purchased. This feature shall receive regular maintenance and upkeep. This is a medium-priority feature.

### Stimulus/Response Sequences

Once a certain product reaches a stock of 20, the management team will be notified to reorder the product.

Hitting the “Contact Supplier” button will take admin to the supplier’s contact information and their rates as well as delivery schedule.

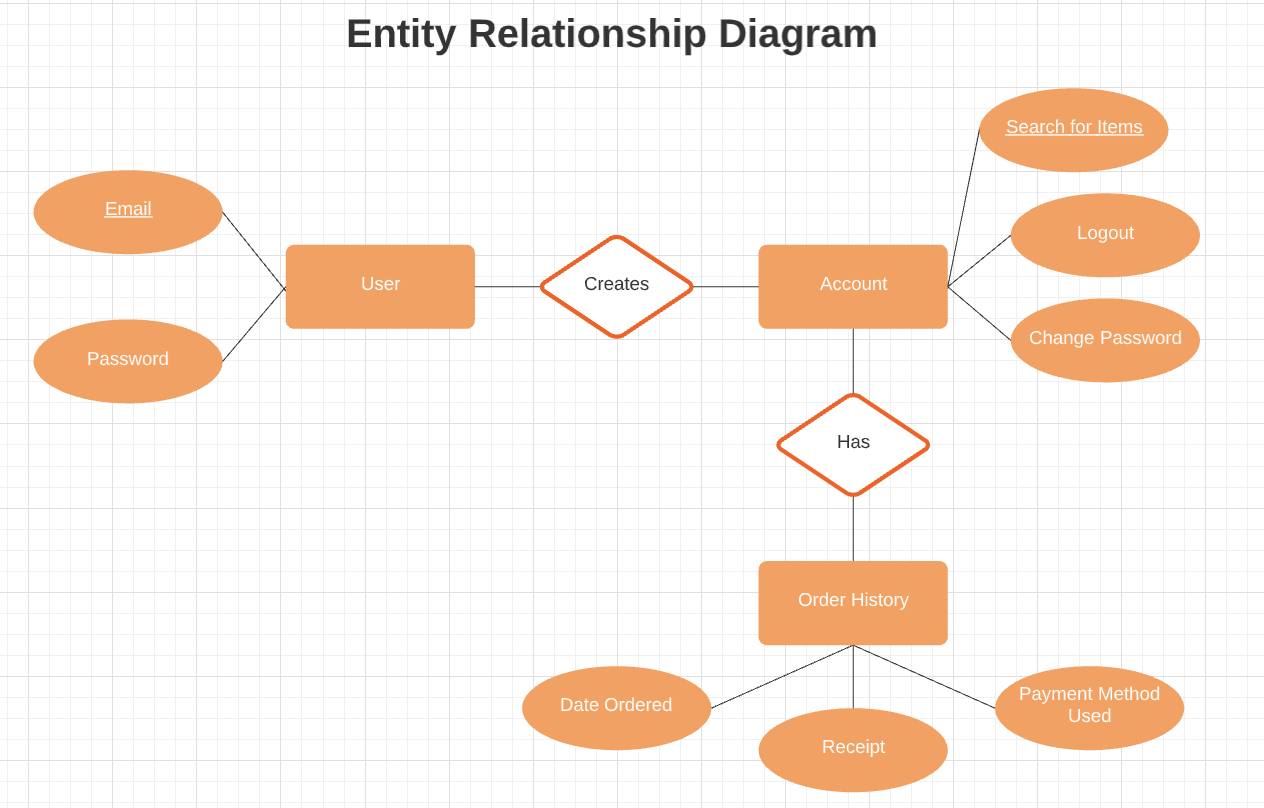
### Functional Requirements

System should allow backend developers to download user activity logs.

System should allow developers to verify shipment/inventory invoices using a passcode.

# Data Requirements

## Logical Data Model



## Data Dictionary

The data dictionary will be included with the artifacts.

## Reports

There will be a couple of reports generated. One will be the user activity log report which will log all activity performed by the user. This includes seeing what they search for so that they can receive relevant ads, and it will also record the date and times they log in and do some shopping. All this information is gathered to learn more about the user and provide them with the best catered shopping experience. And if it has been a couple of weeks since a user has logged in, then the site will send them a coupon to invite them back to the site in hopes of an additional sale.

Another report that will be generated is how long and when the site is down. This lets developers observe the average time the site is down and the reasons why. It helps to prevent similar shutdowns from happening, thereby improving the overall performance of the site. It will also be helpful in the future to keep track of past system failures.

## Data Acquisition, Integrity, Retention, and Disposal

All data shall be stored locally, as well as in persistence storage. If main servers do happen to go down, there will be external servers (backups) that contain all the same information. This allows minimal shutdown of the site itself. There will be several methods used to acquire data. Examples include collecting new data, utilizing legacy data, and sharing/exchanging data.

New data will come from customers, as their activities will be logged and used for future references. Legacy data will be collated through automated collection from other sources. The data received from these sources will be thoroughly reviewed to meet standards as they will need to be current and applicable.

In case the site does go down, it will be prepared because of some application checkpoints that will be in place. This will allow the current state of the site to revert back to a state when it was working, hence keeping the site failure time low.

# External Interface Requirements

## User Interfaces

UI -1: The user interface for this product will share similarities to other online shopping stores so as to not confuse users who have seen other online markets. Our research has shown that by changing the storefront website from what users are used to, less experienced users may become scared or confused as to where to look and may give up.

UI-2: We plan to mimic other online storefront elements, such as the main tab based menu at the top of the storefront. This will include sections for recommended products, new to our inventory, “about to leave,” etc. The menu can lead users to discover new products at a glance. Below that, there will be ads for popular and trending products so that users who purchase these items regularly will not have to look far for what they need.

UI-3: This application will include traditional online store amenities such as descriptive product pages with reviews, a search bar and results page, a shopping cart/wishlist, and an account to save information on the customer as well as track their purchase history.

## Software Interfaces

SI-1: This application will interact continuously with our product inventory database. This database will work concurrently with supplier information to keep the application updated with new products and information on stock availability so that an item can be listed if new inventory has arrived or delisted if inventory has run out.

SI-2: At regular intervals throughout the day (every 2 hours), the application will send a request to the database to check for any changes to inventory. The system, in return, will receive any changes in inventory status.

SI-3: Included within the status update, can be updates to product information that can be displayed on a products page. This information can include specification, sales, stock, price, and images of the product to be displayed. Outside of information included with the update, suppliers registered with the database can update with special notes that they may want to include.

## Hardware Interfaces

This application requires no hardware interfaces on the users' side aside from access to the web through a supported device with a supported browser. The application itself and its databases will be hosted on paid servers, but physical interactions with these servers are not required for this application.

## Communications Interfaces

CI-1: This system will email users who provide a valid email address with invoices generated by our system upon successful purchase of inventory.

CI-2: This email will contain a simple confirmation text of what they have purchased, the price, and contact information to customer service.

CI-3: Attached to the email will be a PDF with a more detailed invoice with itemized details of the purchase.

CI-4: We will use a secure encrypted server to handle in-house payments.

CI-5: We will use PayPal plugins if users wish to use Paypal as a valid payment option.

# Quality Attributes

## Usability

Throughout our storefront UI, we have made it accommodating to as many users as we can be based on focused feedback for different user groups. By incorporating traditional online storefront elements, we have laid out a familiar user experience for the majority of users.

USE-1: For users who are less experienced, we have features such as popular product ads and a search bar to help guide them to products that they need or to help them discover products they want.

USE-2: We also have our customer support number on every website page, so no matter where a user is in our application, they will be able to find the number and talk to someone who can help.

USE-3: We will offer tutorials on how to navigate pages and make purchases through our customer support page

## Performance

PER-1: This system will process payments and send confirmation of purchase within 60 seconds of submission

PER-2: This application will send main page data within 1 second of user request.

PER-3: This system will experience no more than 4 days of downtime per year

PER-4: This system will store account information of at least 70% of average unique page visits per quarter

## Security

SEC-1: This application uses secure servers to track user sessions so scraping data off our website is difficult for non-approved users.

SEC-2: We do not track user data in any way that connects users to their data without their consent. We track global trends, such as how many times a product page has been visited to help with internal decisions, but no data will be linked to users.

SEC-3: Users who create a profile with us will have their information encrypted on verified and secure servers, and we will restrict employees who have access to this data.

SEC-4: We use only secure payment methods such as PayPal and secure transactions in-house.

## Safety

SAF-1: This application will not use flashing imagery that may trigger epilepsy.

SAF-2: For the personal safety of our users, we will secure all personal user data in relation to their accounts.

# Internationalization and Localization Requirements

LOC-1: This application will launch in Canada, the United States of America, and Mexico.

LOC-2: In order to be accessible to a wide range of users, our application will support American English & Spanish as the primary languages during the launch window.

LOC-3: This application will help partner with various local companies to train our system with the local dialects for users who use speech-text features on our website.

LOC-4: This application will bridge the gap between local businesses and customers by managing order delivery into remote areas of the country.

LOC-5: This application will have an option to contact a local liaison to improve communication between customers and the company.

# Other Requirements

OTR-1: ABC Inc will apply for Business Operation License, Seller’s permit, and Sales tax permits in order to incorporate this application as one of its offerings.

OTR-2: This application is built following the regulatory guidelines of the international trading commission and local government rules.

OTR-3: This application will provide links to public legal documents in the “About us” section, where users can fact-check ABC Inc’s offerings.

OTR-4: This application will support all devices that are capable of connecting to the internet and has a browser to render the website.

**Appendix A: Glossary**

**Activity Diagram:** An analysis model that depicts a process flow proceeding from one activity to another. Similar to a flowchart.

**Actor:** A person performing a specific role, a software system, or a hardware device that interacts with a system to achieve a useful goal. Also called a user role.

**Alternative Flow:** A path through a use case that leads to success but that involves a variation from the normal flow in the specifics of the task or in the actor’s interaction with the system.

**Analysis, Requirements:** The process of classifying requirements information into various categories, evaluating requirements for desirable qualities, representing requirements in different forms, deriving detailed requirements from high-level requirements, negotiating priorities, and related activities.

**Architecture:** The structure of a system, including any software, hardware, and human components that make up the system, the interfaces and relationships between those components, and the component behaviors that are visible to other components.

**Backlog, Product:** On an agile project, the prioritized list of work remaining for the project. A backlog can contain user stories, business processes, change requests, infrastructure development, and defect stories. Work items from the backlog are allocated to upcoming iterations based on their priority.

**Business Objective:** A financial or nonfinancial business benefit that an organization expects to receive as a result of a project or some other initiative.

**Business Objectives Model:** A visual representation of a hierarchy of business problems and business objectives.

**Business Requirements:** A set of information that describes a business need that leads to one or more projects to deliver a solution and the desired ultimate business outcomes. The business requirements include business opportunities, business objectives, success metrics, a vision statement, and scope and limitations.

**Business Rule:** A policy, guideline, standard, regulation, or computational formula that defines or constrains some aspect of the business.

**Class Diagram:** An analysis model that shows a set of system or problem domain classes, their interfaces, and their relationships.

**Constraint:** A restriction that is imposed on the choices available to the developer for the design and construction of a product. Other types of constraints can restrict the options available to project managers. Business rules often impose constraints on business operations and hence on software systems.

**Context Diagram:** An analysis model that depicts a system at a high level of abstraction. The context diagram identifies objects outside the system that exchange data with the system, but it shows nothing about the system’s internal structure or behavior.

**Customer:** An individual or organization that derives either direct or indirect benefit from a product. Software customers might request, pay for, select, specify, use, or receive the output generated by a software product.

**Data Dictionary:** A collection of definitions for the data elements and data structures that are relevant to the problem domain.

**Data Flow Diagram:** An analysis model that depicts the processes, data stores, external entities, and flows among them that characterize the behavior of data flowing through business processes or software systems.

**Entity Relationship Diagram:** A graphical relationship illustrating relationships among people, objects, and concepts within a system.

**Exception:** A condition that can prevent a use case from concluding successfully. Unless some recovery mechanism is possible, the use case’s post-conditions are not reached, and the actor’s goal is not achieved.

**External Interface Requirement:** A description of a connection between a software system and a user, another software system, or a hardware device.

**Flowchart:** An analysis model that shows the processing steps and decision points in the logic of a process. Similar to an activity diagram.

**Functional Requirement:** A description of a behavior that a software system will exhibit under specific conditions.

**Nonfunctional Requirement:** A description of a property or characteristic that a system must exhibit or a constraint that it must respect.

**Normal Flow:** The default sequence of steps in a use case, which leads to satisfying the use case’s postconditions and letting the user achieve his goal. Also known as the normal course, main course, basic flow, normal sequence, main success scenario, and happy path.

**Postcondition:** A condition that describes the state of a system after a use case is successfully completed.

**Precondition:** A condition that must be satisfied or a state the system must be in before a use case can begin.

**Prioritization:** The act of determining which requirements for a software product are the most important for achieving business success and the sequence in which requirements should be implemented.

**Procedure:** A step-by-step description of a course of action to be taken to perform a specified activity, describing how the activity is to be accomplished.

**Process:** A sequence of activities performed for a particular purpose. A process description is a documented definition of those activities.

**Process Flow:** The sequential steps of a business process or the operations of a proposed software system. Often represented by using an activity diagram, flowchart, swimlane diagram, or other modeling notation.

**Product:** Whatever ultimate deliverable a project is developing; application, system, or solution.

**Quality Requirement:** A non-functional requirement that describes a service or performance characteristic of a product. Types of quality attributes include usability, portability, maintainability, integrity, efficiency, reliability, and robustness. Quality attribute requirements describe the extent to which a software product must demonstrate desired characteristics.

**Requirement:** A statement of a customer's need or objective, or of a condition or capability that a product must possess to satisfy such a need or objective. A property that a product must have to provide value to a stakeholder.

**Software Requirements Specification (SRS):** A collection of the functional and non-functional requirements for a software product.

**System:** A product that contains multiple software and/or hardware subsystems.

**System Requirement:** A high-level requirement for a product that contains multiple subsystems, which could be all software or software and hardware.

**Unified Modeling Language (UML):** Describes a set of standard notations for creating various visual models of systems, particularly for object-oriented software development.

**Use Case:** A description of a set of logically related possible interactions between an actor and a system that results in an outcome that provides value to the actor. It can encompass multiple scenarios.

**Use Case Diagram**: An analysis model that identifies the actors who can interact with a system to accomplish valuable goals and the various use cases that each actor might be involved with.

**User:** A customer who will interact with a system either directly or indirectly (for example, by using outputs from the system but not generating those outputs personally). Also called end-user.

**User Class**: A group of users for a system who have similar characteristics and requirements for the system. Members of a user class function as actors when interacting with the system through use cases.

**User Requirement:** A goal or task that specific classes of users must be able to perform with a system or the desired product attribute. Use cases, user stories, and scenarios are common ways to represent user requirements.

**User Story:** A format to capture user requirements on agile projects in the form of one or two sentences that articulate a user need or describe a unit of desired functionality, as well as stating the benefit of the functionality to the user.

**Vision:** A statement that describes the strategic concept or the ultimate purpose and form of a new system.

**Vision and Scope Document:** A collection of the business requirements for a new system, including business objectives, success metrics, a product vision statement, and a project scope description.