

# **Software Requirements Specification**

**for**

## **VideoCo Management System**

**Oct. 1st, 2021**

<b>1. Introduction</b>	<b>4</b>
<b>1.1 Purpose</b>	<b>4</b>
<b>1.2 Document Conventions</b>	<b>4</b>
<b>1.3 Intended Audience and Reading Suggestions</b>	<b>4</b>
<b>1.3.1 Stakeholders</b>	<b>5</b>
Internal Stakeholders	5
External Stakeholders	5
<b>1.4 Product Scope</b>	<b>5</b>
<b>2. Overall Description</b>	<b>6</b>
<b>2.1 Product Perspective</b>	<b>6</b>
<b>2.2 Product Functions</b>	<b>6</b>
<b>2.3 Operating Environment</b>	<b>6</b>
<b>2.4 Design and Implementation Constraints</b>	<b>7</b>
<b>2.5 User Documentation</b>	<b>7</b>
<b>3. External Interface Requirements</b>	<b>8</b>
<b>3.1 User Interfaces</b>	<b>8</b>
<b>3.2 Hardware Interfaces</b>	<b>8</b>
<b>3.3 Software Interfaces</b>	<b>8</b>
<b>3.4 Communications Interfaces</b>	<b>9</b>
<b>4. System Features</b>	<b>10</b>
4.0.1 Functional Requirements	10
<b>4.1 System Login/Register</b>	<b>11</b>
<b>4.2 Browse Movie by Name, Genre, or All Movies and View Movie Description</b>	<b>12</b>
<b>4.3 Place an Order to Rent Movies Online from the System</b>	<b>13</b>
<b>4.4 Place an Order to Rent Movies via Operator or in Store</b>	<b>14</b>
<b>4.5 Cancel Orders that Have Not Been Shipped</b>	<b>16</b>
<b>4.6 Manage User Account</b>	<b>17</b>
<b>4.7 System Admin Add/Remove Movies and Update Existing Movie Information</b>	<b>17</b>
<b>4.8 System Admin View/Edit Customer Profiles and Orders</b>	<b>19</b>
<b>4.9 System Add/Delete Admin Accounts</b>	<b>20</b>
<b>4.10 System Processes User Account Authentication and Orders</b>	<b>20</b>
<b>4.11 System Assembles List of Movies Ordered and Dispatches List to Warehouses</b>	<b>21</b>
<b>5. Other Non-functional Requirements</b>	<b>22</b>
<b>5.1 Performance Requirements</b>	<b>22</b>

<b>5.3</b>	<b>Security Requirements</b>	23
<b>5.4</b>	<b>Software Quality Attributes</b>	23
	<b>Availability</b>	24
	<b>Maintainability</b>	24
	<b>Reliability</b>	24
	<b>Usability</b>	24
	<b>Error Handling</b>	24
	<b>Ease of Use</b>	24
	<b>Modifiability</b>	24
	<b>Flexibility</b>	25
	<b>Verifiability</b>	25
<b>5.5</b>	<b>Business Rules</b>	25
<b>6.</b>	<b>Other Requirements</b>	25
	Appendix A: Glossary	26

## Revision History

Date	Reason for Changes	Version
09-20-2021	Initial draft	1.0
09-30-2021	Proofread.	1.1

# 1. Introduction

## 1.1 Purpose

The primary focus of this documentation is to provide a detailed and ***complete list of functional, non-functional, and software requirements needed to implement the VideoCo Management System***. This document describes the VideoCo Management System. The VideoCo Management System is tasked with providing customers a means to rent movies from the VideoCo movie catalog through online, phone dial-in, and in-person services. Movies can be rented out in-person in the physical VideoCo stores in Toronto, Canada. The documentation is intended for, but not limited to, project leads, development teams, stakeholders, and end users.

## 1.2 Document Conventions

### Typographical Conventions

Italics are used to indicate priority of requirements and features throughout this document. Bold text is used to indicate important text throughout the document. Bold requirements named “REQ-X” where x is a number, refer to requirements that are listed in section 4.0.1.

### Priorities

Each requirement within the document consists of a description, and a priority. The description specifies a function or characteristic the software must exhibit and must have a single interpretation. The priorities specify the importance of the requirement and are defined as follows:

**High** = essential

**Medium** = important

**Low** = desirable, but not critical

Priority High items must be designed and built into the software. Priority Medium items should be present, though there may have to be sacrifices in performance or availability may be delayed. The software should fulfil all High Priority and 90% or more of Medium Priority requirements. Low Priority items should be considered for upgrades or development.

## 1.3 Intended Audience and Reading Suggestions

The following sections will cover “Overall Description”, “External Interface Requirement”, “System Features”, “Other Non-Functional Requirements”, and “Other Requirements”. The SRS can be read in chronological order following Section 2 “Overall Description”.

For role specific reading sequences refer to the roles below:

Role – Software Development Team the SRS should be read in chronological order with a focus on sections 3.3, 4 and 5.

Role – Hardware Development Team the SRS should be read in chronological order with a focus on sections 3.1, 3.2, 4 and 5.

Role – Stakeholders the SRS should be read in chronological order with a focus on sections 2 and 5.

Role – End Users the SRS should be read in chronological order with a focus on sections 2.3, 4 and 5.

### **1.3.1 Stakeholders**

#### **Internal Stakeholders**

Internal stakeholders for the system include any employees who are to use the system to be, as they will be interacting with the developed VideoCo Management System on a frequent basis. This includes any employees who are interacting with the system directly. Cashiers are internal stakeholders as they will be accessing the system to update movie stock and complete in person customer orders. The Inventory operators are also internal stakeholders as they will be interacting with the system similar to a customer, fulfilling orders for dial in customers. The warehouse team will also be using the system to get the list of orders to be sent out, as well as updating the system accordingly when orders are sent out.

#### **External Stakeholders**

The most direct external stakeholders are the primary users of the system: Movie consumers. Movie consumers require a method to be able to enjoy movie experiences. Our solution delivers a way to allow movie consumers to be able to enjoy movie experiences from the comfort of their home. As movie consumers use our system, they will be able to order movies to be shipped to them without needing to go to the store. Methods for ordering movies include: ordering online, through the phone, and in-person. For movie consumers to be able to select the movies they choose to order, they must browse the collection of movies available on the online system or ask the operator for the available movie. This will provide movie consumers with the ability to order and have movies shipped to them without the need to go to the store.

Movie suppliers are also direct external stakeholders. Movie suppliers include any business that supplies movie rentals for the stock of the movies to the warehouse team. They will also provide licenses and other information needed for VideoCo Management System to allow customers to rent the movies supplied.

## **1.4 Product Scope**

As online ordering is becoming a standard, VideoCo has decided to create a seamless online and phone ordering system that will allow users to rent movies from the VideoCo catalog. VideoCo Management System will increase both user convenience and accessibility. The system will be implemented to allow users to register for an account, rent a movie via VideoCo's website or phone operators, have the movie shipped to a desired address as well

as allow customers to collect loyalty points. Customers can return rented movies any time before their due date, via drop off or mail, before a per day late fee is applied to their account.

## **2. Overall Description**

### **2.1 Product Perspective**

VideoCo is expanding its business and wants to start renting videos worldwide through the internet. VideoCo wants to develop a new, self-contained computer system. This computer system will interface with the existing and expanding teams of operators and for video shipping. This new system must also interface with the new warehouses that VideoCo is planning to build around the globe. The new computer system will not replace the existing VideoCo video located in Toronto. Instead, these existing stores will coexist and allow customers living nearby to rent and return videos in person.

### **2.2 Product Functions**

The system must allow customers to register accounts. The system must allow registered customers to rent videos. The system must facilitate shipping of rented videos to customers. videos. The system must facilitate returning of videos by customers. The system must automatically calculate and charge customers late fees for unreturned videos based on their location. The system must allow customers to rent and return videos in person if the video they want is available at a desired video store location. The system must support payment through credit card and through loyalty points. The system must support a loyalty program that allows customers to earn loyalty points. The system must track the quantity of each video in stock at each location. The system must track the status of videos. The system must generate a list of videos pending shipping at a preconfigured interval.

### **2.3 Operating Environment**

The VideoCo system must support macOS, Linux, and Windows. More specifically, the VideoCo system must be tested on OS X Yosemite; 32-bit versions of Windows 7, 8.0, 8.1, and 10; 64-bit versions of Windows 7, 8.0, 8.1, and 10; Ubuntu Desktop 14.04, Debian 7, Red Hat Enterprise Linux 7, CentOS 7, and Fedora 23. On Windows systems the Microsoft .NET Framework 4.5.2 is required.

The VideoCo system must support recent versions of common modern browsers including Internet Explorer 11 or higher, Microsoft Edge 79 or higher, Firefox 78 or higher, Chrome 83 or higher, Safari 13 or higher. The VideoCo system must support recent versions of common modern mobile browsers including Chrome for Android 80 or higher, Firefox for Android 75 or higher, iOS Safari 13 or higher, Chrome for iOS 80 or higher, Firefox for iOS 20 or higher.

The system must peacefully coexist with common applications and services present on the above listed operating systems.

## 2.4 Design and Implementation Constraints

The VideoCo system front-end must run on a CPU with a clock speed of 1.6 GHz or faster and at least 1 GB of RAM. The backend server and database subsystems of the VideoCo system must run on commonly available cloud hardware.

The system architecture must provide a number of quality attributes. The system must be scalable so that it can handle thousands of customers simultaneously. The system design must facilitate modifiability and must allow data and schema evolution such that it is possible to add, remove, and modify data attributes without the costs of migrating the database. The system must be recoverable within an hour of database congestion. The system must be easy-to-learn for new users who can learn the user interface as they use it and can refer to documentation for more advanced usage and help if necessary.

The system must obey common design principles and patterns that reduce complexity and coupling and allow for system extension. The system should assign each component with a single responsibility so that changes to one part of the system do not affect other parts. The system components should be open for extension but closed for modification to ensure the system is not brittle but can support change. The system must segregate interfaces to avoid overlapping concerns, to keep the system decoupled and to ease refactoring. The system should depend upon abstractions instead of concretions so that it remains general and easy to extend.

## 2.5 User Documentation

User documentation for the VideoCo system must be available online in the form of a webpage. The user documentation will be packaged with the software such that it is available through the user-facing end of the system. Additionally, the user documentation must be available on hard-copy for users that request it.

The user documentation must include a user manual that helps users to become familiar with the usage of the VideoCo system. The user manual must contain associated images including screenshots of the human-machine interface. The language of the use manual must have minimal jargon and must be explained thoroughly.

There must also be on-line help in the form of an FAQ provided. The FAQ must provide answers to common questions that help users to accomplish common tasks and to overcome common issues.

User documentation must follow the recommendations given in the IEEE/ISO/IEC P26514 document entitled Systems and Software Engineering--Design and development of information for users.

## 3. External Interface Requirements

### 3.1 User Interfaces

The user interface must use a consistent, familiar, and professional style such as the Material Design guidelines by Google (<https://material.io/design/guidelines-overview/>). The components of the VideoCo system requiring a user interface includes the main user-facing frontend for customers to register, browse, rent, and return movies. Users will also need to make payments through this user interface. Additionally, the user interface must allow users to browse the latest version of the user documentation for the VideoCo system. The VideoCo system also needs a user interface, consistent with that for customers, for the operators to be able to help customers browse, choose, and rent videos. The shipping team and other management teams also need a user interface to make changes to the system such as the update interval for the list of to-be-shipped videos. The user interface for the shipping team can be simpler and does not need to follow the Material Design guidelines.

The user interface must be comfortable for use on a variety of common devices and device sizes in use today. The user interface must work well on large computer desktop monitors, on smaller laptop screens, and on even smaller mobile devices. The user interface must be comfortable for use on common aspect ratios including 5:4, 4:3, 3:2, 16:10, 16:9, 1.85:1, 2.35:1.

The user interface must have a consistent appearance and experience across the different browsers supporting including Internet Explorer 11 or higher, Microsoft Edge 79 or higher, Firefox 78 or higher, Chrome 83 or higher, Safari 13 or higher, Chrome for Android 80 or higher, Firefox for Android 75 or higher, iOS Safari 13 or higher, Chrome for iOS 80 or higher, Firefox for iOS 20 or higher.

### 3.2 Hardware Interfaces

The VideoCo system must interface with the point-of-sale devices used at VideoCo stores for making payments. The new VideoCo system must be able to communicate with credit card terminals. When a video is rented or returned, the credit card terminal is used to make payment and to notify the system of the update to the video catalog. The credit card terminal will connect over USB with a computer that will handle the serial communication and then communicate directly with the VideoCo system.

The VideoCo system front-end must run on a CPU with a clock speed of 1.6 GHz or faster and at least 1 GB of RAM. The backend server and database subsystems of the VideoCo system must run on commonly available cloud hardware.

### 3.3 Software Interfaces

The VideoCo system must support macOS, Linux, and Windows. More specifically, the VideoCo system must be tested on OS X Yosemite; 32-bit versions of Windows 7, 8.0, 8.1, and 10; 64-bit versions of Windows 7, 8.0, 8.1, and 10; Ubuntu Desktop 14.04, Debian 7, Red Hat Enterprise Linux 7, CentOS 7, and Fedora 23. On Windows systems the Microsoft .NET Framework 4.5.2 is required. On Linux systems GLIBCXX version 3.4.15 or later and GLIBC version 2.15 or later is required.



The VideoCo system must support recent versions of common modern browsers including Internet Explorer 11 or higher, Microsoft Edge 79 or higher, Firefox 78 or higher, Chrome 83 or higher, Safari 13 or higher. The VideoCo system must support recent versions of common modern mobile browsers including Chrome for Android 80 or higher, Firefox for Android 75 or higher, iOS Safari 13 or higher, Chrome for iOS 80 or higher, Firefox for iOS 20 or higher.

The VideoCo system must provide integration points to interface with different payment providers. Common payment providers for websites include Stripe, PayPal, WePay, Amazon Payments, and more. The VideoCo system must support these payment providers in a generic way such that payment providers can be added and removed.

The components of the system must share data. The user interface must receive data from the server and the server from the database. The video catalog database for example will be a common point of interaction as the server fetches data from the catalog based on user requests. When users rent or return videos through the user interface the system will update the database. The software components must all share a user database as well so that users can be authenticated and authorized for actions they take and data they access.

### **3.4 Communications Interfaces**

The components and services of the system will communicate using HTTP. The HTTP communication must use SSL/TLS to encrypt and authenticate data transmissions and prevent man-in-the-middle attacks and other security risks to communications. The HTTP requests and responses will use the application/json MIME type in most cases to communicate information. In some cases, the response data will be binary data including images and videos for example when populating the user interface with pictures and trailers of the available videos.

The VideoCo system will use a RESTful API and handle common CRUD operations. CRUD stands for Create, Read, Update, and Delete. The VideoCo system will present resources at appropriate endpoints. Subsystems will use the endpoints along with HTTP actions POST, GET, PUT, and DELETE to retrieve information and make changes to shared data.

The VideoCo backend must use an ACID-compliant database that provides consistent views of the data for customers even as many other customers are browsing, renting, and returning videos. Such a database will help support concurrent access to the system by many users and keep data synchronized between many different clients.

## 4. System Features

### 4.0.1 Functional Requirements

**REQ-1:** Users must login to the system with valid and existing credentials. Invalid credentials will result in a login error.

**REQ-2:** Users must be able to register to the system with a unique username and email. Invalid credentials will result in a registration error.

**REQ-3:** Users will be able to search for movies by name, category, or all movies. If no movies match the selected search, a message is displayed indicating no movies match the desired search.

**REQ-4:** Users must be able to create an order by adding a movie to their order. Users will not be able to add a movie which is not in stock.

**REQ-5:** Users must be able to review a tentative order, removing items from or cancelling the order as needed.

**REQ-6:** Users must be able to pay for an order using a payment service or loyalty points if they have enough. Invalid shipping or payment information will show the user an error respective of the invalid credential.

**REQ-7:** Users will be rewarded 1 loyalty point per order. 10 loyalty points is applicable toward a free movie rental.

**REQ-8:** The User should be notified of a late fee based on their location. Canadian Users are charged a late fee of \$9.99 if they are located outside of Ontario. The company is not global, it is Canada only.

**REQ-9:** The system will update the available movie stock according to the corresponding user action (placed order, cancelled order).

**REQ-10:** Users must be able to dial in an operator and place an order for their movie rentals via the operator (after providing shipping/billing address). The user will be given a unique order ID for their completed order.

**REQ-11:** Users must be able to dial in an operator to check the status of their order by using their unique order ID. Invalid order IDs will have no corresponding order.

**REQ-12:** Users must be able to rent out available movies from the two store locations in Toronto if they choose to walk into the store.

**REQ-13:** Users must be able to cancel their movie rental order given that the order status is not "delivered".

**REQ-14:** Users must be able to manage their account/profile. This includes changing any of the following: password, name, email.

**REQ-15:** Users must be able to view their collected loyalty points while managing their account.

**REQ-16:** The System Admin is able to add/remove a movie from the system.

**REQ-17:** The System Admin is able to update movie information (title, actors, directors, date of release, description).

**REQ-18:** A system admin will be able to retrieve and update customer account information including: name, email, password, username, order information, order status.

**REQ-19:** A system admin will be able to delete existing customer accounts.

**REQ-20:** The system can add admin accounts to the system using the administrator's name and email address. An admin must be an employee of the company.

**REQ-21:** The system can remove admin accounts from the system.

**REQ-22:** The system will be able to update the customer's order shipping status.

**REQ-24:** The system will charge customers a late fee of \$1.00 CAD per day for movies which are not returned within two weeks. The \$1.00 CAD charge is a per movie charge.

**REQ-25:** The system will send customer orders to the warehouse after they have been placed.

**REQ-26:** The system will dispatch a list of ordered movies to be shipped to different warehouses based on shortest geographic distance to the customer's destination address.

## **4.1 System Login/Register**

### **4.1.1 Description and Priority**

All users will be required to log on to the system. Before a user can interact with the system, they must first provide his/her system identification information (username and password) to the system. If the user is not a returning user, then they should be able to provide their identification (username, password, email) to register into the system. Using this information, the system will authenticate the user and either grant or prevent the user from accessing the system. Using the online system will be required to register/login to order any movie to be shipped to customers. Not having a Register/Login system will make keeping track of orders, as well customers, and customer loyalty points will be extremely difficult, and thus has High priority.

*Priority:* High

### **4.1.2 Stimulus/Response Sequences**

The stimulus needed to initiate this behaviour would be the user's desire to access the online system. This could be to browse or rent movies, view pending orders, cancel orders, manage loyalty subscriptions, and manage user credentials/information.

1. If the user is a returning user then they are able to login to the system with their credentials.
2. If the user is a new user, they are able to register into the system and login with their new credentials.

#### 4.1.3 Functional Requirements

**REQ-1:** Users must login to the system with valid and existing credentials. Invalid credentials will result in a login error.

Priority: High

The priority of this requirement is high since this is a central functionality of the system, and provides optimal user experience. It also allows for security and privacy concerns to be handled.

**REQ-2:** Users must be able to register to the system with a unique username and email. Invalid credentials will result in a registration error.

Priority: High

The priority of this requirement is high since this is a central functionality of the system, as a user must be able to register for an account so they can use all system functionalities accordingly.

## 4.2 Browse Movie by Name, Genre, or All Movies and View Movie Description

### 4.2.1 Description and Priority

Users that are logged into the system will be able to search for available movies on the online system for rental, by name, genre, or all movies. Movies that are available to be ordered for rental are displayed. If the movie is in the online system, it is available to be ordered for rental, given that there are copies available to be rented. Users will also be able to view the movie description (as well as title, actors, directors) for a selected movie available. Customers will be required to search or browse movies that are available in the online system before they are able to order them. Thus, preventing the customer from ordering a movie that is not available to be rented from the online system, making this feature a High priority.

*Priority:* High

### 4.2.2 Stimulus/Response Sequences

The stimulus needed to initiate this behaviour would be the user's desire to filter and search for a movie that suits their preference.

1. The user has a preference for a movie they want to view.
2. The user logs into the online system.
3. The user will then be able to view all movies available on the system. If the user wishes to browse movies by name, a search tool with a search bar is available to

filter movies by the name entered. To filter by Genres, a category with genres is available to filter movies by the selected category.

#### 4.2.3 Functional Requirements

**REQ-3:** Users will be able to search for movies by name, category, or all movies. If no movies match the selected search, a message is displayed indicating no movies match the desired search.

Priority: High

The priority of this requirement is high since searching and finding movies is a must have functionality in a movie rental system. There is great benefit to this requirement and without it the system suffers.

### 4.3 Place an Order to Rent Movies Online from the System

#### 4.3.1 Description and Priority

All registered users will be able to add movies to a new order. Users will be able to add or remove movies from an order, as well as view their current tentative order if any at all. A user will be able to begin to place an order after they are satisfied with the movies they wish to rent from the system. After the user begins placing the order, they will be required to provide shipping and payment information in order to complete the order. Upon a valid shipping address and payment method being provided, the order will then be placed and a confirmation is shown to the user. The system updates its inventory of movies accordingly. Renting movies from the system is the central functionality of the software system and thus is of highest priority.

Priority: High

#### 4.3.2 Stimulus/Response Sequences

The stimulus needed to initiate this behaviour would be the user's desire to rent a movie from the system by placing an order.

1. A user adds at least one movie to their order
2. User views order before proceeding to shipping and payment
3. User provided payment and shipping information
4. User may pay with loyalty points if they have a sufficient amount
5. Upon valid payment and shipping information the order is placed.
6. The movie inventory stock in the system is updated accordingly

#### 4.3.3 Functional Requirements

**REQ-4:** Users must be able to create an order by adding a movie to their order. Users will not be able to add a movie which is not in stock.

Priority: High

The priority of this requirement is high since ordering movies is a must have functionality in a movie rental system. There is great benefit to this requirement and without it the system does not provide the service as needed.

**REQ-5:** Users must be able to review a tentative order, removing items from or cancelling the order as needed.

Priority: High

The priority of this requirement is high since allowing the revision of tentative orders and cancellations of orders provides an optimal user experience. Without this there is too much room for accidental error on the user's part which will result in a worse customer experience overall.

**REQ-6:** Users must be able to pay for an order using a payment service or loyalty points if they have a sufficient amount. Invalid shipping or payment information will show the user and error respective of the invalid credential.

Priority: High

The priority of this requirement is high since payment is a crucial part of the business's success. Loyalty points must be incorporated into the system as a means of payment to promote customer retention in our system.

**REQ-7:** Users will be rewarded 1 loyalty point per order. 10 loyalty points is applicable toward a free movie rental.

Priority: High

The priority of this requirement is High because loyalty points must be incorporated into the system as a means of payment to promote customer retention in our system. These points must have a value associated with them and this req builds on **REQ-6**.

**REQ-8:** The User should be notified of a late fee based on their location. Canadian Users are charged a late fee of \$9.99 if they are located outside of Ontario. The company is not global, it is Canada only.

Priority: Low

The priority of this requirement is low as charging a late fee based on location is a good feature to have in our system to ensure optimal profit margins, but this does not change the central functionality of the system for the user.

**REQ-9:** The system should update the available movie stock to the corresponding user action (placed order, cancelled order).

Priority: High

The priority of this requirement is high since we need to ensure that the orders customers are making are executable on the part of the business. If the movie stock is not correct a customer may order a movie which we do not currently have, resulting in a rather negative situation for our business and customers.

## 4.4 Place an Order to Rent Movies via Operator or in Store

### 4.4.1 Description and Priority

All users will be able to place an order to rent a movie by dialling an Operator to place the order for them or by going into the stores located in Toronto if their geographical location permits. The dialling the Operator and placing an order is intended for non tech savvy users,

but may be used by any customer who chooses so. In store the customer. When in store the customer simply completes the transaction to rent out the movie as they would in any other store. When a user calls an operator, they are not required to provide login credentials and therefore may not use any loyalty points to pay for their order. No login means no loyalty program subscription or points and this is one pitfall of ordering via the operator. When an operator is dialled, the customer must provide them with the movie(s) they wish to rent, shipping information, and payment, and they are provided with a unique order ID which they can use later to check their order status. To check their order status an user may dial in an operator and provide their unique order ID. The system updates its inventory of movies accordingly. Renting movies is a critical part of the system and thus this feature is of high priority.

*Priority: High*

#### 4.4.2 Stimulus/Response Sequences

The stimulus needed to initiate this behaviour would be the user's desire to rent a movie from the system by placing an order via the Operator or in Store.

Operator:

1. A user dials the operator to order the movie(s) for them.
2. The user confirms the order with the operator.
3. User provides shipping and billing information to the operator.
4. Upon valid billing and shipping information the order is placed and the user is provided an unique order ID
5. The bill is sent to the customer billing/shipping address. They do not need to provide payment information over the phone.
6. The movie inventory stock in the system is updated accordingly
7. User dials in the operator and uses their unique order ID to check the status of their order.

In Store:

1. A user goes into a store in Toronto and rents out a movie
2. The movie inventory stock in the system is updated accordingly

#### 4.4.3 Functional Requirements

**REQ-10:** Users must be able to dial in an operator and place an order for their movie rentals via the operator (after providing shipping/billing address). The user will be given an unique order ID for their completed order.

*Priority: High*

The priority of this requirement is high since we need to target as many customers as possible, and this includes non tech savvy customers. This is analogous to a customer ordering online, as to the business, an order is an order regardless of how it is placed.

**REQ-11:** Users must be able to dial in an operator to check the status of their order by using their unique order ID. Invalid order IDs will have no corresponding order.

*Priority: Medium*

The priority of this requirement is medium because users who call into the operator do not have an account (as they are not tech savvy) and so they need some means of checking the status of their order without an account. The priority is medium since regardless the customer order will be fulfilled at the end.

**REQ-12:** Users must be able to rent out available movies from the two store locations in Toronto if they choose to walk into the store.

Priority: High

The priority of this requirement is high since walking into the store in Toronto is a quick and viable option for those in the Toronto area. This also provides some further incentive for local Toronto residents to rent out movies.

**REQ-7:** Users will be rewarded 1 loyalty point per order. 10 loyalty points is applicable toward a free movie rental.

Priority: Medium

**REQ-8:** The User should be notified of a late fee based on their location. Canadian Users are charged a late fee of \$9.99 if they are located outside of Ontario. The company is not global, it is Canada only.

Priority: Low

**REQ-9:** The system will update the available movie stock to the corresponding user action (placed order, cancelled order).

Priority: High

## 4.5 Cancel Orders that Have Not Been Shipped

### 4.5.1 Description and Priority

Registered users that are logged into the system and have ordered a rental will be able to cancel their order given that the movie has not been shipped yet. The user will be able to view their order status (processed, shipped, delivering, delivered) and if the order status is not “delivered”, the user will be able to cancel their order. Giving the customers the option to be able to cancel orders makes orders more flexible as it ensures customers are ordering the movie they want and choose to rent thus making this a High priority.

*Priority:* High

### 4.5.2 Stimulus/Response Sequences

The stimulus needed to initiate this behaviour would be the user’s desire to cancel an order in which they have placed for a movie rental.

1. The user has a desire to cancel a movie that has been ordered
2. The user must log into the system and navigate to view their order status.
3. Through their order status, they are able to view their order and choose to cancel it.
4. After the order has been cancelled, a refund in the system is initiated, and a message is displayed, indicating that they have cancelled their order and information regarding the movie order cancellation (including cancellation number, movie order that was cancelled, and refund of their order rental) is sent to their email.

### 4.5.3 Functional Requirements

**REQ-9:** The system will update the available movie stock according to the corresponding user action (placed order, cancelled order).

Priority: High



**REQ-13:** Users must be able to cancel their movie rental order given that the order status is not “delivered”.

Priority: High

The priority of this requirement is high since we need to ensure that users can cancel an order which they may no longer need, or was accidentally ordered on their part. Having this as a high priority will allow the user the best experience with the system overall.

## 4.6 Manage User Account

### 4.6.1 Description and Priority

All Users will be able to manage their account. This will include allowing them to change their password, name, and view their current credentials. A user is also able to view how many loyalty points they have in their account. As this is an extension to the system, it is not required, but enhances the experience for customers and creates opportunities for returning customers, thus making its priority as Medium.

*Priority: Medium*

### 4.6.2 Stimulus/Response Sequences

The stimulus needed to initiate this behaviour would be the user’s desire to manage their account as well as view their loyalty points.

1. A returning user accesses the manage profile section
2. User views their loyalty points
3. User changes their last name and their password

### 4.6.3 Functional Requirements

**REQ-14:** Users must be able to manage their account/profile. This includes changing any of the following: password, name, email.

Priority: Medium

The priority of this requirement is medium since user account management allows the user to change their profile according to their needs. This functionality does not change the user's experience renting out a movie and thus it is not of high priority.

**REQ-15:** Users must be able to view their collected loyalty points while managing their account.

Priority: Medium

The priority of this requirement is medium since users should be able to view their loyalty points to see if they are eligible for a free movie. This functionality does not change the user's experience renting out a movie and thus it is not of high priority.

## 4.7 System Admin Add/Remove Movies and Update Existing Movie Information

### 4.7.1 Description and Priority

A System Admin will have a different set of functionalities from normal users. A System Admin will be able to view movies, as well as its movie's description. A System Admin will also be able to add or remove a movie to the online system. A System Admin will also be able to update a selected movie's information (title, actors, directors, description). For customers, they are only able to view the movies that are available in the system. However, a System Admin is the only user with permissions to be able to add/remove movies and to update existing movie information, thus making this a High priority.

*Priority:* High

#### 4.7.2 Stimulus/Response Sequences

The stimulus needed to initiate this behaviour can vary depending on the action required from a System Admin.

##### **Add a movie:**

The stimulus needed to initiate adding a movie to the system will be a System Admin's desire to add a movie that is not currently an existing movie in the system.

1. A System Admin is required to add a movie that is not currently an existing movie in the system.
2. A System Admin adds the information regarding the movie (title, actors, directors, date of release, description)
3. In a successful addition of a movie, a message will display to a System Admin indicating the success of the addition of the movie. On failure of the addition of a movie, an error message is displayed.

In a successful addition of a movie, a message will display to the System Admin indicating the success of the addition of the movie. On failure of the addition of a movie, an error message is displayed.

##### **Remove a movie:**

The stimulus needed to initiate removing a movie from the system will be a System Admin's desire to remove a movie that is not being offered to be rented by the system anymore.

1. A System Admin is required to remove a movie that is not being offered to be rented by the system.
2. A System Admin removes the movie from the online system. In a successful removal of a movie, a message will display to a System Admin indicating the success of the removal of the movie. On failure of the removal of a movie, an error message is displayed.

##### **Updating a movie:**

The stimulus needed to initiate removing a movie from the system will be a System Admin's desire to remove a movie that is not being offered to be rented by the system anymore.

3. A System Admin is required to remove a movie that is not being offered to be rented by the system.

4. A System Admin removes the movie from the online system. In a successful removal of a movie, a message will display to a System Admin indicating the success of the removal of the movie. On failure of the removal of a movie, an error message is displayed.

#### 4.7.3 Functional Requirements

**REQ-3:** Users will be able to search for movies by name, category, or all movies. If no movies match the selected search, a message is displayed indicating no movies match the desired search.

Priority: High

**REQ-16:** The System Admin is able to add/remove a movie from the system.

Priority: High

The priority of this requirement is high since a central and key functionality of the system is to remove and add movies to the system as needed. Without this the movie selection will be outdated quickly.

**REQ-17:** The System Admin is able to update movie information (title, actors, directors, date of release, description).

Priority: Medium

The priority of this requirement is Medium as updating movie information allows the user to be provided the accurate information regarding potential movies they would want to rent out. If a movie's information is initially set incorrectly this will allow flexibility to fix these issues.

## 4.8 System Admin View/Edit Customer Profiles and Orders

### 4.8.1 Description and Priority

System Admin Users will be able to view all customer accounts and their corresponding orders. System admin will be able to change customer account information as well as customer order information/status. System admins will be able to retrieve customer credentials for the customer. A system admin will be able to view a list of movies taken out by a certain customer at any time. A system admin is also able to delete existing customer accounts in the system. As this feature allows System Admin Users to track existing movies that are currently being rented, it is given a Medium priority.

Priority: Medium

### 4.8.2 Stimulus/Response Sequences

The stimulus needed to initiate this behaviour can vary depending on the action required from the Admin user.

The Admin user retrieves/updates customer information:

1. Admin accesses the customer account using their username
2. Admin retrieves any necessary information from the customer account including name, email, password, username, order information, order status.
3. Admin updates any information on the customer account including: name, email, password, username, order information, order status.

The Admin user deletes existing customer account:

1. Admin accesses the customer account using their username.
2. Admin deletes the account.

### 4.8.3 Functional Requirements

**REQ-18:** A system admin will be able to retrieve and update customer account information including: name, email, password, username, order information, order status.

Priority: Medium

The priority of this requirement is medium as an admin should have access to all customer accounts if they need to retrieve any customer information. This is not of high priority as it is not directly affecting the customers overall experience using the system.

**REQ-19:** A system admin will be able to delete existing customer accounts.

Priority: Low

The priority of this requirement is low since deleting customer accounts for inappropriate activity such as constantly ordering and cancelling orders is a good functionality to have in the system, but is not entirely needed.

## 4.9 System Add/Delete Admin Accounts

### 4.9.1 Description and Priority

The system will be able to add/delete admin accounts. An admin account is created when a new administrative employee is hired and accounts are removed when former admins are leaving their job. These accounts allow for administrative control over the system and are essential. Admin accounts are registered using employee names and emails. It is required that an admin be an employee of the company. With all this, it is categorized as High priority.

*Priority:* High

### 4.9.2 Functional Requirements

**REQ-20:** The system can add admin accounts to the system using the administrator's name and email address. An admin must be an employee of the company.

Priority: High

The priority of this requirement is high since admin accounts are needed in the system for optimal functionality when a new system administrator is hired.

**REQ-21:** The system can remove admin accounts from the system.

Priority: High

The priority of this requirement is high since admins who are no longer an employee of the business need to have their accounts deleted accordingly to ensure system privacy and security.

## 4.10 System Processes User Account Authentication and Orders

### 4.10.1 Description and Priority

The system will authenticate user accounts (system admin and customer). The system will also create user accounts when a new customer registers for an account. The system will be able to check and charge a late fee to customers based on order credentials such as shipping address. The system will also charge customer accounts for late return fees when the customer does not return a movie within 2 weeks. The system will also send orders to the warehouse after an order has been completed on the users end.

*Priority: High*

#### 4.10.2 Stimulus/Response Sequences

The stimulus needed to initiate this behaviour is that the system must internally ensure desired functionality of the system is being provided to the user.

1. User logs into system and system authenticates user credentials to allow this behaviour
2. Customer checks out an order and the system charge a late fee based on the customer's location.
3. System sends order to warehouse for processing
4. System charges customers a late fee for not returning the movie(s) within a 2-week window.

#### 4.10.3 Functional Requirements

**REQ-22:** The system will be able to update the customer's order shipping status.

*Priority: High*

The priority of this requirement is high since the customer needs to be able to check their order status, and to ensure that the information they receive is accurate the system needs to be able to update the order status of the customers' orders accordingly.

**REQ-24:** The system will charge customers a late fee of \$1.00 CAD per day for movies which are not returned within two weeks. The \$1.00 CAD charge is a per movie charge. The priority of this requirement is high since customers need to have an incentive to return the movies on time back to the business, otherwise there is nothing stopping them from keeping the movies they rent out indefinitely which is very costly for the business.

**REQ-25:** The system will send customer orders to the warehouse after they have been placed.

*Priority: High*

The priority of this requirement is high as to ensure customer orders are being carried out the system will be able to communicate directly with the warehouses to let them know to begin packaging customer movies.

**REQ-8:** The User should be notified of a late fee based on their location. Canadian Users are charged a late fee of \$9.99 if they are located outside of Ontario. The company is not global, it is Canada only.

*Priority: Low*

### 4.11 System Assembles List of Movies Ordered and Dispatches List to Warehouses

#### 4.11.1 Description and Priority

The system will assemble a list of ordered movies and dispatch the list to different warehouses based on the shortest geographic distance to reach the customer's destination address and whether the movie is available in the specific warehouse. The list of movies ordered is delivered to the different warehouses at 5:00 AM EST and 3:00 PM EST. As ordered movies

are to be shipped from the warehouse to the customer's destination address for customers to consume, this is categorized as having High priority.

*Priority:* High

#### 4.11.2 Stimulus/Response Sequences

The stimulus needed to initiate this behaviour is that the system recognizes the time and compares it with the dispatch times. If both times are the same, the list of ordered movies is dispatched.

1. System time matches one of the dispatch times.
2. System dispatches the ordered movies to the different warehouses.
3. Warehouse receives a list of ordered movies to be shipped.

#### 4.11.3 Functional Requirements

**REQ-26:** The system will dispatch a list of ordered movies to be shipped to different warehouses based on shortest geographic distance to the customer's destination address.

*Priority:* High

## 5. Other Non-functional Requirements

Non-functional requirements, as the name suggests, are requirements that are not directly concerned with the specific services delivered by the system to its users. They may relate to emergent system properties such as reliability, response time, and store occupancy. Alternatively, they may define constraints on the system implementation such as the capabilities of I/O devices or the data representation used in interfaces with other systems. Non-functional requirements, such as performance, security, or availability, usually specify or constrain characteristics of the system as a whole

### 5.1 Performance Requirements

The online system has a high number of users interactions so maintaining the amount of response time reasonable for all users is important. The system response time for every instruction conducted by the user must not exceed more than a minimum of 5-10 seconds. The system should have a high-performance rate when executing the user's input and should be able to provide response within a short time span, usually 5-10 seconds for highly complicated tasks and 3-5 seconds for less complicated tasks.

The system must accommodate 250-300 simultaneous users or less within the peak load period/ day time. The system shall be able to process a notification in 1 second or less, and up to and including 100 notifications in 15-20 seconds or less. Database interaction and data retrieval must execute at a minimum rate of 1-3 seconds per transaction or no more than 8 seconds latency.

### 5.2 Safety Requirements

If there is extensive damage to wide portion of the database due to catastrophic failure, such as a disk crash, the recovery method restores a past copy of the data that was backed up to archival storage (typically tape) and reconstructed a more currently state by reapplying or redoing the operations of committed transaction from the backed up log, up to the time of failure.

### 5.3 Security Requirements

The system provides a login mechanism in place, So The registered users and system users (staffs) have username and password for the system to prevent unauthorized access. The user's password must be greater than eight characters. The subsystem should provide high level security and integrity of the data held by the system, only authorized personnel of the company can gain access to the company's secured page on the system; and only users with valid password and username can login to view the user's page.

Only high tier persons have admin level login credentials. When a user forgot their login credentials, they must be requested from the administrator. For online users such as customers, the system works as administrator. But for staff members the admin should process this request using the system. When password resetting a user cannot be reused his/her initial password. All users must be uniquely identified, this could be done by unique username and password for login. If applicable, grant user category as Visitor or Guest or read-only capabilities for unauthenticated users who are not a valid user.

#### **Password Format:**

- Must contains at least 8 characters
- Must have one UPPERCASE and One LOWERCASE letter
- Must have at least one special character (Non-Alphanumeric)
- Must have at least one number

Data integrity should be assured by limiting access to the database and by appropriate synchronization and back-up functionalities. The user's privacy will be granted by the limited access that the log-in process is going to give.

When a user is unable to reset password through the system, he/she can request the service through email to the admin by providing some more information such as email, last purchase details etc. This service must be handled by the admin separately. The system must automatically log out the users after a period of time inactivity.

The system uses SSL (secured socket layer) in all transactions that include any confidential user information. The system does not leave any cookies on the customer's computer containing the user's password. The system backend servers shall only be accessible to authenticated administrators. Sensitive data will be encrypted before being sent over insecure connections like the internet.

### 5.4 Software Quality Attributes

**Availability**

The system should be available for access all-time 24 hours, 7 days a week. This means users can access it using a web browser, only restricted by the down time of the server on which the system runs. In case of hardware failure or database corruption, a replacement page will be shown. Also, in case of a hardware failure or database corruption, backup of the database should be retrieved from the server and saved by the administrator. Then the service will be restarted. It means 24 x 7 availability.

Also, in the occurrence of any major system malfunctioning, the system should be available in 1 to 2 working days, so that the business process is not severely affected. The system overall availability rating should be 99% or above.

**Maintainability**

A commercial database is used for maintaining the database and the application server takes care of the site. In case of a failure, a re-initialization of the program will be done. Also the software design is being done with modularity in mind so that maintainability can be done efficiently.

**Reliability**

The system will not crash on invalid input data or null search results. The mean time between failures shall exceed 420 hours or more. The mean time to repair shall not exceed 8 hours. The system should provide user requested data with accuracy >95%. The maximum defect rate that the system could support is 3% bugs/KLOC (thousands of lines of code). The reliability of the system is the backup of the database which is continuously maintained and updated to reflect the most recent changes.

**Usability**

The system provides a help and support menu in all interfaces for the user to interact with the system. The user can use the system by reading help and support.

**Error Handling**

Error should be considerably minimized and an appropriate error message that guides the user to recover from an error should be provided. Validation of the user's input is highly essential also the standard time taken to recover from an error should be 15-20 seconds.

**Ease of Use**

Considered the level of knowledge possessed by the user of this system. A simple but quality user interface should be developed to make it easy to understand and require less training.

**Modifiability**

The system must allow data and schema evolution such that it is possible to add, remove, and modify data attributes continuously and without the costs of migrating the database. This helps protect the system against change such as new kinds of events that require tracking new attributes or new features that require new user attributes.



## Flexibility

The system must be able to run on the most common mobile operating systems: Android 4.4+ and iOS 6+.

## Verifiability

The system must have unit test coverage greater than 75% and end-to-end tests with coverage greater than 50%.

## 5.5 Business Rules

**Customer** - Manage own user data, Rentals and purchases, User credential changes (password). The customer must supply a valid credit card information for online renting and purchases.

**Admin** - Can manage or overview all kinds of functions under this system. User management, Order management, Inventory management.

**System** – Responsible for adding Admin level users and other category users, manage order status, manage movie items stock, remove access for users and authenticate user accounts. System can automatically process some tasks related to business needs such as calculate late fee when customer return rented items etc.

**Cashiers** - Can perform in store desk tasks such as handling payments - cash or card, handling return items, rentals and sales. Other tasks are restricted to this level user.

**Inventory operator** - Can perform all tasks as cashier and handle calls from customers and make order requests, Manage order status. Other tasks are restricted to this level user.

**Warehouse Team** - Manage shipment list, manage shipment receiving. Other tasks are restricted to this level user.

**Shipping Team** – Manage shipping, prepares paper work for shipping, manage delivery, update the movie stocks details. Other tasks are restricted to this level user.

**Payment Service** - This is a third-party payment system, that handles all payments requests which are made by the rental system (Original System). Credit card validations.

**User DB, Order DB, Movies DB** - Handling database transactions and Data storage management, Concurrency control, Back up.

## 6. Other Requirements

The system must store all user information into the system database. For each customer user account username, password, email, delivery address, payment details stored with User Id. There must also be a field in the user account database which handles the users accumulated loyalty points. There must be a database to store the current and past orders and one to store

the available movies in the system. These databases can be relational or non-relational left to the preference of the developers.

## Appendix A: Glossary

- REST - Representational State Transfer (API style)
- UI - User interface
- GUI - Graphical User Interface
- DB - Database
- JSON - JavaScript Object Notation
- SOA - Software Oriented Architecture
- SSL - Secure Socket Layer
- HTTP - Hypertext Transfer Protocol
- TCP - Transmission Control Protocol
- ACID - Atomicity, Consistency, Isolation, Durability