**Software Requirement Specification**

**MRS (Movie Rental Software)**

**By:**

**Ryan Askew**

**Quinton Thompson**

**Jared Jones**

**Quentin Higgins**

**3/28/17**

**Table of Contents:**

[**1. Introduction**](#mvghr8jk6z2n)**--------------------------------------------------------------------------------------------------- 3**

[**1.1 Purpose**](#3r4zpnhgipng)**-------------------------------------------------------------------------------------------------- 3**

[**1.2 Scope**](#eqon8zs758ii)**----------------------------------------------------------------------------------------------------- 3**

[**1.3 Definitions**](#808xua1xpw6h)**---------------------------------------------------------------------------------------------- 3**

[**1.4 References**](#sznufs8js0ox)**---------------------------------------------------------------------------------------------- 3**

[**1.5 Overview**](#w3tmegv8zccp)**------------------------------------------------------------------------------------------------- 3**

[**2. Overall Description**](#al6y3w1jmyiy)**----------------------------------------------------------------------------------------- 4**

[**2.1 Product Perspective**](#6rlach170gb)**--------------------------------------------------------------------------------- 4**

[**2.2 Product Functions**](#48uxd6w36uxt)**------------------------------------------------------------------------------------ 4**

[**2.3 User Characteristics**](#m4ujd8xgl7o8)**--------------------------------------------------------------------------------- 7**

[**2.4 Constraints**](#76o0lbbchc4s)**--------------------------------------------------------------------------------------------- 7**

[**2.5 Assumptions**](#rmetgpmf114j)**------------------------------------------------------------------------------------------- 7**

[**2.6 Apportioning of Requirements**](#jz1ifu6vwlke)**--------------------------------------------------------------------7**

[**3. Specific Requirements**](#ou9rrfuphug0)**------------------------------------------------------------------------------------ 7**

[**3.1 External Interfaces**](#k0gm26mohmhs)**----------------------------------------------------------------------------------- 7**

[**3.2 Functions**](#xaiq7ebgr7bu)**------------------------------------------------------------------------------------------------ 8**

[**3.3 Performance Requirements**](#wann9exul782)**--------------------------------------------------------------------- 12**

[**3.4 Logical Database Requirements**](#ypa9duh0lim3)**--------------------------------------------------------------- 12**

[**3.5 Design Constraints**](#utr189cg2xrz)**--------------------------------------------------------------------------------- 12**

[**3.6 Software System Attributes**](#28ff0o1stff8)**--------------------------------------------------------------------- 13**

1. **Introduction**

**1.1 Purpose**

The purpose of this Software Requirement Specification (SRS) document is to provide all of the information, to both the software developers and the recipients of the MRS product (the client), that is necessary to ensure that the final product is of the highest quality possible and meets the needs of the client in the best way possible.

**1.2 Scope**

MRS will be a web-based service operating on a website where customers are able to create an account and sign up for a monthly subscription service for movie rentals where movies will be shipped to them. The software does not include the ability to stream movies directly to a customer’s device - it is simply an online physical DVD rental service that makes it easier for customers to enjoy their favorite movies.

MRS is not a self-contained system. While customer data and movie data will be stored in the system, MRS will rely on external systems for email service and payment processing.

**1.3 Definitions, Acronyms, Abbreviations**

OPSP: Refers to vetted, external Online Payment Service Providers

(OPSP’s),such as Paypal, Authorize.net, etc.

**1.4 References**

[1]FTC Rules

[IEEE-STD-830-1998.pdf](https://tracs.txstate.edu/access/content/group/5a322313-2652-4f38-adc3-5d9407acc02f/IEEE-STD-830-1998.pdf)

https://www.ftc.gov/enforcement/rules

**1.5 Overview**

The remainder of this SRS document consists of two sections - the overall description (section 2) and the specific requirements (section 3). Section 2 provides a summary description of each functional part of the software. Section 3 consists of the functional and nonfunctional requirements of the software.

1. **Overall Description**

**2.1 Product Perspective**

**Perspective**

MRS will be implemented alongside new and existing systems. A server will be used to store movie information so that it can be retrieved quickly when a request is made by the software. User information, which is sensitive and must be kept secure, will be stored on a more secure server. Finally, the site will use an OPSP[1] as a pre-existing way to process credit card payment without having to create a new payment system. Credit Card information associated with an account will be stored on the OPSP[1] servers.

**System Level Constraints**

The system will need to operate two servers, one for customer information and one for movie data. These will require server rooms and server hardware to be installed if the servers are to be owned and maintained by the client. In addition, users will need reliable access to movie data as well as their own data, so the servers must be reliably connected to the internet and be able to handle multiple customers accessing the servers at one time.

**Rationale**

MRS will provide a DVD rental option for those users who prefer to watch movies on DVD rather than through a streaming service. This hybrid service option can help bridge the gap between online and physical movie rentals at a time when physical DVD rental stores are going out of business.

**2.2 Product Functions**

**2.2.1 Register**

The register function creates an account for the user to view, purchase, and have movie rentals shipped to their residence through the website. It is required that the user create a username and password and supply the site with valid personal information. The user should be registered after the successful termination of this function.

**Rationale:** Enables the user to have the option of renting movies on the site

**2.2.2 Subscribe**

The subscribe function allows the user to choose a payment plan for their movie rentals. The first plan is the two movies plan which bills the user $5 per month and allows them to check out two movies at a time. The second plan is the unlimited option which costs $10 dollars and enables the user to check out as many movies as they would like. The fee for each of these plans is billed automatically to the user’s card unless the user opts to unsubscribe.

**Rationale:** Allows the user to select a plan to rent movies

**2.2.3 Unsubscribe**

The unsubscribe function removes the user from the current plan. The user will be unable to rent movies until re-subscribing to another plan.

**Rationale:** Lets the user cancel their current payment plan.

**2.2.4 Log In**

The login function allows the user to re-gain access to their account by providing their username and password. If the login attempt is unsuccessful, the user will be prompted to re-enter their account info. If the login is successful, the user will have control of their account again. The login function can only be used to regain control of an already existing account.

**Rationale:** Keeps user account info secure, deters someone from using someone else’s account info to pay for movie rentals.

**2.2.5 Log Out**

The user can use the logout function to securely leave the website and lock the account. Once the user is logged out, the only way to access the account again is by logging in.

**Rationale:** User can keep their account secure by logging out if using someone else’s computer to pay for rentals.

**2.2.6 Retrieve**

The user will be prompted to enter the email address associated with their account. If the user enters a valid email that is associated with the account, a temporary password is created. An email with both the account username and a temporary password is sent to this email address. The user is told to enter their username and temporary password. The temporary password will eventually expire. If the user re-enters their info successfully, they will be logged in to their account. The user will then be asked if they want to change their password. Whether they choose to or not, the user will have control of their account after making this decision.

**Rationale:** Allows the user to regain access into their account if they have forgotten their username or password.

**2.2.7 Edit Account Information**

If the user is not currently logged in, the website prompts them to login. The user can edit their account info, address, email address, and credit card information with this function.

**Rationale:** Lets user change their address after a move, change credit card info (example: previous card expired), and change their name.

**2.2.8 Edit Queue**

Every movie that the registered user has chosen for rental is put into a Queue. The user can add movies to their Queue, provided that the movie is currently in stock with the website. They can also delete movies from their Queue, and decide the order of shipping. The Queue displays movies that are currently rented by the user, and also displays movies that the user has requested to rent in the future. Once rented movies have been returned by the user, they are removed from the Queue.

**Rationale:** Allows the user to organize and keep track of their movies.

**2.2.9 Search**

The search function allows the user to type in a set of keywords in order to search for specific movie titles. All movies, if any, that are available on the site which match (or closely match) the user’s keywords will be shown in the search results. Each movie search result will contain, at a minimum, the movie’s title.

**Rationale:** The user should be able to quickly find a specific movie on the site.

**2.2.10 Browse DVD**

The browse DVD function allows the user to view movies in several categories: Action, Comedy, Classics, Drama, Foreign, and Horror. Each category will appear on a page in an appealing format, each with a selection of movies associated with that category. Each movie listing will have at least the movie’s title, a short description of the movie, and an option for the user to add that movie to his or her cart.

**Rationale:** The user should have a simple way to browse through the site’s movies without being overwhelmed by the selection.

**2.3 User Characteristics**

**Customers**

Customers will likely be average consumers who either want to rent physical copies of movies without having to go to a movie rental store, or who do not have access to such a store. They will likely have some knowledge of how the internet works, but they should not have to be very technically minded in order to use the site.

**Clients**

Clients will be the users who maintain the databases, add content to the site, and generally manage the way in which MRS is utilized. Clients will likely have extensive technical knowledge that will help them to maintain the site.

**2.4 Constraints**

MRS must follow all FTC rules regarding e-commerce. Security is also a concern when it comes to customer data, so security precautions must be taken. Additionally, the servers containing user data and movie data must each be reliable and fast enough such that they can be accessed quickly most of the time by multiple customers at one time.

**2.5 Assumptions and Dependencies**

The payment part of the system depends on the ability of the OPSP[1] to serve the vast majority of customers.

Users will have an active internet connection.

Customers who create an account have an email address.

**2.6 Apportioning of Requirements**

Any algorithms mentioned in this document will need to be implemented at a later time during development.

1. **Specific Requirements**

**3.1 External Interfaces**

MRS will interface with web browsers, which will display the site to the users in such a way that they are able to use it effectively. The browsers will handle all input and output from the user. The rest of the input will come from the servers containing user and movie data, as well as the OPSP[1] servers. The output will be handled by the browser and displayed on the user’s screen.

Email accounts will be handled by an external server.The address, email addresses, and names of customers are stored on the company servers. Customer credit card info is stored on a secure OPSP[1] server.

**3.2 Functions**

**3.2.1 Function: Register**

**Description:** Allows the user to set up an account with the website. The user inputs their information to create an account. Once the account has been created the user is able to rent movies from the site.

**Inputs:** user name, password, address, email address, and credit card type and number

**Source:** All aforementioned fields come from the user.

**Outputs:** New account and it’s information.

**Destination:** The user’s information is stored within their account and added to the database of accounts.

**Requires:** The user has entered the information

**Pre condition:** The register function is available to the user.

**Post condition:** An account is created for the user which they will need to use in order to use the movie rental service.

**Side effects:** N/A

**3.2.2 Function**: **Subscribe**

**Description**: Sets up one of two payment plans when the user registers for an account. The user selects either the two movie at $5 per month plan or the unlimited plan at $10 a month. In both cases the user is billed automatically monthly.

**Inputs**: The user selects from one of the two plans.

**Source**: The user

**Outputs**: A notification of the user’s subscription information and a message stating they will be billed automatically monthly.

**Destination**: Goes to the user database where in the information is filed away under the specific user’s account.

**Requires**: The user selects a subscription plan.

**Pre condition**: That the user has not already subscribed to a plan. If so, the user will be asked if they would like to change their plan. The Subscribe function is available to the user.

**Post condition**: The user’s plan information is updated and stored within the database.

**Side effects:** N/A

**3.2.3 Function**: **Unsubscribe**

**Description**: If the user unsubscribes from a subscription plan, they won’t be able to purchase rentals anymore and they will no longer be billed for a plan.

**Inputs**: The user selects the option to unsubscribe.

**Source**: The user

**Outputs**: The user will be notified that their subscription has been canceled and the user will not be able to purchase movies.

**Destination**: Goes to the user database where the subscription information is changed under the specific user’s account.

**Requires**: The user selects unsubscribe.

**Pre condition**: That the user is already subscribed to a plan. Unsubscribe function is available to the user.

**Post condition**: The user’s plan information is updated and stored within the database.

**Side effects:** N/A

**3.2.4 Function: Log In**

**Description:** The login function allows the user to access and manage their account. Once the user inputs their credentials, they will have access to their account.

**Input:** Username and password

**Source:** The User

**Output:** Message that tells the user that they have successfully logged in.

**Destination:** User’s web browser and screen

**Requirements:** A database with user account info.

**Pre condition:** The username received in input must match a username associated with a registered account. If the username is valid, the password that the user enters must match the password associated with this account. The Log In function is available to the user.

**Post condition:** The user is logged in.

**Side effects:** N/A

**3.2.5 Function: Log out**

**Description:** The logout function allows the user to securely sign out of their account. Once they have logged out, they cannot see their account info or their Queue until they have logged back in.

**Input:** The user selects the option to log out.

**Source:** The User

**Output:** Message telling the user that they have successfully logged out.

**Destination:** User’s web browser and screen

**Requirements:** The user is logged in.

**Pre condition:** The user must be logged in to an account, and the Log Out function is available to the user.

**Post condition:** The user is logged out of the account.

**Side effects:** N/A

**3.2.6 Function:** **Retrieve**

**Description:** Allows the user to retrieve username/password if they forget this info.

**Inputs:** Email address that is associated with the user account.

**Source:** The user.

**Outputs:** Message displaying that username and password have been reset.

**Destination:** Temporary password is sent to registered email.

**Action:** The user chooses to reset their password and enters their email address. Later on, the user is prompted to log in to their account with their username and the temporary password that has been sent to them by email.

**Requirements:** A database with user email info. An algorithm that can create a temporary password.

**Pre-condition:** The email that the user inputs at the prompt must match the email that is stored in the user’s account info. If it does not match, the user will not receive a temporary password. Also, the temporary password that the user must enter has to match the one sent by email. The Retrieve function is available to the user.

**Post-condition:** The user’s password has been reset to be the temporary password. The user is then asked if they want to change their password to something else.

**Side effects:** N/A

**3.2.7 Function: Edit account information**

**Description:** Allows the user to edit their address, email address, credit card info, or name.

**Inputs:** Address, credit card info, name (entered by the user).

**Source:** Newly entered user information

**Outputs:** “Info has been updated”

**Destination:** The new user information is stored in the user database where in the it is filed away under the specific user’s account. Credit card info will be stored on a secure OPSP[1] server.

**Requirements:** A database for storing user account info

**Pre condition:** For credit card info, the website code checks to see if the user inputted only numbers for credit card info. If not a valid number, user is given an error message. Invalid credit card strings will not be written to the OPSP[1] server. The Edit function is available to the user.

**Post condition:** The new user information is written to the database. Preferably, it should overwrite any old data that the user updated.

**Side effects:** N/A

**3.2.8 Function: Edit Queue**

**Description:** Allows the user to see and edit a list of what they wish to order.

**Inputs:** Any movie added by the user into the queue.

**Actions:** The user can edit their queue. The user can set their shipping order by arranging the movies into an ordered list. Movies at top of list will be shipped before movies that are lower on list. Once a movie has been returned by the user, the returned movie is removed from the queue, and therefore removed from the list.

**Outputs:** A complete list of all of the movies currently in the user queue.

**Destination:** The user’s screen and web browser.

**Requirements:** A database of movies currently in stock by the website.

**Pre condition:** The user can only add movies that are listed as being in stock. The Queue function is available to the user.

**Post condition:** All movies that the user wants to order are listed in their queue. The user has their shipping order arranged as well.

**Side effects:** N/A

**3.2.9 Function: Search**

**Description:** Allows the user to enter a set of keywords, and returns movie search results that closely match the keywords entered.

**Inputs:** Keywords entered by the user. Movie data.

**Source:** The user. The movie database server.

**Outputs:** Search results showing movies related to the user’s keywords.

**Destination:** The user’s web browser and screen.

**Action:** A search algorithm shall take as input the keywords given by the user. These keywords shall then be used to search the movie data server for related movies. Once these movies are found, they shall be displayed on the search results page as described in 2.2.

**Requirements:** A database containing data on all movies that the site has to offer.

**Pre-condition:** The Search function is available to the user.

**Post-condition:** The selected movies, if any, related to the user’s keywords have been fetched from the database and displayed on the user’s screen as the search results.

**Side effects:** N/A

**3.2.10 Function: Browse DVD**

**Description:** Allows the user to browse categories of movies for movies that they may be interested in. Movies have a short description and the option to be added to the user’s queue when displayed. Categories include Action, Comedy, Classics, Drama, Foreign, and Horror.

**Inputs:** movie data.

**Source:**  A database containing all movie data on the site.

**Outputs:** Movies displayed by category.

**Destination:** The user’s web browser and screen.

**Action:** The site shall display the six movie categories, with each category listing movies that belong to it.

**Requirements:** A database containing all movie data on the site.

**Pre-condition:** The Browse function is available to the user.

**Post-condition:** The movie categories are listed on the user’s screen as described in 2.2.

**Side effects:** N/A

**3.3 Performance Requirements**

The system shall load a page in no more than 5 seconds when a user’s internet speed is over 1mb/s.

The system shall respond to a search query in no more than 10 seconds when a user’s internet speed is over 1mb/s.

The site should be able to handle up to 10,000 simultaneous users.

**3.4 Logical Database Requirements**

The data contained in the database servers must be accessible within the response time constraints listed in 3.3. In order to do this, the server hardware must be readily available to quickly handle queries into the databases whenever a user makes a request.

**3.5 Design Constraints**

The software should keep track of and record all subscription purchases in order to keep track of revenue for various financial and legal purposes.

**3.6 Software System Attributes**

**Reliability**

The servers should have a 0.001% chance of data corruption.

**Availability**

The system should be available 99.5% of the time to 95% of users between the hours of 5AM-12AM.

**Security**

A user’s data should not be accessible by other users.

**Maintainability**

The software should be designed in such a way that it can easily be updated during the hours of 12AM-5AM.

**Compatibility**

The site should be compatible with browsers last updated in 2015 or later to ensure that 90% of users will not experience problems using the site.