**Team**:  Paul Boschert, Teresa Creech

**Title**: Tardis-box

**Project Summary**:

Tardis-box is a no-charge media distribution system. It will allow a potential customer view and sort the available media and, if desired, create a user account using a valid email address. The user may, delete this account, if all borrowed media has been checked in.

A registered user may view and sort the available media with the option to check-out an item. If the requested media is unavailable, the request will be waitlisted to be checked out to the user as soon as it is available.

An administrator may view and sort the media. Only an admin can add and delete media. An administrator can delete a user account, if necessary and can handle overdue items.
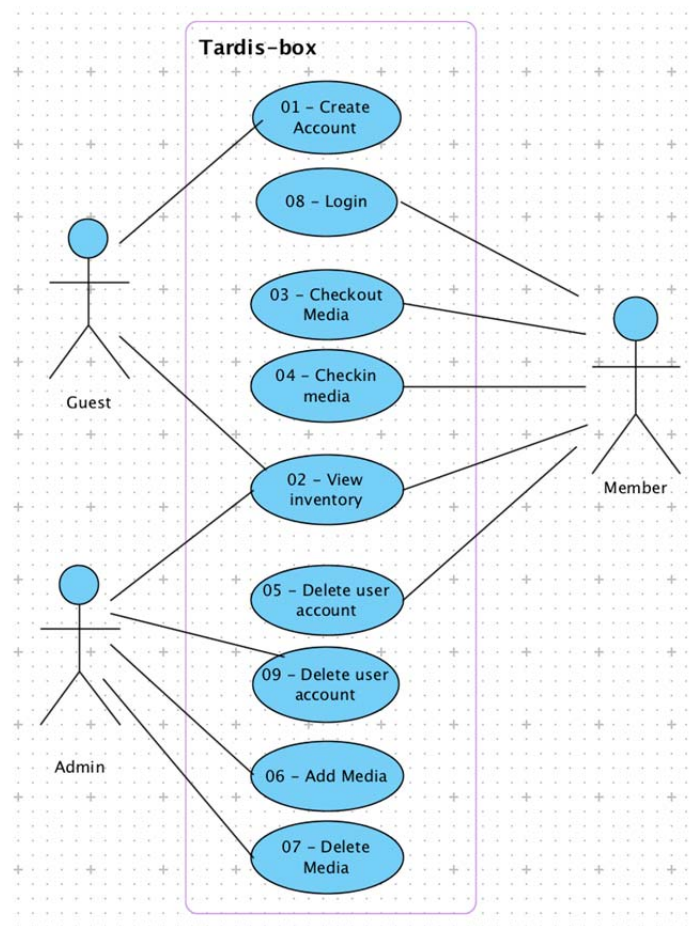
**Project Requirements:**

| Business Requirements | | | | |
|---|---|---|---|---|
| **ID** | **Requirement** | **Topic Area** | **User** | **Priority** |
| **BR-01** | User account usernames must be a valid e-mail address. | Accounts | Registered User | Medium |
| **BR-02** | Only admin can add/delete media items. | Database | Admin | Medium |
| **BR-03** | Potential customers can view available media without having to create an account or login. | Database | Unregistered User | Critical |

## User Requirements

| ID | Requirement | Topic Area | User | Priority |
|---|---|---|---|---|
| **UR-01** | Unregistered user can request and create their own account | Accounts | Unregistered User | Critical |
| **UR-02** | Any user can view a help/contact page | User Interface | All | Low |
| **UR-03** | Any user can view media list | Database | All | Critical |
| **UR-04** | A user can remove their account if certain conditions are met (e.g. no overdue items). | Accounts | Registered User | Medium |
| **UR-05** | A user can make a request to checkout media. | Database | Registered User | Critical |
| **UR-06** | A user can check-in media. | Database | Registered User | Critical |
| **UR-07** | An admin can delete an account | Accounts | Admin | Medium |
| **UR-08** | An admin can handle overdue items | Accounts | Admin | Medium |

## Functional Requirements

| ID | Requirement | Topic Area | User | Priority |
|---|---|---|---|---|
| **FR-01** | There must be a display of the media that shows the title and an associated picture. | Database | All | Critical |
| **FR-02** | There must be a way to view details of a particular media item (besides just a picture and title) that shows a short description and possibly other details e.g. IMDB data). | Database | All | Medium |
| **FR-03** | There must be a way to sort the media for display to the user. | Database | All | Medium |
| **FR-04** | There must be a way to put media on a waitlist, if necessary. | Accounts | Admin | Medium |
| **FR-05** | There must be a way to remove item from waitlist (and check it out to a requestor) | | Admin | Medium |

| Non-Functional Requirements | | | | |
|---|---|---|---|---|
| ID | Requirement | Topic Area | User | Priority |
| **NR-01** | Time to cycle from one page media titles to the next must be reasonable. | User Interface | All | Medium |
| **NR-02** | User credentials must be encrypted. | Accounts | Registered user | Low |
| **NR-03** | Web pages need to be clearly worded and easy to navigate. | User Interface | All | Medium |

**Users and Tasks:**

| Use Case ID: | UC-01 | |
|---:|:---|:---|
| Use Case Name: | Create Account | |
| Description: | Allow a guest to create a user account, which will allow member access to the database of items to check out. | |
| | | |
| Actor: | Guest | |
| Pre-conditions: | Member database must exist | |
| Post-conditions: | Guest becomes a Member | |
| Frequency of Use: | Unknown, potentially 100% of guests | |
| Flow of Events: | **Actor Action** | **System Response** |
| | 1. Guest visits homepage and clicks on "create account" button | Accesses the create account page |
| | 2. Guest enters first and last names, a valid email-id, enters a password, then presses "ok" | System verifies that the email is valid, creates entry in the member database containing this information. System issues an "account created" message |
| Variations: | Guest enters an invalid email-id. System issues error message then clears the create account page. | |
| Notes & Issues: | | |
| Developer Notes: | | |


| Use Case ID: | UC-02 | |
|---:|:---|:---|
| Use Case Name: | View Media | |
| Description: | Allow a guest or member to access the item database. This will allow them to scroll through and sort (by item number, name, or description) | |
| | | |
| Actor: | Guest, Member | |
| Pre-conditions: | Database exists with ability to sort and scroll, Guest/Member has accessed the View Media page | |
| Post-conditions: | Database is scrolled and/or sorted | |
| Frequency of Use: | Arbitrary, depends on guest/member needs | |
| Flow of Events: | **Actor Action** | **System Response** |
| | 1. Uses scroll bar | Moves list up or down, depending on the direction user is moving scroll bar |
| | 2. Sort by clicking one of the column headers (once for ascending order, twice for descending order) | Sorts the list using the selected column, either in ascending or descending order |

| | | | as indicated by the number of clicks. |
|---|---|---|---|
| | 3. | | |
| | 4. | | |
| | 5. | | |
| **Variations:** | | | |
| **Notes & Issues:** | | | |
| **Developer Notes:** | | | |


| Use Case ID: | UC-03 | | |
|---|---|---|---|
| **Use Case Name:** | Check-out Media | | |
| **Description:** | Member may borrow an item if it is available (not already checked-out). | | |
| | | | |
| **Actor:** | Member | | |
| **Pre-conditions:** | Member is signed in and has accessed the View Media page | | |
| **Post-conditions:** | Item is either checked-out to the member or has been placed on a waitlist for the member. | | |
| **Frequency of Use:** | Arbitrary, estimated at least once per member login. | | |
| **Flow of Events:** | | **Actor Action** | **System Response** |
| | 1. | Enters an item number into the check-out field, Clicks OK button | Checks to see if the item is available. If it is, the item is checked-out to the user. If not available, system issues message asking user if he wishes to add the item to the waitlist. |
| | 2. | User selects yes | Item is added to waitlist and system issues confirmation message. |
| | 3. | User selects no | System removes message |
| | 4. | | |
| | 5. | | |
| **Variations:** | Issue error message if the item number is invalid. | | |
| **Notes & Issues:** | | | |
| **Developer Notes:** | | | |


| Use Case ID: | UC-04 |
|---|---|
| **Use Case Name:** | Check-in Media |
| **Description:** | Member returns an item which had been borrowed |

| | | |
|---|---|---|
| | | |
| **Actor:** | Member | |
| **Pre-conditions:** | The item must already be checked-out to the member, member is logged in and has accessed the view items page | |
| **Post-conditions:** | The item is available for check-out | |
| **Frequency of Use:** | Done, at some point, for every item borrowed. | |
| **Flow of Events:** | | **Actor Action** | **System Response** |

| | | **Actor Action** | **System Response** |
|---|---|---|---|
| | 1. | Enters item number in the check-in field | System makes item available for check-out and removes it from the member's account. Issues a "check-in successful" message |
| | 2. | | |
| | 3. | | |
| | 4. | | |
| | 5. | | |

| | |
|---|---|
| **Variations:** | Members enters an invalid item number (either already available in the database or the item does not exist. An appropriate error message is displayed and the check-in field is cleared. |
| **Notes & Issues:** | |
| **Developer Notes:** | |

| | |
|---|---|
| **Use Case ID:** | **UC-05** |
| **Use Case Name:** | Delete User |
| **Description:** | Member may delete account as long as all items have been checked-in |
| | |
| **Actor:** | Member |
| **Pre-conditions:** | Must be on login page |
| **Post-conditions:** | Account no longer exists |
| **Frequency of Use:** | |
| **Flow of Events:** | |

| | | **Actor Action** | **System Response** |
|---|---|---|---|
| | 1. | Enters email-id and password | Verifies information |
| | 2. | Clicks on Delete button | Verifies that no items are checked-out, removes account and issues confirmation email |
| | 3. | | |
| | 4. | | |
| | 5. | | |

| | |
|---|---|
| **Variations:** | Email-id and/or Password invalid – system issues error message and clears screen<br>Items checked out – system issues message instructing Member to return items before account can be deleted |
| **Notes & Issues:** | |
| **Developer Notes:** | |

<br>

| **Use Case ID:** | **UC-06** | | |
|---|---|---|---|
| **Use Case Name:** | Add media | | |
| **Description:** | A new media item can be added to the system. | | |
| | | | |
| **Actor:** | Admin | | |
| **Pre-conditions:** | An admin must be logged in and the media item must not already exist. | | |
| **Post-conditions:** | The new media item was added and is available for checkout. | | |
| **Frequency of Use:** | Arbitrary, anytime an admin needs to add media. Est: weekly | | |
| **Flow of Events:** | | **Actor Action** | **System Response** |
| | 1. | Log in to Tardis-box | User credentials are verified |
| | 2. | Click on administrator panel | Administrative options are displayed |
| | 3. | Click on media sub-panel | A list of media actions is displayed |
| | 4. | Click on add media | The media entry form is displayed |
| | 5. | Enter media details; click submit | The new media item is added |
| **Variations:** | | | |
| **Notes & Issues:** | | | |
| **Developer Notes:** | | | |

<br>

| **Use Case ID:** | **UC-07** |
|---|---|
| **Use Case Name:** | Delete media |
| **Description:** | A media item needs to be removed from the system. |
| | |
| **Actor:** | Admin |
| **Pre-conditions:** | An admin must be logged in, and the media item must exist. |
| **Post-conditions:** | The media item deleted no longer exists in the system. |
| **Frequency of Use:** | Arbitrary, anytime an admin needs to delete media. Est: weekly |

| Flow of Events: | | Actor Action | System Response |
|---|---|---|---|
| | 1. | Log into Tardis-box | User credentials are verified |
| | 2. | Click on administrator panel | Administrative options are displayed |
| | 3. | Click on media sub-panel | A list of media actions is displayed |
| | 4. | Click on delete media | A list of available media items is displayed |
| | 5. | Select the media item; click delete | The media item is deleted |
| Variations: | | | |
| Notes & Issues: | | | |
| Developer Notes: | | | |

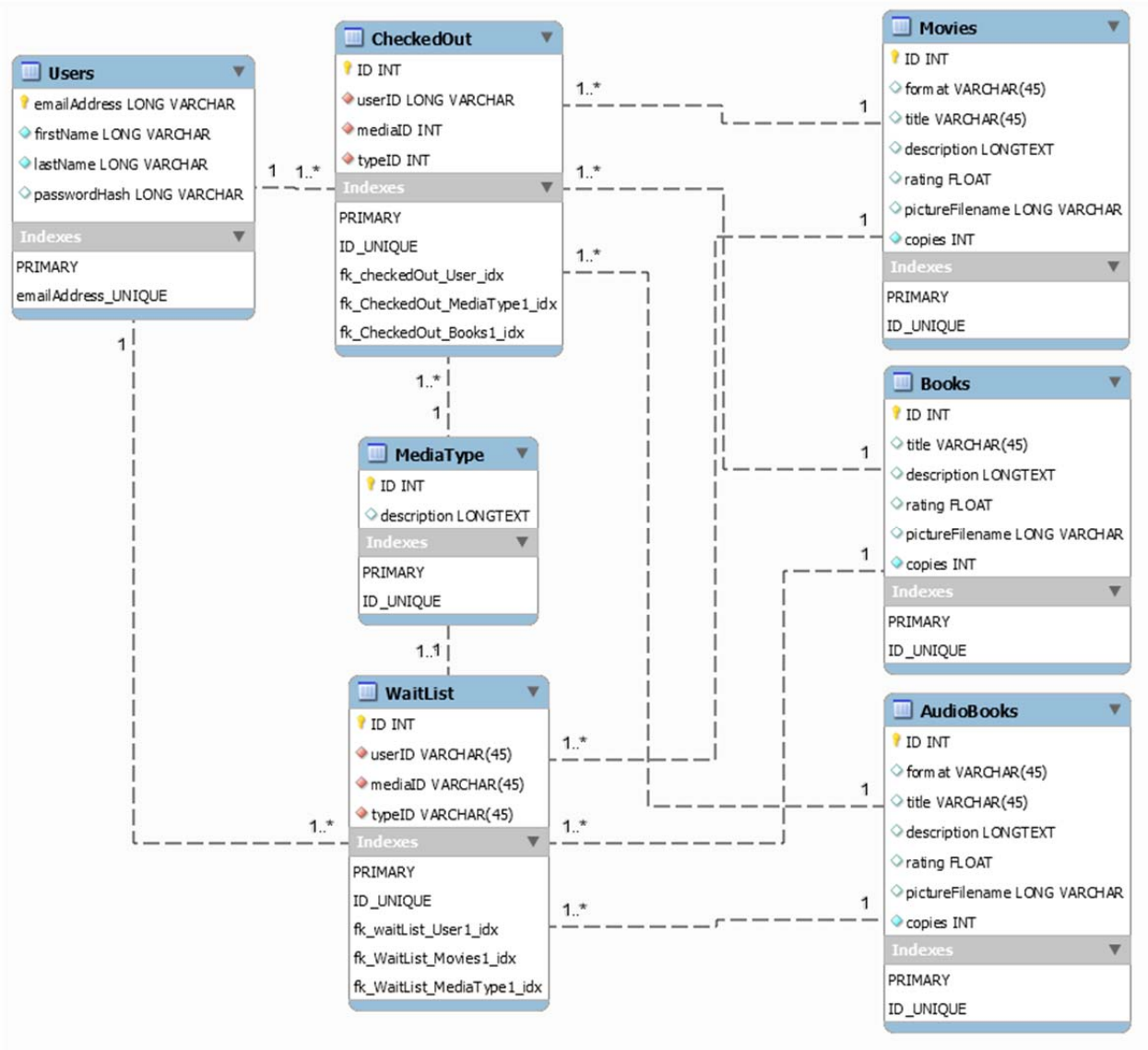| Use Case ID: | UC-08 | | |
|---|---|---|---|
| Use Case Name: | Member Login | | |
| Description: | A member may login to account | | |
| | | | |
| Actor: | Member | | |
| Pre-conditions: | Member is on login page | | |
| Post-conditions: | Member is logged in | | |
| Frequency of Use: | | | |
| Flow of Events: | | Actor Action | System Response |
| | 1. | Enter email-id and password, clicks on Login button | Verifies information, logs user in, and goes to view items page |
| | 2. | | |
| | 3. | | |
| | 4. | | |
| | 5. | | |
| Variations: | Email-id and/or Password invalid – system issues error message and clears screen | | |
| Notes & Issues: | | | |
| Developer Notes: | | | |

| Use Case ID: | UC-09 |
|---|---|
| Use Case Name: | Delete user account |
| Description: | Administrators can remove user accounts |

| | | | |
|---|---|---|---|
| Actor: | Admin | | |
| **Pre-conditions:** | The admin must be logged in and the user must exist. | | |
| **Post-conditions:** | The deleted user no longer exists. | | |
| **Frequency of Use:** | Arbitrary, anytime a user needs to be removed by an administrator. Est: monthly. | | |
| **Flow of Events:** | | **Actor Action** | **System Response** |
| | 1. | Log in to Tardis-box | User credentials are verified |
| | 2. | Click on administrator panel | Administrative options are displayed |
| | 3. | Click on user sub-panel | A list of users is displayed |
| | 4. | Select a user | The user is highlighted |
| | 5. | Click the remove icon | The user is removed |
| **Variations:** | | | |
| **Notes & Issues:** | | | |
| **Developer Notes:** | | | |

**Activity Diagram:**
TODO

**Data Storage:**



**Users**
- 🔑 emailAddress LONG VARCHAR
- 🔷 firstName LONG VARCHAR
- 🔷 lastName LONG VARCHAR
- 🔷 passwordHash LONG VARCHAR

Indexes
- PRIMARY
- emailAddress_UNIQUE

**CheckedOut**
- 🔑 ID INT
- 🔶 userID LONG VARCHAR
- 🔶 mediaID INT
- 🔶 typeID INT

Indexes
- PRIMARY
- ID_UNIQUE
- fk_checkedOut_User_idx
- fk_CheckedOut_MediaType1_idx
- fk_CheckedOut_Books1_idx

**Movies**
- 🔑 ID INT
- 🔷 format VARCHAR(45)
- 🔷 title VARCHAR(45)
- 🔷 description LONGTEXT
- 🔷 rating FLOAT
- 🔷 pictureFilename LONG VARCHAR
- 🔷 copies INT

Indexes
- PRIMARY
- ID_UNIQUE

**MediaType**
- 🔑 ID INT
- 🔷 description LONGTEXT

Indexes
- PRIMARY
- ID_UNIQUE

**Books**
- 🔑 ID INT
- 🔷 title VARCHAR(45)
- 🔷 description LONGTEXT
- 🔷 rating FLOAT
- 🔷 pictureFilename LONG VARCHAR
- 🔷 copies INT

Indexes
- PRIMARY
- ID_UNIQUE

**WaitList**
- 🔑 ID INT
- 🔶 userID VARCHAR(45)
- 🔶 mediaID VARCHAR(45)
- 🔑 typeID VARCHAR(45)

Indexes
- PRIMARY
- ID_UNIQUE
- fk_waitList_User1_idx
- fk_WaitList_Movies1_idx
- fk_WaitList_MediaType1_idx

**AudioBooks**
- 🔑 ID INT
- 🔷 format VARCHAR(45)
- 🔷 title VARCHAR(45)
- 🔷 description LONGTEXT
- 🔷 rating FLOAT
- 🔷 pictureFilename LONG VARCHAR
- 🔷 copies INT

Indexes
- PRIMARY
- ID_UNIQUE

**UI Mockups:**
==TODO==

**User Interactions:**
==TODO==

**Class Diagram:**

==TODO==