

Spoiled Tomatillos Use Cases

Yiwen Dong, Diana Regalbuto, Fiona Tran,
Jiameng (Moe) Wang, Xi (Erin) Zhang

Team 24

April 2018

*Use cases are arranged by feature then by completion status.

Login and Sign Up Features

Use Case:	Sign Up for a Membership COMPLETED
Primary Actor:	Guest
Goal in Context:	The guest is not a member of the site and wants to sign up for an account.
Preconditions:	The user is not logged in to the site and does not have an account.
Trigger:	The user selects the “Sign In/Sign Up” option at the top of the site.
Scenario:	<ul style="list-style-type: none">· The user is prompted with two options: to sign in with an existing account or sign up for a new account.· The user selects “Sign Up”· The user is lead to a form that contains required fields:<ul style="list-style-type: none">○ First and Last Name○ Email○ Password· When the form is filled correctly, the user has successfully created an account.
Exceptions:	<ul style="list-style-type: none">· If the user misses a field or fills in a field incorrectly, an error message will pop up that will prompt the user to fix the mistakes and submit again.
Priority:	This case is required to allow members to join the site.
When available:	This should be available when the site is first released.
Channel to actor:	The system recognizes the actor is not a member of the site and provides limited feature to the actor. The actor will communicate through the web interface.

Use Case:	Login to Account COMPLETED
Primary Actor:	User
Goal in Context:	A user that has a membership with the website should be able to log into their account.
Preconditions:	The user is not currently logged in.
Trigger:	The user selects the “Sign In/Sign Up” option at the top of the site.
Scenario:	<ul style="list-style-type: none"> · The user is prompted with two options: to sign in with an existing account or sign up for a new account. · The user selects “Sign in” · The user is prompted to complete the following fields: <ul style="list-style-type: none"> ○ Username ○ Password · When the information is filled correctly, the user has successfully logged into the account.
Exceptions:	<ul style="list-style-type: none"> · If the user fills in a field incorrectly, an error message will pop up that will prompt the user to fix the mistakes and submit again.
Priority:	This case is required to allow members to log into the site.
When available:	This should be available when the site is first released.
Channel to actor:	The actor will communicate through the web interface.

Use Case:	Edit Profile Page NOT STARTED
Primary Actor:	User
Goal in Context:	Users are able to edit what information they want to display on their profile page.
Preconditions:	<ul style="list-style-type: none"> - A user has signed up for a user account. - A user has logged in to the system. - A user is currently on the view my profile page.
Trigger:	A user clicks on the edit profile button.
Scenario:	<ol style="list-style-type: none"> 1. This use cases starts when a user clicks on the edit profile button. 2. A user can click on the avator to upload a new profile picture. 3. A user can modify their personal information stored on the system. 4. A user can click on the cancel button to discard the changes they have made. 5. A user can click the save button to store their updated information. 6. The use case ends when the updated information is stored to the database.
Exceptions:	<p>V1: A user wants to edit their username.</p> <ul style="list-style-type: none"> - Username is setup when they signed up for the system. Username that stored in the system cannot be changed. However, they can change the name they want to display. <p>V2: Lost internet connection or computer breakdown while in edit mode.</p> <ul style="list-style-type: none"> - User will lose the information they have entered. <p>V3: User opens two windows to edit their profile.</p> <ul style="list-style-type: none"> - The change occurred later will be stored.

Use Case:	User forgot password NOT STARTED
Primary Actor:	User, Moderator, Admin
Goal in Context:	Reset password for the primary actor
Preconditions:	Primary already have an account with us.
Trigger:	Primary actor cannot login due to forgotten password
Scenario:	<ol style="list-style-type: none"> 1. Primary actor navigate to reset password link right under the login form. 2. Primary actor enters their email into text field and submit. 3. In their email, click the link provided navigate that navigate them to a 1 time link to reset password. 4. On our website, enter their new password 5. Enter their password again in another text box. 6. Submit their new password. 7. Wait for system to tell them success and redirect them to login page.
Exceptions:	<ol style="list-style-type: none"> 1. Email they provided was invalid - retry from step 2. 2. Their new password is not strong enough - retry from step 4.
Priority:	This is a necessary feature for this program
When available:	First release

Movie Features

Use Case:	View a Movie Page COMPLETED
Primary Actor:	Guest
Goal in Context:	A guest is not signed into the website. Guests are not members of the site, but are still given the capability to view movie pages and ratings.
Preconditions:	The user is not logged in to the site.
Trigger:	The user uses the search bar to find a movie and selects the desired movie.
Scenario:	<ul style="list-style-type: none">· When the page loads, the user is able to see a page with the title of the movie and an image pertaining to the movie.· The user is able to scroll through the page to view reviews from critics and site users.<ul style="list-style-type: none">○ The ratings the user will see from site users will be randomly generated.· The user is not able to view how friends have rated the movie, write a review, or make a recommendation.
Exceptions:	<ul style="list-style-type: none">· If the user selects a button to 'Write a Review' the guest is prompted to log in or sign up for an account.
Priority:	This use case is the baseline functionality for the site. It is a high priority that this feature is functional.
When available:	This should be available when the site is first released.
Channel to actor:	The system recognizes the actor is not a member of the site and provides limited features to the actor. The actor will communicate through the web interface. There will not be a second actor in this case.

Use Case:	Search for a movie COMPLETED
Primary Actor:	User
Goal in Context:	Find a movie depending on user constraints.
Preconditions:	User with a movie that they want to search and have reached the search page.
Trigger:	User wants to be updated on their friends' activities
Scenario:	<ol style="list-style-type: none"> 1. User enter the search criteria into our website. 2. Select the sorting option for the search "new"/ "recommended"/ "all" 3. Press search button and look at all the movies that fits and sorted by the critia. 4. Scrolling down the user should be able to see more movies slightly less relevant to their searches.
Exceptions:	<ol style="list-style-type: none"> 1. No more relevant results - will no longer be able to scroll and a small prompt to search again.
Priority:	This is a necessary feature for this program
When available:	First release
Channel to actor:	Web interface allowing users to explore the system.

Use Case:	Movie Recommendations COMPLETED
Primary Actor:	User
Goal in Context:	Users can send recommended movies to other users.
Preconditions:	User has to be logged in and both users must exist.
Trigger:	User clicks recommend.
Scenario:	<ol style="list-style-type: none"> 1. User is logged in to site. 2. User goes to the movie page and clicks 'Send Recommendation'
Priority:	High priority
When available:	Phase 1 Prototype
Secondary Actor:	Receiving user
Open Issues:	You must know the name of the user to recommend to them. In the future, a user search should be implemented.

Use Case:	Adding new movies NOT STARTED
Primary Actor:	Administrator
Goal in Context:	Import a new movie that users can search for, rate, and get recommended to.
Preconditions:	An existing movie that is in IMDB but not in our systems.
Trigger:	Movie gets added to IMDB
Scenario:	<ol style="list-style-type: none"> 1. Admin logs into website 2. Admin navigates to a Admin only page by clicking link 3. Enter the IMDB url to import to the import form 4. Gets notified on success
Exceptions:	<ol style="list-style-type: none"> 1. Used wrong login - back to step 1 of this 2. Given IMDB url was wrong - display error message and stay at admin only page 3. The movie was already in database - display error message and stay at admin only page
Priority:	This is a necessary feature for this program
When available:	First release
Channel to actor:	Web interface allowing administrator to communicate with system

External Site Features

Use Case:	Go to external site COMPLETED
Primary Actor:	User
Goal in Context:	User can click on a link in the webpage and be navigated to an external site
Preconditions:	<ul style="list-style-type: none">- User opens a web page that have external link(s).
Trigger:	A user click on the a link to the external site
Scenario:	<ol style="list-style-type: none">1. User sees a link to a content she/he is interested2. User clicks on the link3. The system opens the link4. User gets navigated to the linked web page
Exceptions:	N/A

Review Features

Use Case:	Write a Review COMPLETED
Primary Actor:	User
Goal in Context:	User can post reviews on movies they have watched and share their interests and perspectives online.
Preconditions:	<ul style="list-style-type: none">- User is logged in to the system.- User can only post reviews on a movie that exists on the platform.
Trigger:	A user clicks write a review on a movie page.
Scenario:	<ol style="list-style-type: none">1. The use case starts when a user clicks the write a review button on a movie page.2. User writes the review.3. User saves or gives up changes.<ol style="list-style-type: none">a. User clicks save button to publish the review.b. User clicks cancel button to discard any changes they make.4. User is led back to the movie page. The new review can be seen only by the user if they have posted it.5. Review is under review by the moderator. It can only be seen by everyone once the moderator has finished review.
Exceptions:	N/A

Use Case:	Delete Own Review COMPLETED
Primary Actor:	User
Goal in Context:	User can delete movie reviews that they do not wish to be seen.
Preconditions:	<ul style="list-style-type: none"> - User is logged in to the system. - User has previously posted a review. - User without any admin advantage can only delete their own review.
Trigger:	A user clicks the delete a review button.
Scenario:	<ol style="list-style-type: none"> 1. The user case can start at two locations-either on the movie page where they have posted review on, or the view my review page. 2. User sees the review he/she has posted. 3. User clicks delete button to delete review. 4. A pop-up box shows up to confirm the action in order to prevent any user error. 5. User confirms to delete a review. 6. The system deletes the review. 7. The review is deleted. The review cannot be seen on the movie page nor their review page.
Exceptions:	<p>V1: User deletes a review that has already been deleted.</p> <ul style="list-style-type: none"> - This may be due to a delay in the system. But the review is deleted and will no longer be stored on the system either way.

Use Case:	Open My Reviews COMPLETED
Primary Actor:	User
Goal in Context:	User can view the reviews they have previously posted.
Preconditions:	<ul style="list-style-type: none"> - User is logged in with correct credentials. - User has previously written a review. Otherwise, there will not be any information displayed on the page.
Trigger:	A user clicks View Review tabs.
Scenario:	<ol style="list-style-type: none"> 1. The use case starts when a user wants to view their reviews. 2. User can either go to the movie page where they have posted review under to see their review. 3. User can also go to their personal home page to view previous posts on movies.
Exceptions:	N/A

Use Case:	Approve a Review NOT STARTED
Primary Actor:	Moderator
Goal in Context:	The moderator will examine the content of the review written by a user. The review will be posted to our system if and only if a moderator has approved it.
Preconditions:	<ul style="list-style-type: none"> - A user has posted a review - A movie page under which the review is posted exists - A moderator is an approved admin either from the client or the technical team
Trigger:	A review is posted by a user
Scenario:	<ol style="list-style-type: none"> 1. The use case starts when a review is posted by a user. 2. The system sends a notification of the review to the moderator. 3. The moderator logs on to the system. 4. The moderator receives a notification of the unread review. 5. The moderator examines the review content and decides if the review content is appropriate. 6. The review is stored on the system once the moderator has approved it. 7. The system displays the user's review. 8. This use case ends when the system sends a notification to the user noting that the review has been approved.
Exceptions:	<p>V1: User deletes a review when still not approved.</p> <ul style="list-style-type: none"> - The review is deleted from the moderator's review list. <p>V2: An approved review is edited again by the user.</p> <ul style="list-style-type: none"> - The edited review is treated as a new post and needs to be reviewed again before being displayed on the system.

Use Case:	Reject a Review NOT STARTED
Primary Actor:	Moderator
Goal in Context:	Moderator rejects reviews that contains profanity, racial comments of any kind, violent speech, or any other comments that contain threats or danger to any other individual. Moderator can also reject a review posted by a suspicious user.
Preconditions:	<ul style="list-style-type: none"> - A user has posted a review - A movie page under which the review is posted exists - A moderator is an approved admin either from the client or the technical team
Trigger:	A review is posted by a user.
Scenario:	<ol style="list-style-type: none"> 1. The use case starts when a review is posted by a user. 2. The system sends a notification of the review to the moderator. 3. The moderator logs on to the system. 4. The moderator receives a notification of the unread review. 5. The moderator finds inappropriate content in the review. 6. The moderator rejects the review. 7. The system sends the user a notification that the content has been deleted. 8. This use case ends when the system deletes the record of the review.
Exceptions:	<p>V1: User deletes a review when still not approved.</p> <ul style="list-style-type: none"> - The review is deleted from the moderator's review list. <p>V2: An approved review is edited again by the user.</p> <ul style="list-style-type: none"> - The edited review is treated as a new post and needs to be reviewed again before being displayed on the system.

User-to-User Interaction Features

Use Case:	Add/Delete a friend COMPLETED
Primary Actor:	User
Goal in Context:	Add or delete a friend
Preconditions:	User is logged in and is on another user's page that they wish to add or remove as friend.
Trigger:	User wants to befriend or unfriend another user.
Scenario:	<ol style="list-style-type: none">1. If the user is friends with another user, when visiting their profile should show a unfriend button.2. User should click unfriend button and the unfriend button should turn into a friend button now they are no longer friends.3. If the user is not yet friends with this other user, when visiting their profile should show a friend button.4. User should click friend button and the friend button should go away and informing user that the other needs to confirm the friend request.
Exceptions:	<ol style="list-style-type: none">1. The user trying to befriend/unfriend was banned while the user was trying to friend or unfriend them - refresh the page displaying the appropriate error message.
Priority:	This is a necessary feature for this program
When available:	First release
Channel to actor:	Web interface allowing users to explore the system.

Use Case:	Accept or decline friend requests COMPLETED
Primary Actor:	User
Goal in Context:	Accept or decline friend requests.
Preconditions:	User is logged in and is on their own friends list.
Trigger:	User wants to accept or decline a friend request.
Scenario:	<ol style="list-style-type: none"> 1. In the list of friend should show in a separate section people who sent requests. Buttons should exist next to them with accept or reject. 2. User click accept or reject. 3. The user disappear from the separate section and will now show up alongside the rest of the person's friends if accepted or not if rejected.
Exceptions:	<ol style="list-style-type: none"> 1. User that sent the request was banned - user will disappear and not show up in friend's list even if accepted.
Priority:	This is a necessary feature for this program
When available:	First release
Channel to actor:	Web interface allowing users to explore the system.

Use Case:	Search for a Specific User COMPLETED
Primary Actor:	User
Goal in Context:	A user will be able to find another user by searching their name.
Preconditions:	The user is logged in to the account.
Trigger:	The user enters a name into the search box and hits enter.
Scenario:	<ul style="list-style-type: none"> · After hitting enter, the user is brought to a list of names that match the search query. · The user will be able to distinguish between the different users through the profile pictures or whether or not they are friends. <ul style="list-style-type: none"> ○ If there is an option to add the person, it dictates that they are not already friends. · The user will be able to select a user and it will bring them to that user's profile.
Exceptions:	If there are no users with that name, there will be an error message similar to: "Results not found."
Priority:	This case is necessary to allow users to add friends.
When available:	Should be available with initial release.
Channel to actor:	The actor will communicate through the web interface.
Secondary Actor:	Other actors will not be involved with this scenario.
Channels to Secondary Actors:	None
Open Issues:	User search should provide suggestions and partial results.

Use Case:	Scroll through home page to see recent activity from friends COMPLETED
Primary Actor:	User
Goal in Context:	Get an overview of activities of friends.
Preconditions:	User with an account.
Trigger:	User wants to be updated on their friends' activities
Scenario:	<ol style="list-style-type: none"> 1. User logs into website 2. User click on the home link. 3. Scroll down until there's no more feed to display.
Exceptions:	<ol style="list-style-type: none"> 1. Used wrong login - back to step 1 of this
Priority:	This is a necessary feature for this program
When available:	First release
Channel to actor:	Web interface allowing users to explore the system.

Use Case:	View Another User's Profile COMPLETED
Primary Actor:	User
Goal in Context:	A user will be able to view another user's profile and any reviews that user has posted.
Preconditions:	The user is logged in to the account and has selected a profile to view.
Trigger:	The user selects the profile to view by clicking the profile name, most likely from searching.
Scenario:	<ul style="list-style-type: none"> · When viewing the profile, the user is able to see a profile picture and the full name of the user. · By scrolling, the user is able to see any movie reviews that person has posted – in a feed format. · The reviews are formatted to have a picture of the movie on the left, a movie title and rating on the top, and any comments for the review
Priority:	This case is necessary to allow users to add friends.
When available:	Should be available with initial release.
Channel to actor:	The actor will communicate through the web interface.
Secondary Actor:	Other actors will not be involved with this scenario.
Channels to Secondary Actors:	None
Open Issues:	Will all this information be viewable when the users are not friends? Will users be able to block other users so they can't see their profile?

Use Case:	Get Suggestions for Adding Friends NOT STARTED
Primary Actor:	User
Goal in Context:	A user will be able to see a list of users they may know based on mutual friends or contacts through email.
Preconditions:	The user is logged in to the account.
Trigger:	The user selects the “People You May Know” option on their profile.
Scenario:	<ul style="list-style-type: none"> · The user will be able to scroll through a list of site users. · The list will contain thumbnail profile pictures for these users and their first and last names. · If there are mutual friends, users will be able to see how many there are. i.e. “12 mutual friends” <ul style="list-style-type: none"> ○ By selecting the number, a list will come up showing who these friends are. · Under the names, the user will be able to add the person as a friend by selecting “Add friend” <ul style="list-style-type: none"> ○ Adding a friend only sends a friend request, it does not guarantee a connection. · If the user does not know the person, the suggestion can be removed by selecting the “Remove” button
Priority:	This case is not a high priority/requirement for the product.
When available:	Can be available in a second stage after the initial release.
Channel to actor:	The actor will communicate through the web interface.

Use Case:	Adding new moderator NOT STARTED
Primary Actor:	A moderator
Goal in Context:	Create a new moderator
Preconditions:	A user that is identified to be qualified to be a moderator.
Trigger:	Admin found a user willing to be a moderator.
Scenario:	<ol style="list-style-type: none"> 1. Admin logs into website 2. Admin navigates to a Admin only page by clicking link 3. Enter the username of user to the promote form 4. Gets notified on success
Exceptions:	<ol style="list-style-type: none"> 1. Used wrong login - back to step 1 of this 2. Given username does not exist - display error message and stay at admin only page 3. The user is already admin - display error message and stay at admin only page.
Priority:	This is a necessary feature for this program
When available:	First release
Channel to actor:	Web interface allowing administrator to communicate with system

Use Case:	Ban a User NOT STARTED
Primary Actor:	Administrator
Goal in Context:	Ban user from participating on our website
Preconditions:	A user that is identified to be qualified to be a moderator.
Trigger:	Admin found a user willing to be a moderator.
Scenario:	<ol style="list-style-type: none"> 1. Admin logs into website 2. Admin navigates to a Admin only page by clicking link 3. Enter the username of user to the ban form 4. Gets notified on success
Exceptions:	<ol style="list-style-type: none"> 1. Used wrong login - back to step 1 of this 2. Given username does not exist - display error message and stay at admin only page 3. The user is already banned - display error message and stay at admin only page.
Priority:	This is a necessary feature for this program
When available:	First release
Channel to actor:	Web interface allowing administrator to communicate with system

Group Features

Use Case:	Create a group COMPLETED
Primary Actor:	User
Goal in Context:	A user should be able to successfully make a group.
Preconditions:	The user has an account and is logged in.
Trigger:	A end user clicks on create group button on view group page
Scenario:	<ol style="list-style-type: none">1. user logs into website2. user navigates to a view group page clicking link3. Enter the movie id and invites friends to group

Use Case:	Delete a group COMPLETED
Primary Actor:	User
Goal in Context:	A user should be able to successfully delete an existing group.
Preconditions:	<ol style="list-style-type: none">1. The group exists.2. The user has an account and is logged in
Trigger:	An end user clicks on Delete Group button on View Group page
Scenario:	<ol style="list-style-type: none">1. User logs into website2. User navigates to a view group page by clicking link3. User clicks “Delete Group” button
Exceptions:	If the user is not the creator of the group