

Test Category: System test		Description: This test verifies that a user can create an account and successfully have their information posted to the database.		
Test Case: 3.1.1		Case Name: Account Creation	Version: 1.0	Written By: Cody Rabie
Requirements Fulfilled: (3.1.1.2)		Purpose: This test verifies that a user can create an account through the registration page. Successful account creation requires that the provided username does not already exist in the database and that all required fields (username, password, name, email) are completed.		
Setup Conditions: FlixPicks registration page is open				
Test Case Activity		Pass/Fail	Comments	Expected Result
1	The user clicks the “Register” button on the dashboard (3.1.1.2.1)			Registration page loads with empty fields for the user to input the username, password, name, and email address.
2	The user enters a pre-existing username (3.1.1.2.1)			The user shall be notified that the missing field(s) information is required and will not be able to submit a registration request.
3	The user leaves any of the fields blank (3.1.1.2.1)			The user shall be notified that the missing field(s) information is required and will not be able to submit a registration request.

4	The user fills out all of the form entries and does not pick a pre-existing username (3.1.1.2.1)			The user shall successfully register and their information will be stored in the database as an account.
---	--	--	--	--

Test Category: System test		Description: This test verifies that the user profile settings page and functions display and function properly		
Test Case: 3.1.5		Case Name: User Profile Settings	Version: 1.0	Written By: Cody Rabie
Requirements Fulfilled: (3.1.1.3.2)		Purpose: To ensure the user profile settings page displays correctly. Ensures a user has the options to change their password, view current subscriptions, add subscriptions, and remove subscriptions.		
Setup Conditions: 1. FlixPicks application is open to the Settings page (3.1.1.3.2)				
Test Case Activity		Pass/Fail	Comments	Expected Result
1	The user clicks the “Settings” button on the Userhome page (3.1.1.3.2).			User profile settings page loads and displays options to change password, add subscriptions, and remove subscriptions. Current subscriptions are also displayed on this page. The user’s current username is displayed in the top right header.
2	The user types in the “Change Password” field and presses submit (3.1.1.3.3).			A prompt stating the user password has been successfully changed is displayed, and the user is able to freely access the settings page again.
3	The user types nothing in the “Change Password” field and presses submit (3.1.1.3.3).			An error message stating that the field is empty is displayed.

4	The user clicks the add subscriptions button (3.1.1.3.4).			A text box prompt displays, asking the user for the name of the subscription service they wish to add.
	The user types a valid streaming service that isn't already in their current subscriptions (3.1.1.3.4).			A successfully added subscription message is displayed and the service is added to the users services. The current subscriptions information is updated to include the new service.
	The user instead types in a non valid input (3.1.1.3.4).			An error message stating that this is not a valid streaming service is displayed.
	The user instead types a streaming service that already exists in the list (3.1.1.3.4).			An error message is displayed, and the user is redirected to the settings page.
	The user clicks the remove subscriptions button (3.1.1.3.4).			A text box prompt displays, asking the user for the name of the subscription service they wish to remove.
	The user types a valid streaming service that is already in their current subscriptions (3.1.1.3.4).			A successfully removed subscription message is displayed and the removal is reflected in the current subscriptions information displayed on the page.
	The user instead types non valid input (3.1.1.3.4).			An error message is displayed, and the user is redirected to the settings page.

	The user instead types a streaming service that does not exist in the current subscriptions (3.1.1.3.4).			An error message is displayed, and the user is redirected to the settings page.
--	--	--	--	---

Test Category: System test		Description: This test verifies that the HotPicks media category displays popular movie options to both the guest and registered user UI correctly.		
Test Case: 3.2.1		Case Name: HotPicks Test	Version: 1.0	Written By: Cody Rabie
Requirements Fulfilled: (3.1.2.2)		Purpose: To ensure that the popular movie options are displayed properly in the HotPicks media category. It also ensures that these options can be selected, then loading the mediaInfo page.		
Setup Conditions: <div>1. FlixPicks application is open.</div> <div>2. The FlixPicks guest homepage UI is open.</div> <div>3. The FlixPicks registered user homepage UI is open.</div>				
Test Case Activity		Pass/Fail	Comments	Expected Result
1	The user loads the guest home page.			The HotPicks category displays title and poster information for popular_movies. Each movie is separated into individual tiles that can be selected.
2	The user loads the registered user home page.			The HotPicks category displays title and poster information for popular_movies. Each movie is separated into individual tiles that can be selected.
3	The user selects a tile from the HotPicks category.			<div>The ‘mediaInfo’ page submits a ‘GET’ request to the FlixPicks API to retrieve media information about the selected movie.</div> <div>The mediaInfo page loads with title, plot, release date, genre, and where-to-watch</div>

				information from the selected movie.
--	--	--	--	---