Company   TernOps

Members   Matteo De Angelis Geraldo, Ryan Coones, Tomasz Puzio, Erica Patel, Nolan Tsang

Domain   Celebrity Biographies & their Achievements

Project Overview

Project Name   TernOps

Purpose & Scope   TernOps is a project created by Matteo De Angelis Geraldo, Ryan Coones, Tomasz Puzio, Erica Patel and Nolan Tsang. The project is a catalogue of celebrity biographies, which will help users understand fame and success while preserving stories that resonate with people. It will capture the lives of influential figures in fields like entertainment, sports, and politics, providing a reliable resource for scholars, journalists, and fans. Celebrities serve as role models, inspiring people and greatly shaping public opinion. A catalogue helps track the influence and relevance of certain figures, providing a deeper understanding of various fields or industries.

Introduction

Test Plan Summary

We have a rigorous strategy for the testing of the Celebrity biography catalogue. We have created an in-depth testing suite that tests all major functionalities of the program, such as login/log out, account creation, edit, add, and delete among many other functionalities. By automating these tests using Pythons unittest library, we can efficiently and accurately test the program and every component in it. We took into account clear box, translucent box, and opaque box testing. We used CB testing for testing functions that are account-related, TB testing for any functions that deal with our database (add, edit, delete), and OB for testing functions where we are looking for correct outputs within the system. Our tests cover various scenarios that may come up when being used by an everyday user. By testing edge cases (creating an account with no username, or username that already exists, trying to delete an entry that doesn't exist, etc.), and unexpected use cases, we can swiftly address and fix any errors, and patch any holes or vulnerabilities in the program.

CSCI 2040U - Software Design and Analysis
Phase 3 - Test Plan

Unit Tests

Test ID	Method/Class	Input(s)	Expected Output(s)	Testing Approach	Assigned Team Member
UT-01-CB	register_user()	"John" "abc123"	"Registered"	Automated (unittest)	Tomasz
UT-02-CB	register_user()	"John" "abc123"	"Sorry, the username John is already taken, Please try again.	Automated (unittest)	Tomasz
UT-03-CB	login()	"John" "abc123"	"Logged In"	Automated (unittest)	Tomasz
UT-04-CB	login()	"Bill" "abc123"	"No account found for the username Bill"	Automated (unittest)	Tomasz
UT-05-CB	login()	"John" "123abc"	"Incorrect password"	Automated (unittest)	Tomasz
UT-06-CB	log_out()	null	"Successfully logged out"	Automated (unittest)	Tomasz
UT-07-TB	add_data()	"Eric" "Rapos" "July 20, 1980" "eric_rapos.jpg"	"Successfully added a new entry for Eric Rapos."	Automated (unittest)	Tomasz
UT-08-TB	add_data()	"Eric" "Rapos" "July 20, 1980" "eric_rapos.jpg"	"Entry for Eric Rapos already exists."	Automated (unittest)	Tomasz
UT-09-TB	edit_data()	"first_name" "Eric" "date_of_birth" "July 20, 1970"	"Successfully updated Eric Rapos for "	Automated (unittest)	Tomasz
UT-10-TB	delete_data()	"Eric" "Rapos"	"Successfully deleted the entry for Eric Rapos."	Automated (unittest)	Tomasz

CSCI 2040U - Software Design and Analysis
Phase 3 - Test Plan

Integration Tests

Test ID	Components Involved	Preconditions	Steps	Expected Result	Assigned Team Member
IT-01-CB	`searchCelebrities () & filterCelebrities () & GUI & CSV file`	GUI initialized & celebrity csv is loaded & User can interact with search bar	Enter name into search bar, submit	Search successfully filters and is displayed	Nolan
IT-02-TB	`createEditCelebri tyFrame("Add") & createCelebrityRo w() & GUI`	GUI initialized & celebrity csv loaded & User can press "+" button	Click the "+" button, enter details and save	Addition is successful and appears on display, remains in csv after closure	Nolan
IT-03-TB	`createEditCelebri tyFrame("Edit") & createCelebrityRo w() & GUI & CSV file`	GUI initialized & celebrity csv loaded & User can press the "Edit" button	Click the "Edit" button, edit the details and save	Editing is successful. It appears on display and is saved in csv	Nolan
IT-04-CB	`createFilterFrame () & filterCelebrities () & GUI`	GUI initialized & celebrity csv loaded & User can press filter dropdown	Click the filter dropdown and select filters and press search	Filtering is successful. Displays results based on filter.	Nolan
IT-05-CB	`createCelebrityFr ame() & GUI`	Initialized GUI & Celebrity csv loaded & User can press learn more	Press the "Learn More" button.	Successfully displays all information about celebrity	Nolan
IT-06-TB	`removeCelebrity() & GUI & CSV file`	Initialized GUI & Celebrity csv loaded & User can press delete	Press the "Delete button" and confirm the delete	Successfully deleted from csv and no longer is displayed	Nolan

CSCI 2040U - Software Design and Analysis
Phase 3 - Test Plan

System Tests

Test ID	Scenario	Preconditions	Steps	Expected Outcome	Assigned Team Member
ST-01-TB	Adding a new celebrity to the system	App is running, user can press the "+" button	User presses the "+" button, then they fill in at least the Full name and industry of the celebrity, and press the submit celebrity button	The new celebrity is added to the CSV and is visible in the app's main view. A success message is shown.	Matteo
ST-02-TB	Editing an existing celebrity's data	App is running, user can press the "edit" button, there is an existing celebrity	User presses the "edit" button, then they edit what the need to in the edit celebrity form, and press the edit celebrity button	The celebrity's updated data is saved in the CSV, and the changes are reflected in the app	Matteo
ST-03-CB	Filtering celebrities based on search criteria	App is running, user can press the "filter" button, there are celebrities	User presses the "filter" button, they then select filter criteria (date of birth, genre, etc.)	The display should be a filtered list of celebrities that match the selected criteria	Matteo
ST-04-OB	Displaying detailed celebrity info	App is running, user can press the "filter" button, there is a celebrity	User selects a celebrity from the homepage screen, then they click the "Learn More" button.	The celebrity's full details (biography, achievements, etc.) display in the UI	Matteo

Demonstration of Coverage

Mapping			
Test	Feature	User Story	Risk
UT-01-CB	Register	As a user, I want a dedicated button to create an account/login/sign out so I can log in/out with ease.	Register is a main feature in software that is a high risk and tested thoroughly
UT-02-CB	Register	As a user, I want a dedicated button to create an account/login/sign out so I can log in/out with ease.	
UT-03-CB	Login	As a user, I want to be able to log in, so that I have access to more features of the catalogue, like favouriting. As a user, I want a dedicated button to create an account/login/sign out so I can log in/out with ease.	Login another main feature of software is a high risk that is tested to ensure it is covered
UT-04-CB	Login	As a user, I want to be able to log in, so that I have access to more features of the catalogue, like favouriting. As a user, I want a dedicated button to create an account/login/sign out so I can log in/out with ease.	
UT-05-CB	Login	As a user, I want to be able to log in, so that I have access to more features of the catalogue, like favouriting. As a user, I want a dedicated button to create an account/login/sign out so I can log in/out with ease.	
UT-06-CB	Log out	As a user, I want a dedicated button to create an account/login/sign out so I can log in/out with ease.	Log out is medium risk and tested to not miss a gap in software
UT-07-TB	Add	As a user, I want to be able to add	Add with a main

Mapping			
		new celebrities to the database so the database keeps up with the growth of industries and the emergence of new famous people.	function and high risk, tested to properly add info
UT-08-TB	Add	As a user, I want to be able to add new celebrities to the database so the database keeps up with the growth of industries and the emergence of new famous people.	
UT-09-TB	Edit	As a user, I want to be able to edit the existing information on celebrities, so that the biography of the celebrity is accurate and up to date.	Edit with a main function and high risk, tested to properly edit info
UT-010-TB	Delete	As a user, I want to be able to delete information if it is irrelevant or incorrect, so the information is kept recent and correct.	Delete with a main function and high risk, tested to properly delete info
IT-01-CB	Display Search & Filter	As a user, I want to search celebrities by name, so that I can quickly access the info I need for my research. As a user, I want to be able to filter the celebrity profiles by awards won, industry, etc. so that I can specifically find what I want to find instead of browsing aimlessly.	Search & Filter are another main feature of the software and is a high risk that is tested to ensure it is covered and works
IT-02-TB	Display Added celebrity	As a user, I want to be able to add new celebrities to the database so the database keeps up with the growth of industries and emergence of new famous people.	Add with a main function and high risk, tested to properly add info
IT-03-TB	Display Edited celebrity	As a user, I want to be able to edit the existing information on celebrities, so that the biography of the celebrity is accurate and up to date.	Edit with a main function and high risk, tested to properly edit info
IT-04-CB	Display Filtered celebrity	As a user, I want to be able to filter the celebrity profiles by awards won, industry, etc. so that	Filter is a main feature of the software and is a

Mapping

		I can specifically find what I want to find instead of browsing aimlessly.	high risk that is tested to ensure it is covered and works
IT-05-CB	Display Biography Frame	As a user, I want to have a user-friendly interface so I can easily interact with the application and see the information on a visually appealing medium. As a user, I want to browse celebrities, so that I can find info about celebrities I'm interested in.	The display of celebrity biographies is a must feature of the software and is high risk, thus thoroughly tested
IT-06-TB	Delete celebrity info	As a user, I want to be able to delete information if it is irrelevant or incorrect, so the information is kept recent and correct.	Delete with a main function and high risk, tested to properly delete info
ST-01-TB	Display Added celebrity	As a user, I want to be able to add new celebrities to the database so the database keeps up with growth of industries and emergence of new famous people.	Add with a main function and high risk, tested to properly add info
ST-02-TB	Display Edited celebrity	As a user, I want to be able to edit the existing information on celebrities, so that the biography of the celebrity is accurate and up to date.	Edit with a main function and high risk, tested to properly edit info
ST-03-CB	Display Filtered celebrity	As a user, I want to be able to filter the celebrity profiles by awards won, industry, etc. so that I can specifically find what I want to find instead of browsing aimlessly.	Filter is a main feature of the software and is a high risk that is tested to ensure it is covered and works
ST-04-OB	Display celebrity info	As a user, I want to have a user-friendly interface so I can easily interact with the application and see the information on a visually appealing medium. As a user, I want to browse celebrities, so that I can find info about celebrities I'm interested in.	The display of celebrity info is a must feature of the software and is high risk, thus thoroughly tested

CSCI 2040U - Software Design and Analysis
Phase 3 - Test Plan

Summary

Unit Test	Integration Test	System Test
10 main tests were done on key features of the software (registering, logging in, logging out, adding, editing, and deleting)	6 tests were done to ensure the successful integration of the key features into the software	4 tests were done to ensure the software runs as expected

- All the main must-have features/functions from the first 2 iterations have been tested to make sure of the software's reliability and functionality

Gaps in coverage:
- So far, the third iteration's features/functions have not been added yet, leaving those functions' coverage as a gap
- It will be addressed as they are being implemented and the necessary tests will be made alongside it