Jonas Trepanier CSCI 510

Anirudh Kaluri Group 35

Siddhi Mule

Project 2:

Our Poster:

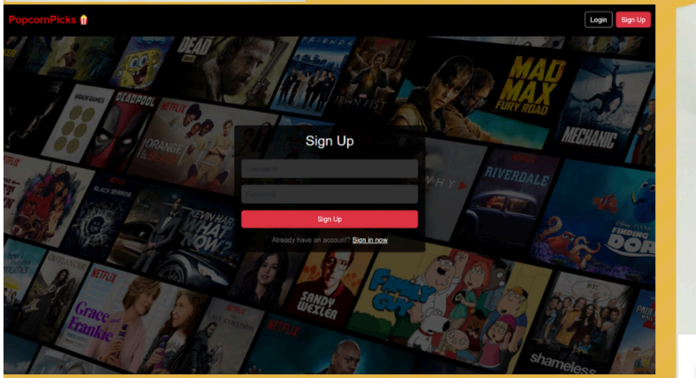
PopcornPicks

Version 2.0

Your one-stop spot for what to watch!

Group 35 - Jonas Trepanier, Anirudh Kaluri, Siddhi Mule

New Profiles!



Sign Up

Username

Password

Already have an account? Sign in now

What's New?

New Infrustructure: Dockerized software, so you can install PopcornPicks with only one command.

Comprehensive Test Coverage: A total of 30 unit test cases covering the new and old services in the backend.

User Profiles: Create an specialized account username and password that is saved in database. Your saved films will be saved to your profile and accessible even after you log out.

Separate Front-End: Added a separate front-end framework using React.js, which improves the overall scalability and maintainability of PopcornPicks.

New Services: New backend services which allow you to add and remove elements from a personalized watchlist and create user tokens for each login session.

```
docker compose up -d --build
```

Single Command Setup!

Repository

https://github.com/CSCI510testerhw1/PopcornPicks

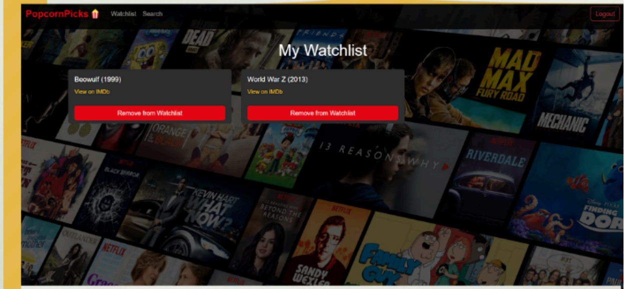
Demo

https://www.youtube.com/watch?v=vaSMBwAT0yw

Introduction

PopcornPicks makes discovering great films effortless. Simple to set up and use, it's the perfect companion for movie enthusiasts looking for their next favorite watch.

Work with a multitude of coding languages, including Python and Javascript, and now features Docker, databases, and a React.js frontend.



My Watchlist

Beowulf (1999)
View on IMDb

World War Z (2013)
View on IMDb

Save movies in new watchlists

What's Still to Do?

Improve Recommendations: Implement more advanced machine learning models in order to provide users with more relevant film recommendations.

Performance Enhancement: Load balancing to ensure robust operations for high traffic.

Streaming Options Integration: Connect recommendations to streaming services information, allowing users to see on what platform their recommended films are playing.

Flexible Recommendation Display: Introduce flexible recommendation displays, allowing users to sort suggestions based on various criteria such as release date or alphabetical order.

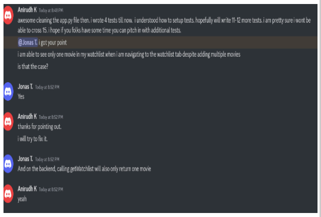
Rating Service: Allow users to rate and review movies themselves and share their thoughts with others.

Our Repository:

https://github.com/CSCI510testerhw1/PopcornPicks

Our Table:

Assesment Points	Self Assesment (Rank 0-3) Sum = 183	Evidence
Workload is spread over the whole team (one team member is often Xtimes more productive than the others…	3	https://github.com/CSCI510testerhw1/PopcornPicks/graphs/contributors?from=9%2F28%2F2024
but nevertheless, here is a track record that everyone is contributing a lot)	Anirudh - 3, Siddhi -3, Jonas -3	https://github.com/CSCI510testerhw1/PopcornPicks/graphs/contributors?from=9%2F28%2F2024
Number of commits	3	https://github.com/CSCI510testerhw1/PopcornPicks/graphs/contributors?from=9%2F28%2F2024
Number of commits: by different people	3	https://github.com/CSCI510testerhw1/PopcornPicks/graphs/contributors?from=9%2F28%2F2024
Issues reports: there are many	3	https://github.com/CSCI510testerhw1/PopcornPicks/issues
Issues are being closed	3	https://github.com/CSCI510testerhw1/PopcornPicks/issues
Docs: doco generated, format not ugly	3	https://github.com/CSCI510testerhw1/PopcornPicks/

		wiki
Docs: what: point descriptions of each class/function (in isolation)	3	https://github.com/CSCI510testerhw1/PopcornPicks/wiki
Docs: how: for common use cases X,Y,Z mini-tutorials showing worked examples on how to do X,Y,Z	3	https://github.com/CSCI510testerhw1/PopcornPicks/blob/master/README.md
Docs: why: docs tell a story, motivate the whole thing, deliver a punchline that makes you want to rush out and use the thing	2	https://github.com/CSCI510testerhw1/PopcornPicks/wiki
Docs: short video, animated, hosted on your repo. That convinces people why they want to work on your code.	3	https://youtu.be/rbT5nZ8c32o
Use of version control tools	3	Used Git
Test cases exist	3	https://github.com/CSCI510testerhw1/PopcornPicks/tree/master/test
Test cases are routinely executed	3	https://github.com/CSCI510testerhw1/PopcornPicks/blob/master/.github/workflows/unittest.yml
Issues are discussed before they are closed	3	See Below:
Chat channel: exists	3	 Anirudh K Today at 8:48 PM awesome cleaning the app.py file then. i wrote 4 tests till now. i understood how to setup tests. hopefully will write 11-12 more tests. i am pretty sure i wont be able to cross 15. i hope if you folks have some time you can pitch in with additional tests. @Jonas T. i got your point i am able to see only one movie in my watchlist when i am navigating to the watchlist tab despite adding multiple movies is that the case? Jonas T. Today at 8:52 PM Yes Anirudh K Today at 8:52 PM thanks for pointing out. i will try to fix it. Jonas T. Today at 8:52 PM And on the backend, calling getWatchlist will also only return one movie Anirudh K Today at 8:52 PM yeah

Test cases: a large proportion of the issues related to handling failing cases.	1	https://github.com/CSCI510testerhw1/PopcornPicks/issues
Evidence that the whole team is using the same tools: everyone can get to all tools and files	3	See Commits in Git
Evidence that the whole team is using the same tools (e.g. config files in the repo, updated by lots of different people)	3	See Commits in Git
Evidence that the whole team is using the same tools (e.g. tutor can ask anyone to share screen, they demonstrate the system running on their computer)	3	See Commits in Git
Evidence that the members of the team are working across multiple places in the code base	3	See Commits in Git
Short release cycles	3	https://github.com/CSCI510testerhw1/PopcornPicks/graphs/code-frequency
The file .gitignore lists what files should not be saved to the repo. See [examples]i(https://github.com/github/gitignore)	3	https://github.com/CSCI510testerhw1/PopcornPicks/blob/master/.gitignore
The file INSTALL.md lists how to install the code	3	https://github.com/CSCI510testerhw1/PopcornPicks/blob/master/INSTALL.md
The file LICENSE.md lists rules of usage for this repo	3	https://github.com/CSCI510testerhw1/PopcornPicks/blob/master/LICENSE
The file CODE-OF-CONDUCT.md lists rules of behavior for this repo; e.g. see example	3	https://github.com/CSCI510testerhw1/PopcornPicks/blob/master/CODE_OF_CONDUCT.md
The file CONTRIBUTING.md lists coding standards and lots of tips on how to extend the system without screwing things up; e.g. see example	3	https://github.com/CSCI510testerhw1/PopcornPicks/blob/master/CONTRIBUTING.md
The file README.md contains all the following	3	https://github.com/CSCI510testerhw1/PopcornPicks/blob/master/README.md
Video	3	https://www.youtube.com/watch?v=vaSMBwAT0yw
DOI badge: exists. To get a Digitial Object Indentifier, regiser the project at Zenodo. DOI badges look like this:	3	https://github.com/CSCI510testerhw1/PopcornPicks/blob/master/README.md
Badges showing your style checkers	3	https://github.com/CSCI510testerhw1/PopcornPicks/

		blob/master/README.md
Badges showing your code formatters.	3	https://github.com/CSCI510testerhw1/PopcornPicks/blob/master/README.md
Badges showing your syntax checkers.	3	https://github.com/CSCI510testerhw1/PopcornPicks/blob/master/README.md
Badges showing your code coverage tools	3	https://github.com/CSCI510testerhw1/PopcornPicks/blob/master/README.md
Badges showing any other Other automated analysis tools	3	https://github.com/CSCI510testerhw1/PopcornPicks/blob/master/README.md
Does your website and documentation provide a clear, high-level overview of your software?	3	https://github.com/CSCI510testerhw1/PopcornPicks/wiki
Does your website and documentation clearly describe the type of user who should use your software?	Yes	https://github.com/CSCI510testerhw1/PopcornPicks/wiki
Do you publish case studies to show how your software has been used by yourself and others?	Yes	Some in wiki, also Demo video
Is the name of your project/software unique?	No	Same name as project we forked from, but different version number
Is your project/software name free from trademark violations?	No	We upheld the terms of the original project's license
Is your software available as a package that can be deployed without building it?	Yes	We dockerized the software, so you can deploy with one command
Is your software available for free?	Yes	See License
Is your source code publicly available to download, either as a downloadable bundle or via access to a source code repository?	Yes	See Git repository
Is your software hosted in an established, third-party repository likeGitHub (https://github.com), BitBucket (https://bitbucket.org),LaunchPad (https://launchpad.net) orSourceForge (https://sourceforge.net)?	Yes	See Git repository
Is your documentation clearly available on your website	Yes	See Git repository

or within your software?		
Does your documentation include a "quick start" guide, that provides a short overview of how to use your software with some basic examples of use?	Yes	See Git repository
If you provide more extensive documentation, does this provide clear, step-by-step instructions on how to deploy and use your software?	Yes	See Git repository
Do you provide a comprehensive guide to all your software’s commands, functions and options?	Yes	See Git repository
Do you provide troubleshooting information that describes the symptoms and step-by-step solutions for problems and error messages?	No	See Git repository
If your software can be used as a library, package or service by other software, do you provide comprehensive API documentation?	Not Applcable	See Git repository
Do you store your documentation under revision control with your source code?	Not Applcable	See Git repository
Do you publish your release history e.g. release data, version numbers, key features of each release etc. on your web site or in your documentation?	Not Applcable	See Git repository
Does your software describe how a user can get help with using your software?	Yes	See Git repository
Does your website and documentation describe what support, if any, you provide to users and developers?	Yes	Yes in Con
Does your project have an e-mail address or forum that is solely for supporting users?	Yes	Email linked in Contributing.md and on README.md
Are e-mails to your support e-mail address received by more than one person?	No	
Does your project have a ticketing system to manage bug reports and feature requests?	Yes	Use github issues
Is your project's ticketing system publicly visible to your users, so they can view bug reports and feature requests?	Yes	Use github issues

Is your software's architecture and design modular?	Yes	See Git. Models, APIs, Frontend and Tests are all in separate files. Each function, test, module handles one specific task
Does your software use an accepted coding standard or convention?	Yes	See Git.
Does your software allow data to be imported and exported using open data formats?	yes	See Git
Does your software allow communications using open communications protocols?	no	
Is your software cross-platform compatible?	yes	See Git
Does your software adhere to appropriate accessibility conventions or standards?	yes	See Git.
Does your documentation adhere to appropriate accessibility conventions or standards?	yes	See Git
Is your source code stored in a repository under revision control?	yes	See Git
Is each source code release a snapshot of the repository?	yes	See Git
Are releases tagged in the repository?	yes	See Git.
Is there a branch of the repository that is always stable? (i.e. tests always pass, code always builds successfully)	yes	See Git
Do you back-up your repository?	yes	See Git
Do you provide publicly-available instructions for building your software from the source code?	yes	See Git
Can you build, or package, your software using an automated tool?	yes	See Git.
Do you provide publicly-available instructions for deploying your software?	yes	See Git
Does your documentation list all third-party dependencies?	yes	See Git
Does your documentation list the version number for all	no	

third-party dependencies?		
Does your software list the web address, and licences for all third-party dependencies and say whether the dependencies are mandatory or optional?	no	
Can you download dependencies using a dependency management tool or package manager?	yes	See Git
Do you have tests that can be run after your software has been built or deployed to show whether the build or deployment has been successful?	no	See Git
Do you have an automated test suite for your software?	yes	See Git
Do you have a framework to periodically (e.g. nightly) run your tests on the latest version of the source code?	no	See Git.
Are your test results publicly visible?	yes	See Git
Are all manually-run tests documented?	yes	See Git
Does your project have resources (e.g. blog, Twitter, RSS feed, Facebook page, wiki, mailing list) that are regularly updated with information about your software?	no	See Git
Does your website state how many projects and users are associated with your project?	yes	See Git.
Do you provide success stories on your website?	no	
Do you list your project's publications on your website or link to a resource where these are available?	no	
Do you list third-party publications that refer to your software on your website or link to a resource where these are available?	no	
Can users subscribe to notifications to changes to your source code repository?	no	See Git.
If your software is developed as an open source project (and, not just a project developing open source software), do you have a governance model?	yes	See Git
Do you accept contributions (e.g. bug fixes, enhancements, documentation updates, tutorials) from people who are not part of your project?	yes	See Git
Do you have a contributions policy?	yes	See Git

Is your contributions' policy publicly available?	yes	See Git.
Do contributors keep the copyright/IP of their contributions?	Not Applcable	
Does your website and documentation clearly state the copyright owners of your software and documentation?	yes	See Git
Does each of your source code files include a copyright statement?	yes	See Git
Does your website and documentation clearly state the licence of your software?	yes	See Git.
Is your software released under an open source licence?	yes	See Git
Is your software released under an OSI-approved open-source licence?	no	
Does each of your source code files include a licence header?	yes	See Git
Do you have a recommended citation for your software?	no	
Does your website or documentation include a project roadmap (a list of project and development milestones for the next 3, 6 and 12 months)?	no	
Does your website or documentation describe how your project is funded, and the period over which funding is guaranteed?	no	
Do you make timely announcements of the deprecation of components, APIs, etc.?	no	