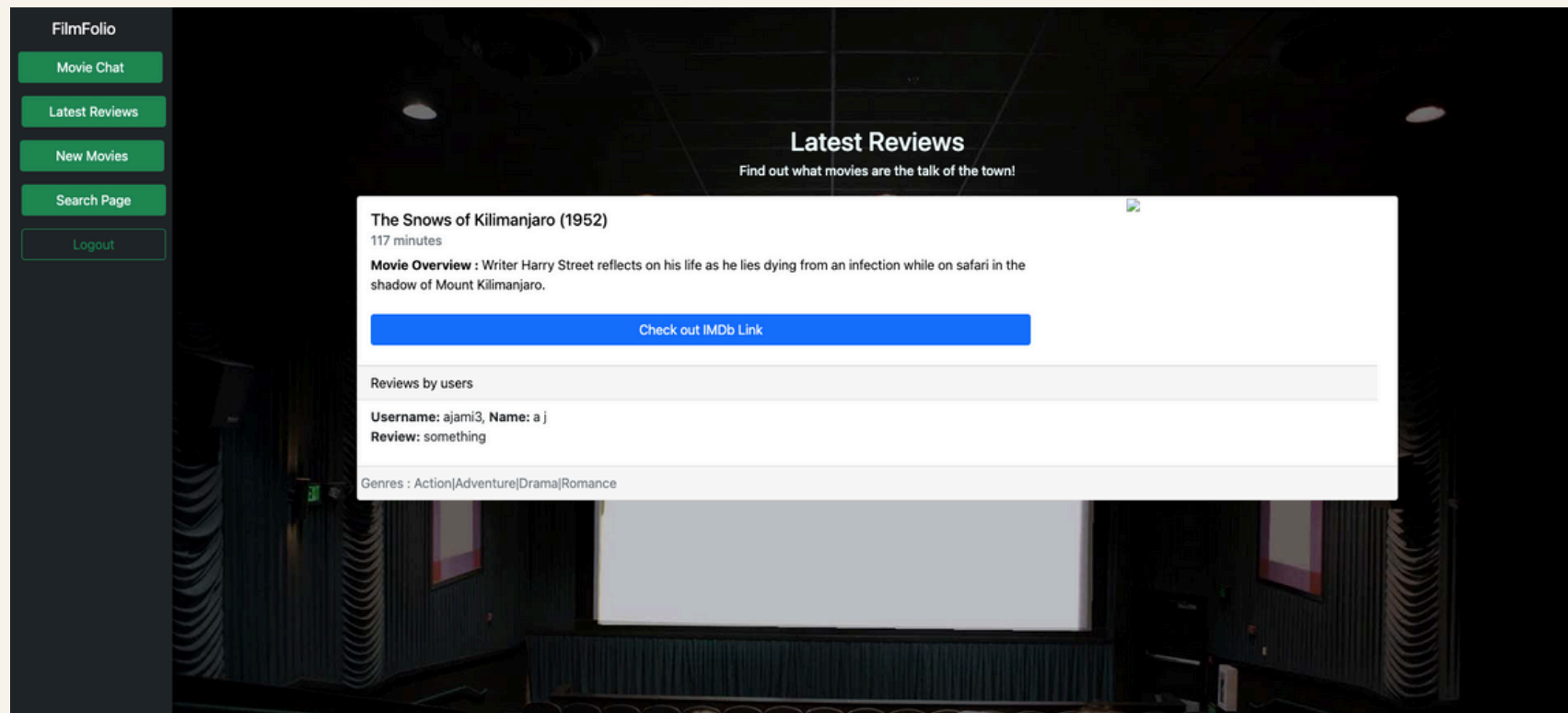


FilmFolio

FilmFolio is more than just a movie recommender system; it's a gateway to a world of cinematic adventures. With an ever-expanding library of films and a powerful recommendation algorithm, FilmFolio is here to transform the way you discover, enjoy, and connect with movies.



New and Improved!

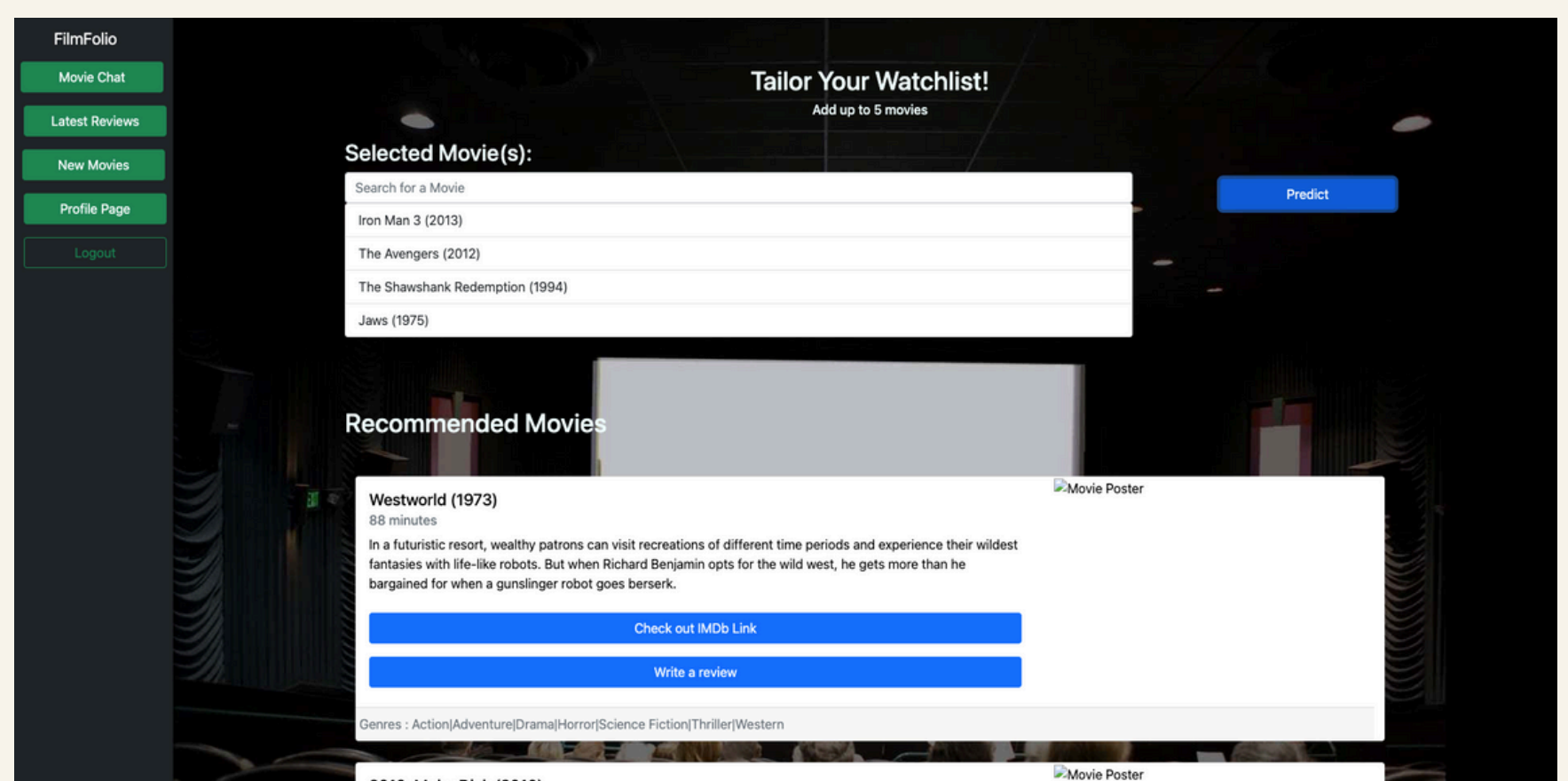
- **Improved Search Algorithm:**
 - Our new and improved search algorithm will help you find movies that are more similar to the movies you liked!
- **Revamped Backend:**
 - We've moved on from your standard MySQL database and are now using the built in SQLite database provided within Flask!
- **Revitalized Frontend:**
 - Say goodbye to clutter and excessive elements that complicate navigation, and embrace a streamlined experience that makes finding what you need effortless.
- **Improved Layout:**
 - We've included dedicated pages for our main features to allow for a more robust experience and ease of use!
- **Extensive Test Cases:**
 - Rest assured! Our system contains extensive test cases to ensure that if anything goes wrong, we'll know about it!
- **Chat Feature:**
 - Users can now chat with one another while watching movies! Stay in the loop with all of your friends and discuss exciting moments!

Future Milestones

- **Milestone 1:**
 - Rework the frontend to use React or Angular.
 - Although FilmFolio has a lightweight database, migration to MongoDB or other alternatives would allow you to scale up with ease.
- **Milestone 2:**
 - Continue to improve user profiles! Features such as private messaging and sharing movie recommendations could be implemented.
- **Milestone 3:**
 - Trends page! Movie watchers always want to know what the latest and trendy movies are. Make a page that easily provides that information!
- **Milestone 4:**
 - Integrate popular streaming services to provide seamless access to the recommended movies.
- **Milestone 5:**
 - Docker implementation would allow you to deploy FilmFolio anywhere!



Check it out for yourself!



- Frontend: HTML, CSS, JavaScript
- Backend: Flask (Python)
- DataBase: SQLite
- Test cases: 75+



Group 100:
Anchita Ramani
Meet Patel
Abhinav Jami

<https://github.com/SE-Group-AR/FilmFolio>

Group 100 Members:

Abhinav Jami

Meet Patel

Anchita Ramani

Notes	Score	Evidence
Video	3	https://drive.google.com/drive/folders/1uMvPk9RZWGQwDKv5eSteUk8YWXE0u5KJ?usp=sharing
workload is spread over the whole team (one team member is often X times more productive than the others but nevertheless, here is a track record that everyone is contributing a lot)	3	GH commits
Number of commits	3	in GH
Number of commits: by different people	3	in GH
Issues reports: there are many	3	https://github.com/SE-Group-AR/FilmFolio/issues
Issues are being closed	3	https://github.com/SE-Group-AR/FilmFolio/issues
DOI badge: exists	3	on README
Docs: doco generated , format not ugly	3	In Docs folder
Docs: what: point descriptions of each class/function (in isolation)	3	In Docs folder
Docs: how: for common use cases X,Y,Z mini-tutorials showing worked examples on how to do X,Y,Z	3	In Docs Folder

Docs: why: docs tell a story, motivate the whole thing, deliver a punchline that makes you want to rush out and use the thing	3	On README.md
Docs: short video, animated, hosted on your repo. That convinces people why they want to work on your code.	3	On README.md
Use of version control tools	3	We use GitHub for this and the trunk flow
Use of style checkers	3	Pylint
Use of code formatters.	3	Black
Use of syntax checkers.	3	Pylint
Use of code coverage	3	Codecov: badge on readme
other automated analysis tools	3	workflows/main.yml and CodeQL
test cases exist	3	/test/ directory
test cases are routinely executed	3	workflows/pytest.yml: see actions
the files CONTRIBUTING.md lists coding standards and lots of tips on how to extend the system without screwing things up	3	In Contributing.md
issues are discussed before they are closed	3	We discussed issues on discord and whatsapp, and summarized on the issue comments
Chat channel: exists	3	We had a chat channel, see CHAT.md
test cases:..a large proportion of the issues related to handling failing cases.	3	see our issues page
evidence that the whole team is using the same tools: everyone can get to all tools and files	3	Used VSCode and everyone has the software installed via the tools in the installation guide
evidence that the whole team is using the same tools (e.g. config files in the repo, updated by lots of different people)	3	

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	
evidence that the members of the team are working across multiple places in the code base	3	
short release cycles	1	visible in commit history
Does your website and documentation provide a clear, high-level overview of your software?	3	In Readme and docs files
Does your website and documentation provide a clear, high-level overview of your software?	3	In Readme and docs files
Do you publish case studies to show how your software has been used by yourself and others?	2	In Readme and the images in the docs. The video and the gifs are great examples of this
Is the name of your project/software unique?	3	
Is your project/software name free from trademark violations?	3	
Is your software available as a package that can be deployed without building it?	0	
Is your software available for free?	3	Yes on github
Is your source code publicly available to download, either as a downloadable bundle or via access to a source code repository?	3	Yes on github
Is your software hosted in an established, third-party repository like GitHub (https://github.com), BitBucket (https://bitbucket.org), LaunchPad (https://launchpad.net) or SourceForge (https://sourceforge.net)?	3	Yes on github
Is your documentation clearly available on your website or within your software?	3	In Readme and docs files

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?	1	Installation guide is included in Readme
If you provide more extensive documentation, does this provide clear, step-by-step instructions on how to deploy and use your software?	3	Instructions in docs folder
Do you provide a comprehensive guide to all your software's commands, functions and options?	3	Instructions in docs folder
Do you provide troubleshooting information that describes the symptoms and step-by-step solutions for problems and error messages?	2	see docs/troubleshoot.md
If your software can be used as a library, package or service by other software, do you provide comprehensive API documentation?	0	
Do you store your documentation under revision control with your source code?	3	Yes, in docs folder and Readme
Do you publish your release history e.g. release date, version numbers, key features of each release etc. on your web site or in your documentation?	2	see badge on github readme and the changes in project 3 docs file
Does your software describe how a user can get help with using your software?	3	Info in Contributing, and on the bottom of readme with a help email
Does your website and documentation describe what support, if any, you provide to users and developers?	3	Info in Contributing, nd on the bottom of readme with a help email
Does your project have an e-mail address or forum that is solely for supporting users?	3	At the bottom of readme
Are e-mails to your support e-mail address received by more than one person?	3	All of our developers check the help email

Does your project have a ticketing system to manage bug reports and feature requests?	3	In issues and projects tab
Is your project's ticketing system publicly visible to your users, so they can view bug reports and feature requests?	3	In issues and projects tab
Is your software's architecture and design modular?	2	Yes, but a little more separation could improve this
Does your software use an accepted coding standard or convention?	2	Consistent standards followed across codebase, conventions are defined in our pylint and black configuration
Does your software allow communications using open communications protocols?	2	we use http (AJAX) for communication
Is your software cross-platform compatible?	3	Has been tested with Edge, Firefox, Chrome
Does your software adhere to appropriate accessibility conventions or standards?	1	
Does your documentation adhere to appropriate accessibility conventions or standards?	1	
Is your source code stored in a repository under revision control?	3	All code in github
Is each source code release a snapshot of the repository?	2	
Are releases tagged in the repository?	2	Releases tagged in github
Is there a branch of the repository that is always stable? (i.e. tests always pass, code always builds successfully)	1	our final release is stable, but during development we did not have this
Do you back-up your repository?	3	On github and local machines
Do you provide publicly-available instructions for building your software from the source code?	3	In Readme and installation guide

Can you build, or package, your software using an automated tool?	0	
Do you provide publicly-available instructions for deploying your software?	3	In Readme and installation guide
Does your documentation list all third-party dependencies?	3	In requirements
Does your documentation list the version number for all third-party dependencies?	3	In requirements
Does your software list the web address, and licenses for all third-party dependencies and say whether the dependencies are mandatory or optional?	3	see dependencies.md
Can you download dependencies using a dependency management tool or package manager?	3	Using pip install requirements
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?	3	Testing through github actions
Do you have an automated test suite for your software?	3	Testing through github actions
Do you have a framework to periodically (e.g. nightly) run your tests on the latest version of the source code?	3	
Do you use continuous integration, automatically running tests whenever changes are made to your source code?	3	On-push testing through github actions
Are your test results publicly visible?	3	Test results in actions build tab
Are all manually-run tests documented?	0	
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?	0	

Does your website state how many projects and users are associated with your project?	0	
Do you provide success stories on your website?	1	the gifs show successful runs of the code
Do you list your important partners and collaborators on your website?	3	Contributors on Readme
Do you list your project's publications on your website or link to a resource where these are available?	0	
Do you list third-party publications that refer to your software on your website or link to a resource where these are available?	0	
Can users subscribe to notifications to changes to your source code repository?	3	
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?	3	laid out in docs
Do you accept contributions (e.g. bug fixes, enhancements, documentation updates, tutorials) from people who are not part of your project?	3	
Do you have a contributions policy?	3	In Contributing
Is your contributions' policy publicly available?	3	In Contributing on Github
Do contributors keep the copyright/IP of their contributions?	3	MIT license is used
Does your website and documentation clearly state the copyright owners of your software and documentation?	3	all files say copyright informations and doi is listed on readme
Does each of your source code files include a copyright statement?	3	At top of each source file
Does your website and documentation clearly state the license of your software?	3	MIT License

Is your software released under an open source license?	3	MIT License
Is your software released under an OSI-approved open-source license?	3	MIT License
Does each of your source code files include a license header?	2	
Do you have a recommended citation for your software?	3	Citation.md
Does your website or documentation include a project roadmap (a list of project and development milestones for the next 3, 6 and 12 months)?	3	In projects tab and open issues tagged
Does your website or documentation describe how your project is funded, and the period over which funding is guaranteed?	0	
Do you make timely announcements of the deprecation of components, APIs, etc.?	0	