



Movie Finder

Spencer Milbrandt, Sam Leon, Tetsumichi Umada



What we wanted to do:

- Add and remove movies to favorites to populate suggestions
- Network based recommendations
- Pull from API to get data



Features

- Communication: Slack
- Planning: Trello
- Version Control: Github
- Languages: PHP, HTML, CSS, Python (attempted)
- API: OMDb
- Website Deployment: Heroku



slack



- No notifications → Long reply times
- Shows who is and isn't online
- Easy to send screenshots (versus on a mobile chat app)



- Good for establishing what things need to be done when
- Time-consuming to constantly be updating this



- Incredible code hosting website
- Easy to track changes
- Versioning is easy
- Github pages is very specific about what it hosts



CGI- Python Script



- Not very cooperative
- Confusing documentation
- Python/HTML coding confusing



- Challenging to store a complex data
- Syntax is simple and easy to learn
- Databases can be complex to transfer from one location to another



OMDb API ★★★★★

- Gives images from movie posters
- Rating
- Genre
- Release Date
- Etc.



HEROKU



- Hosts websites (favors PHP)
- Easy to push new files to website
- Does not update live



Testing:

- Added/Removed data from Database
 - Randomly added users to Sign In database and tested login credentials
- Made multiple queries to API with randomized inputs
- Ensured profile verification would send emails to verify new users
 - Had issues with SMTP Hosting to send email verifications



Challenges

- Three group members
- Nobody worked with MySQL, PHP before class
- Challenging schedules to align



Demo

<https://dry-tor-23979.herokuapp.com/>