Recommender System Projects

Software Engineering I & II

Fall 2019 (T. Marlowe)

Spring 2020 (G. Chang)

What is a Recommender System?

- Source of news, product recommendations, or readings
  - Often for users interested in a particular topic or area
- Makes use of user history and preferences to push articles
- Anyone on Social Media and most people on the Internet have interacted with several of these
  - Often even get recommendations via email

Examples

- Commercial recommender systems
  - Amazon, on-line marketers, …
- Social media providers
  - Facebook, LinkedIn, …
- Research services
  - Academia, ResearchGate, …
- Careers, jobs, and professional development
  - StackOverflow, Pluralsights, Web Code Geeks, …

What could be added?

- Better user preference and configuration information
  - Better user interface and options for display
  - Allow selection via Boolean search on topics to select or omit
  - Provide clustering search options interacting with Boolean search
  - Refine selection after liking or downloading artifacts
  - Tailor delivery: when, where, and how – also ranking
- Limit repeated delivery
  - Keep some user history as well as preferences and user information
  - Do deliver reminders and service notices
  - Do deliver artifacts deemed urgent or critical

Why a Recommender System Project?

Ingredients

- Give students experience in real-world development
- Expose students to Web/mobile development
- Interact with various sources — web, databases, inventories, …— for items / artifacts to evaluate and recommend
- Integrate data science tools to select information to push
    - Allow users to pull information as well
- Maintain user databases with history, preferences, and other configuration information

Why a Recommender System Project?

Developing Viable Products

- Consider business/economic and technical factors
- Focus on areas that need a product
- Provide a more robust set of recommended artifacts
- Provide additional configurability and better user models
- Reduce duplication in recommendations

Why a Recommender System Project?

Develop Technical Skills

- Follow real-world development process models
- Develop for Web/mobile deployment
- Incorporate data analytics and visualization
- Address substantial requirements and implementation challenges
- Utilize modern software development and coding techniques
    - Agile software and management processes
    - Full stack implementation
    - Test-driven development
    - Design patterns, aspects, component-driven development
    - Version control and configuration management

Why a Recommender System Project?

Develop Soft Skills

- Work on a large project with complex requirements and interacting components
- Develop a constructive coding style — informative, easily modified, reusable, clean
- Realize the importance of and improve communication skills — speaking, listening, writing, reading, presenting
- Rely on teamwork, positive interactions, and task-specific leadership
- Refine critical thinking and analysis
- Consider and address social, legal, and ethical issues

A Year-Long Project

- First semester (Marlowe, occasionally with Chang)
  - Conceptual understanding of software engineering, software development, and modern development approaches
  - Business case, product vision, and plan
  - Understanding requirements and risk
  - Development of initial prototype
- Second semester (Chang, occasionally with Marlowe)
  - Incremental iterative development
  - Full-stack development and incorporation of advanced features
  - Professional perspective and case histories
  - Interaction within and among teams, with Chang

The Projects

EVY:
MUSIC
RECOMMENDER

COLLEGE
SELECTION
INFORMATION

GAMES, GAMING
AND GAMIFICATION

**CYBERSECURITY
INCIDENT
REPORTER**

**AUTISM SPECTRUM
DISORDER
LITERATURE**

EVY

By: Yohan Ninan , Brandon Ganesh, Vamsi Avinash Gunji, and Sean King

GOAL OF PRODUCT

- Let users search for music on a topic they identify with which also sounds similar to what they like to hear.
- Give artists exposure to a larger audience for their music

OVERVIEW OF PROJECT

- Created a platform where artists can come and upload their music, whether it is songs or instrumentals.
- Users will be able to create accounts in order to listen to music, as well as add to their library, and get recommended music that fits their taste.
- Incorporates profile pages for users, artists, and advertisers.
- Users can search for songs that are about something and search for songs that sounds similar (have a similar frequency spectrum).

DESIGN MODEL

- Uses Razor Pages, Entity Framework and Tensorflow
- Front End: HTML, CSS
- Back End: C#, Javascript
- A microservice architecture pattern where our main project is monolithic which imports other projects as services, using dependency injection to supply all the classes with their dependencies (services).

A Recommendation System for College Articles

By: Davis Cook, Anthony Mauro, Craig Pelling, Mostafa Sabri, Ryan Sogluizzo

Our Vision

- As a team, we wanted to create an application that allows incoming college students and their families to be able to easily find and pull articles on any college or university. Being able to have all the important information about a school is the best tool in helping young students make their college decisions.

- On the other side of the spectrum, our application is a great tool for aiding professors who are searching for a new school to apply to.

Our Stack

- React Frontend
  - Used OKTA for O-Auth
- Python Backend
  - Used Flask to create 2 APIs, one for read, one for write
- MongoDB database
  - Used because we planned on tagging the data and we wanted to easily have an arbitrary number of tags

Challenges (and Solutions)

- Getting React to talk to our Flask APIs
  - Used http calls via axios
- Running two Flask applications
  - We are still solving this…..
- Getting our data
  - Plan to have a server run our MongoDB database and run a daily job to load new data (currently having issues, may have to use a load script).

What We Learned

- How to work on a somewhat large (ok, this is still pretty small) software project.
- How to take an idea from an idea to a working application.
- Assigning roles within the team to allow us to focus on single aspects of the project.

GMRZ Recommendation

“Whatcha playin?”

A recommendation service that allows gamers to search for new video-games to play

By Michael Kozak, Hans Joshua Yabao,
Derrick Beecke, Justin Tom

Vision of Our Web-Application

- The objective of the Game Recommendation system is to provide new titles for users to play.
  - Recommend games based on a previously played game or on a searched game
  - Compare genres, tags and the developer to recommend similar games

Developmental Stack

Front-End

Python Django:

. Clean template design for users to use web-app with ease.

. Pre-packaged authorization system using *sqlite* for easier management of schema tables.

-Bootstrap: widely used CSS framework to design the website.

Back-End

- Python:
  - API data extracting
  - Data formatting
  - Recommendation calculation
- Utilizing the RAWG API database
  - Vast library of games with a multitude of data on each one

Preview

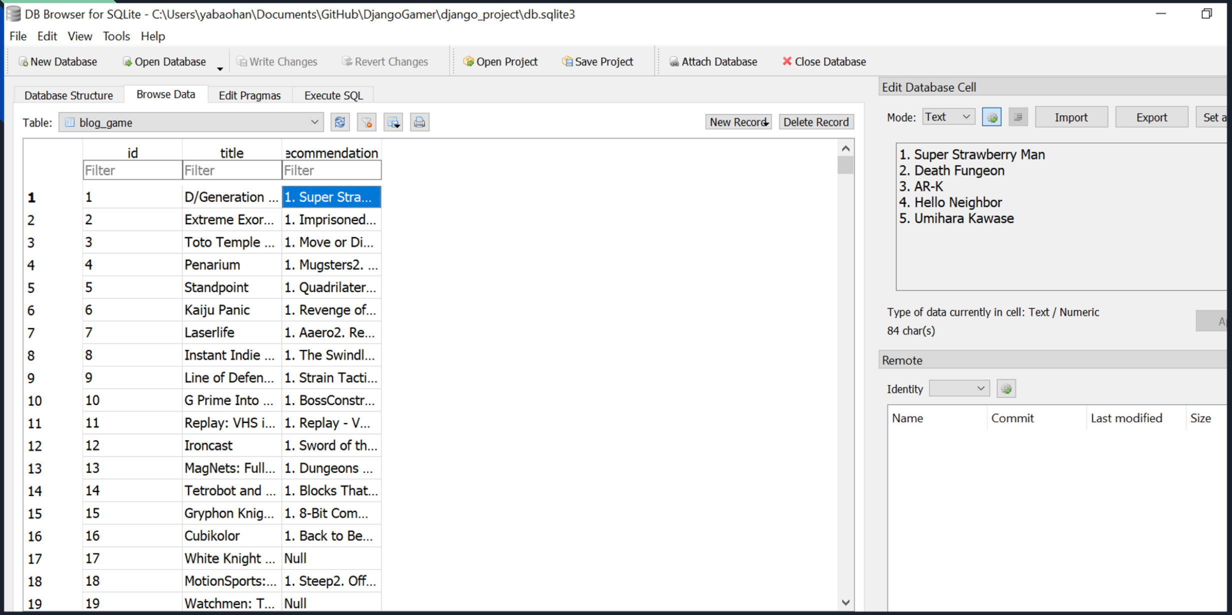


Table: blog_game

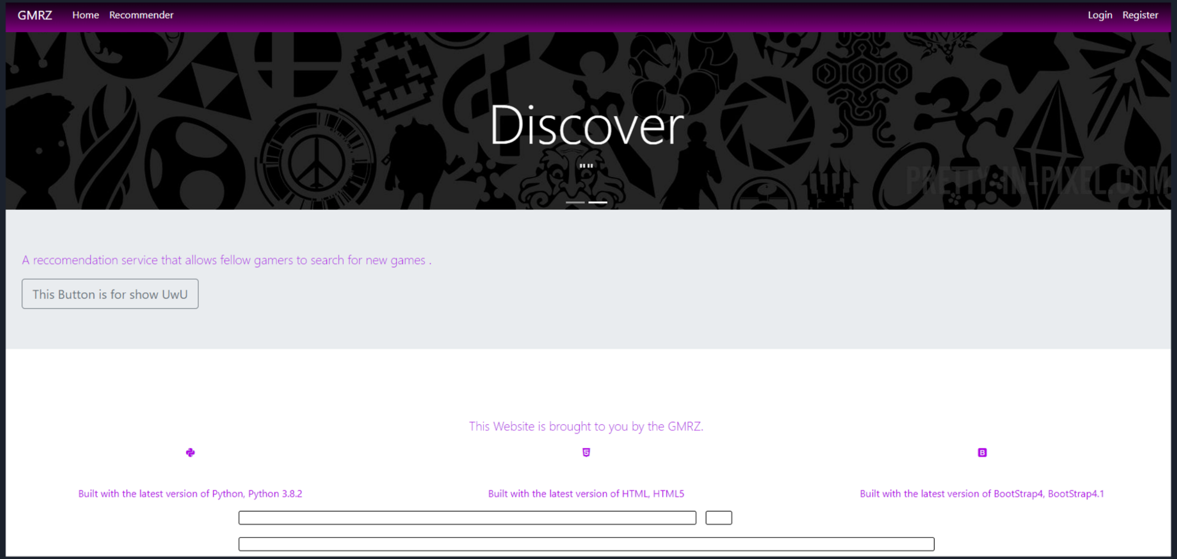
	id	title	ecommendation
	Filter	Filter	Filter
1	1	D/Generation ...	1. Super Stra...
2	2	Extreme Exor...	1. Imprisoned...
3	3	Toto Temple ...	1. Move or Di...
4	4	Penarium	1. Mugsters2. ...
5	5	Standpoint	1. Quadrilater...
6	6	Kaiju Panic	1. Revenge of...
7	7	Laserlife	1. Aaero2. Re...
8	8	Instant Indie ...	1. The Swindl...
9	9	Line of Defen...	1. Strain Tacti...
10	10	G Prime Into ...	1. BossConstr...
11	11	Replay: VHS i...	1. Replay - V...
12	12	Ironcast	1. Sword of th...
13	13	MagNets: Full...	1. Dungeons ...
14	14	Tetrobot and ...	1. Blocks That...
15	15	Gryphon Knig...	1. 8-Bit Com...
16	16	Cubikolor	1. Back to Be...
17	17	White Knight ...	Null
18	18	MotionSports:...	1. Steep2. Off...
19	19	Watchmen: T...	Null

Mode: Text

1. Super Strawberry Man
2. Death Fungeon
3. AR-K
4. Hello Neighbor
5. Umihara Kawase

Type of data currently in cell: Text / Numeric
84 char(s)

Identity



A reccomendation service that allows fellow gamers to search for new games .

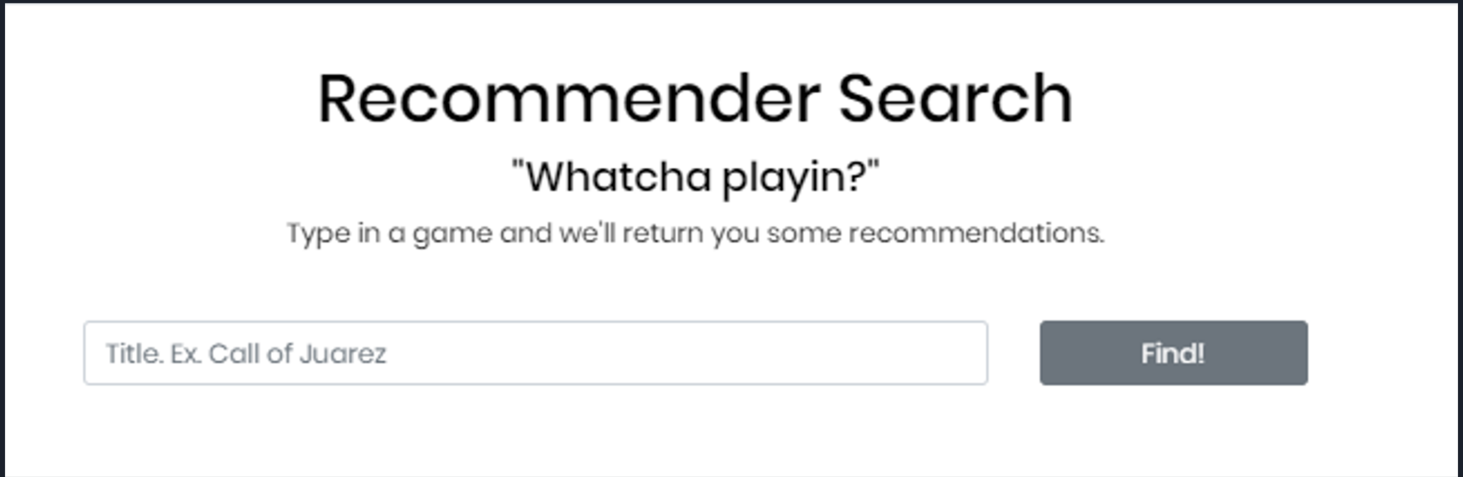
This Button is for show UwU

This Website is brought to you by the GMRZ.

Built with the latest version of Python, Python 3.8.2

Built with the latest version of HTML, HTML5

Built with the latest version of BootStrap4, BootStrap4.1



Recommender Search

"Whatcha playin?"

Type in a game and we'll return you some recommendations.

Title. Ex. Call of Juarez

What we’ve learned from the experience?

- Taking data from an API
- Knowing how databases function and handle data from different endpoints
- Creating login and register features
- Different languages and services/utilities are required to work together in order to create a functional web application
- Learning how to use in-demand frameworks that companies require from software developers

Conclusions

- Five very different projects
- Different implementation decisions and emphases
- All have done substantial work on their projects
- All have resulted in the core of a usable product
- Several can be made marketable

Conclusions

- Everyone worked hard
- All learned valuable intellectual, interpersonal, and career skills
- Everyone is now highly employable
- Part of student portfolios for interviews and resumes