

California State University Monterey Bay

Mike Menendez

Edgaras Slezas

Felix Ramon Romero-Flores

Frank Piva

Wesley Modes

CST 205: Multimedia Design and Programming

Term Project Ideation, Prototyping, and Planning

04/07/2020

Ideation

1. COVID-19 Resources
 - a. <insert tl;dr here>
2. Dog Fart Sound Generator
 - a. Randomize different dog fart sounds when you tap a button.
3. Tinder For Cats
 - a. It's like Tinder, but for cats.
4. Containerized Secure Code Execution (C++)
 - a. Allows submitters to run their C++ code
5. StudyBuddy
 - a. Allows students with similar schedules to find study partners.
6. Free Stuff Locator
 - a. Use Craigslist to look for "free" in an ad listing and compile results in one place.
7. YouTube Ratings
 - a. Allow users to rate YouTube videos and filter out unwanted categories.
8. Skin Care App
 - a. A user enters their allergens and an engine tells them what products to avoid.
9. Recipe Maker
 - a. Enter ingredients you have available and an engine gives you a recipe.
10. Five Buck Angel
 - a. Anonymously match donors with recipients; send a stranger five bucks.
11. Carpool Subsidy
 - a. Post your commute; potential fares can offer on what they're willing to chip in.
12. Virtual Hug
 - a. Tap a button and send a random individual with the app installed a virtual hug.

Prototyping

1. COVID-19 Resources
 - a. TL;DR: Allows for us to see areas that are greatly infected with COVID.

Home

Search

← drop down to pages

image retrieved
from other website

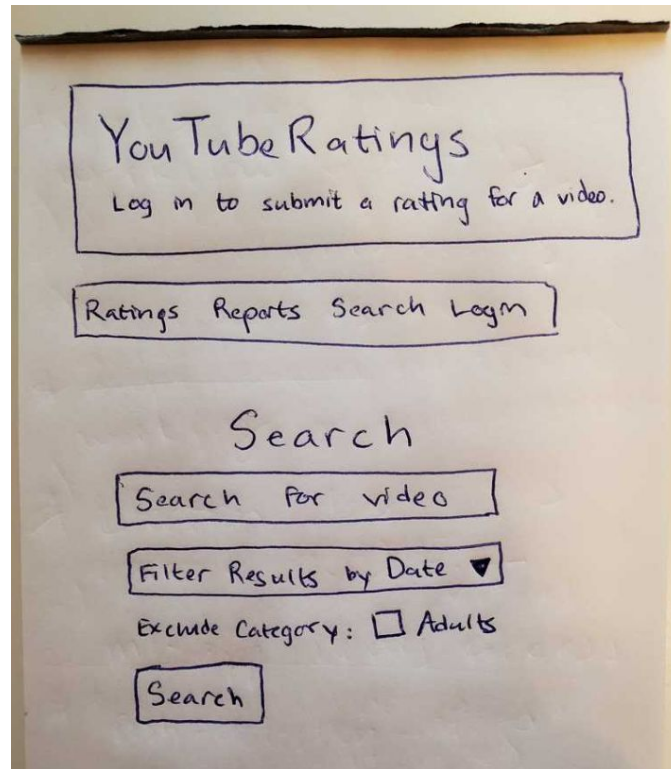
image retrieved
from other website

number of deaths
places in ~~that impacted~~
impacted

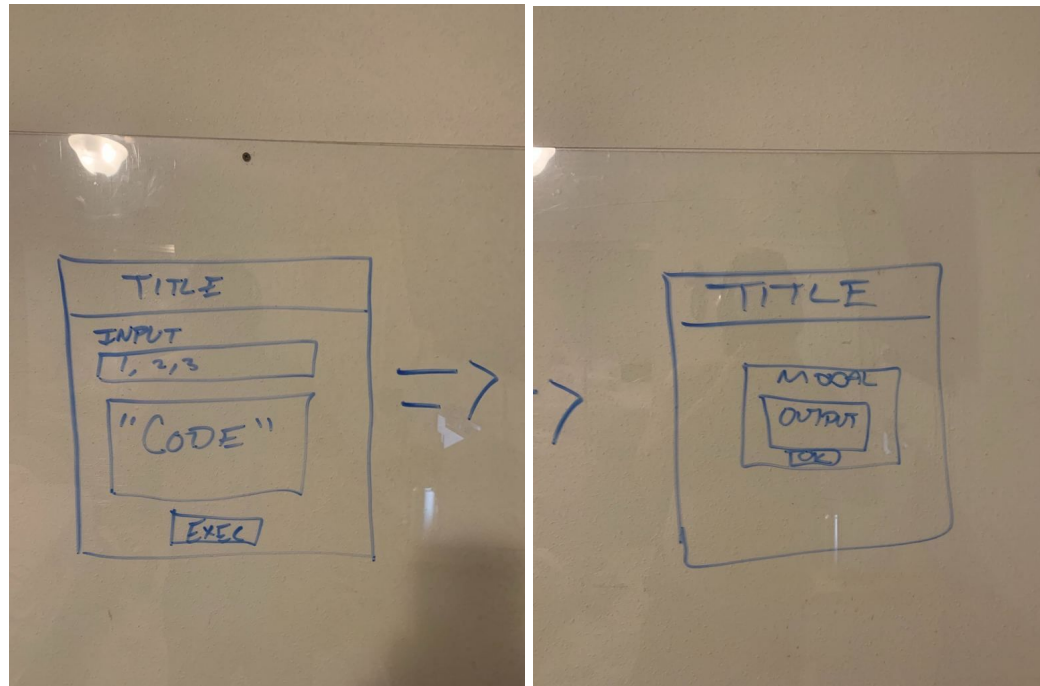
API info

b.

- c. I'm excited about this idea because with this website one will be able to get in hold of important information regarding COVID.
 - d. In this times it's not easy to find reliable information which is also easy to navigate through. Our site however will allow for easy navigation in order to find out what you want to know regarding covid.
 - e. In order for completion, it requires html, api, json, and pillow knowledge.
2. YouTube Ratings
- a. TL;DR: users can rate YouTube videos
 - b.



- c. I am excited about this because I think it is a lucrative financial opportunity. Parents are willing to spend all kinds of money on protecting their kids and I need to pay for university.
 - d. YouTube videos don't have a great rating system. Parents who want to prevent their children from viewing inappropriate content can use this to filter out unwanted search results. If a video is marked as adult content and the user has chosen to exclude that category, it will not appear in the search results. Users with accounts are the only ones who can rate videos (to prevent inaccurate ratings), but the search function does not require an account and is free to use.
 - e. We would need a relational or NoSQL database to store ratings, a YouTube API key, and a server to host the application on.
3. Containerized Secure Code Compilation + Execution (C++)
- a. TL;DR: Allows submitters to run their C++ code



- b.
- c. By providing a consistent environment for all students/users, they will have equal opportunities despite their situation such as not being able to afford a macbook, a laptop (as this can technically be used from their phone), the school computers (which do not have g++).
 - d. Not everyone has a consistent environment to run and test their code, especially for compiled languages such as C++. By having a consistent environment, it guarantees that everyone will be using the same compiler (G++) as not everyone has easy access to it such as windows users having to jump through hoops to use it despite it being one of the most commonly used ones out there. In addition to this, it lets students use the same compiler that is used in their classes here at CSUMB.
 - e. A running server, knowledge of containers, knowledge of C++ compilation, knowledge of bash, knowledge of gnu-utils, knowledge of REST, knowledge of networking, knowledge of DNS, knowledge of Javascript ... etc. etc.

Planning

- Edgaras, start listing resources who will do what
- It doesn't have to be set in stone
- Just fill in the blanks
- You can say what supporting technologies we'll be using
- E.G. GitHub for source control, MariaDB PostgreSQL for relational database
- Decide what questions you need to answer to choose your project.
- Make a schedule for who, how, and by what deadlines in the next several weeks you will answer these questions.

- Do some research to answer the most important of your prototype questions. Document your research.
- What project are you learning toward based on what you know now?