University of CINCINNATI

CodeBase

Yizong Cheng

Project Advisor

Seth Hanusik

Computer Science

Matthew Schlager

Computer Science

James Wilfong

Computer Science

Project Abstract

Currently there are many repository options for programmers to store large and small snippets of code, most notably are GitHub products. GitHub provides a space, GitHub Gist, for users to store and share small snippets of code. However, this feature is under used and lacks standard searching options. Similar repository solutions are subject to similar flaws.

We introduce CodeBase, an online repository designed for storing code blocks both publicly and privately. A given user will be able save and classify a piece of code either publicly or privately, each entry will be able to be tagged by language and purpose.

This will allow users to also search, and filter code solutions based on their given needs, such as language, tags, and reusability. Publicly posted entries will feature a liking and commenting system to promote discussion and the advancement of better code over inefficient code.

Goals

1. CodeBase will implement a modern user interface that will allow for users to easily traverse and make use of the searching and filtering options built into Codebase.

2. CodeBase will use Elasticsearch to search through many code blocks to find the most relevant responses to a user query. These results will also be able to be weighted based on popularity, freshness, and overall activity.

3. CodeBase will provide online compiling for a subset of languages, the extent of languages offered will be based on the time available during development.

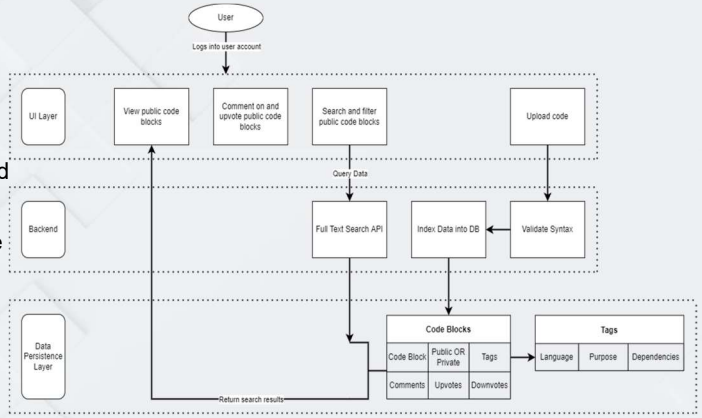
Design

CodeBase is split into three primary layers.

1. **UI Layer**: This is where users can see, interreact, and post code blocks.

2. **Backend**: This houses the APIs that connect the UI and Data Layer.

3. **Data Layer**: Database for storing user code blocks and other information.



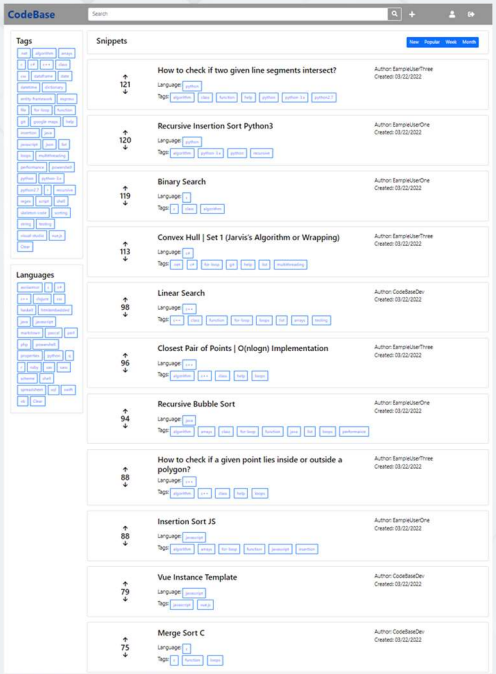


Figure 1: The landing page serves as a space for users to explore and filter trending code blocks. It defaults to showing the most popular code blocks. A search bar is available to find relevant code blocks.

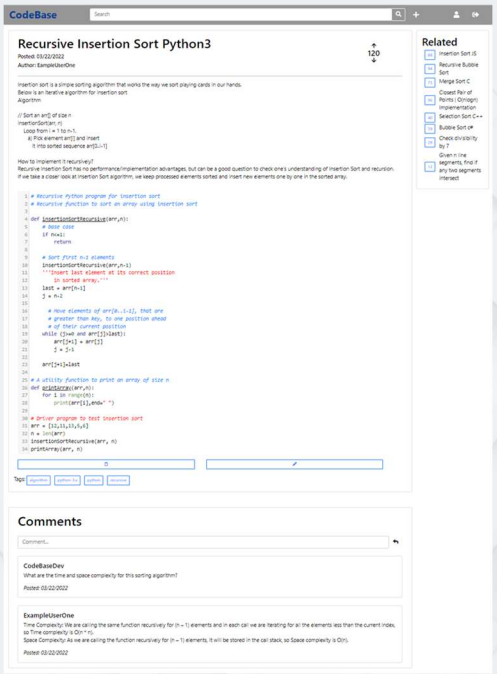


Figure 2: Each code block can be viewed, showing the full code snippet, description, tags, and comment section. Snippets can be copied and experimented with from these pages.

Challenges

Throughout the designing, implementation, and integration of CodeBase the development team was faced with many challenges and obstacles to overcome.

1. Ensuring CodeBase separates itself from other code storage options like GitHub and GitHub Gists.

2. Implementing ways for users to experiment and compile code through a web browser.

3. Maintaining consistent and scheduled development while working as full-time students.

Future Growth

In the future there is continued room for CodeBase to grow and develop.

1. Supporting a larger list of tags for users to select from and support more languages that can be compiled in the browser.

2. Add functionality to make the direct sharing of code blocks easier for users.

3. Add functionality to allow users to follow users or tags that they are active with, so they are notified when new code blocks are uploaded.

4. Implement a more navigable and expansive UI for users; such as being able to create code blocks from the sandbox page.

5. Expand the range of executable languages offered by CodeBase.

Technologies