



Crystal Group
ODU Spring 2024

Team Bio



Robert Geraghty



Sean Allgaier



Dom Hankle



Adam Adam



Abigail Eagan



Ethan Hancock



Drew Rockmore

Elevator Pitch

If you have a goal, and are looking for a piece of software to accomplish it, it can be difficult to find something that fits your needs, especially if you have specific needs. Search results are full of AI-generated top 10 lists, and old forum posts are often outdated. “SoftWhere” aims to fix that!

Search a large list of software solutions, filter your searches to narrow the search down, see user feedback about software! This will make finding the software you want so much easier!

Table of Contents

[Cover Slide](#) (Slide 1)

[Meet the Team](#) (Slide 3)

[Overview and Problem Description](#) (Slides 5-6)

[Solution and Clarifications](#) (Slides 10-15)

[Process Flow](#) (Slide 11)

[Competition Assessment](#) (Slide 15)

[Design Overview](#) (Slides 16-18)

[Risks](#) (Slide 19-26)

[Conclusion](#) (Slide 27)

[Appendix and Citations](#) (Slides 29-31)

Problem Statement

Users face the daunting task of finding software that aligns with their needs and preferences.

- Searching for software may be complicated and time-consuming, since it involves many platforms, websites, and sources.
- Users may find it difficult to assess the accuracy and reliability of software sources, leading to hesitation in making decisions.
- Users may not be aware of all available software options, missing out on potentially useful solutions.
- Users may not have clear guidelines on how to conduct effective software searches, such as the criteria to look for and where to look for relevant information.
- Existing filtering schemes may be ineffective in reducing search results to valid possibilities, resulting in information overload.

Problem Characteristics

- Search engine results contain many subpar sources (Montti).
 - Search Query: “best pdf reader”
 - Top Result: A subpar, possibly AI-generated list (Clark).
 - **“Best Free PDF Readers of 2024”**
 - Issues:
 - Reliability of information (AI generated?)
 - Free-trial software

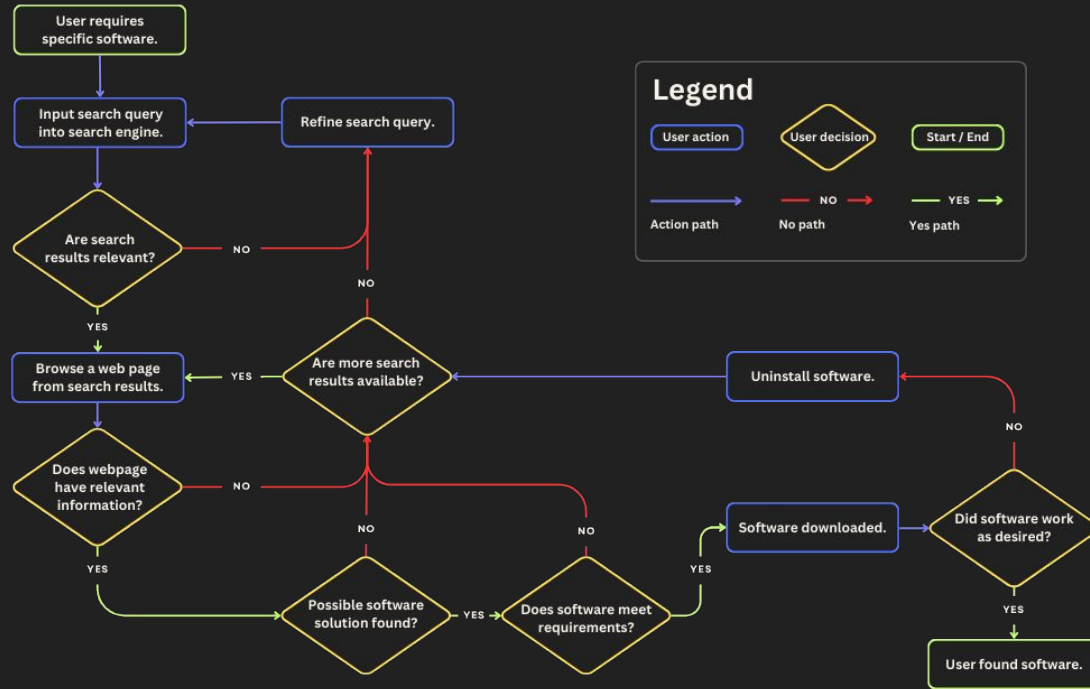
Problem Characteristics

- Old forum posts may be outdated
 - Search Query: “concatenate pdfs cli”
 - Top Result: A StackOverflow post from 2011 (StackOverflow).
 - **“Merge/convert Multiple PDF Files Into One PDF”**
 - Issues:
 - Possibly outdated information
 - Newer solutions may be better

Problem Characteristics

- App stores contain a many options
 - The Microsoft Store has over **800,000** apps (Tyrina)
 - The Mac App Store has over **31,000** apps (Koetsier)
 - The Debian 12 repository has over **63,000** packages (Debian)
- App stores may have subpar search features, contain subpar software, and may not provide user feedback.
- App stores do not contain all software available for the respective platform.
- The main advantage of app stores is that they are a generally a curated list of trustworthy software.

Current Process Flow

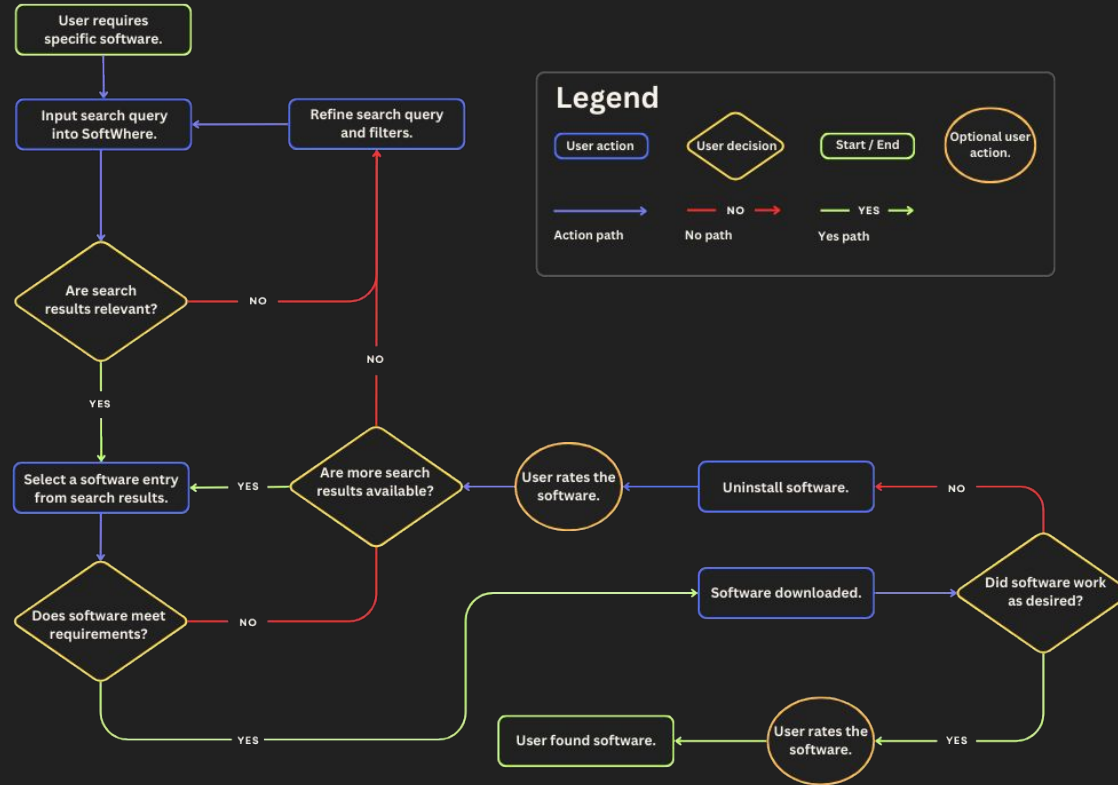


Solution Statement

We will be developing a web application that will provide end users a powerful search engine for software. Users will have the capability to filter software based on specific categories or tags. Some examples of these would be “open source”, “command line”, “Windows”, “MacOS”, “web-based”.

Duplicate search results will be limited and end users will be able to like, dislike, and leave comments about specific search results. Search results will be shown in order of most “liked” to least “liked” from top to bottom.

Solution Process Flow



Where is our data coming from?

- Some of the data will be contributed by the development team.
- The community will have the ability to contribute entries and vote on them.
- Community content will be curated via a combination of automated tools and manual moderation.

What The Solution Will Do

- Keyword search
 - Of only user-ready software
- Powerful tag-based filtering
 - Logical composition (AND, OR, NOT)
- Display like and dislike counts
- Display user comments
- User-Generated pages

What The Solution Will Not Do

- Provide direct means of installation
- Provide bug tracking or user support
- Provide unsolicited recommendations of software
- Search by competitor

Competition Matrix

Feature	<u>SoftWhere</u>	AlternativeTo.net	GitHub.com	Search Engines (Google, Bing, etc.)	App Stores (iOS, Google Play, etc.)
Display user comments	✓	✓	✓		✓
Display likes and dislikes separately	✓	✓	✓		✓
Search by competitor		✓			
Search by use case	✓			✓	✓
Search by programming language	✓		✓	✓	
Search by keyword query	✓	✓	✓	✓	✓
Filter search by user-defined tags	✓				

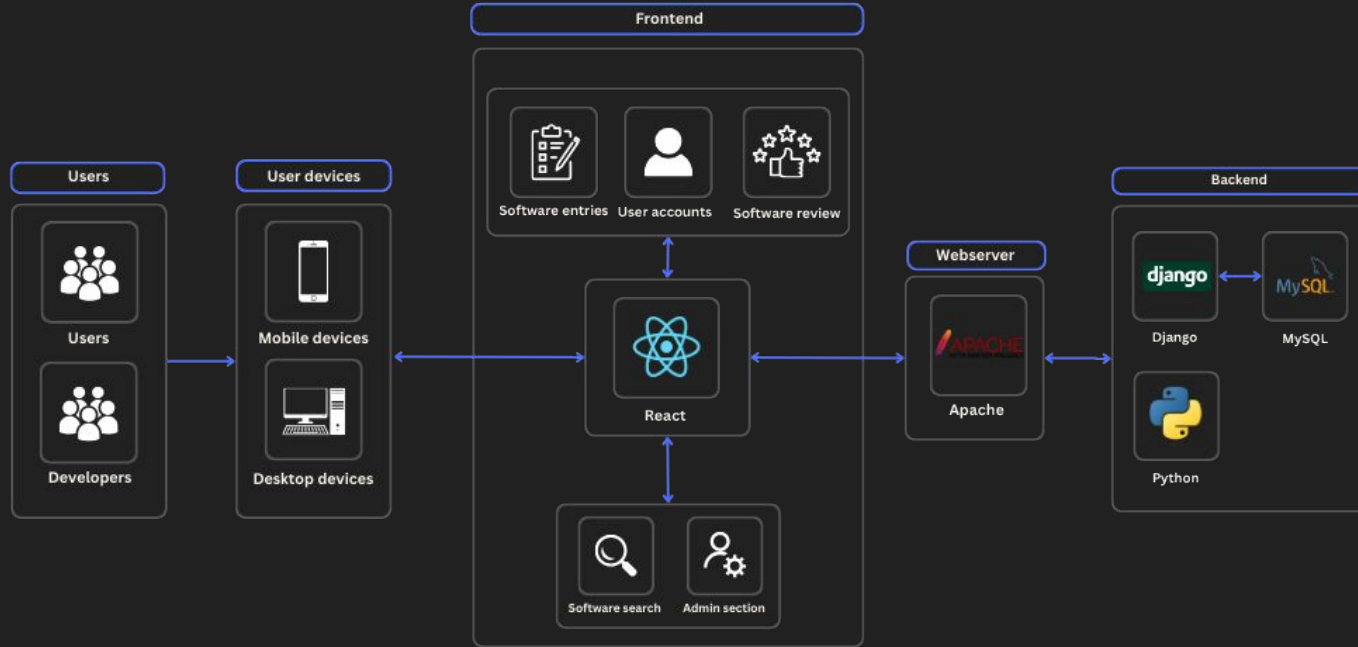
Development Tools

- SoftWhere will be a webapp
- Backend will be Python3 developed in VS Code
- There will be a GitHub repository that will contain the Python3 code, which will sync to the server on commits to 'main'
- Unit testing for the backend will be done with Pytest
- Unit testing for the frontend will be done with Jest
- Python documentation will be Pydoc

Major Functional Components

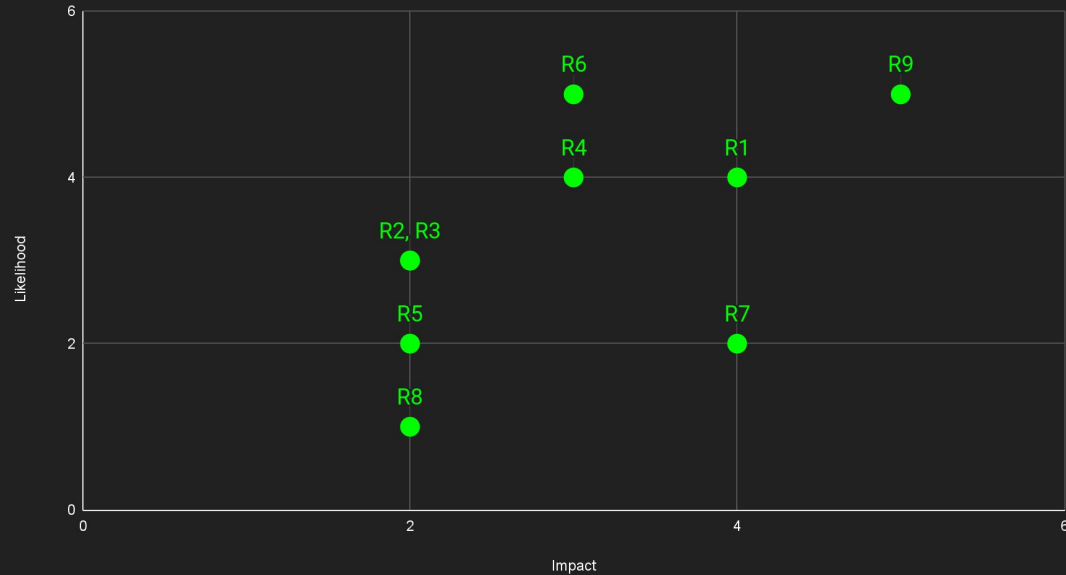
- There will be an Apache web server hosting the service.
- There will be a Django backend.
- There will be a React frontend combined with HTML5.
- There will be a MySQL database.
- There will be Python3 code that handles a lot of the logic.
- There will be an admin section of the webapp for maintainers to maintain things.

Diagram of Major Function Components



Risks - Pre-Mitigation

Risk Likelihood and Impact Chart

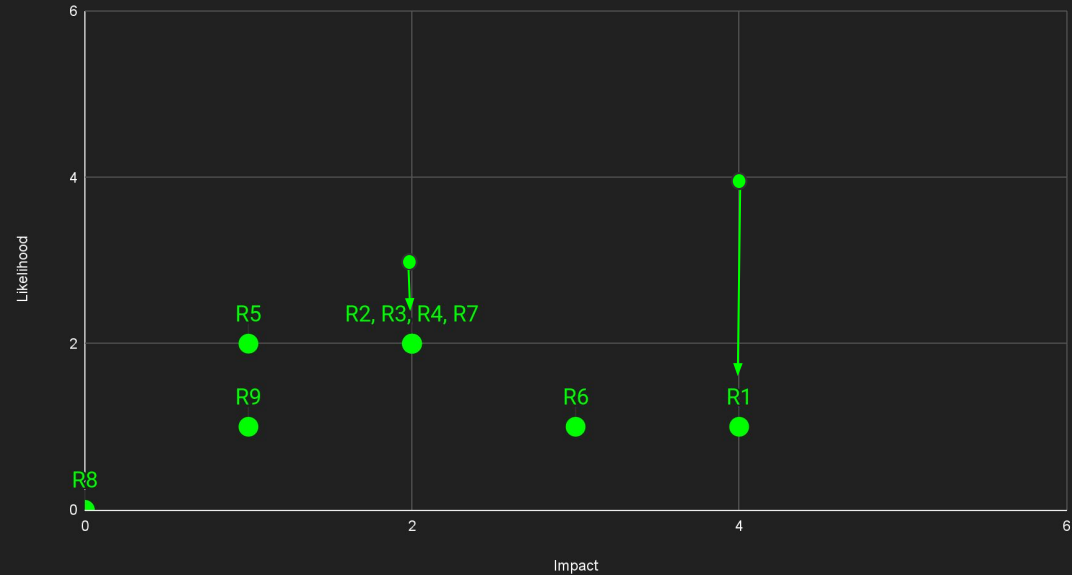


Risks - Technical

- **SQL Injection – R1 (Impact: 4, Likelihood: 4)**
 - SQL injection would be the result of a malicious attacker infiltrating SoftWhere's database and corrupting it. This isn't as likely considering it required an educated hacker, but it's definitely still possible.
 - SoftWhere will attempt to mitigate this through input sanitation and other database security techniques.
- **Distributed Denial-of-Service (DDoS) Attack – R2 (Impact: 2, Likelihood: 3)**
 - This is a cyber attack where the attacker attempts to overwhelm an online service with a sudden flood of traffic, usually done with bots.
 - Implementing a rate limiting mechanisms to restrict the number of requests a single IP address can make in a given time period could mitigate this risk.
- **Backend Server Crash – R3 (Impact: 2, Likelihood: 3)**
 - This is a risk describing the situation that the server hosting SoftWhere's backend server goes down, making SoftWhere unable to query data.
 - This could be mitigated through a secure power supply, ethernet connection, and consistent maintenance for the machine hosting SoftWhere's backend server.

Technical Risks - Post-Mitigation

Risk Likelihood and Impact Chart



Risks - Customer

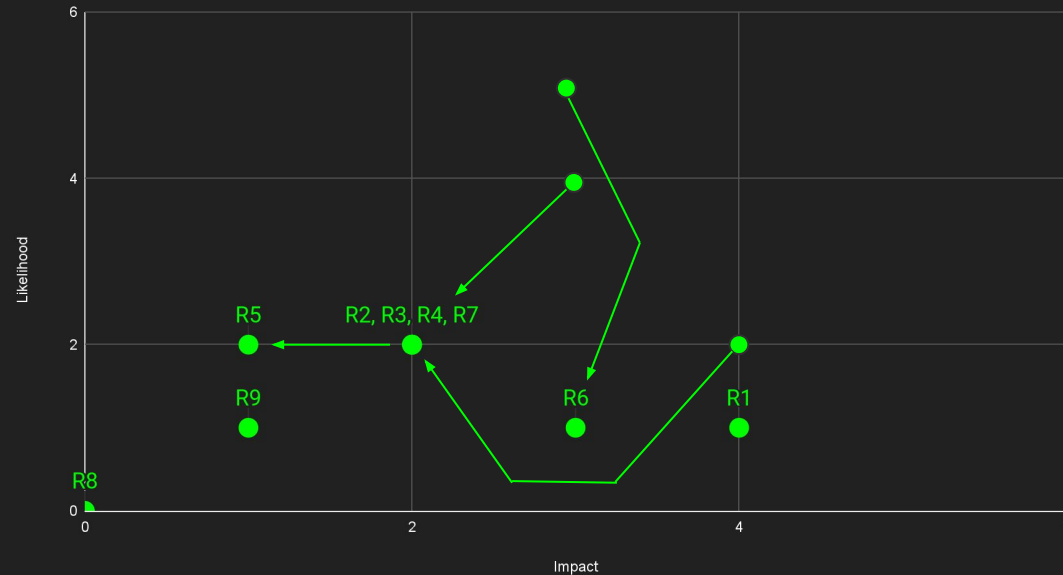
- **Bad User-Generated Content – R4 (Impact: 3, Likelihood: 4)**
 - Search results for software could become unreliable due to users intentionally rating poor search results higher, or creating poor-quality pages.
 - SoftWhere will mitigate this by providing users with the capability to report pages that are rated incorrectly or are spreading misinformation.
- **Dislike Trolls – R5 (Impact: 2, Likelihood: 2)**
 - This risk is also related to search results becoming unreliable, but for the opposite reason of bad user-generated content. In this scenario users will purposely dislike valuable search results in order to lower their ratings in SoftWhere.
 - Our web application will mitigate this by slightly rating “likes” of search results higher than “dislikes” internally.

Risks - Customer

- **Inadequate Results – R6 (Impact: 3, Likelihood: 5)**
 - A user may search for a piece of software, but not find any that suits their needs.
 - This is more likely when SoftWhere is newer, and will be mitigated as the site and the database grow.
 - Advertisements/reaching out to companies to put pages up could help the database grow faster.
- **Malicious Software – R7 (Impact: 4, Likelihood: 2)**
 - A user may run the chance of downloading malicious software from an applications that are linked in the database.
 - Malicious software is unlikely to receive positive ratings, and thus would appear lower on search results, minimizing user exposure.
 - SoftWhere will tell users whether or not an application is from a verifiable reputable source.
 - Verified pages will have a clearly visible label indicating that the source is trustworthy.
 - (i.e. Verified: Microsoft, Apple, etc. Unverified: Joe from Github)

Customer Risks - Post-Mitigation

Risk Likelihood and Impact Chart

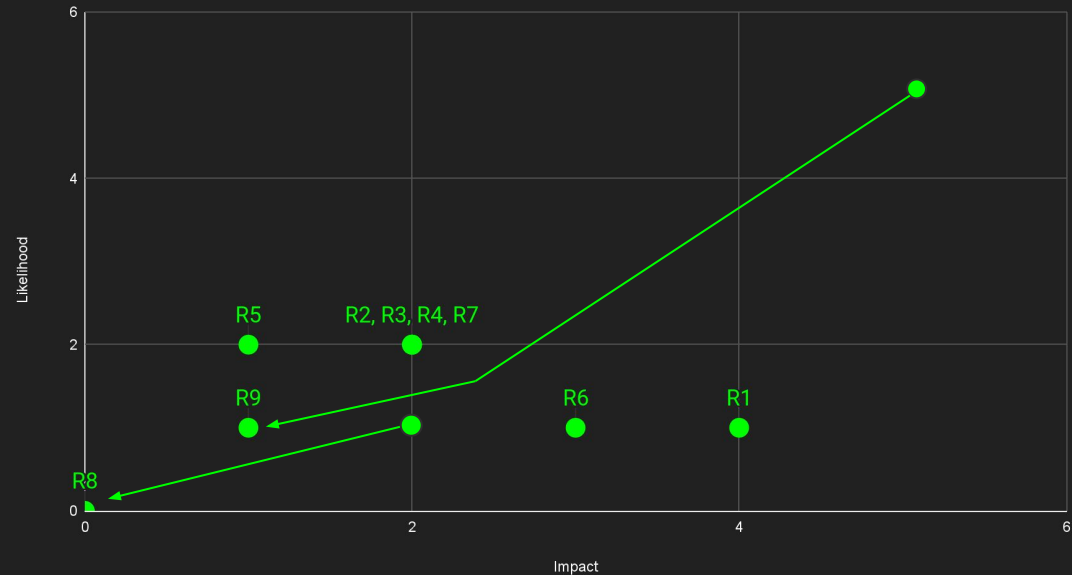


Risks - Legal

- **Misinformation Lawsuit – R8 (Impact: 2, Likelihood: 1)**
 - This risk describes the situation in which a user attempts to sue SoftWhere with the grounds that it is spreading misinformation.
 - Though a lawsuit isn't very likely, SoftWhere will attempt to mitigate this through a "Terms of Use" document that specifies search results are subject to user ratings, which could make them unreliable and contain misinformation.
- **Licensing Issues – R9 (Impact: 5, Likelihood: 5)**
 - This risk describes the likelihood of a company attempting to sue SoftWhere for directly distributing an application in violation of the relevant license.
 - In order to mitigate this, SoftWhere will not directly distribute any executables, instead providing links to the proper download page.

Legal Risks - Post-Mitigation

Risk Likelihood and Impact Chart



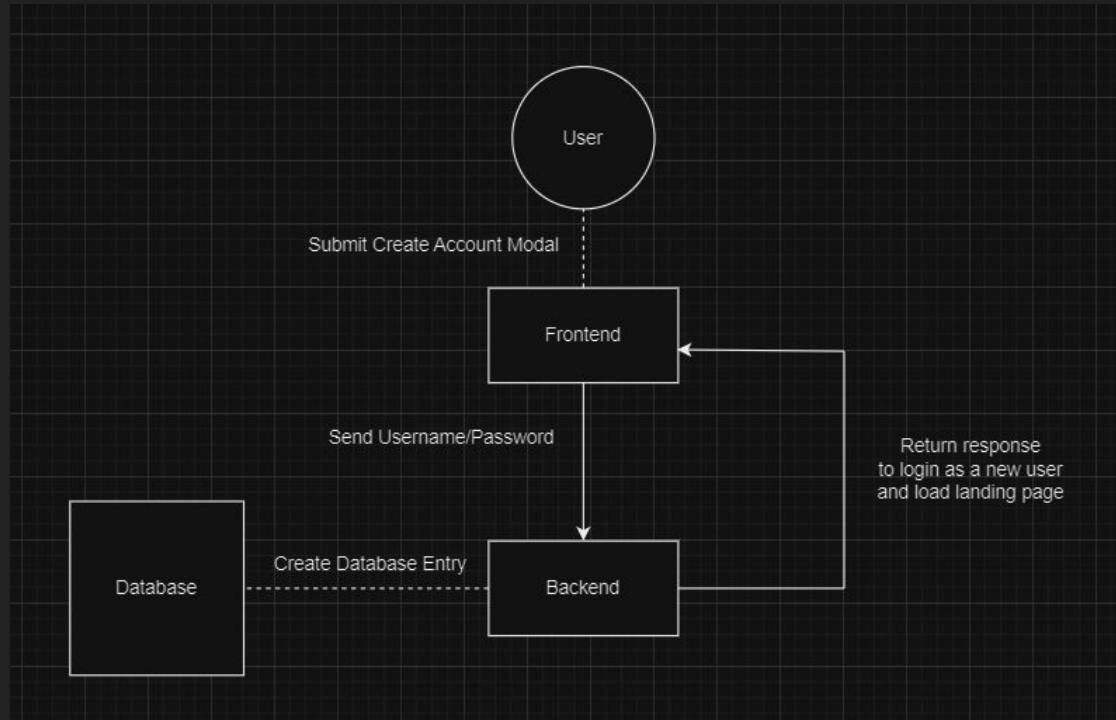
Work Breakdown

1. Apache needs to be setup first so that basic content can be viewed. All of the other components will hinge around the web server itself.
2. MySQL will need to be setup second.
3. Django will need to be setup immediately after MySQL, so that the two can interact.
4. React will be setup after the previous items.
5. The interactions between Django and MySQL, React and Apache, and the frontend between the backend will need to be tested as a whole once the individual components are done. API testing will be done with Postman.
6. The development model would be Agile for each section of work performed.

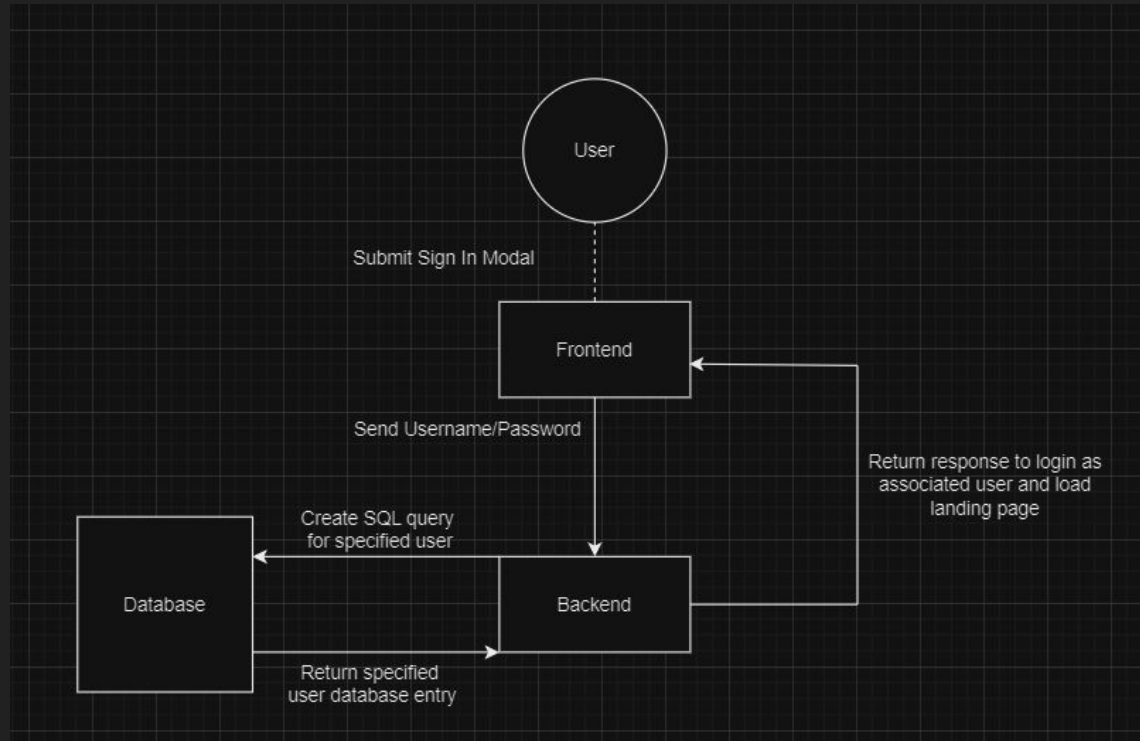
Algorithms

- **Search**
 - A text-matching algorithm must be used to match search terms to page descriptions, and to assign pages a relevance score
- **Filter**
 - A simple filtering algorithm may be used to implement tag filtering of search results
- **Sort**
 - Search results must be ordered considering both search relevance and like/dislike counts
- **Software entry creation**
 - A sanitization and verification algorithm to ensure user input aligns with the software entry schema.
- **Software entry reporting**
 - An algorithm to handle reporting of software entries that will ensure a user submission aligns with the reporting schema, ensuring all appropriate parties are notified.

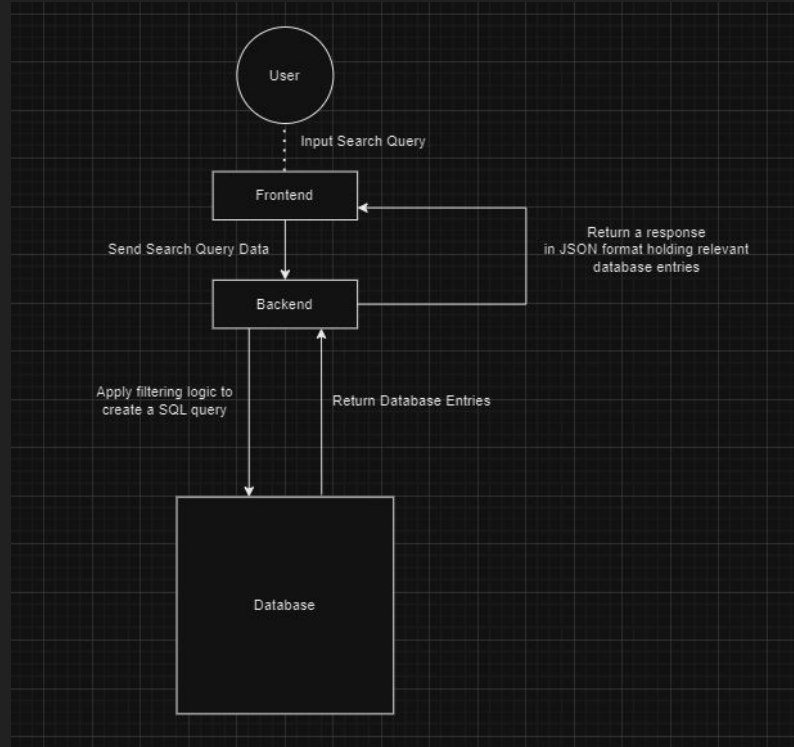
Data Flow Diagram - User Account Creation



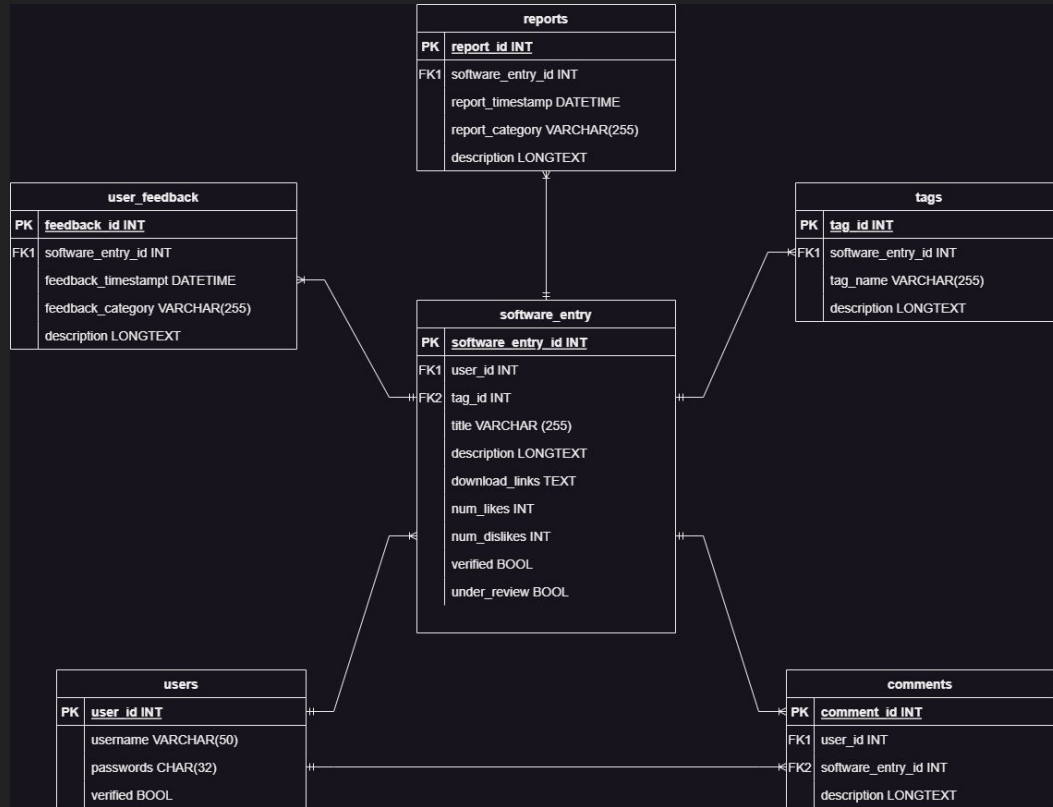
Data Flow Diagram - User Login



Data Flow Diagram - Search Query



Database Schema



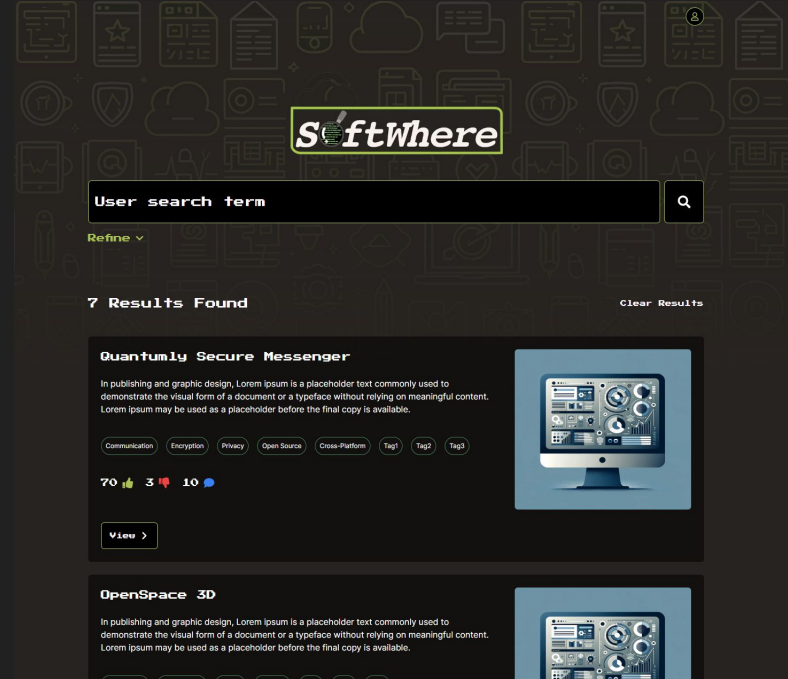
Real World Product vs Prototype

Real World Feature	Present in Prototype (Y/N)	Explanation
Verified Product	N	Correspondence with big tech companies needed
User-Generated Product Pages	Y	Sample product pages will be submitted
Like/Dislike Feature	Y	Essential for main functional requirements
Comment Section	Y	Provide the ability for feedback on each page
User Page Customization	N	Not necessary for main functionality
Account Generation	Y	Allow for user accounts within a database
Password Analysis Tool	N	Will require additional user input verification
Account Modification	N	Managing account data is not necessary for primary functionality
Account Verification	N	The platform offers functions for managing user accounts such as logging in, signing up, overseeing profiles and adjusting account preferences. Verifying accounts is crucial to ensure the safety and authenticity of user registrations.
Tagged Search Results	Y	Essential for main functional requirements

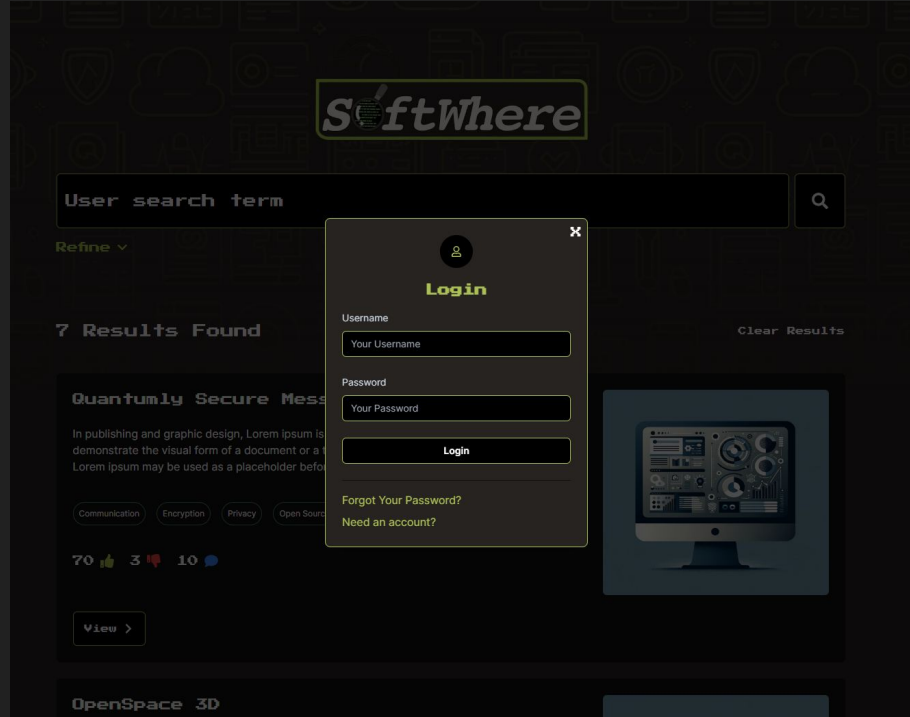
Real World Product vs Prototype cont.

Advanced Search	N	Nonessential, would involve boolean logic searches
Submission Moderator Staff	N	Not feasible for a prototype
Search by Keyword: Developer Name	Y	Essential for main functional requirements
Search by Keyword: Title of the Software	Y	Essential for main functional requirements
Search by Keyword: Anywhere in Description	Y	Essential for main functional requirement
Exhaustive Database	N	Database will be manually populated
Community Driven Results	N	There will not be a community to drive the results, but it will be simulated

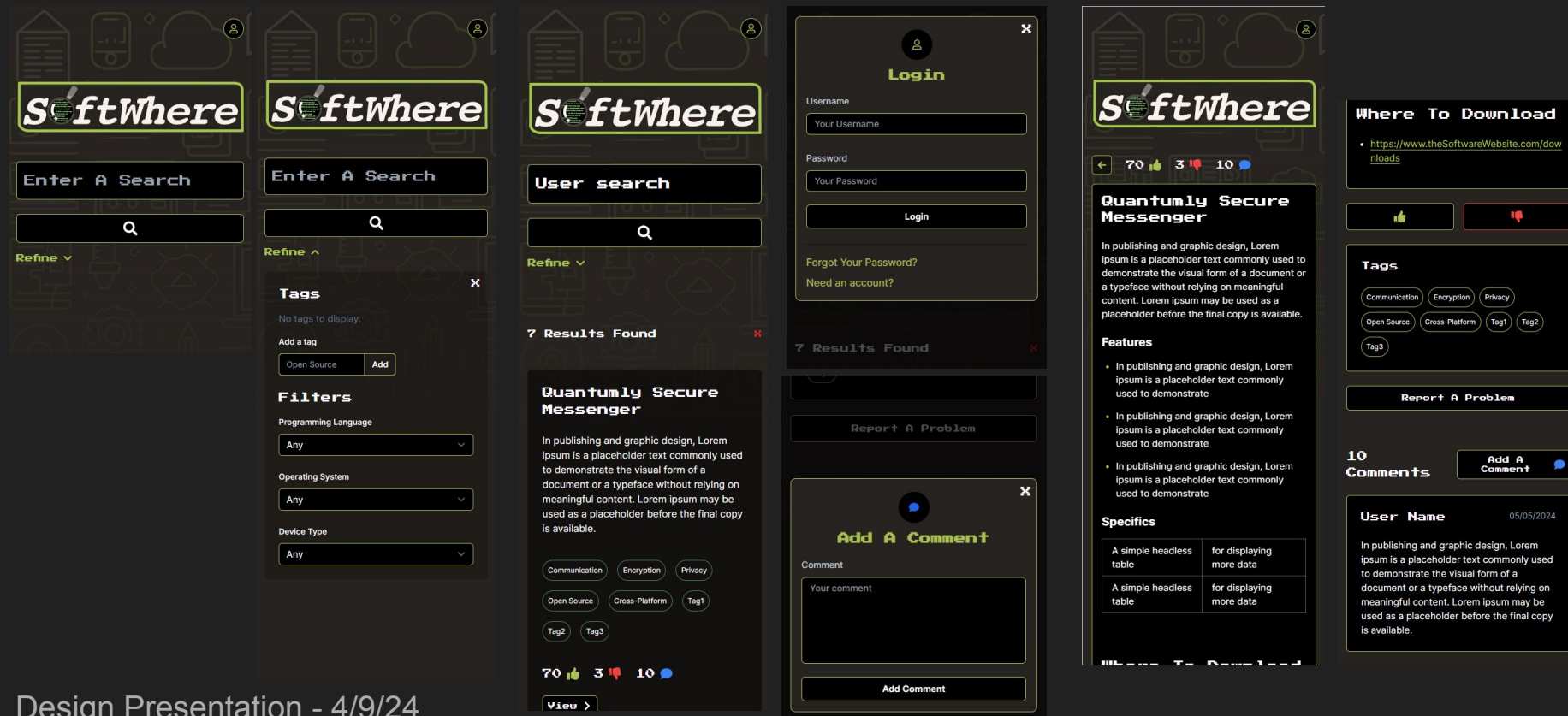
GUI Mockup - Homepage



GUI Mockup - Login



GUI Mockup - Mobile



Conclusion

We hope that our new community-driven web app “SoftWhere” will revolutionize the way that you search for software, reducing the tedium and difficulty of finding software online.

Citations / Sources

“Alternativeto - Crowdsourced Software Recommendations.” Alternativeto, alternativeto.net/. Accessed 21 Feb. 2024.

Montti, Roger. “Ex-Googler Answers Why Google Search Is Getting Worse.” Search Engine Journal, 30 Nov. 2022, www.searchenginejournal.com/whygoogle-search-is-getting-worse/472681/.

“Merge / Convert Multiple PDF Files into One PDF.” Stack Overflow, Jan. 2011, stackoverflow.com/questions/2507766/merge-convert-multiple-pdf-files-into-one-pdf.

Clark, Steve. “Best Free PDF Readers in 2024.” TechRadar, TechRadar pro, 16 Nov. 2021, www.techradar.com/news/the-best-free-pdf-reader.

Tyršina, Radu. “How Many Apps Does Microsoft Store Have: Year-by-Year Guide.” Windows Report, Microsoft, 4 Oct. 2023, windowsreport.com/state-windows-8-apps-windows-store/.

Koetsier, John. “Apple’s Mac App Store Adds Just 15 Apps per Month vs 15 per Minute for the IOS App Store.” Forbes, Forbes Magazine, 8 Nov. 2022, www.forbes.com/sites/johnkoetsier/2022/10/29/apples-mac-app-store-adds-just-15-apps-per-month-vs-15-per-minute-for-the-ios-app-store/?sh=4bcf711654df.

“Packages.” Debian, Debian, www.debian.org/distrib/packages. Accessed 21 Feb. 2024.

Appendix - User Stories

- As someone using SoftWhere, I would like to be able to use a search box to provide search strings.
- As someone using SoftWhere, I would like to be able to use a series of boolean operations to filter searches for software such as “OpenSource”, “Free”.
- As someone using SoftWhere, I would like to be able to provide feedback on a piece of software such as comments and thumbs up/down.
- As someone using SoftWhere, I would like to create a new entry on the website to define a new piece of software.
- As someone using SoftWhere, I would like to be able to log in with an account.
- As an administrator of SoftWhere, I would like to be able to administrate posts in a admin portal, such as banning users, and overriding community member actions.