

Software Design Document for The arqive

Version 2.0 approved

Prepared by Leslie Segovia, Erica Santos, Matthew Frias, Jesus Gonzalez, Kevin Kazaryan, Daniel Lee, Stewart McKenzie, Erica Payne, Bryan Sosa, Elio Vences

California State University, Los Angeles – The arqive

December 9, 2021

Table of Contents.....	pg 2
Revision History.....	pg 5
1. Introduction.....	pg 6
1.1. Purpose.....	pg 6
1.2. Document Conventions.....	pg 6
1.3. Intended Audience and Reading Suggestions.....	pg 6
1.4. System Overview.....	pg 7
2. Design Considerations.....	pg 7
2.1. Assumptions and dependencies.....	pg 7
2.2. General Constraints.....	pg 7
2.3. Goals and Guidelines.....	pg 8
2.4. Development Methods.....	pg 8
3. Architectural Strategies.....	pg 9
4. System Architecture.....	pg 9
4.1. DFD Level 0.....	pg 9
4.2. DFD Level 1.....	pg 10
5. Policies and Tactics.....	pg 11
5.1. Specific Products Used.....	pg 11
5.2. Requirements traceability.....	pg 11
5.3. Testing the software.....	pg 11
6. Detailed System Design.....	pg 11
6.1.1 User Account Requirements	pg 11
6.1.1 Responsibilities.....	pg 11
6.1.2 Constraints.....	pg 11
6.1.3 Composition.....	pg 11
6.1.4 Uses/Interactions.....	pg 11
6.1.5 Resources.....	pg 11
6.1.6 Interface/Exports.....	pg 12
6.2.1 Gamification Requirements	pg 12
6.2.1 Responsibilities.....	pg 12
6.2.2 Constraints.....	pg 12
6.2.3 Composition.....	pg 12
6.2.4 Uses/Interactions.....	pg 12
6.2.5 Resources.....	pg 12
6.2.6 Interface/Exports.....	pg 12
6.3.1 Map Requirements	pg 12
6.3.1 Responsibilities.....	pg 12
6.3.2 Constraints.....	pg 12
6.3.3 Composition.....	pg 12
6.3.4 Uses/Interactions.....	pg 12
6.3.5 Resources.....	pg 13
6.3.6 Interface/Exports.....	pg 13
6.4.1 Story Requirements	pg 13
6.4.1 Responsibilities.....	pg 13
6.4.2 Constraints.....	pg 13

6.4.3	Composition.....	pg 13
6.4.4	Uses/Interactions.....	pg 13
6.4.5	Resources.....	pg 13
6.4.6	Interface/Exports.....	pg 13
6.5.1	Platform Requirements	pg 13
6.5.1	Responsibilities.....	pg 13
6.5.2	Constraints.....	pg 13
6.5.3	Composition.....	pg 13
6.5.4	Uses/Interactions.....	pg 13
6.5.5	Resources.....	pg 14
6.5.6	Interface/Exports.....	pg 14
6.6.1	Multimedia Requirements	pg 14
6.6.1	Responsibilities.....	pg 14
6.6.2	Constraints.....	pg 14
6.6.3	Composition.....	pg 14
6.6.4	Uses/Interactions.....	pg 14
6.6.5	Resources.....	pg 14
6.6.6	Interface/Exports.....	pg 14
6.7.1	Security Requirements	pg 14
6.7.1	Responsibilities.....	pg 14
6.7.2	Constraints.....	pg 14
6.7.3	Composition.....	pg 14
6.7.4	Uses/Interactions.....	pg 14
6.7.5	Resources.....	pg 14
6.7.6	Interface/Exports.....	pg 15
6.8.1	User Role Requirements	pg 15
6.8.1	Responsibilities.....	pg 15
6.8.2	Constraints.....	pg 15
6.8.3	Composition.....	pg 15
6.8.4	Uses/Interactions.....	pg 15
6.8.5	Resources.....	pg 15
6.8.6	Interface/Exports.....	pg 15
6.9.1	FAQ Requirements	pg 15
6.9.1	Responsibilities.....	pg 10
6.9.2	Constraints.....	pg 10
6.9.3	Composition.....	pg 10
6.9.4	Uses/Interactions.....	pg 10
6.9.5	Resources.....	pg 10
6.9.6	Interface/Exports.....	pg 16
6.10.1	About Requirements	pg 16
6.10.1	Responsibilities.....	pg 16
6.10.2	Constraints.....	pg 16
6.10.3	Composition.....	pg 16
6.10.4	Uses/Interactions.....	pg 16
6.10.5	Resources.....	pg 16

6.11.6	Interface/Exports.....	pg 16
6.11.1	Support Requirements	pg 16
6.11.1	Responsibilities.....	pg 16
6.11.2	Constraints.....	pg 16
6.11.3	Composition.....	pg 16
6.11.4	Uses/Interactions.....	pg 16
6.11.5	Resources.....	pg 16
6.11.6	Interface/Exports.....	pg 16
6.12.1	Accessibility Requirements	pg 17
6.12.1	Responsibilities.....	pg 17
6.12.2	Constraints.....	pg 17
6.12.3	Composition.....	pg 17
6.12.4	Uses/Interactions.....	pg 17
6.12.5	Resources.....	pg 17
6.12.6	Interface/Exports.....	pg 17
6.13.1	Contact Us Requirements	pg 17
6.13.1	Responsibilities.....	pg 17
6.13.2	Constraints.....	pg 17
6.13.3	Composition.....	pg 17
6.13.4	Uses/Interactions.....	pg 17
6.13.5	Resources.....	pg 17
6.13.6	Interface/Exports.....	pg 17
7.	Detailed Lower level Component Design.....	pg 18
8.	Database Design.....	pg 18
9.	User Interface	
9.1.	Overview of User Interface.....	pg 19
9.2.	Screen Frameworks or Images.....	pg 19-20
9.3.	User Interface Flow Model.....	pg 21
10.	Requirements Validation and Verification	
11.	Glossary.....	pg 21
12.	References.....	pg 22

Revision History

Name	Date	Reason For Changes	Version
First Rough Draft	November 28, 2021	First document with no specific details	1.0
Second Draft	December 9, 2021	Peer reviewed and expanded on more details.	2.0

1. Introduction

1.1 Purpose

This document will provide the Software Design Specifications as part of the design plan for improving functionality and adding on features to The arqive. This document will expand on the functionality and features as described in the Software Requirements Specification (SRS). Each functionality that has been improved will be described in detail as well as additional features to be implemented. Overall, this document will expand on features the SRS presents along with design issues encountered.

1.2 Document Conventions

SDD	Software Design Document
SRS	Software Requirements Specification
User	The individual using The arqive website and/or mobile app
Administrator	Users with permission to other functions of the website and/or mobile app, not available to regular users

1.3 Intended Audience and Reading Suggestions

This document is intended for Software Developers to have an understanding of the design behind the basic functionalities and implemented features that the SRS describes.

1.4 System Overview

The arqive is a digital online storytelling map for LGBTQ+ stories that seeks to provide the full range of queer stories and geolocates them to digitally preserves the posts. Users have a safe platform where they can share personal, historical, and community stories, as well as have access to information about safe spaces. It is currently available as a web and mobile application.

2. Design Considerations

2.1 Assumptions and Dependencies

- OpenStreetMap
- ReactJS Framework
- User must be sharing their location
- Mobile device or Computer with a web browser
- iOS and/or Android device

2.2 General Constraints

- Privacy for users is not always guaranteed
- Too many story pins on the map may cause a slower load in for the user
- Javascript must be enabled in the user's web browser
- Slow internet connection may affect the users experience

2.3 Goals and Guidelines

- Provide a platform for the LGBTQ+ community to post and share stories, experiences and have access to resources.
- Security and protection from malevolent agents that might cause harm to the archive's users
- Implementing gamification to encourage and reward users for their activity on the app and/or web application.
 - o Have no ranking or competitive system that other social media platforms reward users for.
- Highlight and display local stories
- Ability for administrators and moderators to moderate flagged posts with the help of integrated content moderation.
- The mobile apps should implement additional functionality and be created with the established style of the archive website, according to the guidelines provided.

2.4 Development Methods

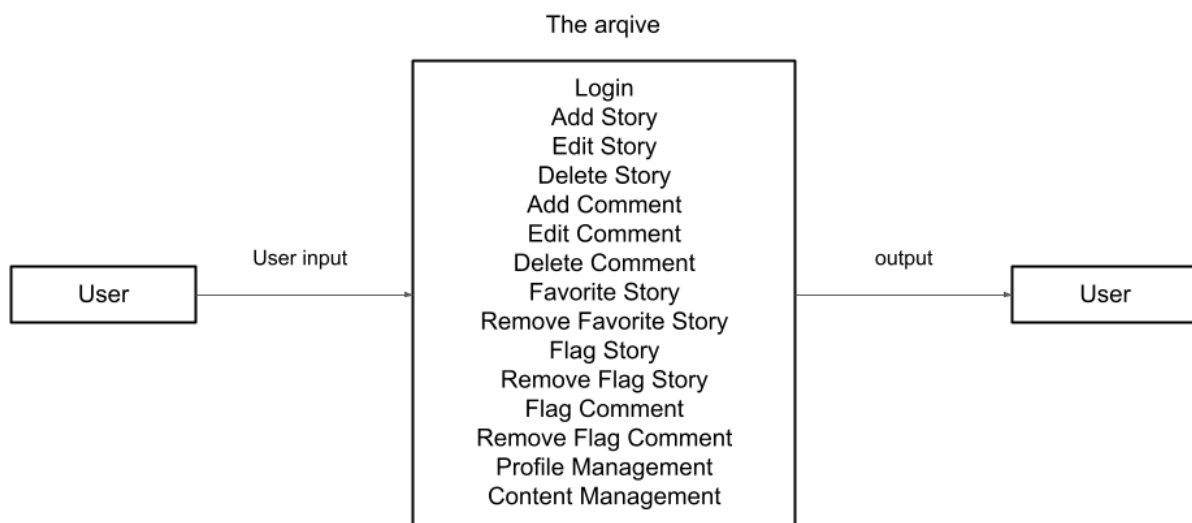
The agile development method was used in the design of this project. Members of the team handled bug fixes and deployed the changes after resolving them. To implement new features to The archive, we must take a waterfall development method by merging and refactoring the front-end and back-end code to 1 database. After multiple changes and implementations to the applications, the team then demonstrates the new changes to the sponsors to provide feedback and approval.

3. Architectural Strategies

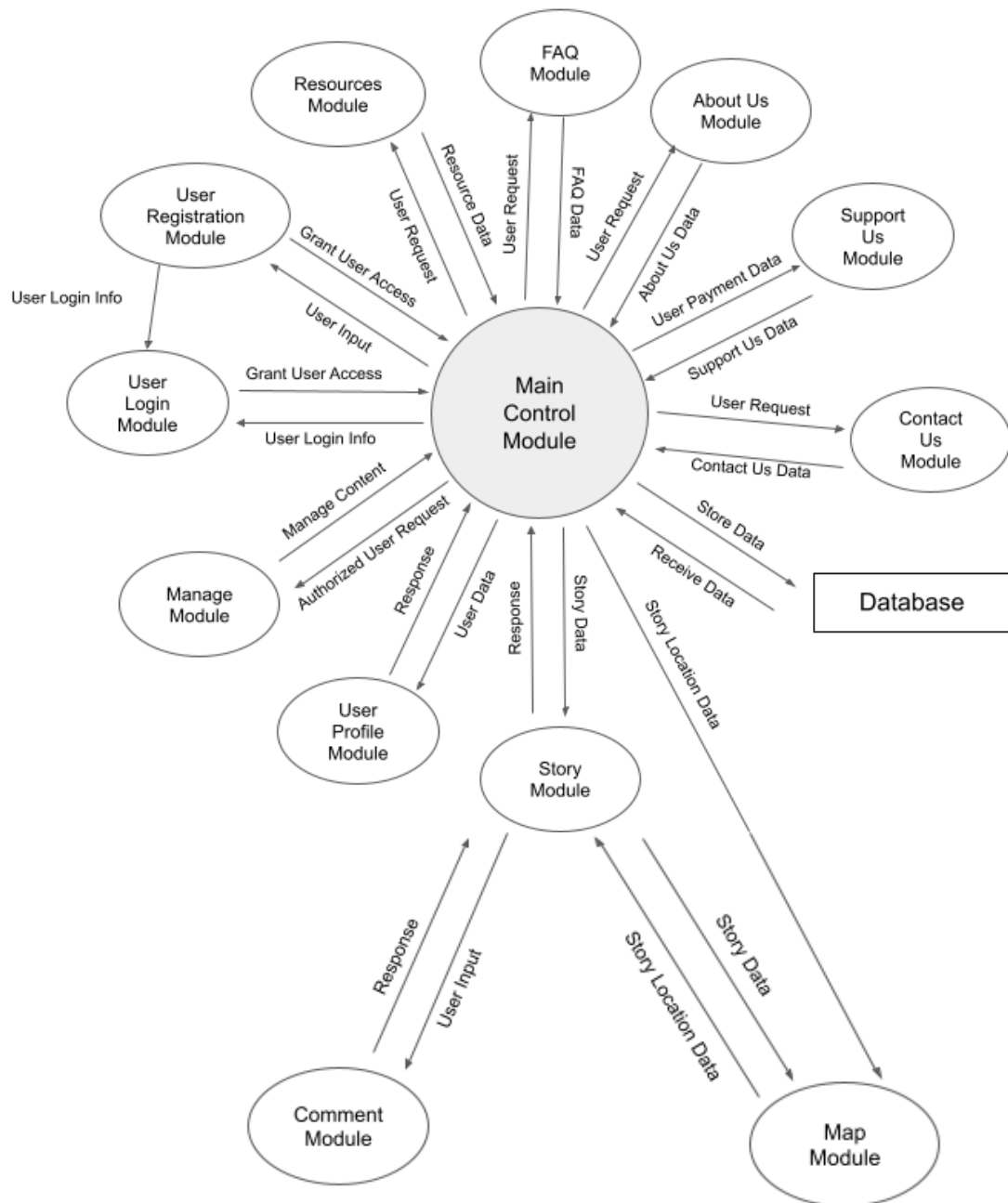
The applications of The arqive were built using Python, JavaScript, Django and React which resulted in having a flexible and responsive website. In addition, Django and React are libraries that are able to provide a more enhanced functionality to the website. The mobile app was built prior using React Native which helps to focus on one code base for both iOS and Android devices.

4. System Architecture

DFD Level 0



DFD Level 1



5. Policies and Tactics

5.1 Choice of which specific products used

PostgreSQL, Github, Digital Ocean, ReactJS, React Native, Django REST, OpenstreetMap

5.2 Plans for ensuring requirements traceability

Each developer will create test cases that will map to requirements on any effort they are working on.

5.3 Plans for testing the software

Testing the website and mobile app will be done by the developers, sponsors and third-party users asked to use the program to identify potential bugs or UI improvements. Testing the software may occur throughout the developmental process of both the website and mobile app.

6. Detailed System Design

6.1 User Account Requirements

6.1.1 Responsibilities

Separate the users into regular users, anonymous users, moderators, and administrators with unique roles.

6.1.2 Constraints

We must assume that the roles with more power do not abuse others with their status. We must also make sure that actions other than reading a post are restricted to registered users.

6.1.3 Composition

No current subcomponents.

6.1.4 Uses/Interactions

The user's role allows them to create or modify posts based on their role. Admin's have the same basic functions that users have but can moderate and delete any posts, comments and/or users that may have violated the code of conduct.

6.1.5 Resources

Modification of the database of pins is dependent on the user's role.

6.1.6 Interface/Exports

No current exports.

6.2 Gamification Requirements

6.2.1 Responsibilities

To establish a system to encourage users to be active on the app. Gamification will not consist of a competitive system but rather reward users for individual and unique activities they may encounter while using the app.

6.2.2 Constraints

We must assume that users with achievement badges understand that they are only a representation of the activity they did and do not hold a value against other users.

6.2.3 Composition

No current subcomponents.

6.2.4 Uses/Interactions

The user's experience with gamification will result in receiving badges after doing unique or consistent activity on the website or mobile app.

6.2.5 Resources

No current resources.

6.2.6 Interface/Exports

No current exports.

6.3 Map Requirements

6.3.1 Responsibilities

The map is able to show all pins from the database.

6.3.2 Constraints

The size of the map is set to be 100% of the viewing screen's size.

6.3.3 Composition

No current subcomponents.

6.3.4 Uses/Interactions

The map will interact with the database in order to show and store location based pins via form submission.

6.3.5 Resources

Digital Ocean PostgreSQL Development Database

6.1.6 Interface/Exports

No current exports.

6.4 Story Requirements

6.4.1 Responsibilities

The story page shows all the data of the pin. It allows users to favorite, report, or comment on the post. Users can create anonymous posts to hide their identity.

6.4.2 Constraints

Content containing explicit material will be flagged for moderation. Moderators will then decide on the appropriate action for flagged content.

6.4.3 Composition

Favoriting a post

6.4.4 Uses/Interactions

Users will be able to view, favorite, bookmark or report the stories posted.

6.4.5 Resources

Digital Ocean PostgreSQL Development Database

6.4.6 Interface/Exports

No current exports.

6.5 Platform Requirements

6.5.1 Responsibilities

This allows the user to access the site with different browsers like Microsoft Edge, Firefox, Chrome, etc.

6.5.2 Constraints

Not all available browsers will be covered.

6.5.3 Composition

No current composition.

6.5.4 Uses/Interactions

Users will be able to access and use the platform via the web and on mobile.

6.5.5 Resources

No specific resources.

6.5.6 Interface/Exports

No current exports.

6.6 Multimedia Requirements

6.6.1 Responsibilities

Allows users to embed pictures, videos, and other media onto their posts.

6.6.2 Constraints

The file size must be reasonable.

6.6.3 Composition

No current composition.

6.6.4 Uses/Interactions

Users can upload media about the pin to let other people view it.

6.6.5 Resources

No specific resources.

6.6.6 Interface/Exports

It may be a part of the pin creation process.

6.7 Security Requirements

6.7.1 Responsibilities

The security will protect the site, its data and users.

6.7.2 Constraints

Security can not be fully guaranteed but the security implemented will still offer safety.

6.7.3 Composition

No current composition.

6.7.4 Uses/Interactions

Ideally, the security would not be used for anything major. The users will not interact with the security unless the user causes a problem.

6.7.5 Resources

No specific resources.

6.7.6 Interface/Exports

No current exports.

6.8 User Role Requirements

6.8.1 Responsibilities

User roles will give people a sense of membership and provide moderation of content.

6.8.2 Constraints

Currently, user roles only include registered users, administrators, and moderators

6.8.3 Composition

No current composition.

6.8.4 Uses/Interactions

Allows users to have specific privileges within the application.

6.8.5 Resources

Database

6.8.6 Interface/Exports

No current exports.

6.9 FAQ Requirements

6.9.1 Responsibilities

Provide users with information regarding the site.

6.9.2 Constraints

Limited to what Administrators post.

6.9.3 Composition

No current composition.

6.9.4 Uses/Interactions

Allows Administrators to post FAQs and Users to read them.

6.9.5 Resources

Database

6.9.6 Interface/Exports

No current exports.

6.10 About Requirements

6.10.1 Responsibilities

Provide users with information on the mission of the archive.

6.10.2 Constraints

Administrators are the only ones allowed to post in the “About Us” section.

6.10.3 Composition

No current composition.

6.10.4 Uses/Interactions

Allows administrators to edit the “About Us” and users to read them.

6.10.5 Resources

Database

6.10.6 Interface/Exports

No current exports.

6.11 Resources Requirements

6.11.1 Responsibilities

Provides users with access to various valuable resources.

6.11.2 Constraints

We may not provide all available hotlines and websites are not enough.

6.11.3 Composition

No current composition.

6.11.4 Uses/Interactions

Hotlines and Websites that mainly aid LGBTQ+ individuals will be posted for users to use.

6.11.5 Resources

No current resources.

6.11.6 Interface/Exports

No current exports.

6.12 About Requirements

6.12.1 Responsibilities

Allow users with disabilities to use the web app to its full potential.

6.12.2 Constraints

We are using UserWay Widget, so we do not have control over their functionalities.

6.12.3 Composition

No current composition.

6.12.4 Uses/Interactions

Users will be able to use the UserWay Widget which allows them to have text spoken aloud among other things.

6.12.5 Resources

UserWay Widget

6.12.6 Interface/Exports

No current exports.

6.13 Contact Us Requirements

6.13.1 Responsibilities

Allows users to contact us directly through email.

6.13.2 Constraints

No current constraints.

6.13.3 Composition

No current composition.

6.13.4 Uses/Interactions

The user will input a message, and if they want a response, they could provide an email (optional).

6.13.5 Resources

SMTP, Django Documentation

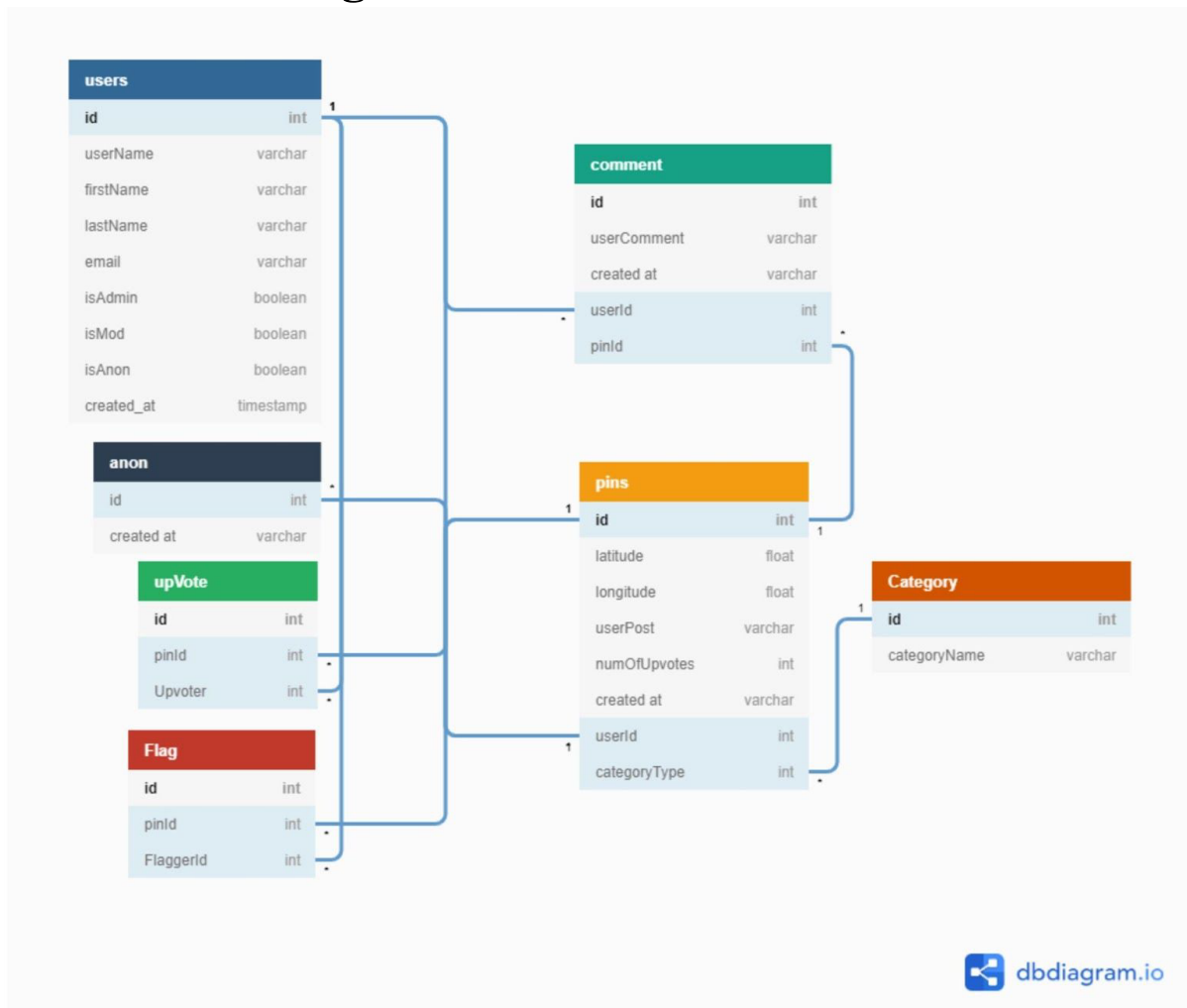
6.13.6 Interface/Exports

A simple and secure way to contact us

7. Detailed Lower level Component Design

Refer to Section 4 of this Document.

8. Database Design



9. User Interface

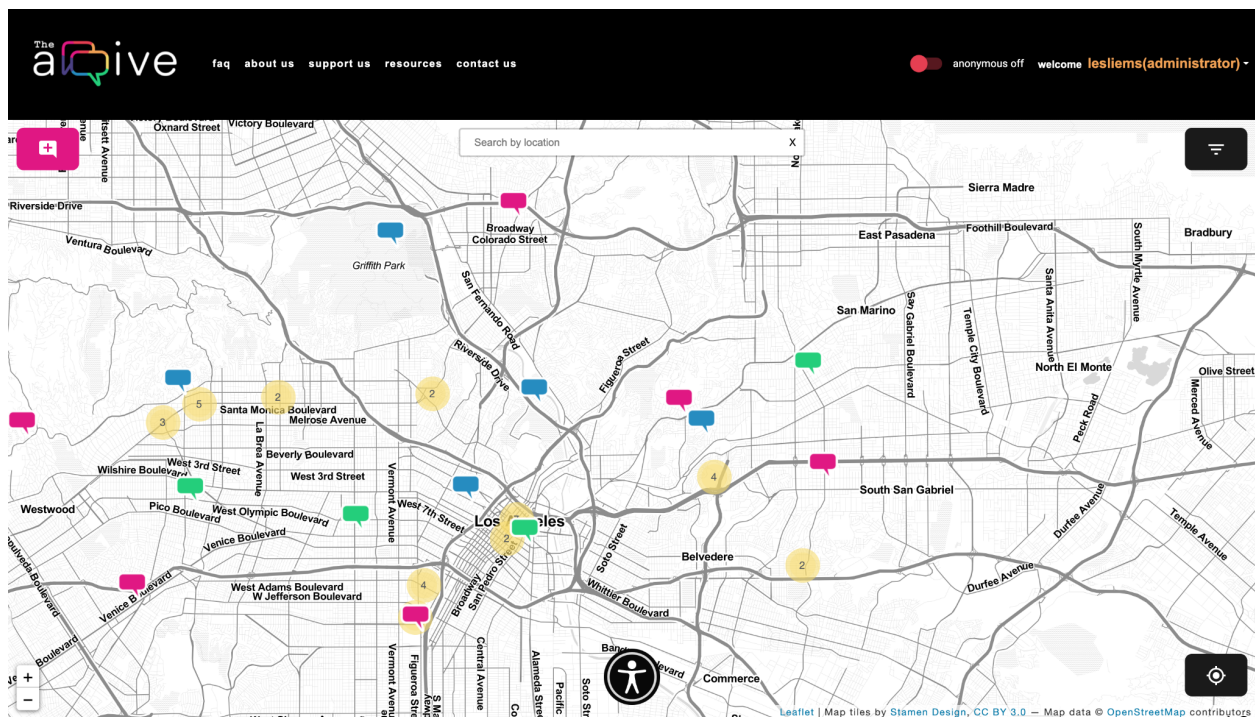
9.1 Overview of User Interface

The user will access the main content of the site by navigating to the homepage (<https://thearchive.com>). Within this page, users can post pins on the map by category (personal, historical, or community). Users can also access other stories by navigating around the map and clicking on pins, which will direct them to the individualized story page of the selected pin. Inside this individualized story page, users can read more detailed information about the pin, i.e., the story author. If clicked, the author's name leads to their user profile which displays their profile picture, name, biography, and all previous stories created.

In the site header, users have the ability to click the site logo to navigate back to the homepage, click the search button to search the website, click the login/logout button to login or logout, or click the Register button to sign up on the site. Also located in the site header are links that take users to the About Us, FAQs, Help, Contact Us, or Support Us pages.

In the About Us page, users can read about the archive and its mission. In the FAQs page, users can read frequently asked questions and their respective answers. Within the Help page, users can access useful information and resources that help support the LGBTQ+ community. In the Contact Us page, users can submit a message directly to the archive email. Lastly, within the Support Us page, users can support the website by sharing the #thearchive on other social media.

9.2 Screen Frameworks or Images



The **archive**
[faq](#) [about us](#)

anonymous off
welcome **lesliems(administrator)**

add a story

address

locality (city, township, etc.)

region (state, province, etc.)

country

postcode

title

**please enter a story title*

category personal

description

**please enter a story description*

File Edit View Insert Format

The **archive**
[faq](#) [about us](#) [support us](#) [resources](#) [contact us](#)

anonymous off
welcome **lesliems(administrator)**

"The Princess" and "The Duchess"

griffith park los angeles ca

Posted by: **Anonymous**

03/17/1968 - no end date

Two drag queens known as "The Princess" and "The Duchess" held a St. Patrick's Day party at Griffith Park, a popular cruising spot and a frequent target of police activity. More than 200 gay men socialized through the day.

Originally a place to protest police entrapment, this event led to a "gay-in" which took place on May 30, 1968.

9.3 User Interface Flow Model

Homepage → Story Page
Story Page → Story Page
Story Page → User Profile
Site Header → Login
Login → Register
Login → Homepage
Site Header → Register
Register → Login
Register → Homepage
Site Header → Registration
Site Header → FAQ
Site Header → About Us
Site Header → Contact Us
Site Header → Help Site
Header → Support Us
Site Header → Homepage
Site Header → Profile
Profile → Settings

10. Requirements Validation and Verification

Not applicable at this time.

11. Glossary

Django REST	Toolkit for building Web APIs in Python
DFD	Data Flow Diagram
LGBTQ+	Lesbian, Gay, Bisexual, Transgender, Queer, and more
SRS	Software Requirement Specification
Openstreetmap	An editable geographic database of the world
React JS	An Open source front end framework for developing websites

React Native	Open source mobile application framework for iOS and Android devices
Digital Ocean	Cloud infrastructure provider
SciKit-learn	A Machine learning framework

12. References

Brad Appleton <brad@bradapp.net>

<http://www.bradapp.net>

https://www.cs.purdue.edu/homes/cs307/ExampleDocs/DesignTemplate_Fall08.doc

The arqive's 2020 Development Team (SRS)

<https://csns.cysun.org/department/cs/project/resource/view?projectId=7808908&resourceId=7837898>