Department of Computer Science & Engineering
The University of Texas at Arlington

Detailed Design Specification
CSE 4317: Senior Design II
Spring 2023

Team ALL STAARS
STAR Website

Ethan Nguyen
Sumaita Sabaha
Kabigya Raj Karki
Victor Ong
Jeremy Jones

Revision History

Revision	Date	Author(s)	Description
1.0	3.24.2023	EN, SS, KK, VO, JJ	official release

Contents

1 Introduction 5

2 System Overview 5

3 Front-end Layer Subsystems 6

3.1 Layer Hardware 6

3.2 Layer Operating System 6

3.3 Layer Software Dependencies 6

3.4 Public Pages 6

3.5 Admin Pages 7

4 Back-end Layer Subsystems 9

4.1 Layer Hardware 9

4.2 Layer Operating System 9

4.3 Layer Software Dependencies 9

4.4 Database Subsystem 9

4.5 Access Control Subsystem 10

4.6 Third-Party Integration 11

5 Appendix A 13

List of Figures

1 System Architecture 5
2 Public pages layer diagram 6
3 Admin Pages subsystem diagram 7
4 Back-end Layer Diagram 10
5 Access Control Diagram 11
6 Third-party integration Diagram 12

List of Tables

1 Introduction

The project is a website using Firebase as a platform. The website will have many customize-able aspects that do not require technical skills to modify. The goal is an easy-to-maintain professional internet presence for our sponsor.

2 System Overview

There are two layers, the front-end and the back-end. The back-end encompasses all of the data processing hidden from users, while the front-end is composed of the user-visible display. The database subsystem covers storage, the access control subsystem determines who can view the admin pages, and third-party integration allows for social-media interaction and payment processing. The public pages subsystem represents the pages accessible by anyone, while the admin pages require a validated administrator to be viewed.

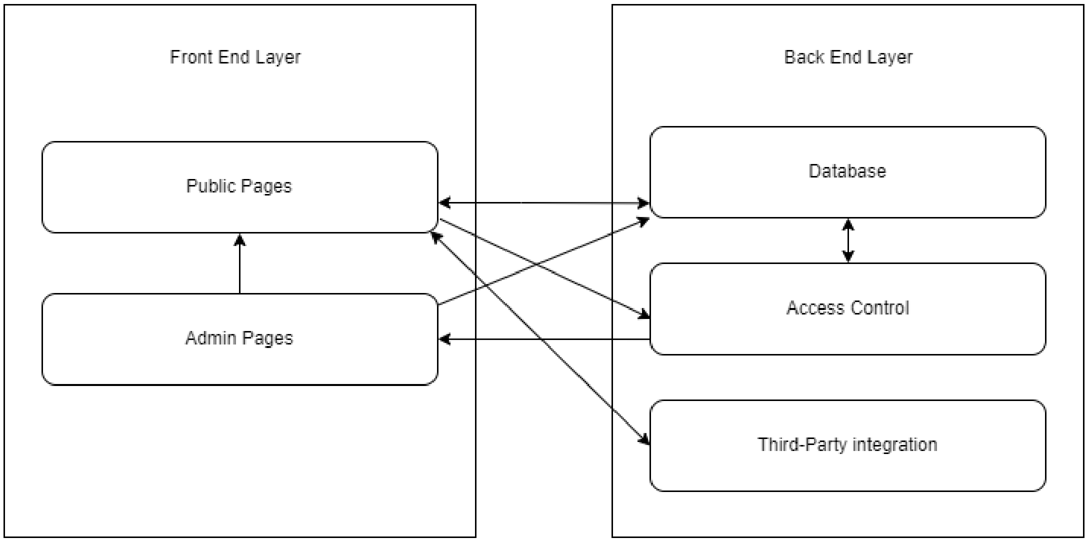

Figure 1: System Architecture

3 Front-end Layer Subsystems

3.1 Layer Hardware

This is a web application so it doesn't require any specific hardware component. It can be accessed from any device.

3.2 Layer Operating System

There are no operating systems required for our end product. It can be accessed through any modern browser with any operating system that supports it.

3.3 Layer Software Dependencies

The UI design and the visual aspects of the website depends on JavaScript, HTML, CSS and a library such as JQuery.

3.4 Public Pages

These pages are the publicly accessible pages that consist mostly of viewable content with very little input.

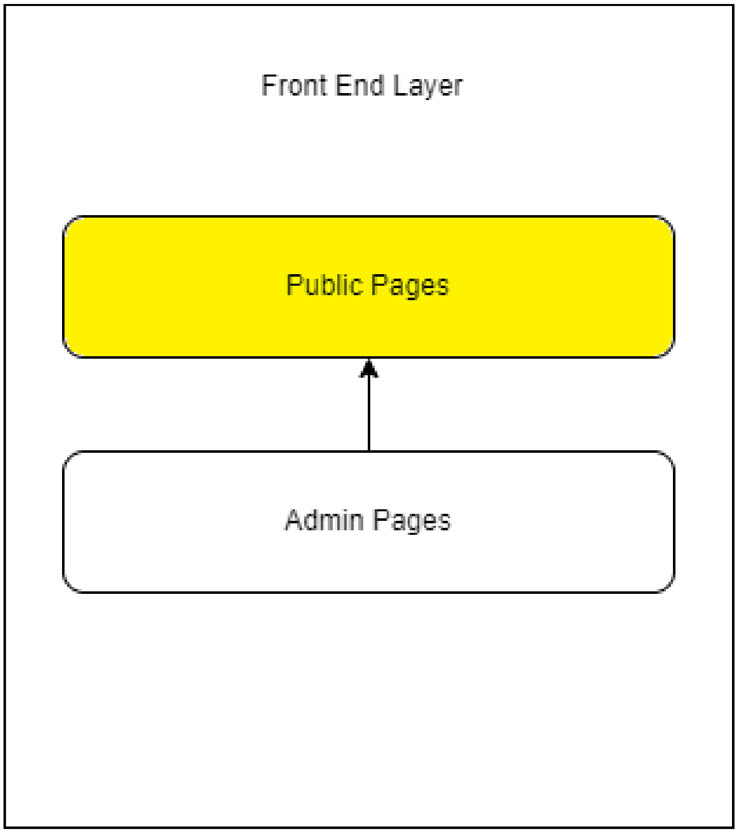

Figure 2: Public pages layer diagram

3.4.1 Subsystem Hardware

The public pages are compatible with any hardware capable of using a browser.

3.4.2 Subsystem Operating System

The public pages are not dependant on an operating system. Any operating system that can run a browser can display the public pages.

3.4.3 Subsystem Software Dependencies

The public pages dependencies are Firebase, ConstantContact, Javascript, HTML, and CSS.

3.4.4 Subsystem Programming Languages

The programming languages we are using are Javascript, HTML, and CSS.

3.4.5 Subsystem Data Structures

Front-end development usually uses various data structures such as arrays, strings, maps, objects and booleans.

3.4.6 Subsystem Data Processing

The public pages receive input from public users who sign up for the newsletter or want to send an email. The page will take input from the user and then provide it to the Third-party Integration subsystem.

3.5 Admin Pages

These pages are the admin-only pages and the admin version of the public pages. These pages allow the admin users to customize the website.

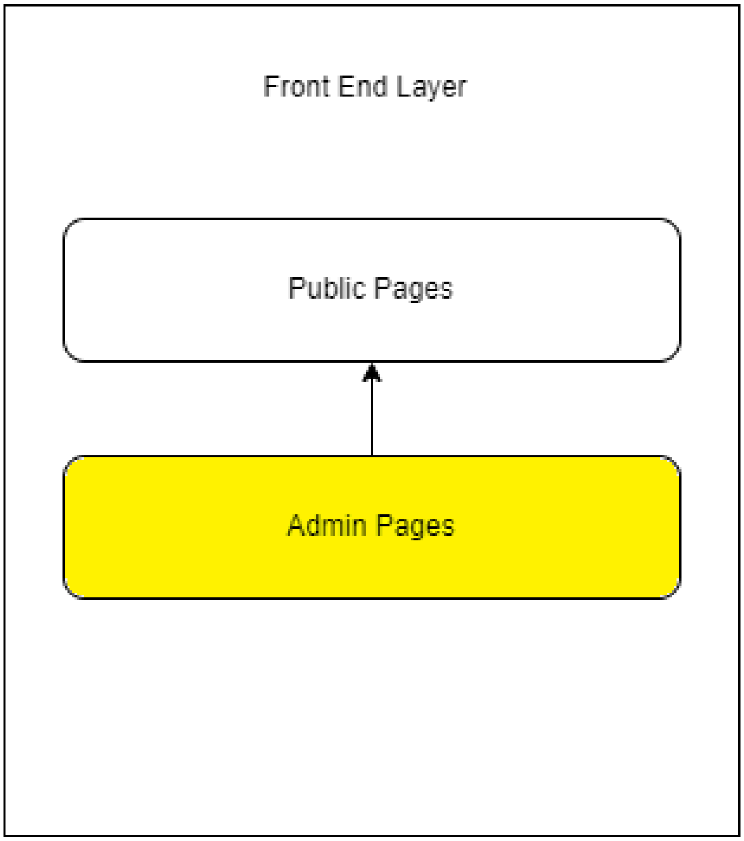

Figure 3: Admin Pages subsystem diagram

3.5.1 Subsystem Hardware

The admin pages are compatible with any hardware capable of using a browser.

3.5.2 Subsystem Operating System

The admin pages are not dependant on an operating system. Any operating system that can run a browser can display the admin pages.

3.5.3 Subsystem Software Dependencies

The admin page dependencies are Firebase, Javascript, HTML, and CSS.

3.5.4 Subsystem Programming Languages

The programming languages we are using are Javascript, HTML, and CSS.

3.5.5 Subsystem Data Structures

Front-end development usually uses various data structures such as arrays, strings, maps, objects and booleans.

3.5.6 Subsystem Data Processing

The admin pages have submission fields to submit new page elements to the database, as well as submission fields for admin settings.

4 Back-end Layer Subsystems

4.1 Layer Hardware

We are building a online web app which has no physical components. We do interact with physical components such as making API calls to our Firebase or making API calls to Social Media platforms and interacting with their servers but because we can not manage those physical servers we will exclude them here.

4.2 Layer Operating System

Our end product will not rely on any one operating system. Development will Happen on a windows operating System, while end users can open any browser and be able to reach our domain.

4.3 Layer Software Dependencies

We unfortunately are not using and framework to develop but we are relying on a few libraries such as

- The bootstrap library for CSS and styling
- Firebase Library / API calls
- Firebase Website hosting
- Google Fonts for HTML styling
- JQuery Scripts Library

4.4 Database Subsystem

The database subsystem covers all of the persistent storage required for the function of the website.

4.4.1 Subsystem Hardware

Our team does not maintenance the hardware required to keep the Firebase servers up. We are utilizing the cloud service so we are not responsible for the hardware or informed of the hardware specifications.

4.4.2 Subsystem Operating System

No specific operating system is required to run, maintain, call, or store any information to or from our Firebase.

4.4.3 Subsystem Software Dependencies

Our program heavily relies on the Firebase libraries and API calls. Our authentication comes from their library and function calls as well as our real time database storing and cloud storage. Not only that but our Google analytics is also linked to the Star Sponsorship Firebase.

4.4.4 Subsystem Programming Languages

Our program is a majority JavaScript and HTML with CSS to help with styling. These are the only languages we use currently.

Our database has the option to be implemented with multiple languages, include the standard JavaScript. There is documentation on their website on how to get started using Firebase with JavaScript and even incorporate it into HTML.

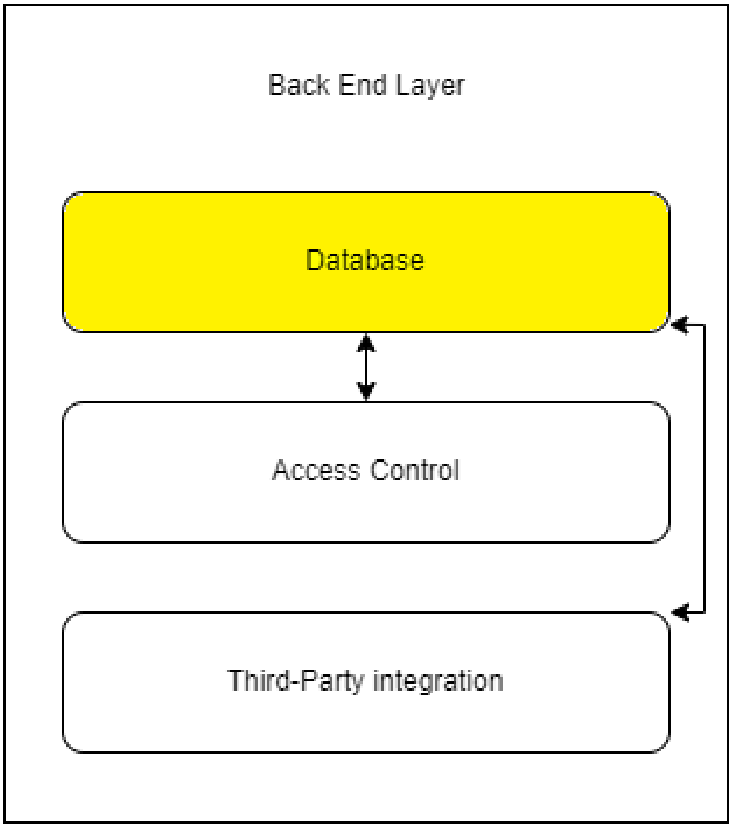

Figure 4: Back-end Layer Diagram

4.4.5 Subsystem Data Structures

We store all our data on either the Real Time database or the firestore. Both have its benefits. The realtime database is fast and doesn't need to hold extremely large amounts of data while the firestore Can hold much larger data but is not nearly as quick as the Real time Database. In addition, there is a cloud storage for media files.

We currently have no schema for the data as we are still developing the website.

4.4.6 Subsystem Data Processing

Our data processing is the standard CRUD operation. We have our edit, create, delete and retrieve. This will be all the necessary data processing for interacting with out data on the Firebase. Other parts of the program may rely on the Firebase libraries which we will not keep track of here

4.5 Access Control Subsystem

This figure shows that different sub layer and modules that are involved in the front-end and back-end layers of the application. The front end is divided into different modules related to main functionality of the web app, admin page and public page. The back end is separated into three layers data base, access control and third party integration for interacting with the front-end.

4.5.1 Subsystem Hardware

We don't have any subsystem hardware to use for the access control.

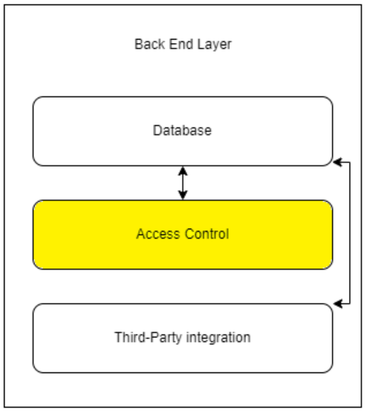

Figure 5: Access Control Diagram

4.5.2 Subsystem Operating System

Not any specific operating system used for the access conrol.

4.5.3 Subsystem Software Dependencies

The software dependencies of the subsystem are: Firebase, VS code,bootstraps library for CSS and styling and JavaScript libraries

4.5.4 Subsystem Programming Languages

The programming languages we use for this projects are: HTML, JavaScript and we used CSS for the styling purpose.

4.5.5 Subsystem Data Structures

The data structure of the system often interconnected and work together to implement access control system. Using these data structure, access control system can provide some sort of security and control over who can access sensitive data or resources of the user from the data base. The access control determines which part of the database will be accessible to the public page and admin page simultaneously, by regulating the data access. In access control, data structure are used to represent the various element of the access control system, such as public users information, admin information, donating transaction information and many more.

4.5.6 Subsystem Data Processing

Data processing in access control involves management of the data that are used to enforce access control policies. This task includes identifying users and determining the level of access based on their role and permission to prevent unauthorized access. The data processing is performed by making API call or CRUD operation. To access the data, we retrieve it from the database that we have collected from Firebase.

4.6 Third-Party Integration

Third-Party Integration in our product allows our application to access another programming interface which allows us features such as displaying information stored in the database and posting from social media platforms to our website without developing our own processes.

4.6.1 Subsystem Hardware

No hardware components are being used for our product since it is completely software.

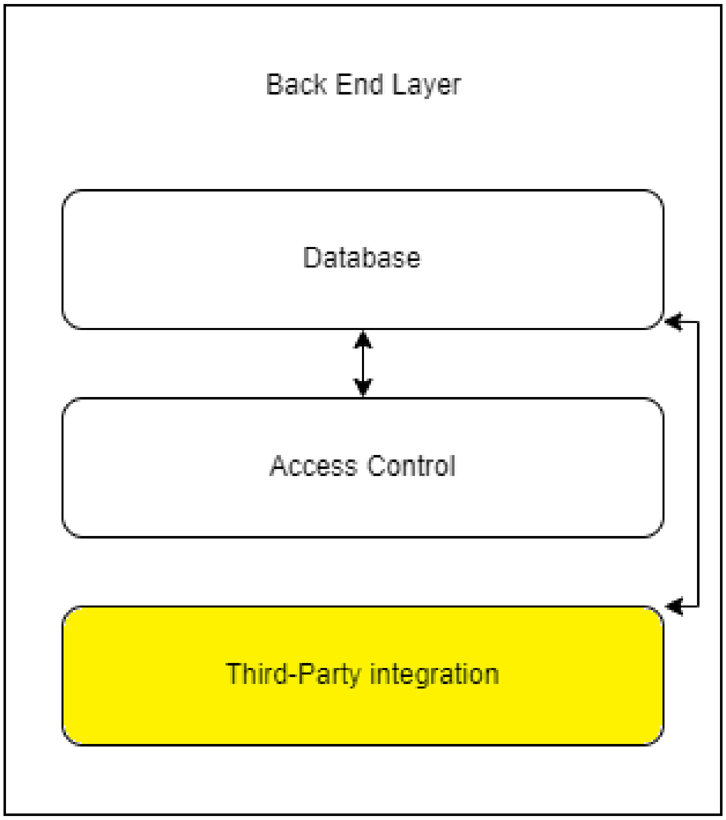

Figure 6: Third-party integration Diagram

4.6.2 Subsystem Operating System

No operating system for our subsystem.

4.6.3 Subsystem Software Dependencies

The subsystem software dependencies are Firebase, ConstantContact, API Calls.

4.6.4 Subsystem Programming Languages

The programming languages we use are Javascript, HTML, and CSS.

4.6.5 Subsystem Data Structures

Information is stored through our Database which is accessed when called. There may be additional information stored through a Third-Party program (such as ConstantContact) which can hold customer's information.

4.6.6 Subsystem Data Processing

We are making API calls in our subsystem for Third-Party Integration. We present data in our product by accessing the database which we call collected data from Firebase. Other features such as social media posting/ConstantContact will most likely require API calls but are not finalized yet.

5 Appendix A

Include any additional documents (CAD design, circuit schematics, etc) as an appendix as necessary.

References