

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
THE UNIVERSITY OF TEXAS AT ARLINGTON**

**PROJECT CHARTER
CSE 4316: SENIOR DESIGN I
FALL 2022**



**ALL STAAR
STAAR SPONSORSHIP WEBSITES**

**ETHAN NGUYEN
VICTOR ONG
JEREMY JONES
SUMAITA SABAHA
KABIGYA RAJ KARKI**

REVISION HISTORY

Revision	Date	Author(s)	Description
0.1	10.01.2015	GH	document creation
0.2	10.05.2015	AT, GH	complete draft
0.3	10.12.2015	AT, GH	release candidate 1
1.0	10.20.2015	AT, GH, CB	official release
1.1	10.31.2015	AL	added design review requests

CONTENTS

1 Introduction	5
2 System Overview	6
2.1 Layer Database Description	6
2.2 Layer Back End Computing Description	6
2.3 Layer Front End User Interface Description	6
3 Subsystem Definitions & Data Flow	7
4 Front End Layer Subsystems	8
4.1 Public Pages	8
4.2 Admin Pages	9
5 Back End Layer Subsystems	11
5.1 Database Management System(DBMS)	11
5.2 Third party integration	12
5.3 Access Control	13

LIST OF FIGURES

1	A simple architectural layer diagram	6
2	A simple data flow diagram	7
3	Example subsystem description diagram	8
4	Example subsystem description diagram	10
5	Database subsystem description diagram	11
6	Example subsystem description diagram	13
7	Access Control diagram	14

LIST OF TABLES

2	Subsystem interfaces	9
3	Subsystem interfaces	10
4	Database Management System interfaces	12
5	Subsystem interfaces	13
6	Access Control interfaces	14

1 INTRODUCTION

The STAR Sponsorship website product concept is a system that will allow users to browse and gather information about the STAR Sponsorship program. User will be able to donate and become a sponsor or read about stories on the website. The scope of the project includes developing a user friendly front end interface and finishing the functions of the back end which will include, secure donations, access control for admins and admin editing features, as well as database rule setting. The main key requirements are that we finished the functionality of the website as well as add the admin features.

2 SYSTEM OVERVIEW

Our system overview will include the front end user facing side of the website and the back end computing/functions. The front end will include the user interface and all visuals for users to interact with. For example, the home page should be the first page the user is directed too and they are able to interact with any buttons to move to any of the different pages. The back end will include data storage, donation features, and other items such as webpage editing.

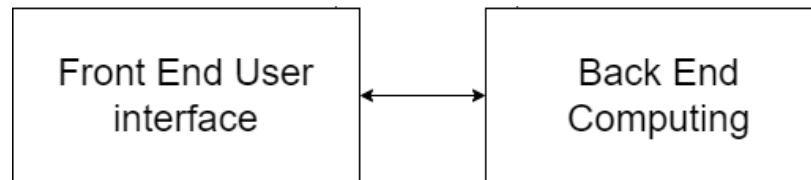


Figure 1: A simple architectural layer diagram

2.1 LAYER DATABASE DESCRIPTION

The database should be hidden from end users and should not have any interface to interact with. The database is only here to store data and display data when admins access it through the website. The database needs to have set rules for security purposes and will be used to track website traffic as well as keep the website pages saved so they can be edited by admins in the future. They should also be able to keep track of simple stored data such as user accounts, admin accounts, emails, and any information on pages necessary.

2.2 LAYER BACK END COMPUTING DESCRIPTION

The backed of the website includes the functionality and computing. The features needed will be basic page functions like buttons to interact with or input fields for users to fill out forms. Other features include page editing for admins and social media posting. All of these features and functions will tie in with the front end user interface. Inputs will be taken in as a form format and these inputs are what are used for these back end features. We tie these layers together by having one js file for every html file and link them together.

2.3 LAYER FRONT END USER INTERFACE DESCRIPTION

The front end of the website includes the user interface and what the end user will see and interact with. These features are mostly finished. The pages should allow users to traverse the website seamlessly and should be easy to use and follow. It is critical that no page should crash and if errors do occur, it should pull up a error page instead.

3 SUBSYSTEM DEFINITIONS & DATA FLOW

This section breaks down your layer abstraction to another level of detail. Here you graphically represent the logical subsystems that compose each layer and show the interactions/interfaces between those subsystems. A subsystem can be thought of as a programming unit that implements one of the major functions of the layer. It, therefore, has data elements that serve as source/sinks for other subsystems. The logical data elements that flow between subsystems need to be explicitly defined at this point, beginning with a data flow-like diagram based on the block diagram.

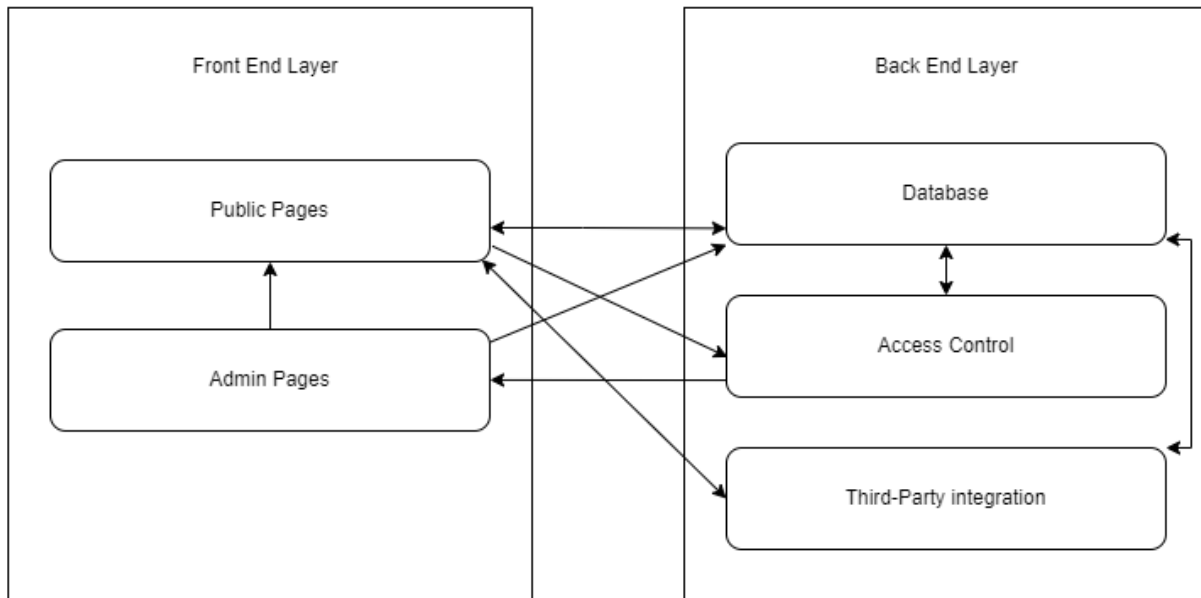


Figure 2: A simple data flow diagram

4 FRONT END LAYER SUBSYSTEMS

In this section, the layer is described in some detail in terms of its specific subsystems. Describe each of the layers and its subsystems in a separate chapter/major subsection of this document. The content of each subsystem description should be similar. Include in this section any special considerations and/or trade-offs considered for the approach you have chosen.

4.1 PUBLIC PAGES

The public pages are what the end users will see when they visit our website. These are the pages users will navigate and interact with to learn about the star sponsorship program. These pages do not include any sort of privileges to allow users to make any changes to them. They should only be interacted in a way that displays and allows for user donations if they want.

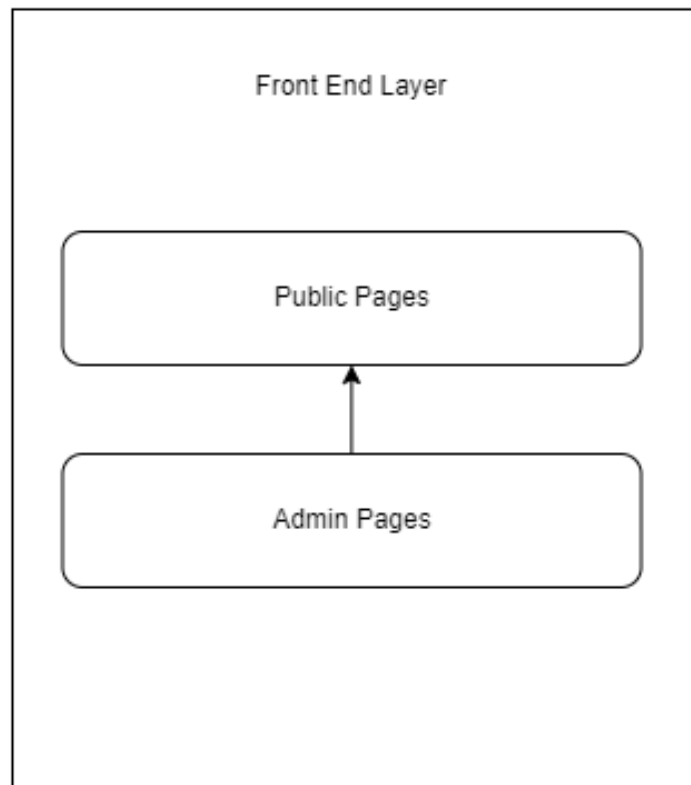


Figure 3: Example subsystem description diagram

4.1.1 ASSUMPTIONS

We are assuming these pages are the first pages the end users will see and all pages are subject to change based on what our client wants.

4.1.2 RESPONSIBILITIES

The responsibilities of our public pages is to allow users to navigate the Star Sponsorship website easily and allow users to learn or gather information regarding the program. Some features of our public pages are the Navigation bar to allow quick access to other pages and the footer to display contact information, social media links and other information. All the public pages will be linked to the home page and Navigation bar so navigation is effortless and easily understandable.

4.1.3 SUBSYSTEM INTERFACES

The public pages will allow users to interact with the page with buttons and form inputs. The table below shows each output given on the depending input for each page

Table 2: Subsystem interfaces

ID	Description	Inputs	Outputs
#01	Mouse click on Nav bar	Mouse click	Redirects to page
#02	Mouse click on input field (forms)	Mouse click	Allows for keyboard input
#03	Mouse click on footer	Mouse Click on social media	redirects user to social media
#04	Mouse click on footer	Mouse Click on newsletter	redirects user to newsletter page
#05	Mouse click on Success stories	Mouse click	Redirects to success story
#06	Mouse click on embedded objects (youtube video)	Mouse click	Plays youtube video
#01	Mouse click on FAQ Dropdown	Mouse click	opens answer to FAQ
#01	Mouse click on Newsletter	Mouse click	Opens the newsletter
#01	Mouse click on Donation button	Mouse click	Redirects to donation page

4.2 ADMIN PAGES

The admin pages are for the users who log in with user admin credentials should see. These pages interact and behave exactly like the public pages with the exception that every page should have an extra form for input. This form allows the admin user to edit the page.

4.2.1 ASSUMPTIONS

We are assuming these pages are only available once the user has logged in using admin credentials. We also assume all pages interact the same as public pages with the exception of the extra details only available to admins such as page editing forms and page creation/deletion

4.2.2 RESPONSIBILITIES

The responsibilities of our admin pages is to allow admins to have extra features not given to the public users. These features include page editing, page creation, page deletion, website traffic statistics, and social media posting. Each of these extra features should only appear once the admin account has been authorized and the account is signed into.

4.2.3 SUBSYSTEM INTERFACES

The Admin pages are assumed to interact and behave exactly the same as the public pages so we will omit the already added inputs and outputs and solely focus on the new features and behaviors.

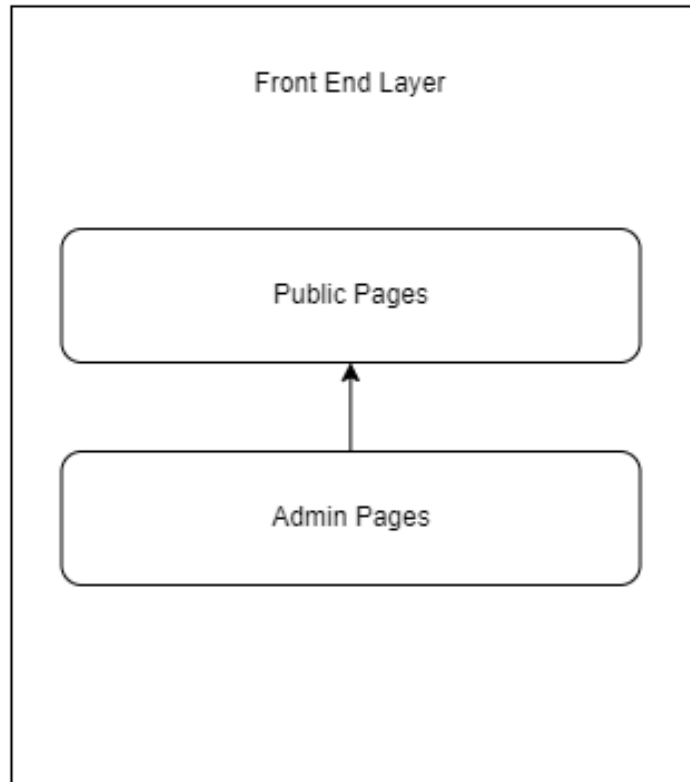


Figure 4: Example subsystem description diagram

Table 3: Subsystem interfaces

ID	Description	Inputs	Outputs
#01	Mouse click on Editing form	Mouse click	Allows for keyboard input and changes
#02	Mouse click on page creation	Mouse click	redirects to page creation
#03	Mouse click on page deletion	Mouse click	redirects to page deletion
#04	Mouse click on social media posting	Mouse Click	allows for social media post

5 BACK END LAYER SUBSYSTEMS

The back end layer is composed of the functions and components that will make our website function properly. Without these components, our web page would just be still images and would not interact with the user in any way. This layer should link to the front end layer to create one cohesive webpage.

5.1 DATABASE MANAGEMENT SYSTEM(DBMS)

Database Management System is the system software which will help to manage and create the data base. It is the software which will manage the whole database to store the data in the organized way. Moreover, it will provide the instruction to the database to store the data. Also, it will allow data base to store or extract the data from it. Database Management System is the software which, will instruct and guide the database to be store the data in the proper format.

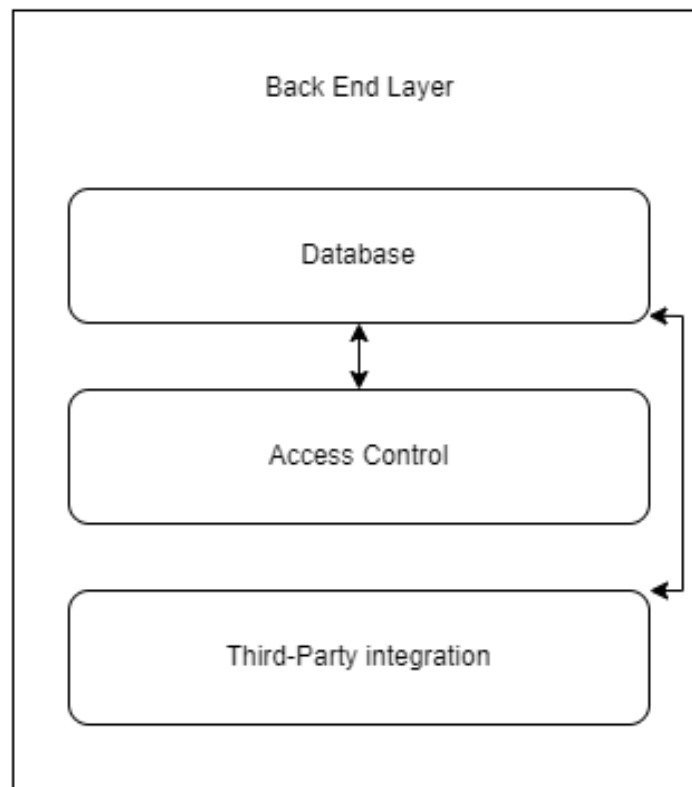


Figure 5: Database subsystem description diagram

5.1.1 ASSUMPTIONS

It might be interact with the outside source to get the information for the insertion or extraction to the database.

5.1.2 RESPONSIBILITIES

Database Management System is one of the most important part of the database. It do have the responsibility and function like Interaction with the file manager, integrity enforcement, security enforcement, Backup and recovery, and concurrency control. To be in more detailed:

Interaction with File Manager:

It will help to store and retrieved the data in the file manager.

Integrity Enforcement:

It will allow to constrain or restrict the data value that users can insert, delete or update in the database.

Security Enforcement: It will help to secure the database from the unauthorized user. Since, it will create and user account which will secure from the unauthorized user.

Backup and recovery:

It will allow to recover and backup the data in the database if the data will be lost or stolen. Moreover, it will create an copy of the data inside the database which will be helpful to recover the data.

Concurrency control:

It is the procedure of managing simultaneous transaction ensuring their atomicity, serializability, consistency, and isolation.

5.1.3 SUBSYSTEM INTERFACES

It is the software which will help to create the database in the organized way. So, it will have basically two entry one is to interact with the File manager and another with the out source.

Table 4: Database Management System interfaces

ID	Description	Inputs	Outputs
#01	Interact with the Outside source	Insertion of data User Login info	Extracting the data
#02	Interacting with the File Manager	Send the data	Extract the data

5.2 THIRD PARTY INTEGRATION

Third-party integration is having a relationship between at least two or multiple applications that are built on an outside platform. Simply, it is the addition of an application programming interface (API) to another application. This gives us the advantage to connect tools and add features to our processes instead of developing our own which can be time consuming.

The Third-Party-Integration system will interact with the database system by collecting information from the user and storing it into the database. The Database interacts with Third-Party Integration by communicating what should be stored such as tables, queries, etc to precisely locate and store information. The Third-Party-Integration then outputs the information to the Front-End(public pages) through an application program interface(API) so users can view the information.

5.2.1 ASSUMPTIONS

It will interact with multiple applications to meet the requirements of a feature for the application, however, there may be restrictions from the third-party that may limit every aspect.

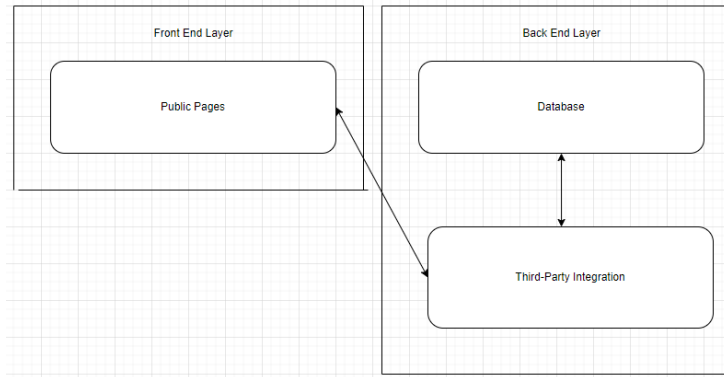


Figure 6: Example subsystem description diagram

5.2.2 RESPONSIBILITIES

Third-party integration has the responsibility of fulfilling the requirements and providing a solution for a business feature or application. It usually develops, designs, and implements the feature which can lead to possible constraints for a business. However, Third-party integration approach an issue from a unique perspective which saves time and provides an optimal solution.

5.2.3 SUBSYSTEM INTERFACES

Collection information will be the user interacting to the public page by inputting and submitting their information with the "Sign up now" button. The output is receiving an automated response through Team Name - Fall 2015 page 8 of 15 email to confirm the business receiving an email. Social media post will take the input of a post of media (Instagram, Facebook, etc) and then output by linking the post on the platforms latest news tab

Table 5: Subsystem interfaces

ID	Description	Inputs	Outputs
#01	collecting information (Constant Contact)	User information Message	Automated Re-sponse to email provided
#02	Social media post	Post from media	Link then post on platform

5.3 ACCESS CONTROL

The Access control system will identify users by verifying various login credentials, like usernames and passwords. This is how it interacts with the Admin Pages and public pages because the login credentials give access to these pages. The login credentials are only authorized if they match with the information in the database system.

5.3.1 ASSUMPTIONS

If login credentials are correct then the user will be able to login, however, there may be restrictions if they are incorrect.

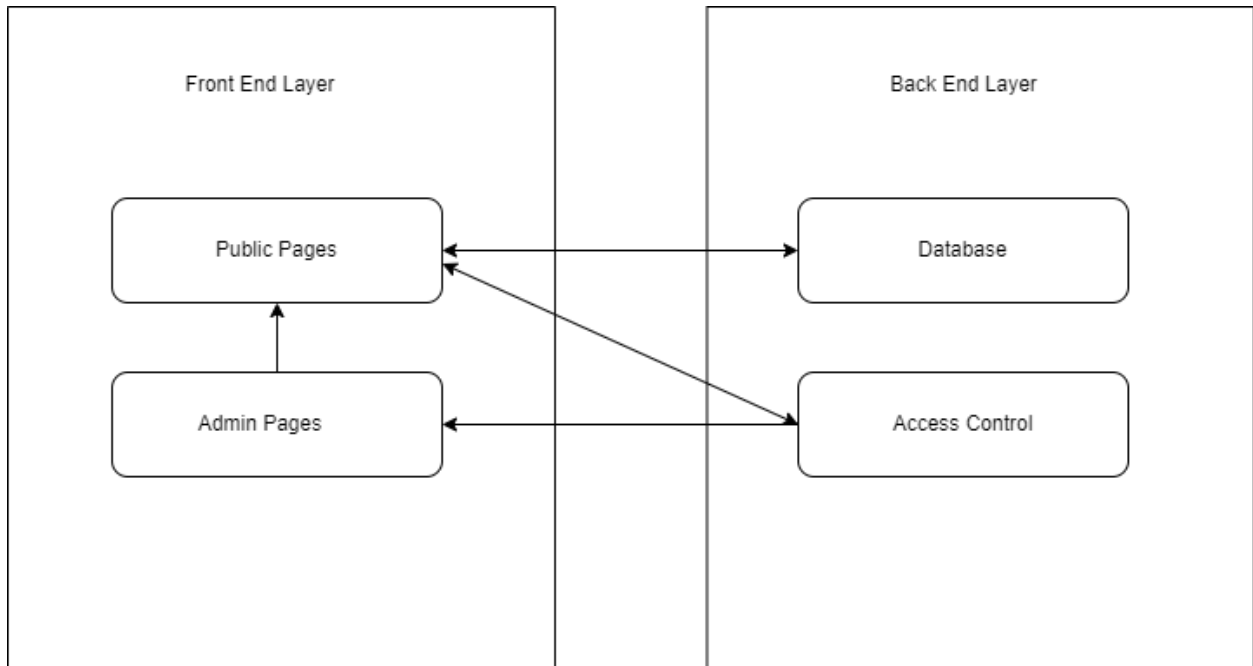


Figure 7: Access Control diagram

5.3.2 RESPONSIBILITIES

The responsibilities of access control is to identify login credentials such as username, passwords, pins, bio-metric scan etc. Many access controls also include multi-factor authentication. This is a method that requires various authentication methods to verify a user’s identity

5.3.3 SUBSYSTEM INTERFACES

When the user inputs the login credentials and clicks the "login" button, the output will be to check the info and match it with the information in the database. Then the user will be able to access the admin page if the credentials match.

Table 6: Access Control interfaces

ID	Description	Inputs	Outputs
#1	Authorizing user information	Login credentials	Access to admin page

REFERENCES