Requiem Registry - Design Document

By: Samantha Cook, Lucas Gamboa, Bricen Hicks

Table of Contents

Project Name

When brainstorming a name for a cemetery management system, we wanted to pick something unique that didn't sound like every other cemetery name. We concluded that Requiem Registry would be a good fit. Requiem means an act or token of remembrance; in the roman catholic church, it means a" Mass for the repose of the souls of the dead," according to the Oxford Dictionary. The registry part fits the idea that a database keeps track of all the plots. This name seems to capture the idea of what our website will do.

Team Members Names

See Figure 1 in the appendix for a detailed description.

Abstract

Requiem Registry will be a web-accessible database that aids in managing cemeteries, with the addition of a public front where people can search for those buried in that cemetery by name. The website will be divided into two sections: one for administrators and one for users. The administrators must log in through a Google account to access administrator privileges and additional web pages. Users, however, will not need to log in, as the website will be available for the public in a read-only state with only the search page and information page visible. The website will have features to help the administrators easily keep and adjust the records for the cemetery, plots, and people as needed. Requiem Registry is specifically geared toward small cemeteries that might only have the funding or support to keep physical records. This will be a simple solution to get their records on something more consistent than multiple copies on poster-sized pieces of paper.

The administrators can add, edit, and remove cemeteries. Each cemetery will have its name, description, address, and capacity—the number of plots it contains. Once an instance of a cemetery is created in the database, the administrator can add people to empty plots, and a map view is provided for quickly moving to plot numbers. The public side of the website will have a drop-down if there are multiple cemeteries available; beside the drop-down will be a search bar to enter the names of people. Once the cemetery selection is made, a name and description of the cemetery will be displayed above the search bar. On a successful search, the website will display that person's information.

Tools & Technologies

See Figure 2 in the appendix for a detailed description.

Requirements list

1. All Pages
   1.1. Admin Login Button in the top right corner
   1.2. Navigation Bar on the left edge of the screen
2. Admin Login Button
   2.1. User Side
       2.1.1. On click, loads the Google login API to validate email and password
       2.1.2. On successful login, the user becomes an admin user and is redirected to the Main Page
       2.1.3. On failed login, the user remains a user and is not redirected
   2.2. Admin Side
       2.2.1. On click, logs out admin user and redirects to the Main Page
3. Navigation Bar
   3.1. User Side
       3.1.1. Requiem Registry Logo
           3.1.1.1. On click, redirect the user to the Main Page
       3.1.2. Map
           3.1.2.1. On click, redirect admin user to Map Page
   3.2. Admin Side
       3.2.1. Requiem Registry Logo
           3.2.1.1. On click, redirect admin user to Main Page
       3.2.2. Add Person
           3.2.2.1. On click, redirect admin user to Create Person Page
       3.2.3. Add Cemetery
           3.2.3.1. On click, redirect admin user to Create Cemetery Page
       3.2.4. Lists
           3.2.4.1. On click, expands a dropdown menu with three items
               3.2.4.1.1. Cemetery
                   3.2.4.1.1.1. On click, redirect admin user to Cemetery List Page
               3.2.4.1.2. Person
                   3.2.4.1.2.1. On click, redirect admin user to Person List Page
               3.2.4.1.3. Plot
                   3.2.4.1.3.1. On click, redirect admin user to Plot List Page
           3.2.4.2. Clicking on the Lists again closes the dropdown menu
       3.2.5. Map
           3.2.5.1. On click, redirect admin user to Map Page

4. Main page
    4.1. User Side
        4.1.1. Cemetery Selection and Information Widget in the top center
        4.1.2. Search bar for people based on selected cemetery
            4.1.2.1. When typing in the search bar, a list of possible names that match the search bar text will be displayed in a dropdown under the search bar.
            4.1.2.2. Clicking a name from the dropdown or hitting enter with a valid name in the search bar will redirect to the Person Page.
            4.1.2.3. Hitting enter with an invalid name in the search bar will do nothing.
    4.2. Admin Side
        4.2.1. Cemetery Selection and Information Widget in the top center
        4.2.2. Search bar for people based on selected cemetery
            4.2.2.1. When typing in the search bar, a list of possible names that match the search bar text will be displayed in a dropdown under the search bar.
            4.2.2.2. Clicking a name from the dropdown or hitting enter with a valid name in the search bar will redirect to the Person Page.
            4.2.2.3. Hitting enter with an invalid name in the search bar will do nothing.
5. Cemetery Selection and Information Widget
    5.1. User Side and Admin Side
        5.1.1. Name of cemetery as header
            5.1.1.1. Cemetery's phone number, address, email, and description as a subheader
                5.1.1.1.1. Phone number, email, and description will not display if left empty
        5.1.2. Dropdown to select a cemetery.
            5.1.2.1. On click, the dropdown expands to show a list of cemeteries and their address in alphabetical order.
                5.1.2.1.1. Clicking on a cemetery in the list updates the header and subheader on the main page.
                5.1.2.1.2. Clicking on the current cemetery in the drop-down list does nothing
6. Map Page
    6.1. User Side
        6.1.1. Cemetery Selection and Information Widget in the top center
        6.1.2. Top-down view of the currently selected cemetery (supported by Google Map API)
            6.1.2.1. Map markers for each occupied plot
                6.1.2.1.1. On click, display a miniature popup that contains
                    6.1.2.1.1.1. Tenant's name, year of birth, and year of death

     6.1.2.1.1.2. Go To Button
      6.1.2.1.1.2.1. On click, redirect to tenant's Person Page

6.2. Admin Side

 6.2.1. Cemetery Selection and Information Widget in the top center
 6.2.2. Top-down view of the currently selected cemetery (supported by Google Map API)
  6.2.2.1. Map markers for each occupied plot
   6.2.2.1.1. On click, display a miniature popup that contains
    6.2.2.1.1.1. Tenant's name, year of birth, and year of death
    6.2.2.1.1.2. Go To Button
     6.2.2.1.1.2.1. On click, redirect to tenant's Person Page
    6.2.2.1.1.3. Edit Plot Link Button
     6.2.2.1.1.3.1. On click, it displays a search bar for people not linked to a plot
      6.2.2.1.1.3.1.1. The first result will always read "unlink current person."
       6.2.2.1.1.3.1.1.1. On click, unlink the current person from the plot.
      6.2.2.1.1.3.1.2. When typing in the search bar, a list of possible names that match the search bar text will be displayed in a dropdown under the search bar.
      6.2.2.1.1.3.1.3. Clicking a name from the dropdown or hitting enter with a valid name in the search bar will add that person to the plot.
      6.2.2.1.1.3.1.4. Hitting enter with an invalid name in the search bar will do nothing.
  6.2.2.2. Map markers for each empty plot
   6.2.2.2.1. On click, display a miniature popup with
    6.2.2.2.1.1. The label reads, "Add person to plot."
    6.2.2.2.1.2. Search bar for people not linked to a plot
     6.2.2.2.1.2.1. When typing in the search bar, a list of possible names that match the search bar text will be displayed in a dropdown under the search bar.
     6.2.2.2.1.2.2. Clicking a name from the dropdown or hitting enter with a valid name in the search bar will link that person to the plot.
     6.2.2.2.1.2.3. Hitting enter with an invalid name in the search bar will do nothing.
 6.2.3. Add Plot Button
  6.2.3.1. On click, the admin user's next click on the map will add a plot.
   6.2.3.1.1. A plot will not be added if the admin user clicks

6.2.3.1.1.1. Outside the map

6.2.3.1.1.2. On an existing plot

7. List Pages

7.1. User Side

7.1.1. Users will not have access to this page

7.2. Admin Side

7.2.1. There will be three different pages, each containing a list

7.2.2. Person List

7.2.2.1. This will be a list of all the people in the database

7.2.2.2. The list will be displayed below a search bar

7.2.2.2.1. The search bar will filter the list based on the name

7.2.3. Cemetery List

7.2.3.1. This will be a list of all the cemeteries in the database

7.2.3.2. The cemeteries in the list can be clicked on to route the user to the cemetery page.

7.2.4. Plot List

7.2.4.1. This page will have a dropdown menu with the Cemeteries

7.2.4.2. Selecting a cemetery will then display a list of the plots in the cemetery

7.2.4.3. The list will show the plot number and first and last name of the person associated with the plot.

7.2.4.3.1. The information will be brief about the person connected.

7.2.4.4. The search bar above the list will take input of plot number to find a plot more quickly.

8. Person Page

8.1. User Side

8.1.1. Person's name as header

8.1.1.1. Person's plot number, date of birth, date of death, and biography as a subheader

8.2. Admin Side

8.2.1. Person's name as header

8.2.1.1. Person's plot number, date of birth, date of death, biography, next of kin's name, next of kin's phone number, and next of kin's email as subheader

8.2.2. Edit Button in the bottom right.

8.2.2.1. On click, a person's information will appear as a series of text fields and text areas.

8.2.2.2. On click, Submit Button appears in the bottom center

8.2.2.2.1. On click, commit changes

8.2.2.3. Leaving the page without clicking the Submit Button will discard changes

9. Cemetery Page
   9.1. User Side
      9.1.1. Users will not have access to this page
   9.2. Admin Side
      9.2.1. Information about each cemetery will appear here. It will show the name, contact information, and description. Below, there will be an edit button.
         9.2.1.1. If the admin user hits the edit button, the data will appear in text fields so it can be edited.
         9.2.1.2. The submit button will appear where the edit button is.
            9.2.1.2.1. This will submit changes to the database.

10. Create Person Page
    10.1. User Side
       10.1.1. Users will not have access to this page
    10.2. Admin Side
       10.2.1. Form with the following inputs
          10.2.1.1. First, Middle, and Last name
          10.2.1.2. Suffix
          10.2.1.3. Birth date, Death date
          10.2.1.4. Biography
          10.2.1.5. Next of Kin's Name
          10.2.1.6. Next of Kin's Phone Number
          10.2.1.7. Next of Kin's Email
          10.2.1.8. Submit Button
             10.2.1.8.1. On click creates a new person with the provided input
             10.2.1.8.2. On click, while either the first name or last name are left empty, empty fields will be highlighted in red
             10.2.1.8.3. On click, while birth date and death date are left empty, empty fields will be highlighted in red

11. Create Cemetery Page
    11.1. User Side
       11.1.1. Users will not have access to this page
    11.2. Admin Side
       11.2.1. Form with the following inputs
          11.2.1.1. Cemetery Name
          11.2.1.2. Cemetery contact information
             11.2.1.2.1. Phone number, email, and address
          11.2.1.3. Cemetery Image
          11.2.1.4. Description
          11.2.1.5. Submit Button

11.2.1.5.1. On click, creates a new cemetery with the provided input
11.2.1.5.2. On click, while name or address are left empty, empty fields will be highlighted in red

Updated Timeline

See Figure 3 in the appendix for a detailed description.

Design Description

UI Design:

Main Page:

The main page will be where all users are directed after accessing the URL. The main page will show a logo in the top left corner, and the website's name, Requiem Registry, will be beside it. Underneath the website, information will be in a selection box for the user to select which cemetery they would like to query. After a selection, the cemetery's name, description, address, and phone number will be displayed. Underneath the description data of the cemetery will be a search bar; as a name is entered, a list of matching results will populate underneath the search bar. The user can then click on the search result that matches, and it will direct them to the person page. See the Person Page below for more description of what that looks like. The user can change which cemetery they have selected in the choice box to change the data and where the search bar is querying, or the user can select all.

There will be a navigation bar on the left side of the screen, which all users will see. The information inside the navigation bar will change depending on the user. A general user will only see an option to look at the map. Admin users can add a cemetery, add a person, go to the list page - people list, cemetery list, plot list, or go to the map page. The logo will double to reroute the user back to the home page.

Login Page:

The login page will be accessed by clicking on the login button on the top right side of the main page. The login page will have the user log in using a Google Account, which is the only login option. This will create a simple, clean, and secure login to our website.

Person Page:

Either type of user will access the person page. The general user will see the person's name at the top of the page. Below the name will be the person's birth date and death date. Below is the description of the person, if there is one. If the admin user looks at this page, the same data will appear, but it will be in text fields that they can edit. At the bottom right corner of this page for admin is a submit button, which will need to be pressed if the admin user changes the person's information.

Add Person form:

This form is where the users will be directed if they click the add person button from the navigation bar or the add person button on the person page. Another way the user gets to this page is if they click on a person's information from the person list or go to the person page from the map. This is where adding, editing, or removing the person happens. The UI will show

entries for first name, middle name, last name(s), suffix, birth date, death date, and a description.

If the person object already exists, the data will appear in those entries and be editable. Buttons on the bottom will be removed or submitted; removing them will delete a person, and submitting them will update the person in the database.

People List Page:

This page contains a list of people in the database. A button to add person - that button will route users to the add person form. Clicking on a person from the list will bring the user to the add person form, but the form will have the preexisting data shown here; the admin user can edit and then click the submit button or delete the person from the list. To edit or delete a cemetery, the user will click the person to take them to that person's page - here, edits can be made.

Admin Home Page:

This page will look the same as the user's home page; the navigation bar will just update, coordinating with admin privileges, and the rest of the data will look the same. This will be like the admin hub page. The logo will double to reroute the user back to the home page. The navigation bar will update to include the add person button, add cemetery button, lists submenu, and the map button. The list button will contain options for people, cemeteries, and plots.

Cemetery List Page:

This page contains a list of cemeteries in the database. A button to add cemeteries - that button will route users to the add cemetery form. Clicking on a cemetery from the list will bring the user to the add cemetery form, but the form will have the preexisting data shown here; the admin user can edit and then click the submit button or delete the cemetery from the list. To edit or delete a cemetery, the user will click the cemetery to take them to that cemetery's page. Edits can be made here.

Add Cemetary Form:

This form will be blank if it is a new entry. The form will contain entries for cemetery name, description, address, contact email, phone number, and number of plots.

These entries will show the data from the database. Users will see this form if they click on a preexisting cemetery. The information will appear in the form, making it easy to edit. The bottom of the form will show a remove button and a submit button.

Plot List Page:

This page will list the plots and preview information about each. This page will have a choice to show a plot list based on the selected cemetery. Search the bar at the top to quickly find a plot number from the list. This page will just be a uniform view of the data on the map page. There will not be any editing data on this page.

Map Page:

The map page for the admin user will contain quite a bit of information and functionality. The map page will have a button "add plot." Click this button, and the user will hoover and click a spot on the map. The map will be Google Maps, and the user can navigate the map to the correct area or choose a PNG image of the map instead. After the user clicks the point to add a new plot to the map, there will be a pop-up window. This pop-up window will have a choice box with a list of the people in the database. This is how the admin connects a person to a plot. If the user clicks on a plot and someone is already associated, the person's information will be previewed in the window. The admin user can remove the person from the plot, leave it empty, or associate another person with it.

Navigation Bar:

The navigation bar will stay on the left side of the screen. The navigation bar will change which options are available depending on the user. General users can only see an option to see the map. The admin user will see the add cemetery, add person, lists, and map page. Clicking on an option from the navigation bar will redirect the user to the corresponding page.

Server Design:

Main Page:

The main page will have the server requesting data from the database on each cemetery. Once the UI has selected a cemetery, the server will request that cemetery's data. The server will also send requests based on what the user is typing in the search bar to see if that information is contained in the cemetery.

Login Page:

The server will have to supervise the connection to the database to see if the admin credentials entered are correct when the user tries to log in.

Person Page/ Add Person form:

The person page will request the database to add a person to the table. If the person is being edited, it will send that data to the database to update. If a person's information is clicked on to see it, this will send a query to the database to get the information. The add person form will have the server sending data to the database to add a row. Deleting a person will be sent from over sever to the database.

Admin Home Page:

The admin page will have several queries to the database to load different pages selected in the navigation bar. More requests from those pages will be sent from the UI to the server.

Cemetery Page/ Add Cemetery Form:

 The cemetery page will send several data sets on each new cemetery. When the user submits the data, the server will send it to the database to store it. Editing data on cemeteries will be sent over the server to the database, and a cemetery will be deleted.

Lists Page (People, Cemetery, or Plot):

The list pages will send a request to the server when the user selects which object they want to look at. The server will request the data for that object from the server. Once the UI receives that data, it will all be displayed. It will be filtered depending on the information entered in the search bar. This will be requested over sever to the database.

Map Page:

The map page will have the server get the data from Google Maps API; when the user adds a new plot, the server will send the latitude and longitude points to the database for storage. Once the user selects a person to add to a plot, that person's information will be sent to the database for the set plot coordinates. If a person is removed from a plot, that person will be sent to the database to update the table.

Navigation Bar:

The navigation bar will have the server send updates to the UI based on the pressed button. The admin user will have seven options. Clicking the logo above the navigation bar will send several requests to update the UI on the main page. The add person button will bring up the add person form, the person page will request the person page, and the add cemetery will have the server update the UI to add the cemetery form. The cemetery, plot, and map pages will have the server load their corresponding pages to the UI.

Database Design:

Main Page:

The main page will have the database receiving requests for data and then sending matching results back. The request will be for cemeteries, then people in the cemeteries. If the user selects to search all the cemeteries, the database will send all that information to the server to display in UI.

Login Page:

The login button will request that the database check the saved credentials table. If the credentials are in the database, it will send that back to the server so the admin user can successfully log in.

Person Page/ Add Person form:

The database will get data from these pages when adding a new person. The database will add the person's information to the person's table. The person page will be sent updates to the database if a person's information is changed. The database will receive this information and update tables accordingly. If a person is deleted, a boolean attribute will be set to true so the data will not appear in queries. This is like a safe delete.

Admin Home Page:

The admin page is not doing anything but serving as a placeholder, and the admin home page would be sent the same request as the main page if the admin uses it. See the main page above.

Cemetery Page/Add Cemetery Form:

The database will get data from these pages when adding a new cemetery. The database will add the cemetery's information to the cemetery table. The cemetery page will update the database if a cemetery's information changes. The database will receive this information and update tables accordingly. If a cemetery is deleted, a boolean attribute will be set to true so the data will not appear in queries. This is like a safe delete.

List Page (People, Cemetery, or Plot):

The plot page will send a request for information to the database; the database will send back information about the object. When something is searched for, that will update what is sent back from the database. Only query for results that match the search.

Map Page:

The map page will send plot information to the database, and the database will store this data and connect it with the correct cemetery. The map page will also establish a connection between the plots and people; the database will receive that data and update connections accordingly. The database will also remove connections and reconnect data if needed. This would be needed if a person is removed from a plot and connects with a different one.

Navigation Bar:

The navigation bar will not directly interact with the database. The database will have functionality based on pages selected from the navigation bar.

Appendix

Figure 1: Team Members

Name	Role
Samantha Cook	Front End Developer
Lucas Gamboa	Database Developer
Bricen Hicks	Backend Developer

Figure 2: Requiem Registry Tech-Stack

Component	Dependency Name	Dependency Type	Explantation
Framework	Next.js	Web Framework	React-based framework for building server-rendered and static websites.
Hosting & Deployment	Vercel	Hosting Platform	Deploys and hosts the Next.js application with serverless functions.
Database	PostgreSQL	Relational Database	Stores and manages application data.
Front-end	Material Design	UI Styling Tool	Provides prebuilt React components following Material Design principles.
Styling	Tailwind CSS	CSS Framework	Utility-first framework for designing responsive and modern UI

Authentication	Google Identity API	3rd Party API	Enables Admin users to log in using Gmail
Version Control	GitHub	Version Control	Used for code collaboration and version tracking.
Google Maps	Maps JavaScript API	3rd Party API	Provides interactive maps for the application

Figure 3: Tentative Schedule

Week	Tasks
Week 1: February 3rd - February 7th	**All Members:** Coming to a conclusion on exactly how everything will look and user flow. Finishing up requirements document and preparing the presentation. **Samantha:** Getting familiar with React. **Lucas:** Getting familiar with PostgreSQL **Bricen:** Creating a base project to test the validity of the tech stack. Also, creating a Figma page for the whole UI
Week 2: February 10th - February 14th	**All Members:** Getting familiar with each component we will be responsible for **Samantha:** Using prototyping tools to understand what UI will look like. **Lucas:** Create an Entity Relationship Diagram with specified data types and constraints. Ensure the database design is in the third normal form. **Bricen:** Making simple tests to show a connection between the database and web page
Week 3: February 17th - February 21st	**All Members:** Getting each portion of the tech stack connected

	Samantha: I began working on development, starting with Google login. **Lucas:** Identity and pseudo code required triggers, stored procedures, and views. **Bricen:** Helping with the structure of the front end to ensure ease of connectivity on the server side part
Week 4: February 24th - February 28th	**All Members:** Agree on exactly how the website and logo will look. **Samantha:** Begin work on frontend **Lucas**: Create a database and populate it with mock data. **Bricen:** Altering code to work with the new information from the database
Week 5: March 3rd - March 7th	**All Members:** Getting started integrating **Samantha:** Start to see about integration between frontend, backend, and database **Lucas:** Test triggers, stored procedures, and views with mock data. **Bricen:** Scaling up the server-side code to work with each cemeteries. Most likely, learning maps
Week 6: March 10th - March 14th (Spring Break)	**All Members:** **Samantha:** **Lucas:** **Bricen:** goin' camping
Week 7: March 17th - March 21st	**All Members:** Continued work on coding**.** **Samantha:** Adding more components to UI now that there is an established connection between components. **Lucas:** Begin integrating the database with the rest of the project. **Bricen:** Adding the completed database information to the UI
Week 8: March 24th - March 28th	**All Members:** More individual coding and integration of portions. **Samantha:** Start implementing Google Maps API. **Lucas:** Continue integrating the database

	with the rest of the project and address emerging bugs. **Bricen:** Making a connection point between Google Maps and the values of the points saved in the database
Week 9: March 31st - April 4th	**All Members:** More individual coding and integration of portions. **Samantha:** Continued work on map integration. **Lucas:** Continue integrating the database with the rest of the project, address emerging bugs, and alert the group to possible design flaws. **Bricen:** Continued work on the connection between map information and the map API
Week 10: April 7th - April 11th	**All Members:** Finishing code starting posters **Samantha:** Begin testing and styling the front-end UI. **Lucas:** Continue integrating the database with the rest of the project, address emerging bugs, and alert the group to possible design flaws. **Bricen:** editing and maximizing code efficiency
Week 11: April 14th - April 18th	**All Members:** Poster design. **Samantha:** Begin working on poster ideas. **Lucas:** Review poster ideas and give feedback if needed. **Bricen:** Drawing possible poster plans
Week 12: April 21st - April 25th	**All Members:** Testing code **Samantha:** Finishing up all development. **Lucas:** Ensuring formalization of database documentation and its functions. **Bricen:** Helping with any development that has yet to be figured out
Week 13: April 28th - May 2nd	**All Members:** Testing website **Samantha:** Working with the team to see how the website responds during testing. **Lucas:** Coordinate with Bricen to stress test

	the project to determine its limits. **Bricen:** Adding in plenty of different cemeteries to see the limit of the project
Week 14: May 5th - May 9th	**All Members:** Finishing up the poster, last-minute testing, and presentation. **Samantha:** Any last-minute things that must be done, including the presentation. **Lucas:** Ensuring the team is ready for the final presentation. **Bricen:** learning how to public speak

Figure 4: Block Diagram

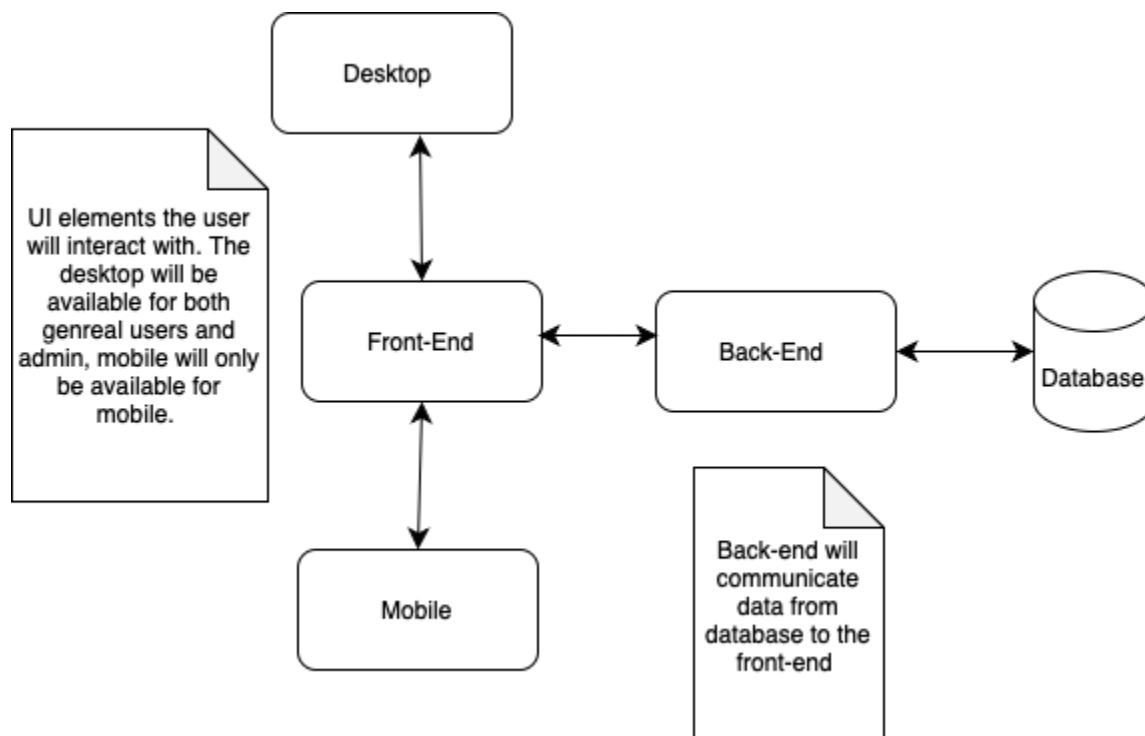## Figure 5: Component Diagram

See Component_Diagram

Figure 6: UI Storyboard

See UI_Storyboard

See [mobileUI_Storyboard](#)

Message Documentation:

Adding cemetery:

Purpose: The purpose of this is to get information from UI about adding a cemetery to the database.
Field Type:
```
{
  "name": "string",
  "description": "string",
  "address": {
    "street": "string",
    "city": "string",
    "state": "string",
    "zip": "string"
  },
  "capacity": int,
  "phoneNumber": "string"
  "email": "string"
}
{
  "success": boolean,
  "message": " string",
  "cemeteryId": int
}
```

Example:
```
{
  "name": "Evergreen Cemetery",
  "description": "A peaceful resting place.",
  "address": {
    "street": "456 Peaceful Rd",
    "city": "Los Angeles",
    "state": "CA",
    "zip": "90002"
  },
  "capacity": 5000,
  "phoneNumber": "+123456789"
}
```

Response:
```
{
```

```
  "success": true,
  "message": "Cemetery successfully added.",
  "cemeteryId": 5
}
```

Adding Person:

Purpose:
This will occur when the user submits the new person form. That will take the data from the UI and send it over server to the database.
Field Type:

```
{
 "Id": "int",
  "firstName": "string",
  "middleName": "string",
  "lastName": "string",
  "suffix": " string",
  "birthDate": "date",
  "deathDate": "date",
  "bio": "string",

}
```

```
{
  "success": boolean,
  "message": " string",
  "personId": int,
  "plotId": int
}
```
Example:
```
{
  "firstName": "John",
  "middleName": "A.",
  "lastName": "Doe",
  "suffix": "Jr.",
  "birthDate": "1950-06-15",
  "deathDate": "2023-01-05",
  "bio": "Loving father and husband.",
}
```

Response:
```
{
  "success": true,
  "message": "Person successfully registered.",
  "personId": 3001,
  "plotId": 101
}
```

Adding a plot to the cemetery:

Purpose: Adding a plot will send the data entered in the UI to the server, which will send it to the database.
Field Type:
```
{
  "plotId": int,
  "cemeteryId": int,
  "coordinates": {
    "latitude": decimal,
    "longitude":  decimal
  },
  "tenantId": null
}
```

Example:
```
{
  "plotId": 101,
  "cemeteryId": 5,
  "coordinates": {
    "latitude": 34.052235,
    "longitude": -118.243683
  },
  "tenantId": null
}
```

Response:
```
{
  "success": true,
  "message": "Plot successfully added.",
  "plotId": 101
}
```

Getting a cemetery from the database:

Purpose: Data on a cemetery for the UI will be collected so the website can update and display information to the user. The UI can then filter and use this data to be displayed all across the website.

```json
{
  "cemeteryId": 5,
  "name": "Evergreen Cemetery",
  "description": "A peaceful resting place.",
  "address": {
    "street": "456 Peaceful Rd",
    "city": "Los Angeles",
    "state": "CA",
    "zip": "90002"
  },
  "capacity": 5000,
  "phoneNumber": "+123456789",
  "plots": [
    {
      "plotId": 101,
      "coordinates": {
        "latitude": 34.052235,
        "longitude": -118.243683
      },
      "tenant": {
        "personId": 3001,
        "fullName": "John A. Doe Jr.",
        "birthDate": "1950-06-15",
        "deathDate": "2023-01-05"
      }
    }
  ]
}
```

Storage Documentation:

Figure 7: ER Diagram

Plot

PK	PLOT_ID int NOT NULL
PK, FK1	CEME_ID int NOT NULL
	COORD_LAT decimal(8,6) NOT NULL
	COORD_LONG decimal(9,6) NOT NULL
FK	TENANT_ID int

Cemetery

PK	CEME_ID int NOT NULL
	NAME varchar(50) NOT NULL
	DESC clob
	IMG clob
FK	CON_ID int
	CAPACITY int NOT NULL
	STREET varchar(20) NOT NULL
	CITY varchar(20) NOT NULL
	STATE varchar(20) NOT NULL
	ZIP varchar(20) NOT NULL

Person

PK	PER_ID int NOT NULL
	F_NAME varchar(50) NOT NULL
	M_NAME varchar(50)
	L_NAME varchar(50) NOT NULL
	SUFFIX varchar(50)
	BIRTH_DATE date
	DEATH_DATE date
	BIO clob
	KIN_NAME varchar(50)
FK	CON_ID int

Contact

PK	CON_ID int NOT NULL
	PHONE_NUM varchar(20)
	EMAIL varchar(50)