

Software Design Document (SDD)
CalPERs Interactive Kiosk
Team Crimson Tigers

Members:

Tommy Kardach
Gadfrey Balacy
Dennis Liu

1. Introduction

This system will provide CalPERS staff and guests to search through a database of CalPERS employees. The system will contain information pertaining to these employees, such as contact information and location within the building relative to the kiosk.

2. Revised Requirements

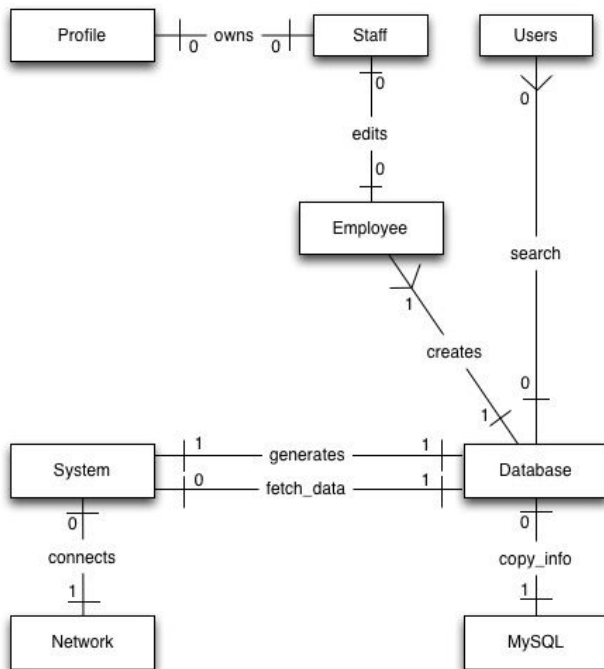
- Major change that we did mid-way in the design phase as a team was deciding that the software Unity(c# game engine) we were going to use as our main development tool for the kiosk interface had too many complexities for it to provide useful results before the deadline as such it was removed. As this was a major setback in our design, many modules will be either modified or removed as transferring work that we have already created would not work or use up too much time and resources to be deemed useful.
- Major change is switching our main languages, as time is very valuable we had to drop and use HTML, Javascript and CSS as our major tools of development. This sudden change in our development environment was a challenge, but an enjoyable challenge as the many resources and toolkits we could use with these languages and scripts made it possible to complete what we had planned in a much faster time with more simple and easy to understand implementations.
- **FR2 Admin Login/FR7 Sync Database/FR15 Remove Employee:** Major change was removing the Administrator top-down control hierarchy module. This module would have provided the Admin access to removing/adding/searching employee profiles and editing their attributes. Time-constraints have concluded that this will not be included in the final build.
- Major change was adding the Kiosk Map Interface. Including the Map module would provide better functionality for the user to access kiosk and find the employee the user was looking for. Because of this modification, some minor modules were either pushed back or removed. With time-constraints lots of optional functionality was also removed from the Kiosk Map interface which will be listed and provided with explanation.
- **FR23 Contact Employee via SMS:** Minor change was the SMS module for text and email was removed because of time constraints. We had to allocate our main resources on the functional requirements before spending man-power on optional functionality.
- **FR11 Search Filter:** Minor change was removing the Search Filter that was meant for easier user interaction, in which without knowing all information of the

employee the user can filter out and limit all the employees that would fit the specifications. This module was removed because of time constraints and all the extra man-power to implement this design in our current set-up without jeopardizing the deadline.

- **FR5 Create Profile:** Minor change was removing the User Register module, as it was deemed unnecessary to have. As users would only use kiosk to find specified employee and location.
- Minor change regarding access door. As this was not included in our provided .csv example file, we could not approach this problem and therefore it was not included in the final design.
- **FR 22 Sync Kiosks:** Minor change Sync Kiosks was removed due to the time needed to learn and implement this algorithm in our design.

3. Data Design

A. Entity-Relationship Diagram



Entities: Users(Guest, Staff), Profile, Database, Employee, Resources, Network, System, Network, MySQL

Assumptions:

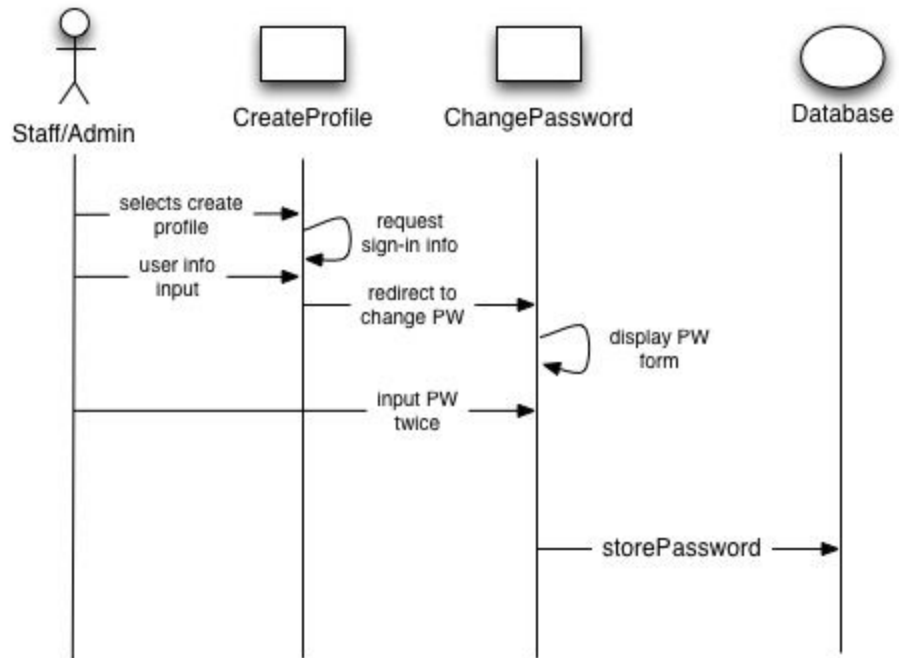
The System must be connected to the Network in order to access the SQL Server
 The Database will be created through the client's MySQL
 An Admin is a CalPERs employee, and has their information on the CalPERs database
 The System's database will be a copy of CalPERs's database, and will be hosted on its own SQL server

Requirements:

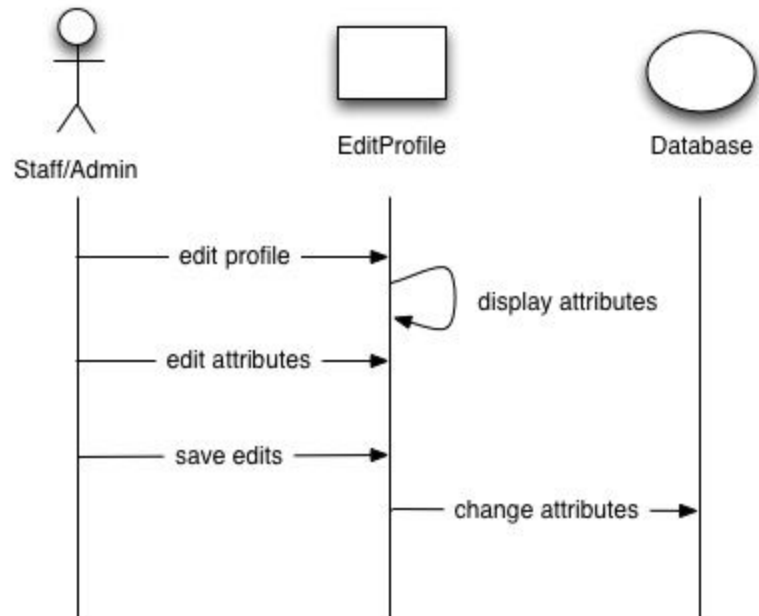
- All Users (Guest, Staff) can search the database
- Staff Users can own a profile, which will allow them to edit their own Employee information.
- System will connect to a Network. It will then generate a Database which will receive Queries from MySQL.
- The Database, having received information from MySQL, will create Employee objects and Resource objects, which is what Users will be search the Database for.
- Users have the option to contact Employees via SMS, or to receive that employee's contact information via SMS to the User's phone.

B. Sequence Diagrams

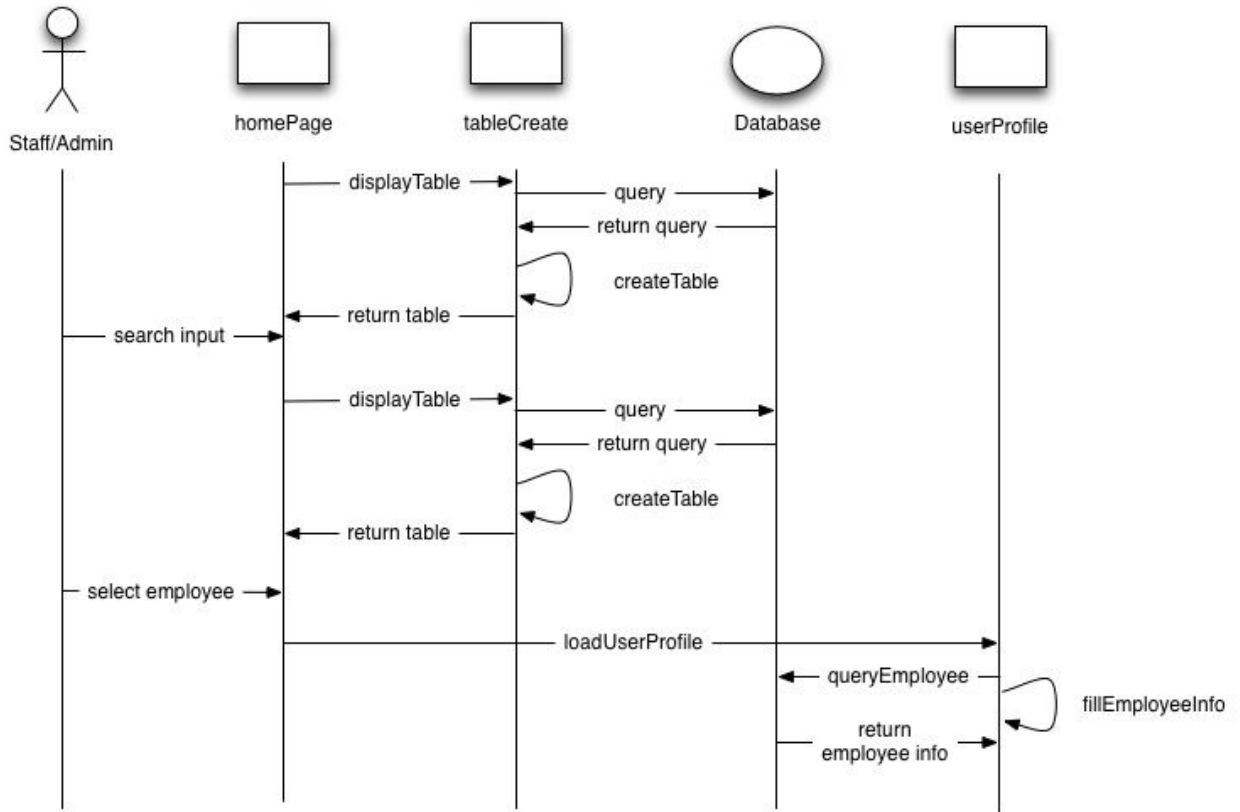
i. CreateProfile:



ii. EditProfile:



iii. DisplayEmployees

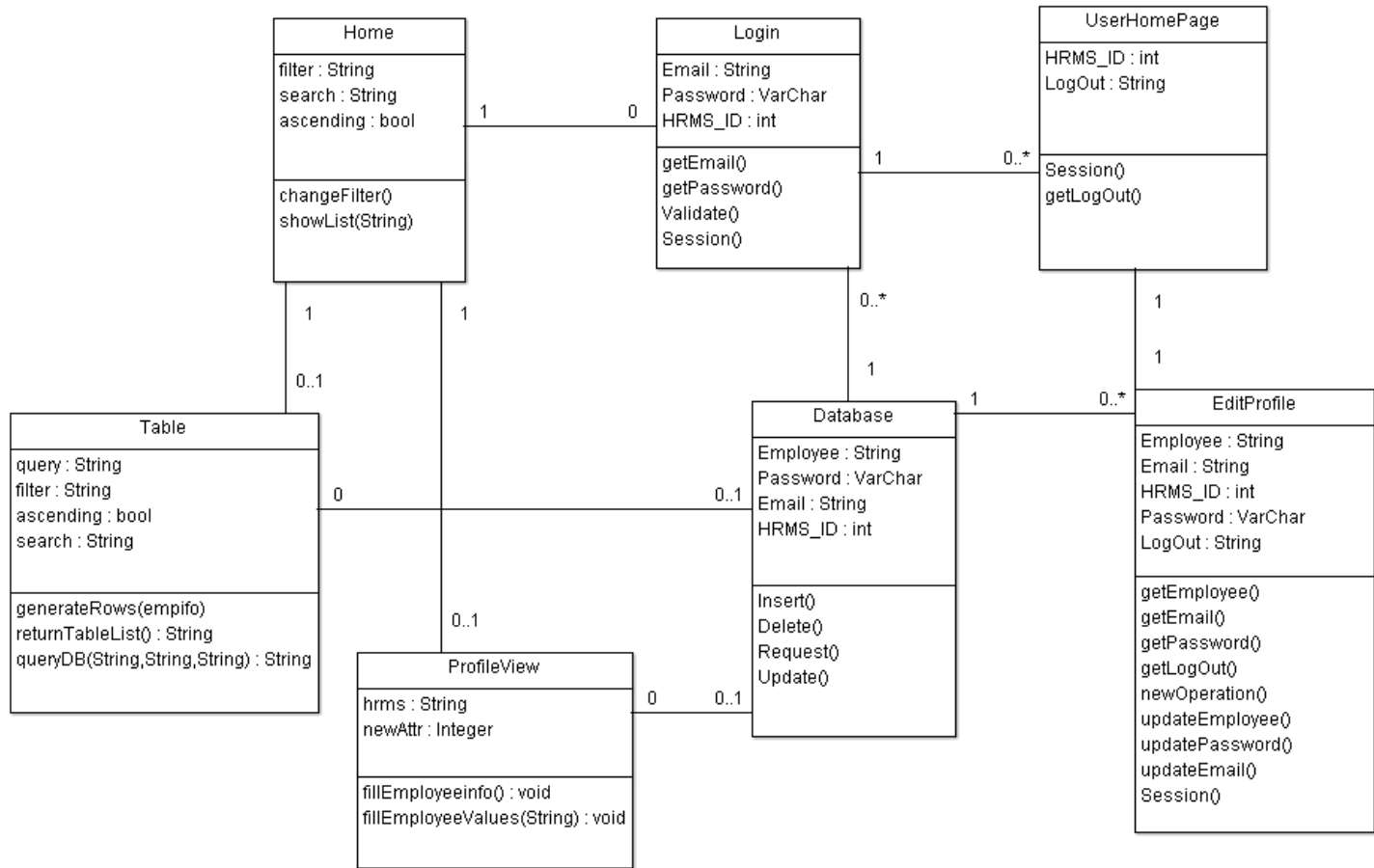


C. Updated Data Dictionary

Data Item Number	Data Item	Alias	Data Type	Description	Range	Default Value
1	First Name	first_name	String	Employee First Name	0-11	null
3	Last Name	last_name	String	Employee Last Name	0-11	null
4	Photo	photo	Binary	Employee Photo	0-11	null
5	Employee Classification	empl_classification	String	Classification of the Employee	0-30	null
6	Job Title	job_title	String	Employee Job	0-24	null

				Title		
7	Desk Phone	desk_phone	Int	Employee Desk Phone Number	11	null
8	Mobile Phone	mobile_phone	Int	Employee Mobile Phone Number	12	null
9	Grid Location	grid_location	String	Location of object on the map grid	0-13	null
10	Building	building	String	Building object is located in	0-8	null
11	Floor	floor	Int	Floor object is located on	1-5	null
12	Email	email	String	E-Mail address used by Employee	0-35	null
13	Access Door	access_door	String	Nearest Access door to get to Employee location	0-11	null
14	Division	division	String	Division name Employee belongs to	0-8	null
15	Unit	unit	String	Unit name employee belongs to	0-30	null
16	Manager	manager	String	Manager of Employee	0-10	null
17	HRMS_ID	HRMS_ID	string	Hrms id of employee	0-7	null
18	Employee	N/A	Associative Array			

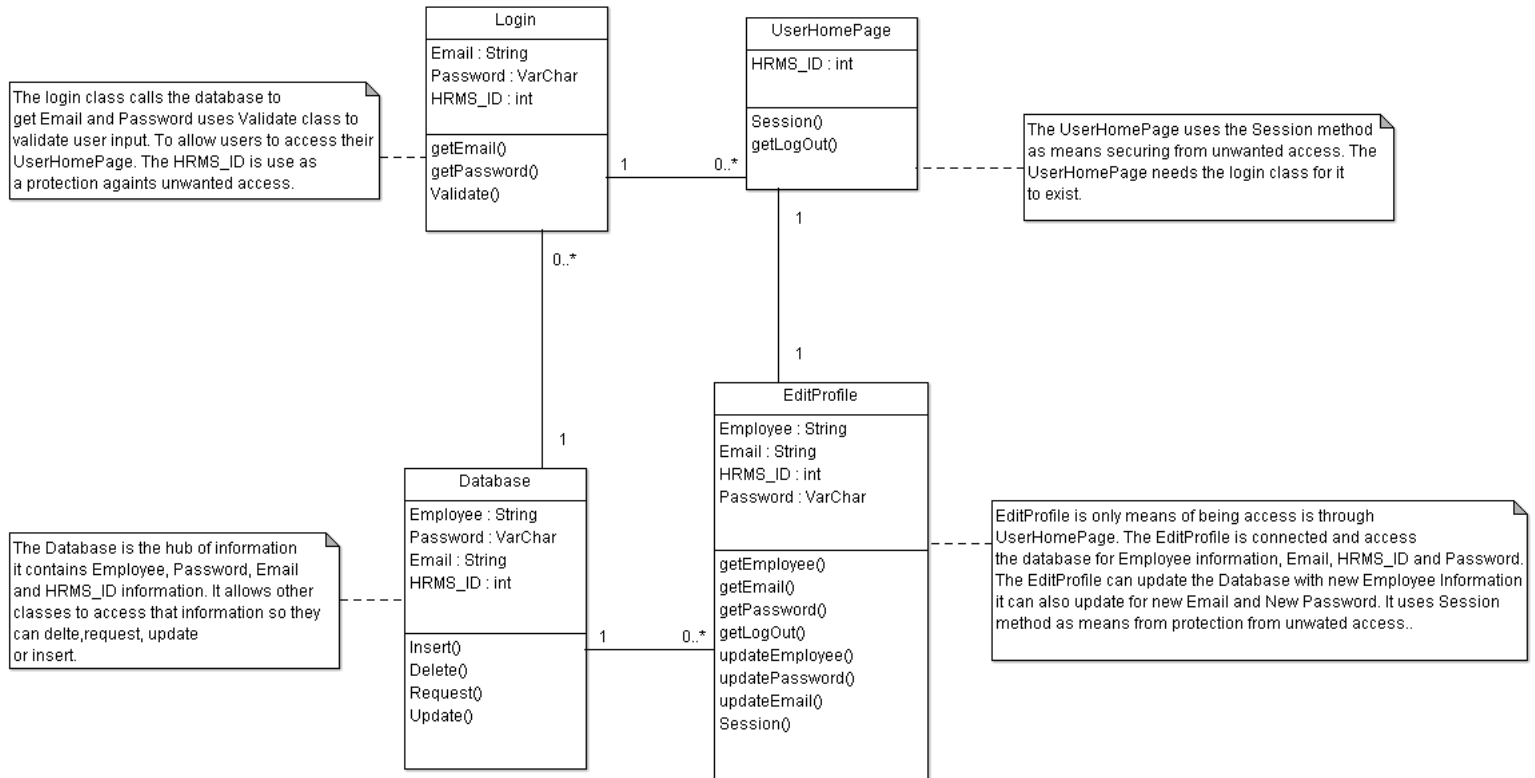
4. Architecture Design



5. Detailed Class Design

A. Profile

i. Class Diagram with Narrative



ii. Class Interface Description

When Login is loaded it display itself with two parameters email and password. Once the user inputs inside those two parameters and click login. The Login class calls the database and check if the email address exist. Then if it does it looks at the associated password and id with the email. Then it pulls the password and id and store it in a local variable. And that variable gets compared to the password that user input. If it matches it allows the user to go the UserHomePage. The UserHomePage is protected by a session method and that uses the user Id as validation. Then with the UserHomePage loaded you could access EditProfile. And once the EditProfile is loaded it access the database to request employee, email and password to populate the profile page. Then the user can change the information in profile page. Then the database gets updated with the new information.

iii. Pseudo-Code Description

Login:

Get Password from Database // No input no default value
Get Email from Database // No input no default value
Get HRMS_ID from Database // No input no default value
Validate()

UserHomePage:

Session() // Check if the user has access

EditProfile:

Get Employee from Database // No default value
Get Password from Database // Default Value User Password
Get Email from Database // Default Value User Email
fillEmployeeInfo() // Fill the employee info
Session() // Check if the user has access

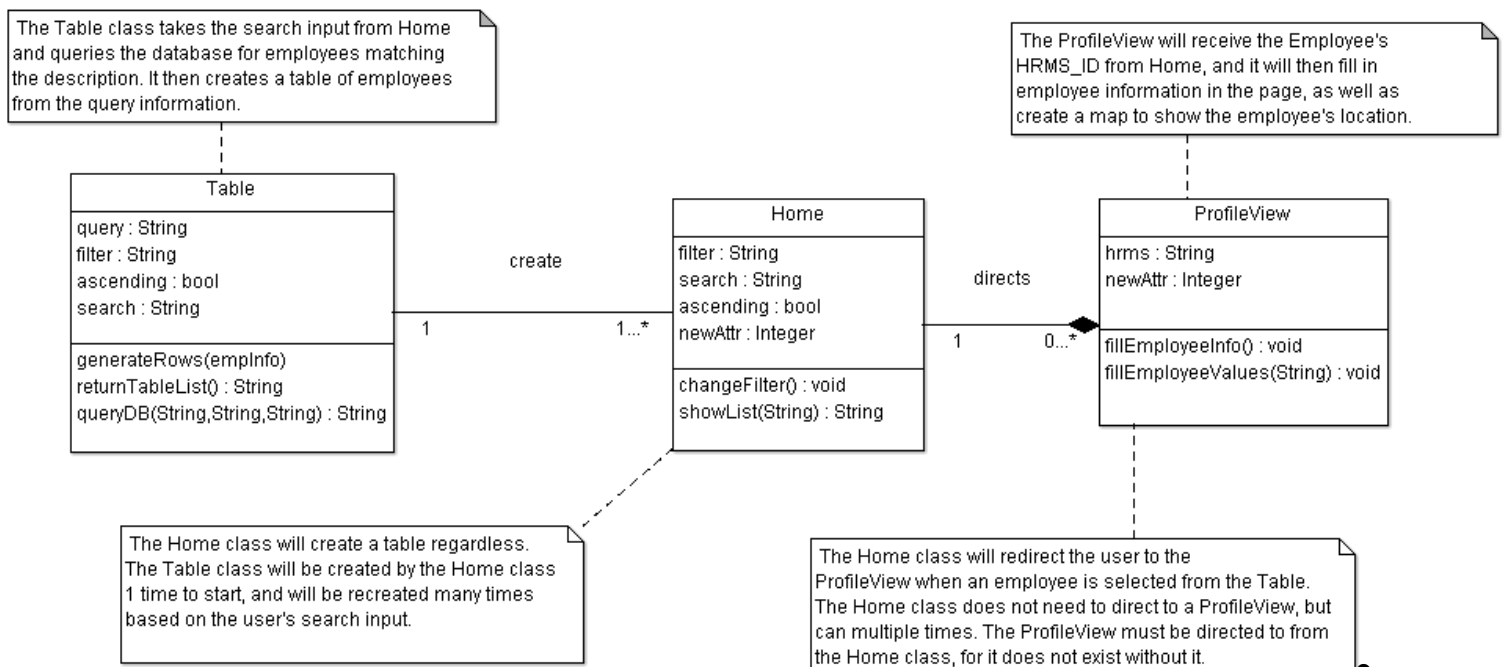
iv. Class Restrictions/Limitations

There must be a connection to the database for there to be any true functionality from these classes.

The user must login to get access to the UserHomePage and EditProfile..

B. Searching

i. Class Diagram with Narrative



ii. Class Interface Description

When Home is loaded Home will take search input from the user. This input is in the form of search input (search for employee names), and organizing the columns alphabetically (ascending or descending) by clicking on the column title. This information is sent to Table, where it is used to create a query. The Table class connects to the database, and uses the newly created query. It then creates HTML code (as a String), which is used in Home to display the list of employees.

If a row from the table created by the Table class is selected, it will redirect to the ProfileView page. The HRMS_ID of the employee depicted by the clicked row will be sent to ProfileView, where it will be used to retrieve the employee's information and fill the page with the information.

iii. Pseudo-Code Description

Home:

showList(); // calls Table to create a table

While still on page:

Ask for user input

showList(in); // calls Table to create a table with new input

If (user selects employee)

Redirect to ProfileView

Table:

GET info from Home // if no input, default values are set

Create query from information received by Home

Create table from query // all employees by default

Return HTML code to Home

ProfileView:

GET hrms_id from Home

fillEmployeeInfo(); // retrieves the employee info from database

fillEmployeeValues(); // fills page with employee's information

iv. Class Restrictions/Limitations

- There must be a connection to the database for there to be any true functionality from these classes.
- ProfileView will display an empty page if it is not called from Home.

6. User Interface Design

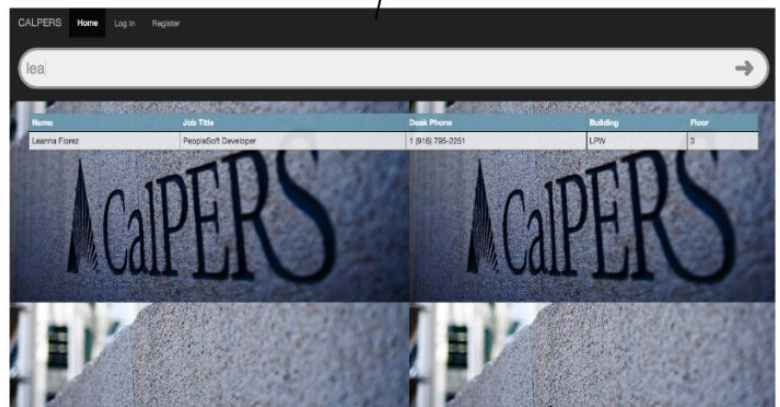
A. Searching



The screenshot shows the CALPERS website with a search bar at the top. Below the search bar is a table listing employees. The table has five columns: Name, Job Title, Desk Phone, Building, and Floor. The table contains 15 rows of employee data.

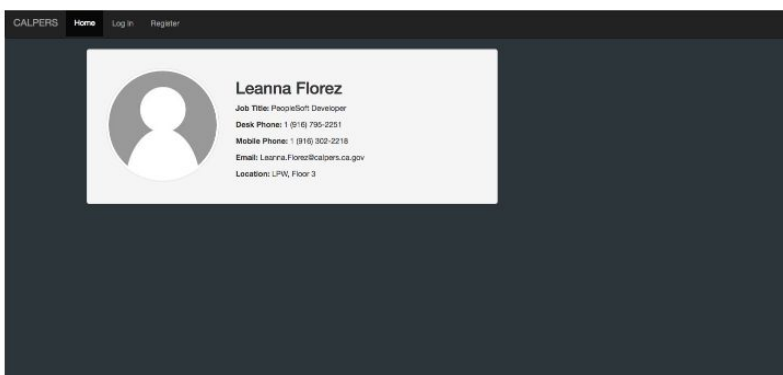
Name	Job Title	Desk Phone	Building	Floor
Aaron White	Section Manager	1 (916) 795-2538	LPW	3
Adeline Crockett	Java Developer	1 (916) 795-3355	LPW	3
Adrian Weaver	System Tester	1 (916) 795-9412	LPW	3
Alan Maguire	System Tester	1 (916) 795-2318	LPW	3
Alexandria Comstock	Java Developer	1 (916) 795-2397	LPW	3
Alexis Dickey	Application Architect	1 (916) 795-2889	LPW	3
Alexis Ward	Business Analyst	1 (916) 795-0895	LPW	3
Alex Mandojin	Java Developer	1 (916) 795-0235	LPW	3
Alex Rowland	Java Developer	1 (916) 795-0598	LPW	3
Arlie Wentworth	Java Developer	1 (916) 795-0853	LPW	3
Andrea Ham	Java Developer	1 (916) 795-3241	LPW	3
Angel Miller	Section Manager	1 (916) 795-1474	LPW	3
Angel Moody	Business Analyst	1 (916) 795-0162	LPW	3
Angelic Brinder	Application Architect	1 (916) 795-3552	LPW	3
Aime Ramon	Integration Developer	1 (916) 795-7261	LPW	3

User Enters "lea" in search engine. List created with any matching results



The screenshot shows the CALPERS website with the search bar containing the text "lea". Below the search bar, a single row of employee data is displayed, matching the search criteria.

Name	Job Title	Desk Phone	Building	Floor
Leanna Florez	PeopleSoft Developer	1 (916) 795-2251	LPW	3



The screenshot shows the CALPERS website with the ProfileView page for Leanna Florez. The page displays a profile card with a placeholder for a photo and the following information:

Leanna Florez
Job Title: PeopleSoft Developer
Desk Phone: 1 (916) 795-2251
Mobile Phone: 1 (916) 302-2218
Email: Leanna.Florez@calpers.ca.gov
Location: LPW, Floor 3

User clicks employee in the table and is redirected to the ProfileView, where the employee's information is displayed

B. Login

Log-In

Email: bubba@sparks.com

Password: *****

Log In

Name	Job Title	Desk Phone	Building	Floor
Aaron White	Section Manager	1 (916) 795-2538	LPW	3
Adeline Crockett	Java Developer	1 (916) 795-3355	LPW	3
Adrian Weaver	System Tester	1 (916) 795-9412	LPW	3
Alan McGuire	System Tester	1 (916) 795-2316	LPW	3
Alexandria Comstock	Java Developer	1 (916) 795-2367	LPW	3
Alexia Dickey	Application Architect	1 (916) 795-2698	LPW	3
Alexis Ward	Business Analyst	1 (916) 795-0895	LPW	3
Alia Marroquin	Java Developer	1 (916) 795-0236	LPW	3
Alli Rowland	Java Developer	1 (916) 795-0698	LPW	3
Anne Wentworth	Java Developer	1 (916) 795-0859	LPW	3
Andrea Ham	Java Developer	1 (916) 795-3241	LPW	3
Angel Miller	Section Manager	1 (916) 795-1474	LPW	3
Angel Moody	Business Analyst	1 (916) 795-0162	LPW	3
Angelic Binder	Application Architect	1 (916) 795-3552	LPW	3
Annie Warren	Integration Developer	1 (916) 795-7061	LPW	3
Annie Freeman	Integration Developer	1 (916) 795-8163	LPW	3
Anthony Goodman	Section Manager	1 (916) 795-4268	LPW	3
Armand Downey	Java Developer	1 (916) 795-3722	LPW	3
Ann Woodward	PowerShell Developer	1 (916) 795-2098	LPW	3

Login is selected (in this scenario current parameters are entered) and user is redirected to the same page, logged in.

Edit-Profile is selected and the user is redirected to their information page, where they can edit their information.

CALPERS Home Edit Profile Log Out

Search 1

Name	Job Title	Desk Phone	Building	Floor
Aaron White	Section Manager	1 (916) 795-2538	LPW	3
Adeline Crockett	Java Developer	1 (916) 795-3355	LPW	3
Adrian Weaver	System Tester	1 (916) 795-9412	LPW	3
Alan McGuire	System Tester	1 (916) 795-2316	LPW	3
Alexandria Comstock	Java Developer	1 (916) 795-2367	LPW	3
Alexia Dickey	Application Architect	1 (916) 795-2698	LPW	3
Alexis Ward	Business Analyst	1 (916) 795-0895	LPW	3
Alia Marroquin	Java Developer	1 (916) 795-0236	LPW	3
Alli Rowland	Java Developer	1 (916) 795-0698	LPW	3
Anne Wentworth	Java Developer	1 (916) 795-0859	LPW	3
Andrea Ham	Java Developer	1 (916) 795-3241	LPW	3
Angel Miller	Section Manager	1 (916) 795-1474	LPW	3
Angel Moody	Business Analyst	1 (916) 795-0162	LPW	3
Angelic Binder	Application Architect	1 (916) 795-3552	LPW	3
Annie Warren	Integration Developer	1 (916) 795-7061	LPW	3
Annie Freeman	Integration Developer	1 (916) 795-8163	LPW	3
Anthony Goodman	Section Manager	1 (916) 795-4268	LPW	3
Armand Downey	Java Developer	1 (916) 795-3722	LPW	3
Ann Woodward	PowerShell Developer	1 (916) 795-2098	LPW	3

CALPERS Edit Profile User Log Out

Edit Profile

Upload a different photo...
Browse... no file selected

Personal info

First name: Bubba

Last name: Sparks

Desk Phone: 1 (233) 345-6789

Mobile Phone: 1 (555) 532-4444

Email: bubba@sparks.com

Change Password:

Confirm Password:

Save Changes

User can now enter new information, and select "Save Changes" when done to save their changes.

C. Map View

Name	Job Title	Desk Phone	Building	# floor
Aaron White	Section Manager	1 (916) 795-2538	LPW	3
Adeline Crockett	Java Developer	1 (916) 795-3355	LPW	3
Adrian Weaver	System Tester	1 (916) 795-9412	LPW	3
Arian McGuire	System Tester	1 (916) 795-2316	LPW	3
Ariandrea Comstock	Java Developer	1 (916) 795-2367	LPW	3
Alesia Dickey	Application Architect	1 (916) 795-2898	LPW	3
Alesia Ward	Business Analyst	1 (916) 795-0695	LPW	3
Aita Marroquin	Java Developer	1 (916) 795-0236	LPW	3
Amy Rowland	Java Developer	1 (916) 795-0638	LPW	3
Amie Worthworth	Java Developer	1 (916) 795-0859	LPW	3
Andrea Lam	Java Developer	1 (916) 795-3241	LPW	3
Angel Miller	Section Manager	1 (916) 795-1474	LPW	3
Angel Moody	Business Analyst	1 (916) 795-0162	LPW	3
Angelic Bender	Application Architect	1 (916) 795-3552	LPW	3
Anne Warren	Integration Developer	1 (916) 795-7361	LPW	3
Anne Freeman	Integration Developer	1 (916) 795-3163	LPW	3
Anthony Goodman	Section Manager	1 (916) 795-0259	LPW	3
Armond Downey	Java Developer	1 (916) 795-3722	LPW	3
Arne Winkefort	Frontend Developer	1 (916) 795-2026	LPW	3

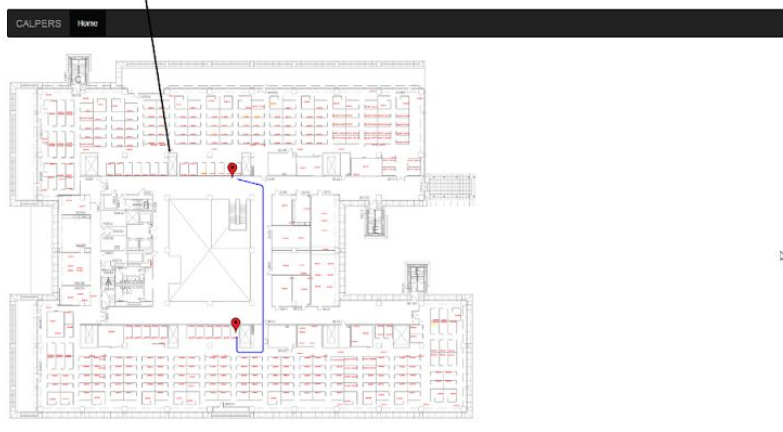
Employee is selected from the list, and user is redirected to employee information page.

Aaron White
Job Title: Section Manager
Desk Phone: 1 (916) 795-2538
Mobile Phone: 1 (916) 315-0058
Email: Aaron.White@calpers.ca.gov
Location: LPW, Floor 3

Receive Contact Information via Email

View Location

User selects "View Location" from the options on the employee information page, and is redirected to the map view of the office.



7. Programming/Development Tools

- **Notepad++:** IDE used to create all HTML, PHP, and Javascript code.
 - notepad++ was used because it is a great IDE when using many different languages. Using it allows us to have all of our code in the same area.
- **Paint.net:** Photo editing software used to create many of the images used in the SDD.
 - This was used because it is a great, free photo editing software.
- **Omnigraffle:** Graphics creator used to make ERD and Sequence Diagrams.
 - Omnigraffle was used due to its universal application in making diagrams.
- **ArgoUML:** Software used to create Class Diagrams.
 - ArgoUML was used for its specialized purpose in making class diagrams.
- **XAMPP:** Used to create a localhost for simulating connection to an actual database.
- **Bootstrap:** Used for creating the basic layout of the website.
- **Brackets:** Editor for nearly all languages(HTML, Javascript, JQuery, CSS, ...) with a built in-live preview for websites, and many plugin customizations.
- **Adobe Illustrator 2015:** Scalable Vector Graphics created and used for Kiosk Map interface.
- **Inkscape:** Free software that works with SVG just like Adobe Illustrator.
- **Google Chrome Web Development Debugger:** To find errors and debug code that was used by HTML.

8. Restrictions, Limitations, and Constraints

- Internet Explorer cannot be used, as it is formatted differently than other popular browsers, such as Google Chrome and Firefox.

9. References or Resources Used

Bootstrap and Stylesheets:

<http://snipplicious.com/css/bootstrap-3.2.0.min.css>

<http://snipplicious.com/css/font-awesome-4.1.0.min.css>

<https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js>

<http://maxcdn.bootstrapcdn.com/bootstrap/3.3.4/js/bootstrap.min.js>

<http://snipplicious.com/js/jquery.js>

<http://snipplicious.com/js/bootstrap.min.js>

<https://github.com/ucdavis/wayfinding>

10. Team Member's Roles and Approvals

Section 1: Tommy

Section 2: Dennis

Section 3: Tommy

Section 4: Gadfrey

Section 5: Gadfrey/Tommy

Section 6: Gadfrey/Dennis/Tommy

Section 7: Tommy/Dennis

Approval:

Tommy Kardach

Gadfrey Balacy

Dennis Liu
