

CMPE 332 Deliverable 3: Final Report

Dennis Grajo (20017666)

Jacob Denyes (20017035)

Trevor McLellan (20011091)

Assumptions

1. Each sponsoring company gets to send a maximum number of emails depending on sponsorship level.
2. Sponsoring Companies do not have to post job applications.
3. Companies do not have to post the pay rate, pay rate can remain null.
4. All Attendees must include their name.
5. Students are not required to include their school name and are not required to stay in a hotel room.
6. Jobs offers must list the location, but not the salary.
7. Each Committee must have a chair but does not have a minimum number of members outside of the chair themselves.
8. Only one session can occur in a room at a single time.
9. Speakers can be students, professionals or sponsors.

Discussion

Problems encountered and how we solved them

One of the problems we encountered when developing our application was that it was difficult to find errors when creating the website. When writing the php to access the database sometimes the website would not display the information we were expecting and it wouldn't throw an error because it was a problem with the SQL query. The solution we found for this was to write all of our queries in phpMyAdmin first to make sure they work and then copy them from there into our code. This ensured that our queries were not the part causing problems and we could narrow our search in finding the error.

Another problem that was encountered was that the original SQL file that the database was constructed on had some flaws and imperfections that the team was not aware of before. These imperfections had to be fixed and the database SQL file had to be updated in order to fix the problems that arose from them.

One such example was that the original SQL file did not include a different attribute for an attendee's first Name and last Name, only one attribute called Name. Although this was a minor problem, it is important to have a different attribute for first name and last name for a variety of reasons such as sorting or searching by first or last name.

Additionally, version control was not originally used. When the database was updated, the files then had to be sent to each member, and then each member had to manually replace them. This created room for human errors, which were made. Progress was slowed down by fixing bugs that shouldn't have existed if proper version control was used from the beginning.

Another such example is that the ID attribute of attendees did not have the auto-increment flag. Once this was added, it was much easier to add a new attendee to the database, without running into difficulties as a result of duplicate IDs as the database would increment it automatically.

Another such example is that a few foreign keys, such as the ID in Sponsors, did not have the flag "on delete cascade", which made it difficult to delete their referenced tuples (the default is to restrict deletion of their referenced tuples). The addition of this flag solved this problem.

One group member was operating on a Macintosh system. When implementing CSS, the program Bootstrapper was used, and XAMPP was used for viewing the database and website. These programs interacted slightly differently with the Mac system, creating errors and hindering the ability for the group to work on the project together.

Important design and implementation decisions

PHP code is unsupported by HTML files whereas PHP files support both PHP and HTML files, making them a more versatile option. Therefore, we came to the conclusion to build the website using PHP files exclusively, rather than HTML files.

Initially, once a user submitted a form, it would take them to a different page with the result or query they were looking for. They would then be able to navigate back to the page they were on using a back button. However, this method was found to be inefficient, as time is spent clicking the back button for a new query as well as loading different pages, as well as aesthetically displeasing. The improved method displayed the result on the same page as the forms, rather than take the user to a different page. This allowed the user to submit a new form quickly, as it was located on the same page, rather than having to click the back button first. This was done by adding php code that displayed a default query (for example: a view of all of the companies) upon entering one of the main pages. Then, upon form submission, it would take the user to a page that looked exactly the same as the previous one except with a different PHP code, based on the nature of the query performed.

Another design decision was to have certain pages display information once the user navigates to them, rather than just be a blank page. Take, for example, the jobs page. Initially, it consisted of only the submission form asking the user to specify the company whose jobs they want to see. Now, the page consists of the form as well as a table of all of the available jobs. This improved the aesthetics of the page as well as provided the user with information.

Using the drop down forms, it was possible for the user to select the default option, usually called “select [option]” and click submit. This would then lead to problems within the code since this option did not have an associated value to it - variables would be undefined. To prevent this, a switch case was used in which the submission of the “default” option would redirect the user to the same page - effectively doing nothing.

SQL injections are a possible threat that must be considered when allowing the user to submit forms where they are required to write text. To avoid this, all of the written input forms use prepared statements. The user is also unable to submit a blank or invalid text form, with an error message being displayed upon detection of invalid input.

One of the functional requirements states that the user must be allowed to add the different types of attendees. To add a sponsor, the user must also specify a company name. Initially, the user could input any company name, regardless of whether they were valid or actually at the conference, and a new sponsor with this company name would appear in the database. However, since this is clearly not ideal, a solution was required. Now, the PHP code checks to see if the company that the user entered exists within the database. Then, if it exists, the sponsor is created and added to the database.

Initially, each page had a link titled “Home” that would send the user back to a home screen with links to all of the other main pages. However, not only was this not aesthetically pleasing, this was also inefficient. Therefore, the team decided to add a navbar at the top of the page. Rather than clicking home and then clicking their desired option, the navbar allowed the user to navigate directly to the page they wanted to go to, cutting out the middleman entirely. This increased ease of access for the user as well as prevented the user from feeling lost when navigating the website.

The website uses the Bootstrap 4 framework for the website’s CSS. Bootstrap 4 was chosen, not only for its popularity, but also for its ease of use and the availability of documentation and integration examples of its different features. Initially, the website was comprised solely of HTML and PHP code. However, once this framework was laid out, most features (headers, forms, buttons, navbar) were updated with CSS using the documentation found in Bootstrap’s website to improve the overall aesthetics of the website.

The technology and tools used in developing the application

The main technology used in creating the website was a simple text editor to write the PHP files - Notepad++ and Sublime Text Editor. The second most important technology used was XAMPP and phpMyAdmin. These were very important pieces of software as they allowed us to create our database. It was also very useful for testing our queries to make sure they work and output the expected results. None of the team had past experience using XAMPP or phpMyAdmin at the beginning of the semester but we were able to quickly learn using the class notes for assistance. As stated in the previous section, Bootstrap 4 was used for the CSS component of the website.

Why you chose them and what your experience was using them

Notepad++ and Sublime Text Editor was chosen due to the team's familiarity with them. These text editors allowed the team to create and edit the PHP files efficiently due to their colour-coding as well as Sublime's auto-finish function. Bootstrap 4 was chosen due to its popularity and simple but pleasing aesthetics as well as the availability of documentation for it. The documentation available on the website made it easy to use Bootstrap 4 and implement it for the different features on the website.

Things we would like to change or go back and do differently

Within the navbar, there is a tab called Sponsors that takes the user to a page named Sponsorship Information that displays the different attending companies and their respective sponsorship levels. However, within the Company Information page (which can be accessed through the Company tab in the navbar) is a table that displays not only the company and their respective sponsor level, but also their max emails and emails sent, effectively making the table in Sponsorship Information redundant. As such, a change would be to remove the Sponsorship Information page, as well as the Sponsors tab to avoid redundancy.

Another change that we would like to implement would be adding the Bootstrap CSS to the tables that were displayed by the PHP code. Most of the other features of the website use Bootstrap CSS except for the tables and so this change would improve the overall look of the website.

Rather than have a Home tab in the navbar, a change that we would like to make would be to remove the Home tab and instead have the logo "Group 69's Database" in the navbar direct to the homescreen. Most websites have their logo in the navbar redirect to the homepage and so making this change would bring our website closer to industry standard.

One more change that would be welcome is the revamping of the website's home page. Currently, the homepage is just a photo of a landscape. The home page could be changed to a wallpaper featuring the logo of Group 69, as that would be more relevant. It could also include a short synopsis about the conference, as well as short descriptions for each of the navigation tabs to help the user better understand what is inside each tab and navigate the page more easily.

Finally use of version control would have made the assignment much easier to work on as a team. Much of the assignment was done individually, then extra time had to be spent to put together what had been done. Version control would have reduced the time to integrate the work greatly.

Relational Schema

Attendees(
 ID int not null primary key,
 Fee int not null,
 Name VarChar(20) not null)

Sponsors(
 ID int not null primary key,
 companyName VarChar(20) not null,
 foreign key (ID) references (Attendees),
 foreign key(companyName) references (Company))

Students(
 ID int not null primary key,
 schoolName VarChar(20),
 Program VarChar(20),
 roomNumber int,
 foreign key (ID) references (Attendees),
 foreign key (roomNumber) references (HotelRoom))

Professionals(
 ID int not null primary key,
 Job VarChar(20),
 foreign key (ID) references (Attendees))

Session(
 sessionID int not null primary key,
 sessionName not null,
 startTime time not null,
 endTime time not null,
 roomLocation VarChar(20) not null)

Company(
 companyName VarChar(20) not null primary key,
 sponsorLevel VarChar(20) not null,
 maxEmails int not null,
 emailsSent int not null)

HotelRooms(
 roomNumber int not null primary key,
 numberOfBeds int not null)

JobAdds(

jobID int not null primary key,
jobTitle VarChar(20) not null,
City VarChar(20) not null,
Province VarChar(20) not null,
payRate int,
companyName VarChar(20) not null,
foreign key (companyName) references (Company))

Speaks(

speakerID int not null,
sessionID int not null,
primary key(speakerID, sessionID),
foreign key (speakerID) references (Attendees),
foreign key (sessionID) references (Session))

CommitteeMembers(

ID int not null primary key,
Name VarChar(20) not null)

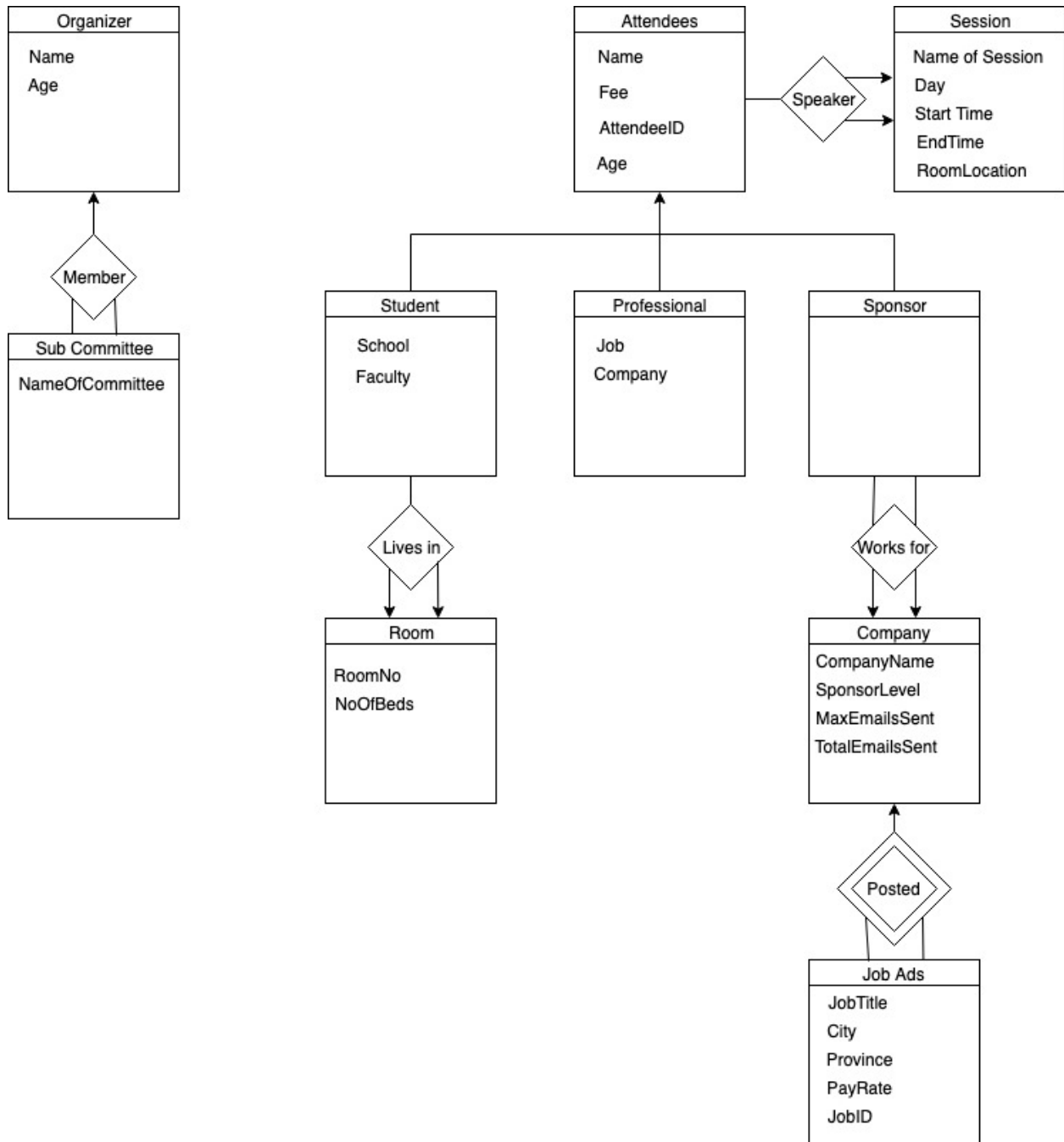
Committees(

chairID int not null primary key,
committeeName VarChar(20) not null,
foreign key (chairID) references (CommitteeMembers))

memberOnComittee(

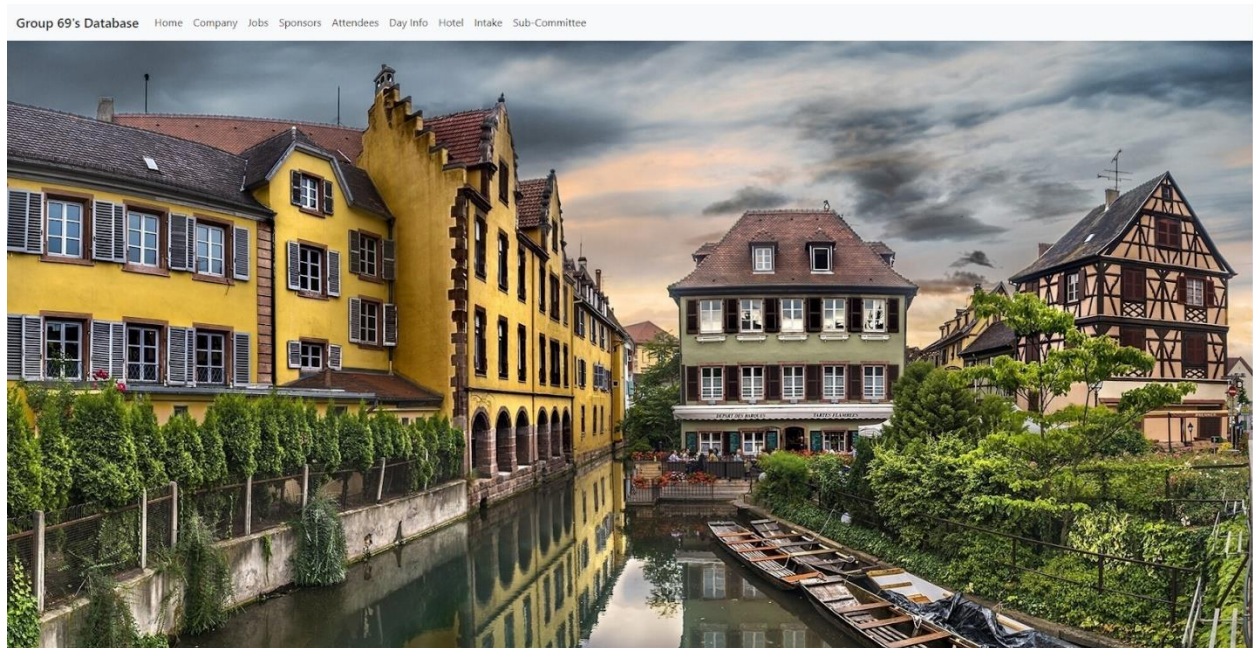
memberID int not null,
chairID int not null,
primary keys (memberID, chairID),
foreign key (memberID) references (CommitteeMembers),
foreign key (chairID) references (Committees))

ER Schema



User Guide

The opening screen of our website has a navigation bar at that top that allows the user to choose which action they would like to take.



Should the user click on the Company page, it will bring them to the Company home page where they can add a company to the database. Once they submit the required information, the table at the bottom showing the current enrolled companies will be updated.

Group 69's Database Home Company Jobs Sponsors Attendees Day Info Hotel Intake Sub-Committee

Company Information

Name: What is the company's sponsor level?

Please write the name of the sponsoring company to delete.

[Home](#)

Company	Sponsor Level	Max Emails	Emails Sent
Clarke Pub	Bronze	0	0
Deloitte	Platinum	5	0
Faskens	Gold	4	0
Pixar	Platinum	5	0

Should the user click on the Job page, it will bring them to the Jobs home page where they can write the company they would like to see jobs for and it will list the current jobs that company has available in a table format.

[Group 69's Database](#) [Home](#) [Company](#) [Jobs](#) [Sponsors](#) [Attendees](#) [Day Info](#) [Hotel](#) [Intake](#) [Sub-Committee](#)

Job Information

Please specify the company you want to see jobs for:

Submit

Listed are all of the jobs available

Company Name	Job Title	Province	City	Pay Per Hour (\$)
Clarke Pub	Bartender	Ontario	Kingston	16
Clarke Pub	Janitor	Ontario	Kingston	15
Deloitte	Accountant	Ontario	Toronto	25

Should the user click on the Sponsor page it will bring them to the Sponsor home page that will list all the companies currently enrolled and their sponsorship level.

[Group 69's Database](#) [Home](#) [Company](#) [Jobs](#) [Sponsors](#) [Attendees](#) [Day Info](#) [Hotel](#) [Intake](#) [Sub-Committee](#)

Sponsorship Information

Company	Level
Clarke Pub	Bronze
Deloitte	Platinum
Faskens	Gold
Pixar	Platinum

Should the user click on the Attendees page it will bring them to the Attendee home page. This is where they can either choose to see who is currently attending by selecting a demographic or add a new attendee by first selecting which demographic the new attendee will be. Once they select the demographic, a table will appear at the bottom of the page listing all those in that demographic who are attending.

Attendee Information

Please select which demographic you wish to view.

Select attendee type ▼

Submit

What kind of attendee would you like to add?.

Select attendee type ▼

Submit

Attendee Information

Please select which demographic you wish to view.

Student ▼

Submit

What kind of attendee would you like to add?.

Select attendee type ▼

Submit

First Name	Last Name	School	Program	Room Number
Jake	Peralta	Queen's	Eng	201
Amy	Santiago	Queen's	CompSci	201
Quentin	Tarantino	Guelph	Arts	202
Dennis	Grajo	Queen's	Eng	203
Jacob	Denyes	Queen's	Arts	204

Once they submit the chosen demographic to add, they will be redirected to a new page specific to that demographic so they can input the relevant information. For example: if they select the Student demographic, they will be redirected to the addStudent page, where they will input all relevant information, and when they submit, a table will appear at the bottom listing all the students attending, including the one they just added.

Student Information

First Name:

Last Name:

School Name:

Program Name:

Submit

First Name	Last Name	School	Program	Room Number
Jake	Peralta	Queen's	Eng	201
Amy	Santiago	Queen's	CompSci	201
Quentin	Tarantino	Guelph	Arts	202
Dennis	Grajo	Queen's	Eng	203
Jacob	Denyes	Queen's	Arts	204

Back to Attendees

Professional Information

First Name:

Last Name:

Job Title:

Submit

First Name	Last Name	Job Title
Michael	Scott	Developer
Jim	Halpert	Advertiser
Pam	Beasley	Recruiter
Oscar	Martinez	Artist
Creed	Bratton	Accountant
Patrick	Chan	Figure Skater

Back to Attendees

Sponsor Information

First Name: Last Name: Company Name:

First Name	Last Name	Company
Meredith	Grey	Clarke Pub
Elon	Musk	Clarke Pub
Peter	Jackson	Deloitte
Jack	Black	Deloitte
Martin	Short	Faskens
John	Smith	Faskens

Should the user click on the Day Info page, it will bring them to the Conference Schedule page where they can see the schedule for a certain day, or edit existing sessions. If they want to see the current schedule for a day, they can select a day and submit, and the table at the bottom will show the current schedule for that day. If they want to edit a session, they select the session they want to edit and then input the new location, start time, and end time. Once they submit the previously mentioned information, the table at the bottom will be updated to show the new schedule for that day.

Conference Schedule by Day

Please select the day that you want to see the schedule for.

Please select the session that you want to edit.

Please enter the new location of the session:

Start Time:

End Time

Schedule for all days

First Name	Last Name	Session	Start	End	Room
Quentin	Tarantino	Understanding Nature	13:00:00	14:00:00	Main Atrium
Amy	Santiago	Understanding Finance	12:00:00	13:00:00	Main Atrium
Jake	Peralta	Understanding You	13:00:00	14:00:00	Tertiary Atrium
Jamie	Fox	Using Knowledge	13:00:00	14:00:00	Main Atrium
Jake	Peralta	Implementing Skills	13:00:00	14:00:00	Main Atrium

Should the user click on the Hotel page, it will bring them to the Hotel main page where they can see all the current hotel rooms and how many beds are in each room. If they select a certain room, it will show a new table which lists all the students that are in each room.

Room Tenant Information

Enter Hotel Room:

Hotel Room Number 201

First Name	Last Name
Jake	Peralta
Amy	Santiago

Room Number	Number of Beds
201	2
202	1
203	1
204	1

If the user clicks on the Intake page, it will bring them to the Total Intake page where they can see how much money has been raised from registration as well as how much money has been raised through sponsorship.

Total Intake

Type	Total
Registration	900
Sponsorship	26000

If the user clicks on the Sub-Committee page, it will bring them to the Sub-Committee information page where they can see all of the sub committees.

Sub-Committee Information

Display which sub-committee?

Select Sub-Committee

Submit

Committee Name
Ticket Entrance Committee
Advertisement Committee
Budget Committee

If they select a sub-committee, it will list all of the people who are in that committee.

Sub-Committee Information

Display sub-committee?

Select Sub-Committee

Submit

Ticket Entrance Committee

First Name	Last Name
Franz	Ferdinand
Gavrilo	Princip

```

1<!DOCTYPE html>
2<html>
3<head>
4<link href="firstPage.css" type="text/css" rel="stylesheet">
5<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css" integrity="sha384-ggOyR0iXCbMVx3Ipa34ND+H1fQ784/j6cY/
6l3TQOHCW7x3V0v8XT2Zw1t" crossorigin="anonymous">
7</head>
8
9<nav class="navbar navbar-expand-lg navbar-light bg-light">
10<span class="navbar-brand mb-0 h1">Group 63's Database</span>
11
12<div class="collapse navbar-collapse" id="navbarNav">
13<div class="navbar-nav">
14<a class="nav-item nav-link active" href="index.php?home">Home</a>
15<a class="nav-item nav-link active" href="companyMainPage.php">Company</a>
16<a class="nav-item nav-link active" href="allJobs.php">Jobs</a>
17<a class="nav-item nav-link active" href="sponsors.php">Sponsors</a>
18<a class="nav-item nav-link active" href="attendeeMain.php">Attendees</a>
19<a class="nav-item nav-link active" href="dayMainPage.php">Day Info</a>
20<a class="nav-item nav-link active" href="hotel.php">Hotel</a>
21<a class="nav-item nav-link active" href="totalIntake.php">Intake</a>
22<a class="nav-item nav-link active" href="subcomFullpage.php">Sub-committee</a>
23</div>
24</div>
25</nav>
26<body>
27
28<div class="container">
29
30<h3>Company Information</h3>
31<form action="companyMainPageAdd.php" method="post">
32<div class="row">
33<div class="form-group col-md-4 align-items-center">
34<label for="addName">Name:</label>
35<input type="text" class="form-control" name="addName" placeholder="Company Name">
36</div>
37<div class="col-4" class="form-group">
38<label for="formLevel">What is the company's sponsor level?</label>
39<select class="form-control" name="formLevel" width: 50%;>
40<option value="">Select sponsor level</option>
41<option value="Bronze">Bronze</option>
42<option value="Silver">Silver</option>
43<option value="Gold">Gold</option>
44<option value="Platinum">Platinum</option>
45</select>
46</div>
47</div>
48<button type="submit" class="btn btn-primary">Submit</button>
49</form>
50
51<form action="companyMainPageDelete.php" method="post">
52<div class="row">
53<div class="form-group col-md-4">
54<label for="formDelComp">Please write the name of the sponsoring company to delete.</label>
55<input type="text" class="form-control" name="formDelComp" placeholder="Company Name">
56</div>
57</div>
58<button type="submit" class="btn btn-primary">Submit</button>
59</form>
60
61<?php
62#
63
64
65
66$addedCompanyName = $_POST["addName"];
67if(isset($_POST["formLevel"]))
68{
69$sponsorLevel = $_POST["formLevel"];
70switch($sponsorLevel)
71{
72case "Bronze": $maxEmails = 0; break;
73case "Silver": $maxEmails = 3; break;
74case "Gold": $maxEmails = 4; break;
75case "Platinum": $maxEmails = 5; break;
76default: break;
77}
78}
79
80
81echo "<table><tr><th>Company</th><th>Sponsor Level</th><th>Max Emails</th><th>Emails Sent</th></tr>";
82
83#connect to the database
84$pdo = new PDO("mysql:host=localhost;dbname=Conference", "root", "");
85
86if ($$sponsorLevel == "" || $addedCompanyName == "")
87{
88echo "<br>Invalid information<br><br>";
89}
90
91else{
92$sql = "insert into company (companyName, sponsorLevel, maxEmails, emailsSent) values (?, ?, ?, 0)";
93$stmt = $pdo->prepare($sql); #create the query
94$stmt->execute([$addedCompanyName, $sponsorLevel, $maxEmails]); #bind the parameters
95
96
97$rows = $pdo->query("select * from company");
98
99foreach($rows as $row) {
100echo "<tr><td>".$row["companyName"]."</td><td>".$row["sponsorLevel"]."</td><td>".$row["maxEmails"]."</td><td>".$row["emailsSent"]."</td></tr>";
101
102}
103
104}
105
106<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-q8i/X+965DzO0rT7abK41JstQAIAqgRvPzbzoSMxYYf3RMVx3Q" crossorigin="anonymous">
107<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js" integrity="sha384-Uo2eTTCm7MiHtbX3SzWArFnhd4" crossorigin="anonymous">
108<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" integrity="sha384-JjSmWgyb001/J3lnp4R14J56hH86MHjpRHH81JDH0G45n7368" crossorigin="anonymous">
109
110</table>
111</div>
112</body>
113</html>

```


Database Building SQL file: Conference.sql

```
1 DROP DATABASE IF EXISTS Conference;
2 CREATE DATABASE Conference;
3 USE Conference;
4
5 create table Attendees( -- attendees, students, professionals and sponsors
6 ID int not null auto_increment,
7 firstName varchar(40) not null,
8 lastName varchar(40) not null,
9 fee int not null, -- 50 for students, 100 for professionals, free for sponsors
10 PRIMARY KEY(ID)
11 );
12
13 create table Professionals(
14 ID int not null,
15 jobTitle varchar(40),
16 PRIMARY KEY (ID),
17 FOREIGN KEY (ID) REFERENCES Attendees(ID)
18 );
19
20 create table HotelRoom(
21 roomNumber int not null auto_increment,
22 numberOfBeds int not null, -- 1 or two beds only
23 PRIMARY KEY(roomNumber)
24 );
25
26 alter table HotelRoom auto_increment = 203;
27
28 create table Students(
29 ID int not null,
30 schoolName varchar(40),
31 program varchar(40),
32 roomNumber int,
33 PRIMARY KEY (ID),
34 FOREIGN KEY (ID) REFERENCES Attendees(ID),
35 FOREIGN KEY (roomNumber) REFERENCES HotelRoom(roomNumber)
36 );
37
38 create table Company(
39 companyName varchar(40) not null,
40 sponsorLevel varchar(40) not null, -- plat can send 5 emails, gold 4 emails, silver 3 emails and bronze 0 emails
41 maxEmails int not null,
42 emailSent int not null,
43 PRIMARY KEY (companyName)
44 );
45
46 create table Sponsors(
47 ID int not null,
48 companyName varchar(40) not null,
49 PRIMARY KEY (ID),
50 FOREIGN KEY (ID) REFERENCES Attendees(ID) on delete cascade,
51 FOREIGN KEY (companyName) REFERENCES Company(companyName) on delete cascade
52 );
53
54 create table JobAds(
55 jobId int not null,
56 companyName varchar(40) not null,
57 jobTitle varchar(40) not null,
58 province varchar(40) not null,
59 city varchar(40) not null,
60 province varchar(40) not null,
61 city varchar(40) not null,
62 payRate int,
63 PRIMARY KEY (jobID),
64 FOREIGN KEY (companyName) REFERENCES Company(companyName) on delete cascade
65 );
66
67 create table CommitteeMembers(
68 ID int not null,
69 firstName varchar(40) not null,
70 lastName varchar(40) not null,
71 PRIMARY KEY(ID)
72 );
73
74 create table Committees(
75 chairID int not null,
76 committeeName varchar(40),
77 committeeID int,
78 PRIMARY KEY (committeeID)
79 );
80
81 create table memberOnCommittee(
82 memberID int,
83 committeeID int,
84 PRIMARY KEY(memberID, committeeID),
85 FOREIGN KEY (memberID) REFERENCES CommitteeMembers(ID),
86 FOREIGN KEY (committeeID) REFERENCES Committees(committeeID)
87 );
88
89 create table Sessions(
90 sessionName varchar(40) not null,
91 sessionID int not null,
92 day int,
93 startTime time not null,
94 endTime time not null,
95 roomLocation varchar(20) not null,
96 PRIMARY KEY (sessionID)
97 );
98
99 create table Speaks(
100 speakerID int not null,
101 sessionID int not null,
102 PRIMARY KEY(speakerID, sessionID),
103 FOREIGN KEY (speakerID) REFERENCES Attendees(ID),
104 FOREIGN KEY (sessionID) REFERENCES Sessions(sessionID)
105 );
106
107 insert into Attendees values(1000,"Jake", "Peralta", 50);
108 insert into Attendees values(1001,"Amy", "Santiago", 50);
109 insert into Attendees values(1002,"Quentin", "Tarantino", 50);
110 insert into Attendees values(1003,"Jamie", "Fox", 50);
111
112 insert into Attendees values(2000,"Michael", "Scott", 100);
113 insert into Attendees values(2001,"Jim", "Halpert", 100);
114 insert into Attendees values(2002,"Pam", "Beasley", 100);
115 insert into Attendees values(2003,"Oscar", "Martinez", 100);
116
117 insert into Attendees values(3000,"Peter", "Jackson", 0);
118 insert into Attendees values(3001,"Martin", "Short", 0);
```

```

115 insert into Attendees values(3000,"Peter", "Jackson", 0);
116 insert into Attendees values(3001,"Martin", "Short", 0);
117 insert into Attendees values(3002,"Meredith", "Grey", 0);
118 insert into Attendees values(3003, "Jack", "Black", 0);
119
120 insert into HotelRoom values(201, 2);
121 insert into HotelRoom values(202, 1);
122
123
124 insert into Students values(1000,"Queen's", "Eng",201);
125 insert into Students values(1001,"Queen's", "CompSci",201);
126 insert into Students values(1002,"Guelph", "Arts",202);
127
128 insert into Professionals values(2000,"Developer");
129 insert into Professionals values(2001,"Advertiser");
130 insert into Professionals values(2002,"Recruiter");
131 insert into Professionals values(2003,"Artist");
132
133 insert into Company values("Deloitte","Platinum", 5, 0);
134 insert into Company values("Faskens","Gold", 4, 0);
135 insert into Company values("Clarke Pub","Bronze", 0, 0);
136
137 insert into Sponsors values(3000, "Deloitte");
138 insert into Sponsors values(3001, "Faskens");
139 insert into Sponsors values(3002, "Clarke Pub");
140 insert into Sponsors values(3003, "Deloitte");
141
142 insert into JobAds values(00001, "Clarke Pub", "Bartender", "Ontario", "Kingston", 15.75);
143 insert into JobAds values(00002, "Clarke Pub", "Janitor", "Ontario", "Kingston", 14.75);
144 insert into JobAds values(00003, "Deloitte", "Accountant", "Ontario", "Toronto", 25.25);
145
146 insert into CommitteeMembers values(001, "Mary", "Jane");
147 insert into CommitteeMembers values(002, "Elon", "Zuckerberg");
148 insert into CommitteeMembers values(003, "Franz", "Ferdinand");
149 insert into CommitteeMembers values(004, "Peter", "Parker");
150 insert into CommitteeMembers values(005, "Steve", "Gates");
151 insert into CommitteeMembers values(006, "Gavrilo", "Princip");
152
153 insert into Committees values(001, "Tricket Entrance Committee", 001);
154 insert into Committees values(002, "Advertisement Committee", 002);
155 insert into Committees values(003, "Budget Committee", 003);
156
157
158 insert into memberOnCommittee values(001, 002);
159 insert into memberOnCommittee values(002, 003);
160 insert into memberOnCommittee values(003, 001);
161 insert into memberOnCommittee values(004, 002);
162 insert into memberOnCommittee values(005, 003);
163 insert into memberOnCommittee values(006, 001);
164
165
166 insert into Sessions values("Understanding Nature", 01, 01, "13:00:00", "14:00:00", "Main Atrium");
167 insert into Sessions values("Understanding Finance", 02, 01, "13:00:00", "14:00:00", "Secondary Atrium");
168 insert into Sessions values("Understanding You", 03, 01, "13:00:00", "14:00:00", "Tertiary Atrium");
169 insert into Sessions values("Using Knowledge", 04, 02, "13:00:00", "14:00:00", "Main Atrium");
170 insert into Sessions values("Implementing Skills", 05, 03, "13:00:00", "14:00:00", "Main Atrium");
171
172 insert into Speaks values(1000, 02);
173 insert into Speaks values(1001, 02);
174
175 insert into Speaks values(1001, 02);
174 insert into Speaks values(1002, 01);
175 insert into Speaks values(1003, 04);
176 insert into Speaks values(1000, 05);

```