

# Welcome to VibEvents

A real time event management application

<http://vibeevents.x10host.com/>

CSCI 455 Senior Project

Professor Maherukh Akhtar

## Team Members:

NAME	STUDENT ID	PHONE #	EMAIL
<b>Mansi Parikh</b> (Team Leader)	1078420	929-328-4021	mparik15@nyit.edu
<b>Alexander Khemai</b>	1072075	917-968-5962	akhemai@nyit.edu
<b>Amitoj Johar</b>	1069158	347-233-5923	ajohar01@nyit.edu
<b>Danny Rivas</b>	1068812	347-990-0846	drivas02@nyit.edu
<b>Alexander Gronowski</b> (Lead Developer)	1076186	347-336-5962	agronows@nyit.edu

# **INDEX**

# **Pg. No**

<b>1.0 - ABSTRACT</b>	<b>5</b>
<b>1.1 - Motivation</b>	<b>5</b>
<b>1.2 - Introduction</b>	<b>6</b>
<b>1.3 - Purpose</b>	<b>6</b>
<b>1.4 - Existing System</b>	<b>6</b>
<b>1.5 - Proposed System</b>	<b>7</b>
<b>1.5.1- Scope</b>	<b>8</b>
<b>1.5.2- Users</b>	<b>8</b>
<b>1.6 - Software Model</b>	<b>9</b>
 <b>2.0- DATABASE DESIGN</b>	 <b>10</b>
<b>2.1 - Entity Relationship Diagram</b>	<b>10</b>
<b>2.2 - Architecture Design</b>	<b>11</b>
 <b>3.0- IMPLEMENTATION</b>	 <b>11</b>
<b>3.1 - Technologies Used</b>	<b>11</b>
<b>3.2 - Code Samples</b>	<b>14</b>

<b>3.3 - Timeline</b>	<b>30</b>
<b>4.0- LIMITATIONS</b>	<b>31</b>
<b>5.0- DIFFICULTIES FACED</b>	<b>31</b>
<b>6.0- FUTURE EXPECTATIONS</b>	<b>32</b>
<b>7.0- CONTRIBUTION AND LEARNING OUTCOMES</b>	<b>33</b>

## 1.0 ABSTRACT:

Event management is similar to project management, but in the sense of creation and advancement of events of any scale, whether they are large-scale like concerts or conferences, or small-scale like friendly meetups or a study session. The act of planning and coordinating is also a substantial part of event management. Every event is different in its own nature, therefore event management operations can easily be stressful or intimidating. The industry of event management usually caters events of all sizes, many corporations, organizations, groups and individuals, orchestrate events in order to advertise or market, convene business relationships, raise money or capital, or just for recreation.

## 1.1 MOTIVATION:

The motivation behind coming up with the idea to build this application is that if a person is new to the city, or maybe even currently living in the city, and they want to plan an event, there should be a place where they can do that which has everything they need to plan an event under one roof. For any kind of event, whatever services they need, they should be provided a unified application that can be a go-to place for them.

The process of event planning can be a daunting task, either for a business professional or the average person. The application designed was focused on putting less time in planning an event and more time in experiencing and enjoying it. The intention of the application is to bring a fresh, new take on event planning. The technologies that will be used are: React Native, HTML, CSS, SQL, JavaScript, Bootstrap. React Native was chosen due to its ability to run on multiple platforms. The recommendation of vendors will be done based on filters to allow users to find the best vendors to plan the event. The users will have vendors be made available to them based on the categories they choose, and the type of event they are planning. Also, vendors will be able to list their business to have their services utilised, by listing the services and location they can provide to the users.

## **1.2 INTRODUCTION:**

The application was created to make event planning easier for individuals, that can be used in the professional world, or for the average person who's trying to get together with friends. Also, this application is to be more personalized than its competitors, specifically "caring" about the user's choices. Cross platform support is also to be utilized as it is important to appeal to as many users as possible. Integrating technology is important to the user's experience. According to Enterprise Event Marketing, event technology can improve event attendance by as much as 30%, increase productivity by 20% and decrease costs.

## **1.3 PURPOSE:**

The purpose of the application is to implement a real time event management application with seamless mobile integration. There is a web application and a mobile application. The mobile application can run smoothly on both Android and iOS. On the other hand, the web application scales well to mobile screens and is mobile responsive.

## **1.4 EXISTING SYSTEMS:**

Some existing systems are: EventBrite and Meetup. These technologies do provide a good method of organizing and planning events. Eventbrite is a very popular event management service, although this service isn't very personalized. It lets users create an event and generate a number of tickets for it, which can be paid or free. With Meetup, users must register and join a group before attending an actual event, which can be a burden for one-time or casual users.

## 1.5 PROPOSED SYSTEM:

The proposed system is a personalized event planning application which will run on mobile phones.

- The user will be able to select the category and type of event that they want to create, or even create the category of event themselves. Based on the event that they wish to plan, recommendations from vendors for different categories will be shown to them, like food, decorators, cake, places, entertainment etc..
- Another aspect of the application is the vendors. After a vendor is selected, the vendor will be booked for them. Vendors who wish to sell their services can register themselves in the application and provide them to the users planning the events. The services of the vendors will be listed for the user to select from.
- When a vendor signs up, it can either host events at their location, or just provide their services, like catered food, cakes or decorations and security.

What makes the proposed system different from existing systems is that the proposed system is more of a planning app, while existing systems, like EventBrite is a management app. The proposed system is more of a personalized application, rather than commercial one like EventBrite. EventBrite is about getting people to the event and promoting it, this one is about helping people plan it. It also helps the different vendors in their businesses by them listing their service here and getting paid for it when selected the user.

Once planners can create an event users are able to view a page where they see pictures of the proposed event, documents by the planners for the event, location of the event, cost of the event, attire, type of event, and other information about the event. The user will also be allowed to make comments on the event as a whole and view them.

Events requires venues, food, staff, furniture, and other items. Planners for events can make tens or hundreds of calls planning an event trying to get the best price, the app would aggregate those vendors and create a simple and efficient

process for vendors and planners to communicate. Vendors can offer their services to planners and planners can ask vendors for their services. Furthermore, planners can view all vendors and the services they provide to see what they offer that planners may not know what they want. The interface of the app would connect the vendors and users ensuring that the planner gets the best possible service quickly.

### 1.5.1 SCOPE

Our scope is limited to getting a few vendors already existing in our database. We are limiting the functionality of our to accepting vendors from only New York City, because we thought it would be a good start to focus in a specific area first.

### 1.5.2 USERS

We have two main kinds of users of our App:

1. **User** : Users are individuals that want to create an event. They sign up, create an event, and then choose from a list of available vendors that provide the service that they are looking for, eg: vendors for cakes, vendors for security, etc.
2. **Vendor** : Companies or Individuals who want to provide their services and location for events created by the user. They sign up, then list their services, and/or provide a location for their services for the users to use for an event they wish to host.



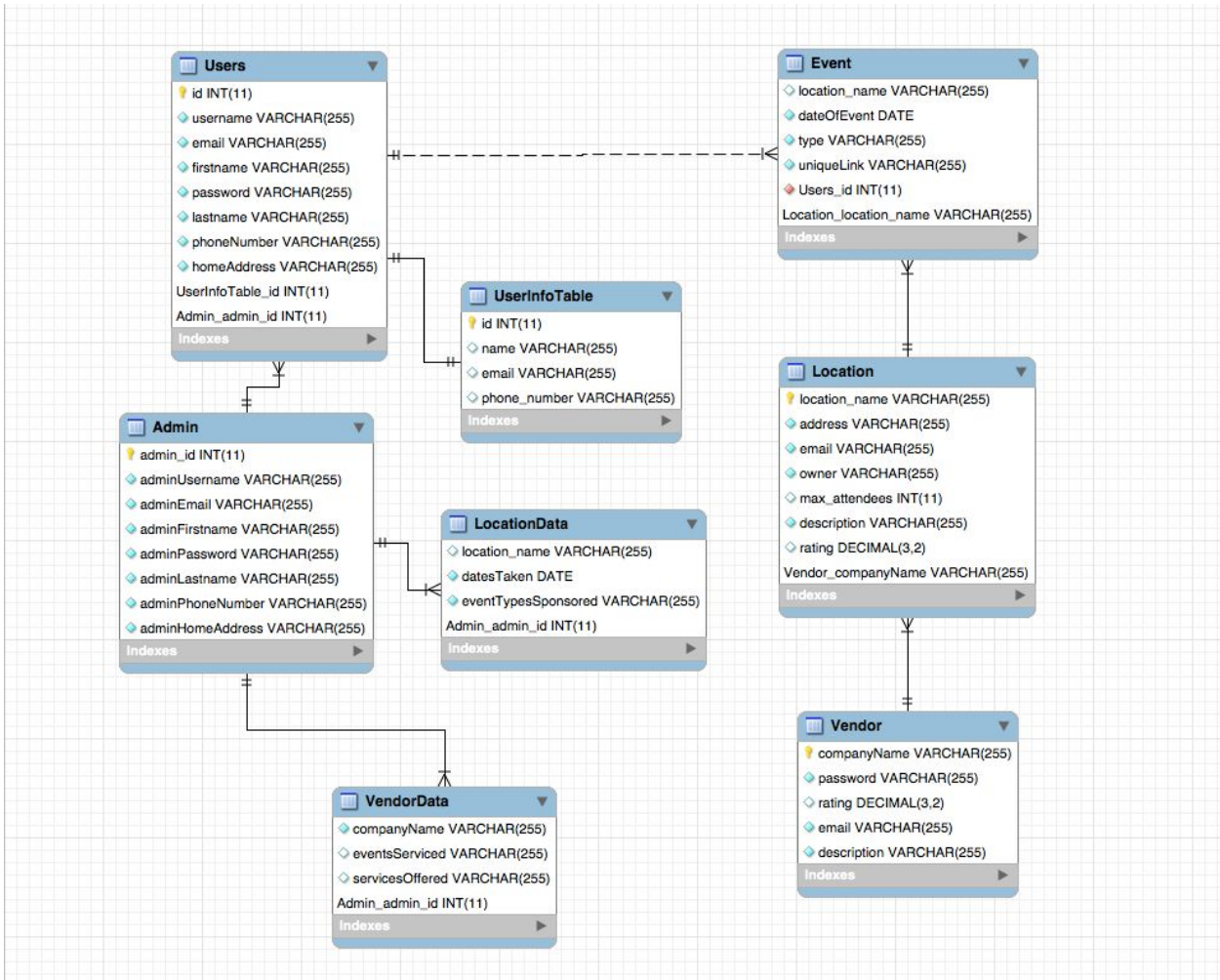
## 1.6 SOFTWARE MODEL

We've decided that the Agile software development model would be the best fit for the project. The reason being, it's not only fast but it requires that component parts of the application are built, tested and deployed at every stage of development. It is also feature driven which means that a team member can focus on a feature. The only limitation might be that it can require multiple team members to work on some phases of code construction like the UI/Front-end design and database/back-end implementation. Agile is designed to accommodate change and the need for faster software development. It is less about milestones than it is about hours, feature selection, prioritization, and meetings.

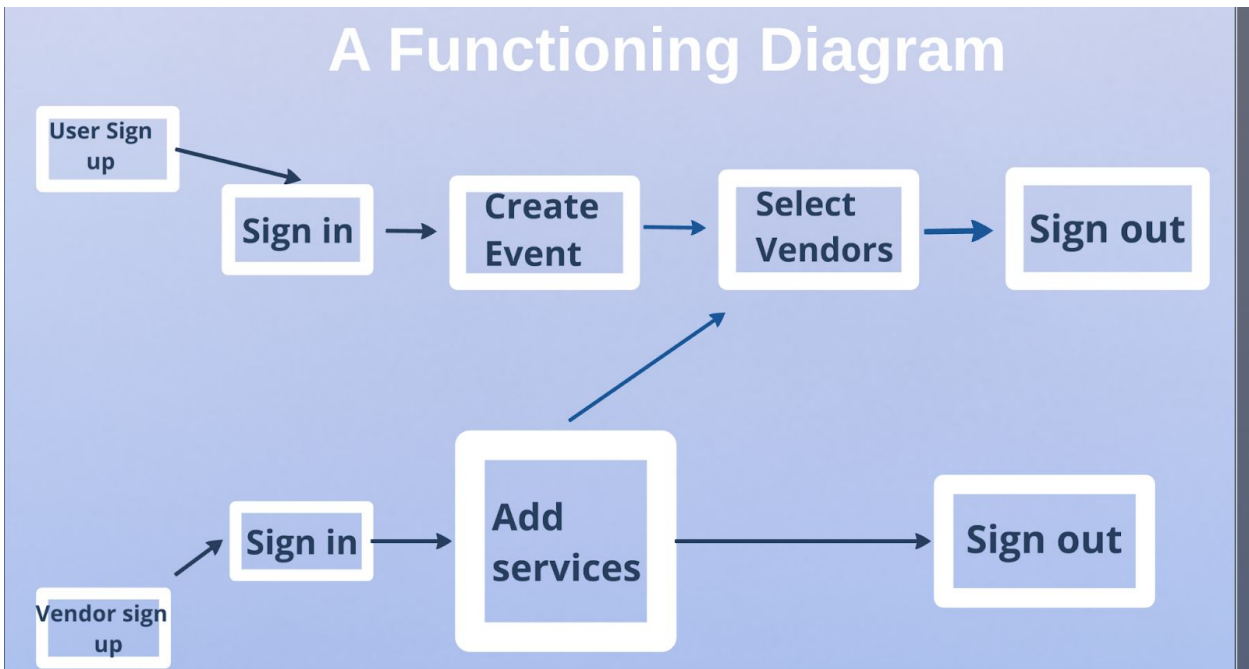
The entire project was divided into small iterations or phases, where development and testing of each iteration is done at the end of that iteration, not waiting for the entire project to finish. The overall goal of each Agile method was to adapt to change and deliver working software as quickly as possible.

## 2.0 DATABASE DESIGN:

### 2.1 ENTITY RELATIONSHIP DIAGRAM:



## 2.2 ARCHITECTURE DESIGN:



## 3.0 IMPLEMENTATION:

### 3.1 TECHNOLOGIES USED:

#### -WEB APPLICATION

The front-end of the website was created with HTML and CSS. Bootstrap, an HTML, CSS and JS framework, was also utilized to aid to with the creation of



the interface, styling and scaling.

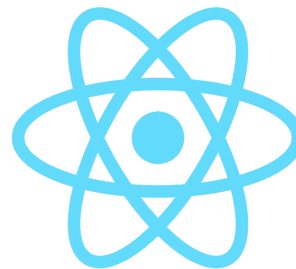
Bootstrap was adopted because it is the most popular front-end framework as well as one of the most popular frameworks in web development in general.

#### **-MOBILE APPLICATION:**

The front-end of the mobile app was created with React Native, a framework for building cross platform, native applications with just JavaScript. React Native is built off of the ReactJS JavaScript library, both developed by Facebook. React Native uses the same fundamental interface building blocks as regular iOS and Android apps use, just by using JavaScript, instead of using Swift or Java. React Native was delegated for the construction of the mobile app because of its engaging interface it promised. However, React Native has a steep learning curve, making it difficult to learn in the beginning. Companies like Facebook, Uber, Instagram, and Walmart utilize React Native. Android Studio and Xcode were used to build the mobile application on Android and iOS, respectively.

#### **-BACKEND:**

The database used was MySQL. It was chosen because relations between data were needed, especially since much of the data in this project relies on some other data. The back-end was built from PHP. It was used to make REST API endpoints, and connect the front-end to the back-end. PHP is also very commonly associated with MySQL.



X10hosting.com was used for hosting the web application and the respective API endpoints. X10hosting also supports PHP and MySQL out of the box. Additional features also include a user friendly control panel and built in code editor.

#### **-COMMUNICATION:**

For Communication and Code sharing, GitHub was used. Discord was used to conduct weekly group meetings and remotely working with each other.

#### **-ADDITIONAL TECHNOLOGIES AND TOOLS**

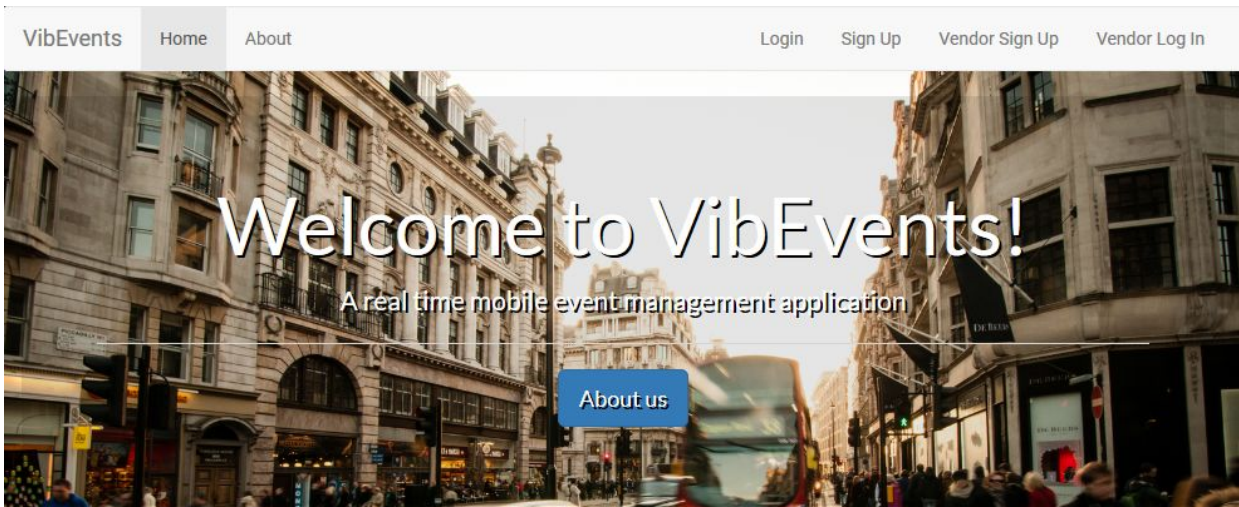
- Windows 10/Mac OS X
- Sublime Text 3/Atom
- Adobe PhotoShop
- React Native
- MySQL
- phpMYADMIN
- JavaScript
- JQuery
- SMTP Server

## 3.2 CODE SAMPLES:

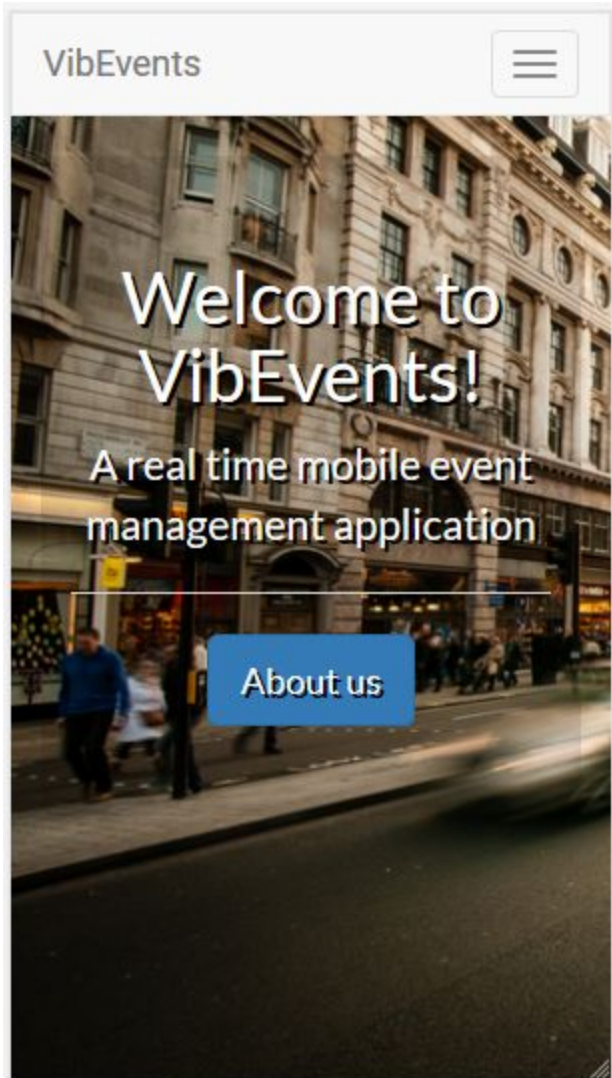
- MOBILE APPLICATION:

```
34 }
35
36 async saveKey(value) {
37   try {
38     await AsyncStorage.setItem('sessionUsername', value);
39   } catch (error) {
40     console.log("Error saving data" + error);
41   }
42 }
43
44 InsertDataToServer = () =>{
45
46   const { TextUserName } = this.state ;
47
48
49   fetch('http://vibeevents.x10host.com/restApi/reset.php', {
50     method: 'POST',
51     headers: {
52       'Accept': 'application/json',
53       'Content-Type': 'application/json',
54     },
55     body: JSON.stringify({
56
57       username: TextUserName,
58       type: "company"
59     })
60   })
61
62   }).then((response) => response.json())
63   .then((responseJson) => {
64
65     // Showing response message coming from server after inserting r
66     // Alert.alert(this.state.sessionUsername);
67
68     Alert.alert(responseJson);
69
70
71   }).catch((error) => {
72     console.error(error);
73   });
74
75 }
```

- **WEB APPLICATION:**





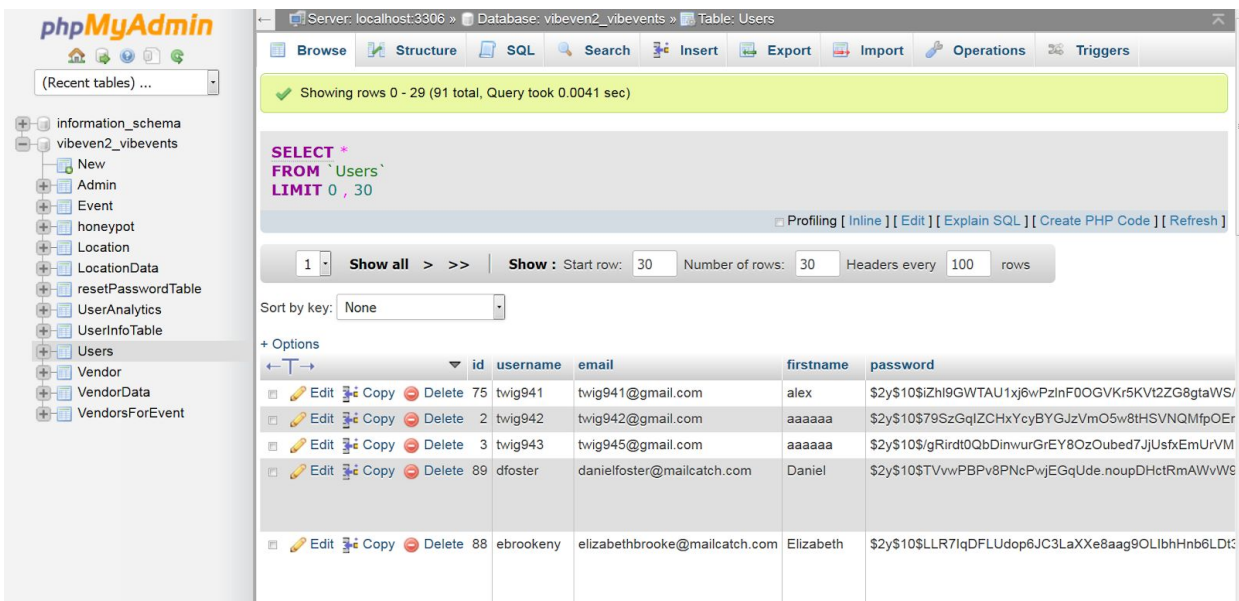


- **APP LOGO DESIGNS:**





- Back-end: We used PHP and MySQL together. We also used phpMyAdmin to view all of our tables in our database, to better monitor our website.



This is how we connected to our MySQL database with PHP using our host name, database name, username, and database to connect to our MySQL database and run queries on it.

```
<?php
require_once 'UserLogin.php';

if ($_SERVER['REQUEST_METHOD'] == 'POST') {
    $login = new
    UserLogin('localhost','vibeven2_alexgronowski1','testPW','vibeve
n2_vibevents');

    $login->login($_POST["username"], $_POST["password"]);
}

?>
```

We heavily used object oriented programming to create reusable classes. One example of this is that we created an email class that sent emails to users to welcome them and reset their passwords regardless of if they were a user or vendor.

```
public function resetPassword($username, $type) {

    // Subject
    $subject = 'Password Reset VibEvents';

    // Message
    $message = 'See code for full HTML and CSS email code';

    // To send HTML mail, the Content-type header must be set
    $headers[] = 'MIME-Version: 1.0';
    $headers[] = 'Content-type: text/html; charset=iso-8859-1';

    // Additional headers
    $headers[] = 'From: Password Reset
    <PasswordReset@vibeevents.x10host.com>';

    // Mail it

    //find email where to send the message
    $email = "";
    if($type === "user") {
        $email = $this->getUserEmail($username);
    }
    else if($type === "company") {
        $email = $this->getCompanyEmail($username);
    }
}
```

```

}
if($email !== "") {
if(mail($email, $subject, $message, implode("\r\n", $headers)))
{
return "We Sent An Email to " . $email;
}
else {
return "Something Went Wrong When Sending The Email";
}
}
else {
return "No Such Username or Company Name Exists";
}
}

```

We then implemented these functions depending on the url query string in order to make the code more reusable and abstract. In the code snippet below when the user clicked on forgot password they would be redirected to PasswordReset.php where the url query string would be set depending on what type of account they want to reset the password with. Then given the username or company name a reset password email would be sent from our SMTP server.

```

require_once'Mail.php';
if ($_SERVER['REQUEST_METHOD'] == 'POST') {
    $mail = new
Mail('localhost','vibeven2_alexgronowski1','testPW','vibeven2_vibe
vents');
if(!empty($_GET["type"])) {

```

```

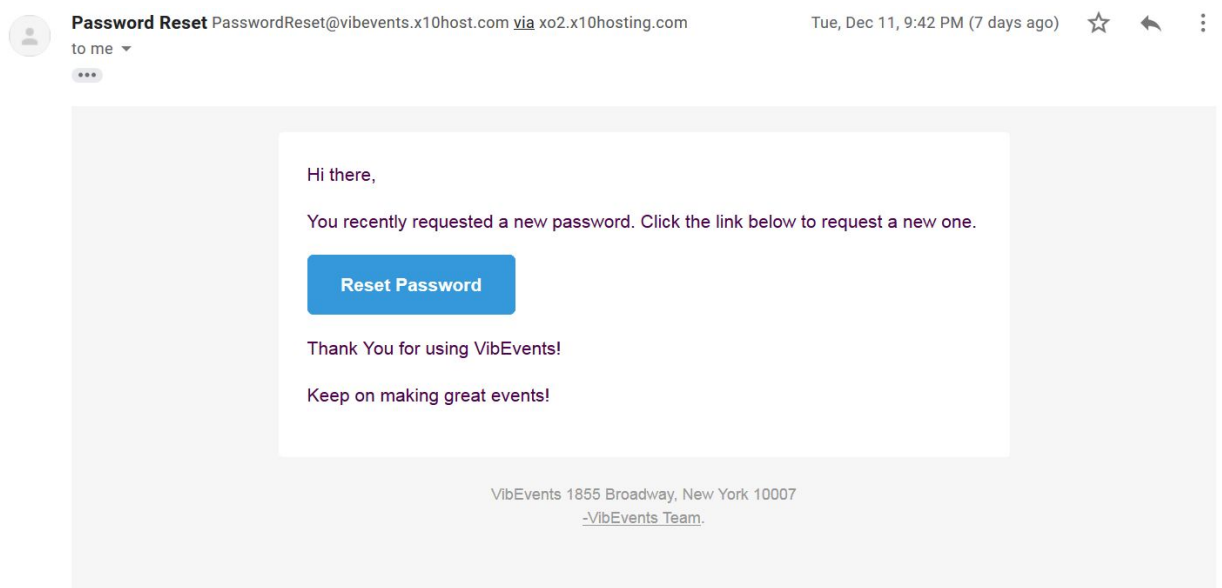
if($_GET["type"] === "user") {
echo "<div class='text-primary' style = 'text-align:center;
font-size: 150%;font-weight: 600;'>" .
$mail->resetPassword($_POST["username"], "user") . "</div>";
}

else if($_GET["type"] === "company") {
echo "<div class='text-primary' style = 'text-align:center;
font-size: 150%;font-weight: 600;'>" .
$mail->resetPassword($_POST["username"], "company") . "</div>";

}
}
}

```

### Example of an email sent from our server:



Our reset password table that contains the unique link that the user would click to reset their password. After they reset their password their link changes or if 24 hours pass their link changes automatically for security reasons.

+ information_schema - vibeven2_vibeevents + New + Admin + Event + honeypot + Location + LocationData + resetPasswordTable + UserAnalytics + UserInfoTable + Users + Vendor + VendorData + VendorsForEvent	+ Options		
	usernameOrCompanyName	type	resetLink
	tttttt	company	a11af750315dc
	alexg1	user	4540905697527
	realEmail	user	137d8f4c936cb
	AlexG	company	ec523ba5f0b75
	iforgotpas	user	85922794d2722
	forgotpas2	user	61724c108c1a2
	NYIT	company	418341e77d74a
	alexgrono	user	74d5d2b71cf30
	alexgron	user	b52b994c3e9da
	Michael	company	43f349e0e10fa
	zzz	company	bd6f42e68fb5e
	qqqqqq	company	32858c3299e9f
	twig941	user	eec3e583a84c3
	amitoj	user	d2c41aeb57995
	Amitoj	company	d2c41aeb57995
	ilovepandas	company	661a39432f314
	Jordan's Bridal Supplies	company	7416f2922dd33
	hclayton	user	2ee1ab0a0a89e
	ekoch	user	76d71634b22ed
	cperez	user	2c33fecdb872a
	cdamon	user	85372bfcdb88f
	cclark	user	a16e85892953b
	chrissum2	user	0675d3af3e887
	covington	user	4a3ca67946d28

In the code we also used sessions to control what the user sees, the best example of this is in our navigation menu where we check if someone is logged in and then display a logout button, their name, and links to traverse the website.

```
<nav class="navbar navbar-default">
  <div class="container-fluid">
    <div class="navbar-header">
```

```

        <button type="button" class="navbar-toggle collapsed"
data-toggle="collapse" data-target="#navbar"
aria-expanded="false" aria-controls="navbar">
            <span class="sr-only">Toggle navigation</span>
            <span class="icon-bar"></span>
            <span class="icon-bar"></span>
            <span class="icon-bar"></span>
        </button>
        <a class="navbar-brand" href="">VibEvents</a>
    </div>
    <div id="navbar" class="navbar-collapse collapse">
        <ul class="nav navbar-nav">
            <li class="active"><a
href="Home.php">Home</a></li>
            <li><a href="About.php">About</a></li>
            <?php
                if (isset($_SESSION["username"]) &&
!isset($_SESSION["companyName"])) {
                    echo '<li><a href="ViewAllEvents.php">View
All Your Events</a></li>';
                }
            ?>
            <!-- <li><a href="ViewAllEvents.php">View All
Your Events</a></li> -->
        </ul>
        <ul class="nav navbar-nav navbar-right">
            <?php

```

```

        if (!isset($_SESSION["username"])) &&
!isset($_SESSION["companyName"])) {
            echo '<li><a href="Login.php">Login</a></li>
<li><a href="Signup.php">Sign Up</a></li>';
            echo '<li><a href="VendorSignup.php">Vendor Sign
Up</a></li>';
            echo '<li><a href="VendorLogin.php">Vendor Log
In</a></li>';
        }

        if (isset($_SESSION["username"])) {
            echo '<li><a href="ProfilePage.php">Welcome ' .
$_SESSION["username"] . '</a></li>';
            echo '<li><a href="CreateEvent.php">Create An
Event ' . '</a></li>';
            echo '<li><a href="Signout.php">Sign
Out</a></li>';
        }

        if(isset($_SESSION["companyName"])) {
            echo '<li><a href="ProfilePage.php">Welcome
' . $_SESSION["companyName"] . '</a></li>';
            echo '<li><a href="Signout.php">Sign
Out</a></li>';

        }

```

```

?>

</ul>
</div><!--/.nav-collapse -->
</div><!--/.container-fluid -->
</nav>

```

- **Front-end/Back-end Integration**

Our Mobile application sends a request to an api endpoint on our server, then our server replies with a response.

An example of inserting an item into the database using our REST API, we use the same methods as our web application to insert the data into the database:

```

<?php
require_once 'AddServicesCompany.php';

$json = file_get_contents('php://input');

// decoding the received JSON and store into $obj
variable.

$obj = json_decode($json,true);

$create = new
AddServices('localhost','vibeven2_alexgronowski1','testPW',
'vibeven2_vibevents', $obj["owner"], $obj["eventType"],
$obj["serviceoffered"]);

$msg = 'Your Service Could Not Be Inserted';

```



```

if($create->insertService('')) {
    $MSG = 'Your Service Has Been Inserted';
}
// Converting the message into JSON format.
$json = json_encode($MSG);
// Echo the message.
echo $json;
?>

```

- **SECURITY:**

-Added protection in the application against SQL injections & cross-site scripting.

SQL Injections and cross-site scripting are the two most common application security vulnerabilities. We fixed these by sanitizing all user input using industry tested functions.

```

class User extends db {
    private $username;
    private $email;
    private $firstName;
    private $lastName;
    private $password;
    private $phoneNumber;
    private $homeAddress;

    public function __construct($host, $username, $password, $db_name, $Username, $Email, $FirstName, $Password, $LastName,
$PhoneNumber, $HomeAddress) {
        parent::__construct($host, $username, $password, $db_name);
        $this->username = htmlspecialchars($this->connection->real_escape_string($Username));
        $this->email = htmlspecialchars($this->connection->real_escape_string($Email));
        $this->firstName = htmlspecialchars($this->connection->real_escape_string($FirstName));
        $this->password = password_hash(htmlspecialchars($this->connection->real_escape_string($Password)),
PASSWORD_DEFAULT);
        $this->lastName = htmlspecialchars($this->connection->real_escape_string($LastName));
        $this->phoneNumber = htmlspecialchars($this->connection->real_escape_string($PhoneNumber));
        $this->homeAddress = htmlspecialchars($this->connection->real_escape_string($HomeAddress));
    }
}

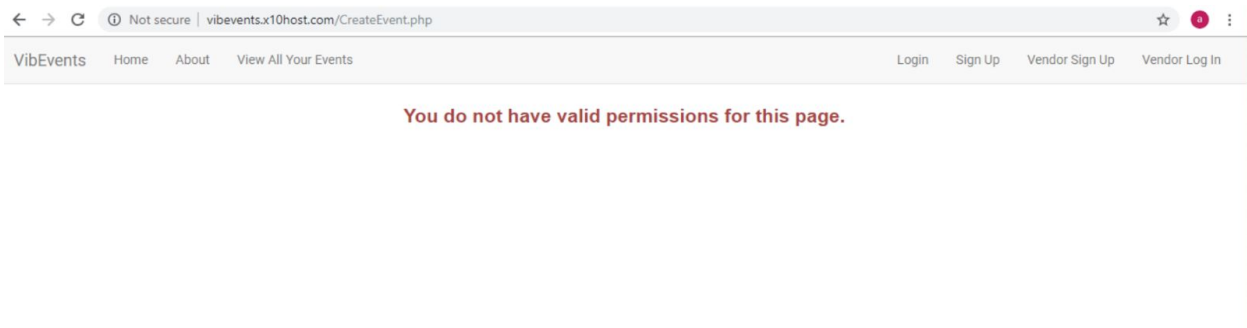
```

We also Sanitize user input from unexpected places like URL query string.

```
if(!empty($_GET["reset"])) {  
if(isset($_POST["resetpass"])) {  
echo $mail->resetPasswordAndUpdate($_GET["reset"], $_POST["password"]);  
}  
  
echo '    <form method = "POST" action = "">  
        <div class="form-group col-md-4 col-md-offset-4">  
            <label for="formGroupExampleInput2">Password</label>  
            <input pattern=".{5,10}" required title="5 to 10 characters" name = "password" type="password" class="form-control"  
placeholder="Enter your Password (5 to 10 Characters)" required>  
        </div>  
  
        <div class="form-group col-md-4 col-md-offset-4">  
<input style = "padding-left: 25px; padding-right: 25px; font-weight: 600" type = "submit" class="btn btn-primary" value =  
"Reset Your Password" name = "resetpass">  
        </div>  
</form>';  
}
```

### -Enabled Access control:

Implemented access control depending on whether someone is logged in or not.  
If and only if someone is logged in as a user can they can access this page.



If someone logs in as a vendor they also can't access the pages since it is only for users, and vice versa. This has been implemented across the whole website.

**-Implemented validation on both the front-end and the back-end of the application.**

**-Prevent outside websites from using content for security reasons.**

**-Hashed Passwords for additional security**

Hashed passwords so even if the website is hacked, hackers can never gain access to the original passwords.

9	ekoch	ekoch@mailcatch.com	Estene	\$2y\$10\$2F1KncibbqinDKuRAm3pdnWt04HEvK39ZsKV1Zn3a1...
0	cperez	cperez@mailcatch.com	Clara	\$2y\$10\$CjJyTGXx8zK6zPpZuThm0O/aDzcL6iYcoEKbv3KOWpC...
1	cdamon	cdamon@mailcatch.com	Connie	\$2y\$10\$3.aK2WzE/8tulStRBLA9hOgKny2kJuyze2OLaG6IdEv...
2	cclark	cclark@mailcatch.com	Christopher	\$2y\$10\$h5vfCLqrEtU0m/SP0ygeEeBvzrl5yN241YI9YSgzjui...
3	chrissum2	chrissummers@mailcatch.com	Christopher	\$2y\$10\$IK5Kkvln/sX7e5J2Xo1x4.hBFUo7vblAlk6HC4.uwuL...
4	covington	pcove@mailcatch.com	Patricia	\$2y\$10\$7IG2TxLCiOptHocGoS1FPOAtdMPBI0GTDLWW8Y/LVHA...
5	mildredtor	mtorres3@mailcatch.com	Mildred	\$2y\$10\$9omrJMBX6ohQr.9BabRz8u0QT4C3wduZ8/BNJfaGzLa...
6	twig10000	twig10000@gmail.com	alex	\$2y\$10\$iqn9wsCG4zaYCYzXhWO4SO1pLtssaO.JYWzSMTaGUQd...

## -Honey Pot

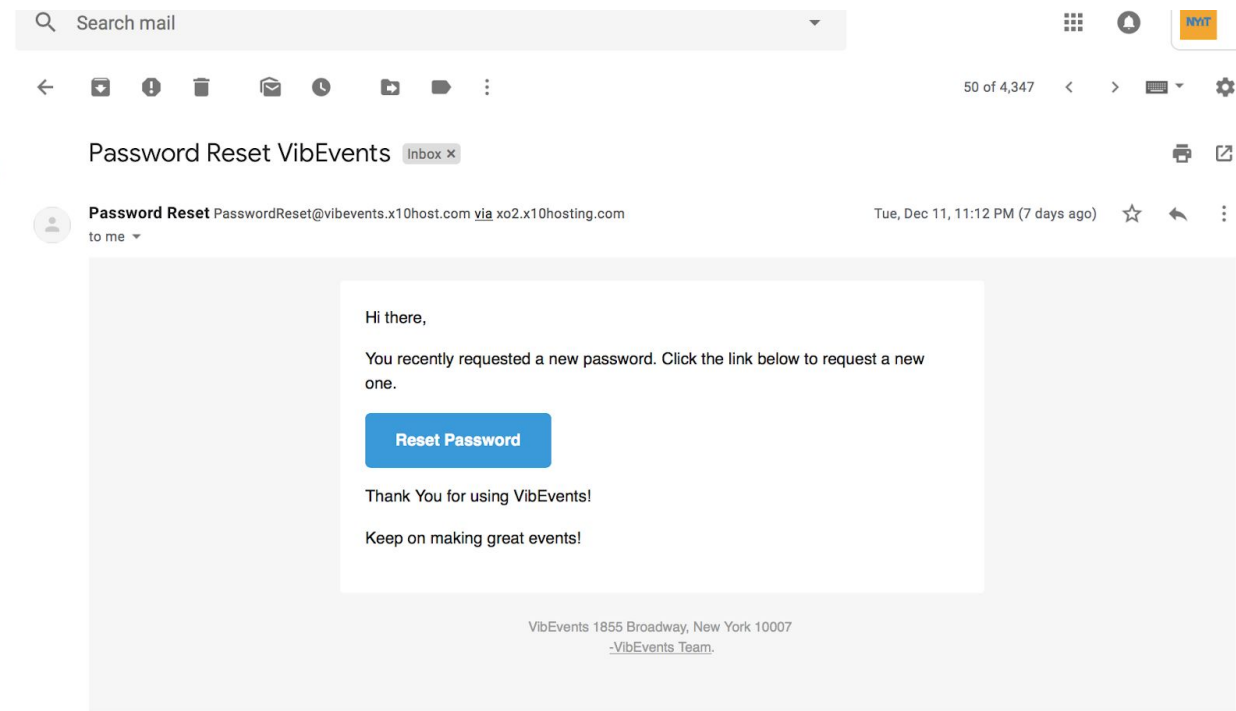
An additional level of security feature added into our Web Application:

Capture Information of users attempting to access a fake admin page. The time, IP address, web browser etc are noted down when someone tries to gain access to our admin page, in order to protect and prevent our Applications to fall under the hands to hackers.

198.91.81.4	Windows 7	Firefox	1544624714
	Windows 7	Firefox	1544624850
98.113.59.130	Windows 7	Firefox	1544624885
172.56.34.58	Android	Handheld Browser	1544625504
172.58.225.218	iPhone	Handheld Browser	1544625600
172.58.225.218	iPhone	Handheld Browser	1544625617
172.58.225.218	iPhone	Handheld Browser	1544625634
172.58.225.218	iPhone	Handheld Browser	1544625664
172.56.34.58	Android	Handheld Browser	1544626979
172.56.34.58	Android	Handheld Browser	1544627517
216.37.98.38, 64.35.176.178	Chrome		1544630119
216.37.98.38, 64.35.176.178	Chrome		1544630119
64.9.249.67	Chrome	http://vibeevents.x10host.com/Home.php	1544630145
64.9.249.67	Chrome	http://vibeevents.x10host.com/VendorLogin.php	1544630192
216.37.98.38, 64.35.176.178	Chrome		1544630209

- **PASSWORD RESET AND WELCOME**

Whenever someone register for the VibEvents for the first time, they get a welcome email from in the email they used to register. This is done to give the user a more personalised feel while using the app.



When users forget their password, they can click on forgot password link, and email will be sent to their registered email address with a link to reset their password.

Our REST API worked by processing a request with data which worked on both the frontend and backend and then with that data we would create an appropriate class to take that data and then save it to the database.

### 3.3 TIMELINE:

#### **September 12-20th :**

Concept- Project topic is decided.

It is envisioned and prioritized.

#### **September 20-23 :**

Inception - Roles of Team members are identified, initial environments and requirements are gathered

#### **September 23- 25 :**

Designing overview : Main entities, relations and ER diagram is worked upon, and the project is designed.

#### **September 25 to October 15 :**

Learning of new technologies, like React Native, Xcode by team members. More research and learning of the new technologies, watching videos, discussions and setting it up with all the group members.

#### **October 10 to December 1:**

Construction of the working model:

Making the front end for the mobile application

Making the front end for the Web App

Creating backend services

Integrating Back end with Front end

#### **December 1 to december 11 :**

Integration

Adding more functionality and security

Testing and simultaneously making changes along with working on presentation

#### **4.0 LIMITATIONS:**

Our limitations were primarily the time constraints. Learning new languages in React Native , while implementing it to deliver a minimum viable product proved to be more challenging than we had initially anticipated, but we got the job done and were able to deliver a functioning product; complete with database structure and front end design work.

#### **5.0 DIFFICULTIES FACED:**

- Steep learning curve with React Native, which took a long time to set up and get working on.
- Figuring how navigation worked in React Native, since it was rather complex.
- Working with React Native's asynchronous nature.
- Implementing user authentication in React Native & PHP.
- Integrating the back end services with the front end.
- Creating code that runs on both the web application and mobile application.
- Creating a system that allows users to reset their passwords and emails them a secure link.
- Creating a mobile responsive web application.

## 6.0 FUTURE EXPECTATIONS:

In the future, given more time and resources, we would aim to do the following things:

1. Set up more details about our vendors, like pictures of the events they serviced, pictures of their services, detailed description of the services they offer.  
Basically, add links to the vendor's websites and more pages into our application to help the users creating the event pick the perfect vendors for themselves.
2. UI for main admin page: Although there is a admin control of what is going on in the application in the backend side from our database and php servers, in the near future we can add an UI interphase to see what is going on and log in as a main Admin to control and see all the events and vendors.
3. Have for categories for our vendors, and expand it to areas further than NYC.
4. Add more filters for the users to select their vendors.
5. Add a Google Maps API to get user statistics and to suggest vendors automatically to users based on their location.



## 7.0 CONTRIBUTION AND LEARNING OUTCOMES

TEAM MEMBER	CONTRIBUTIONS	LEARNING OUTCOME
Mansi Parikh (Team Leader)	Team Leader, Front end development for mobile application using react native, assisted in integration of back end/front end, testing. Documentation.	For me, the biggest challenge was learning React Native as it was a completely new technology for me. It took a long time to set up and get started on. But later, once we figured how it works, it became easier. As a Team leader I learned how to work well with a group of people with such diverse ideas and thoughts. Getting everything and everyone together gave me a big hint of what the real world outside is when we graduate. Never have I worked in such a big project, and never managed it too. It was a wonderful learning experience, that taught me how to work under pressure and yet not compromise the value of work. I shall remember this experience as a highlight of my Undergraduate degree experience
Alexander Khemai	Front-end/User Interface for web application, assisted on back-end design, QA and testing,	After working on this project, I understood and experienced how a full stack application works extensively. It also prepared me further for working professionally, as if I were a software

	research	<p>engineer for a company, making an application for a client. Also, I learned how to work with a team and dealing with different aspects of the application. I thought it was very intriguing to sit down with my team and start brainstorming from scratch. On top of that, this project also coached me on how to satisfy deadlines better and aiding my teammates in satisfying those deadlines as well. This project also helped me strengthen my conceptual programming skills, as I had to learn PHP because I did not have any experience with it prior.</p>
Amitoj Johar	Front-end on mobile, front-end/back-end integration, project design, research	<p>The learning outcome for me was that I learned how to strengthen my learning skills when it comes to programming, especially since I had to learn React Native and React Native is tough to begin with. Also, I learned how a mobile application can work with a back-end server to make it more fully functional. This project also strengthened my communication skills within a team, especially since this is very important when working in the real world.</p>
Danny Rivas	Front-end, testing on different devices, Design for the app	<p>The learning outcome for me was being exposed to new technologies. For example I learned React Native, a new language</p>

	using PhotoShop	that I was excited to learn and it taught me new ways to develop on mobile. Even though React Native had a steep learning curve, I felt like in the end I learned a good amount of the language and especially how to troubleshoot certain error efficiently. I also learned new design techniques with PhotoShop and helped my interest grow in app design.
Alexander Gronowski	Created the backend for the web and mobile application, Frontend for the web application. Created session management for the web application and mobile application. Database creation and implementation. Created the REST API for our applications. Security for our applications. Backend implementation of sending welcome emails and resetting	I learned a lot about development and all the work that goes into producing an application. I never used PHP to send emails before, so it was extremely challenging to get our server setup to send emails, and then also implementing the ability for users to reset their passwords was tough due to all the security considerations that had to go into it. I also learned a lot about security doing this, I never really covered any security topics in any courses, so I wanted to cover some of the OWASP Top 10 Most Critical Web Application Security Risks. It was very challenging to read about topics like SQL injections, Cross site scripting, user authentication, and many other topics, but also then finding how to implement the security functions in PHP and testing to

	passwords.	<p>see if they worked was a great experience. I also never worked with React Native before so writing code that would work on both our web application and mobile application was difficult. I originally wrote code that was only for the web application, but because some of the code was outputting HTML and CSS it would not work with React Native, so I had to go back and rewrite the code so it worked on both our mobile and web applications. I also learned a lot about how React Native works in terms of sending requests to our server, parsing the response from the server, how to implement and handle sessions in React Native, and how to deal with React Native's asynchronous nature by using promises and callback functions.</p> <p>I also learned how to work with a team to produce a finished application. I think one of the most valuable things I learned was how to do pair programming effectively and how to communicate during that process to make sure everything worked and was done effectively.</p>
Overall Learning Outcome		Our overall learning outcome of the project was we learnt a lot about developing skills,

		<p>how to gather ideas, data, and work on something that none of us have done before. How to step by step delivery and maintain the deadlines. How to prepare for the presentation. Working in a team was really good thing we all learnt some new technologies, new languages, and added to each other's ideas which was really a plus point.</p>
--	--	--