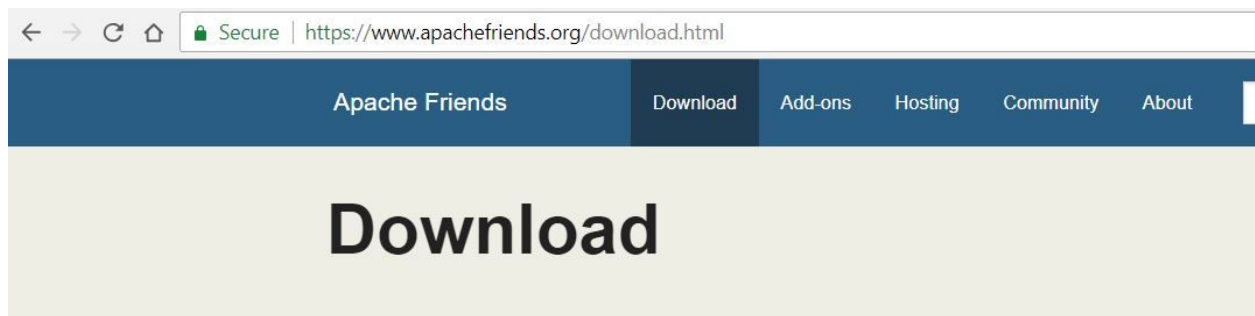Welcome to the World fitness membership management application. Based on XAMPP software(Serves Apache server to host, MySQL for the database and *php*).

Steps to start the app:

1. Download the XAMPP software from (https://www.apachefriends.org/download.html) for your operating system.
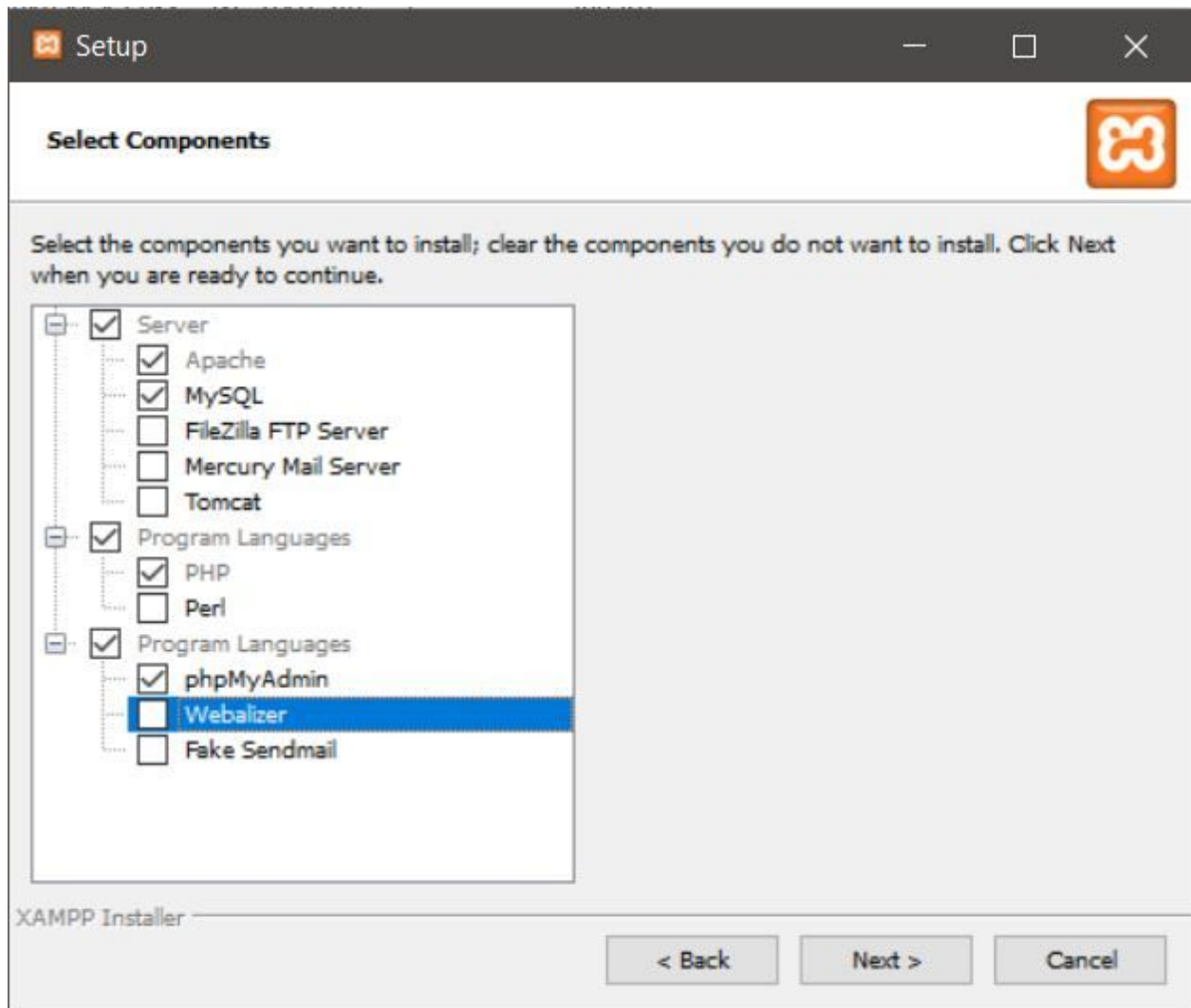
2. Install the XAMPP software with the options below:

3. Wait for the software to install and allow any firewall blocks.
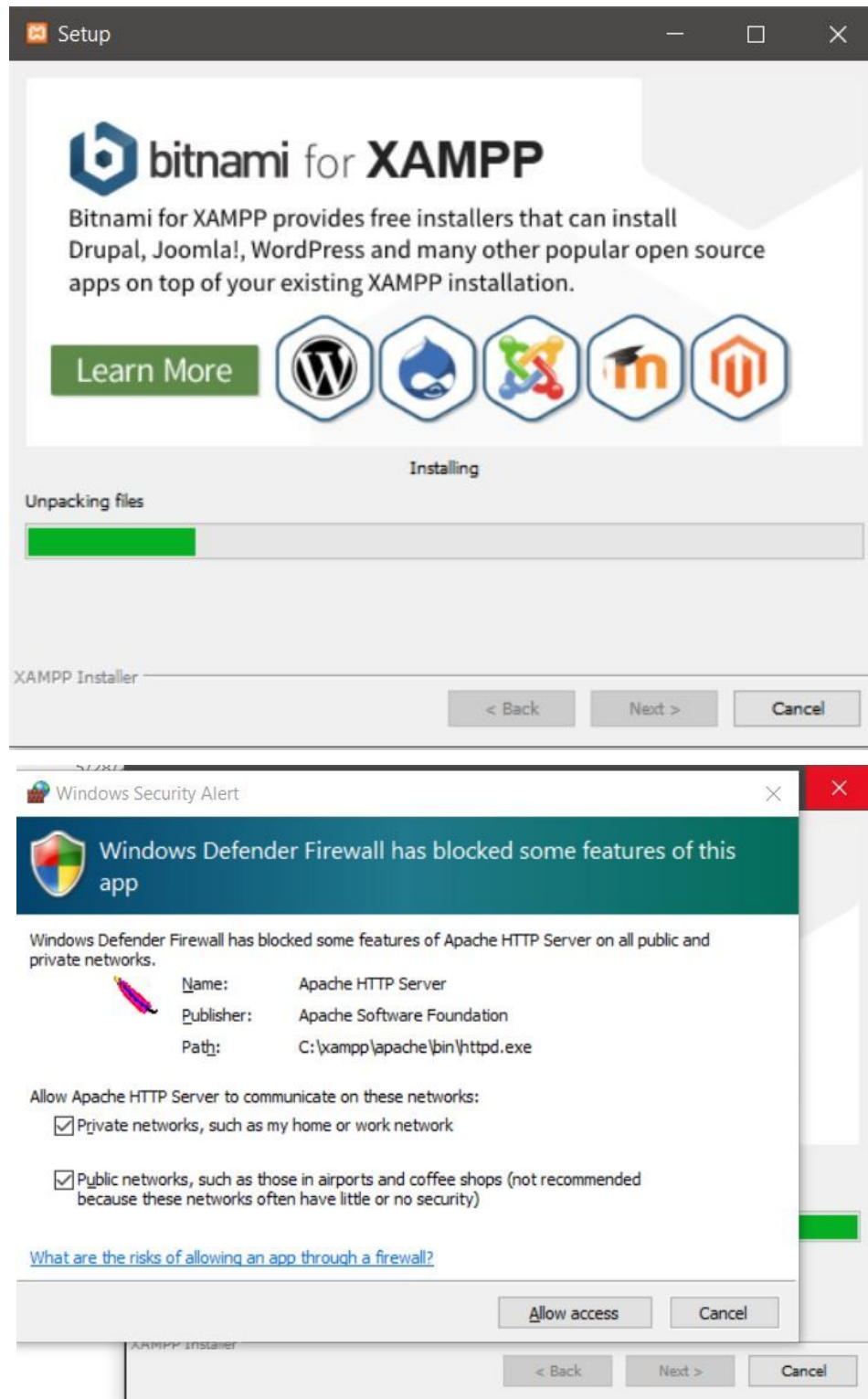
4. After installing the XAMPP, open the XAMPP Control Panel.
   Click on "Start" for Apache and MySQL.
5. Extract the "membership_management" folder from the zip file and copy to the "C:/xampp/htdocs" directory.
6. Open your browser, preferably Google Chrome.
7. Navigate to (localhost:80/phpmyadmin), and add a new database using the "New" button in the left hand corner.
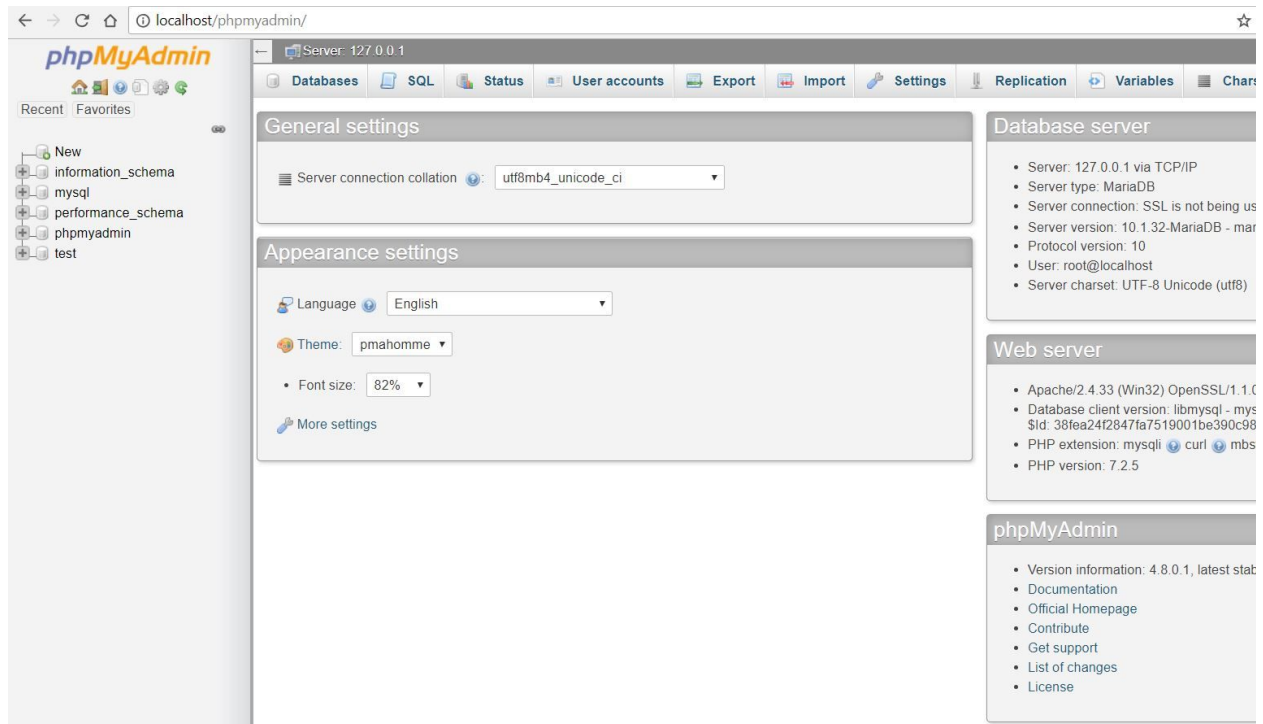
8. Create the database called "membership_management" and click on "Create" button.

9. Navigate to the "membership_management" database and click on it.
10. Click the import tab, when you are in the "membership_management" database.

11. Click on "Choose file" button and select the only SQL file "membership_management.sql". This should import two tables called "users" and "members"
12. Once the database is imported, open a new tab in the browser and navigate to "**http://localhost:80/membership_management/index.php**", you should see the landing page for the app.
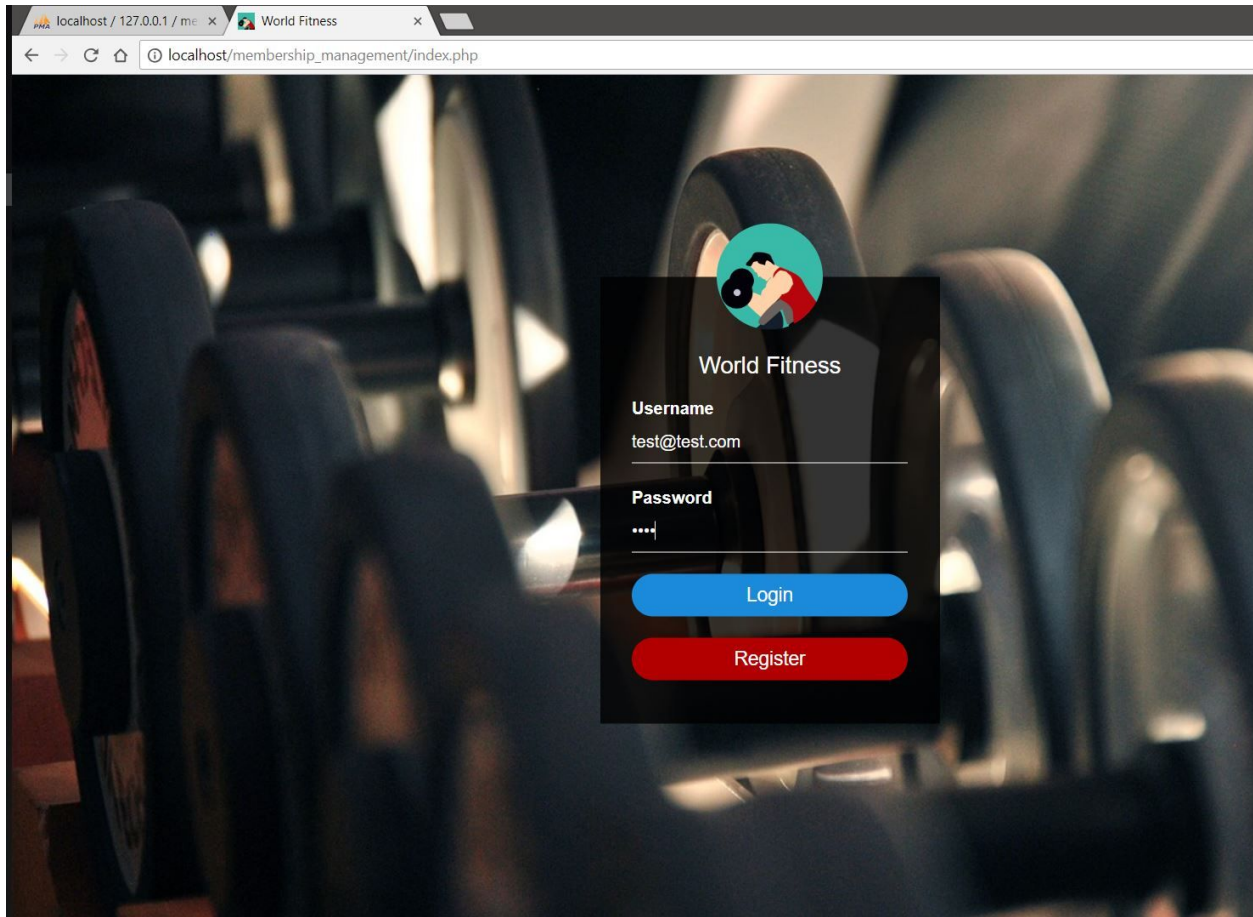
13. Login using credentials as following:
Username: *test@test.com*
Password: ***test***

14. You can also register with a new username and password.
Note: Remember the password as the password is hashed while storing it in the database.

15. Once you login you should be able to see the Dashboard page which has a pie chart (using Google charts library) and a quick table view of the members in the gym and their application status.
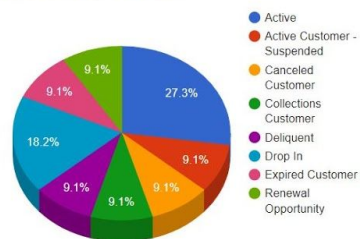
Name	Origin	Contact number	Application Status
John Doe	Maryland	1234567890	Deliquent
Joseph M. McCloud	Illinois	6304196171	Active
George E. Quincy	California	4567328198	Active Customer - Suspended
Charlie Oppen	Pennsylvania	4567898423	Drop In

16. Navigate to Members page by clicking on "Members" in the navigation bar.

World Fitness Dashboard **Members** Logout

Add Member

MEMBERS

Name	Origin	Created Date	Contact number	Calender Events	Recurring Service	Balance	Sessions Remaining	Agreement Status	Alert	Application Status	Action
John Doe	Maryland	2017-12-31	1234567890	2019-12-31	Yes	80	5	Good	On Going	Deliquent	Edit Delete
Joseph M. McCloud	Illinois	2018-05-01	6304196171	2018-05-16	No	0	10	Good	On Going	Active	Edit Delete
George E. Quincy	California	2018-02-14	4567328198	2018-05-15	Yes	80	0	Suspended	On Going	Active Customer - Suspended	Edit Delete
Charlie Oppen	Pennsylvania	2017-05-24	4567898423	2018-03-15	No	90	0	Good	On Going	Drop In	Edit Delete
Karlo Poljak	Wyoming	2016-06-07	0987678765	2018-01-17	No	80	3	Good	On Going	Renewal Opportunity	Edit Delete
Thomas Marcoux	New York	2014-08-15	8765434567	2015-06-10	Yes	0	5	Inactive	Canceled	Canceled Customer	Edit Delete
Rodrigo	Portland	2008-	2345678987	2011-03-	Yes	0	0	Inactive	Expired	Expired	Edit

17. You can add members by clicking on the "Add Member" button.

World Fitness Dashboard **Members** Logout

ADD NEW MEMBER

Name *

Enter Name

Origin *

Enter Origin

Created Date *

mm/dd/yyyy

Phone number *

999-999-9999

Calender Events *

mm/dd/yyyy

Recurring Service Option *

Yes ▾

Ammount Paid in $ *

Enter Paid ammount

Purchase Sessions *

Enter Sessions purchase quantity

Remaining Sessions *

18. You can even update the member or delete them by pressing the "**Edit**" or "**Delete**" link in the Action column.
19. To logout press the "**Logout**" button in the navigation bar.

Notes/Features:
1. The app provides complete security to the users password as the password is hashed and then stored in the database.
2. Once user is *logged in*, they cannot go back to the login page until they are logged out.
3. The website is responsive and can be viewed in mobile once its hosted on some *url*.
4. The user cannot pay an amount higher than the balance left for the user.
5. All the 8 categories of the members work as per their "***application status***" and "***alert***".

Scaling Consideration

How does your approach handle millions of requests per hour?

● I have not tested for a million requests per hour.
● Hopefully, ***phpmyadmin*** can handle the stress.
● If not I would like to consider using middlewares and web server which can support million requests.

How would you monitor the health of this service?

● Using ***phpMyAdmin*** the health of the service can be monitored.
● Go to (**http://localhost:80/phpmyadmin**) and click on Status tab.

How are you going to handle many users changing data?

● Different users are placed in different sessions.
● Two users will have their own user_id and cannot see the other users clients/members.