Dependency/Module installations:

1. Install Jupyter Notebook and create a folder
2. Install Pysyft, pandas, numpy and sklearn libraries/modules using pip as below:

   pip install syft
   pip install pandas
   pip install numpy
   pip install sklearn

When running the Jupyter Notebook:

1. Restart kernel before running
2. Run one cell at a time and not all together as there are some initialization/configuration cells that have to be run only once, skip the next times you run the code again
3. Cell 5: Creating an admin account for each local site using default credentials
   a. Run this cell just once initially

```
#Login to these servers with Bob and Alice as admins with default credentials
#Run this once initially, do not do so after client/admin details have been changed from default ---- RUN AGAIN AFTER RESTARTING NOTEBOOK
Bob_Client = sy.login(url=server1.url,port=server1.port, email="info@openmined.org", password="changethis")
Alice_Client = sy.login(url=server2.url,port=server2.port, email="info@openmined.org", password="changethis")
```

4. Cell 7: Change the account credentials to make the site more secure
   a. Run this cell too just once, skip the next time you run the code again

```
#Change the default admin credentials for both user devices/users --- RUN THIS TOO JUST ONCE, DO NOT UNLESS RESTART JUPYTER NOTEBOOK
BOB_EMAIL = "bob-user1@eps.s3d"
BOB_PASSWD = "bob_doesnt_share_data"
Bob_Client.account.set_email(BOB_EMAIL)

# we can bypass the confirmation by using the confirm=False parameter
Bob_Client.account.set_password(BOB_PASSWD, confirm=False)

ALICE_EMAIL = "alice-user2@eps.s3d"
ALICE_PASSWD = "alice_doesnt_share_data"

Alice_Client.account.set_email(ALICE_EMAIL)

# we can bypass the confirmation by using the confirm=False parameter
Alice_Client.account.set_password(ALICE_PASSWD, confirm=False)
```

5. Cells 17 and 19: Change the path that accesses the user dataset csv files as per your system before running

```
#Access dataset variables and set them into target and features -- BOB Dataset
df1 = pd.read_csv(r"C:\Users\shiva\Documents\CMU\CMU Year 1\Semester 2\EPS\user1_fitness_data.csv")
features1 = ['Max_BPM', 'Avg_BPM', 'Resting_BPM', 'Workout_Frequency (days/week)', 'Experience_Level', 'Session_Duration (hours)']
target1 = 'Calories_Burned'

X1 = df1[features1].values
y1 = df1[target1].values.reshape(-1, 1)
```

```
#Access dataset variables and set them into target and features -- ALICE Dataset
df2 = pd.read_csv(r"C:\Users\shiva\Documents\CMU\CMU Year 1\Semester 2\EPS\user2_fitness_data.csv")
features2 = ['Max_BPM', 'Avg_BPM', 'Resting_BPM', 'Workout_Frequency (days/week)', 'Experience_Level', 'Session_Duration (hours)']
target2 = 'Calories_Burned'

X2 = df2[features2].values
y2 = df2[target2].values.reshape(-1, 1)
```

6. Cell 99 - Creating projects
    a. Run this cell once for initializing the projects and only when you run the code from scratch.
    b. Other times, access the project using the below code snippets:
    fitcheck_project1 = global_client1.projects[0]
    fitcheck_project2 = global_client2.projects[0]

7. Cell 111: Creating code requests
    a. Run this cell only once, unless you have restarted a Jupyter notebook and run the code from scratch. Otherwise skip.
    b. Running this multiple times creates multiple code requests which hinders sharing the results with the global model due to approval status:Pending.

```
[111]: global_client1.code #run only once - creates code requests for the admin accounts from global model's end
       global_client2.code
```

8. Everything else has to be run.
9. If any errors occur especially when running cells with client initialization or database uploads, shut down and restart Jupyter Notebook and run the code from scratch.