Social Lite

Archit Sood, Bingqing Huang, Gauravkumar Wadhwani, Weidong Li

# Core web services

1. **Login**

@app.route('/login')

def login()

# The method for checking user authentication

# components: templates/login.html

# input : user id , password

# output : boolean(success/fail)

# addition : holds user session

1. **meetup\_recommendation**

@app.route('/meetup\_recommendation')

def get\_recommendations():

# Some methods to get recommendations from server, maybe with some parameters

# components: templates/meetup\_recommendation.html

# input : user id

# output : recommendation for this user (JSON)

1. **user\_profile**

@app.route('/user\_profile')

def get\_user\_profile():

# some methods to get profiles from the server

# components: templates/user\_profile.html

# input : user id

# output : JSON

def update\_user\_profile():

# Some methods to update server records

# components: templates/user\_profile.html

# input : user id, profile JSON

# output : boolean(success or fail)

# List of packages

**1. Flask**

Beginner's python web framework

Unless more than half of the team members agree to use Django, we will keep using Flask.

1. Flask-Login 0.2.11

“Flask-Login provides user session management for Flask. It handles the common tasks of logging in, logging out, and remembering your users’ sessions over extended periods of time.”

As our project requires users to logging in and logging out of their accounts and this package also provides basic security. This package will be used for user session management

1. Flask-Mail 0.9.1

“The **Flask-Mail** extension provides a simple interface to set up SMTP with the [Flask](http://flask.pocoo.org/) application and helps to send messages from views and scripts.”

As we will be sending invites to users in our project. If a meeting is set up we can send mails to all the involved users.

1. Flask-User 0.5.4

“Customizable User Account Management for Flask: Register, Confirm email, Login, Change username, Change password, Forgot password and more.”

The initial part of the project would be adding/registering users. This package will help in User Account Management.

**2. SQLAlchemy**

SQLAlchemy is the Python SQL toolkit and Object Relational Mapper that gives application developers the full power and flexibility of SQL.

We will use database. So we use the best~!

**3. Requests**

Requests is an elegant and simple HTTP library for Python, built for human beings.

Looks like a must for every python developer. So why don't we use it? It's for the utilities with web stuffs.

**4. Flask-SocketIO**

Socket.IO integration for Flask applications

We may try to add live-chat function to the app. So we may need the socket integration for Flask.

**5. nltk**

Natural Language Toolkit

We may want to do something with NLP if we have time. This will surely improve the recommendation system if properly implemented.