Meeting 1

Monday, 01.11.2016

Attendees

Mathieu McMurray
Jorell Linsangan
Steven Bairos
Evan Hunzinger
Keane Kraus
Caleb Mueller

Agenda

Last Meeting Follow-up

1. Decide on tech stack
2. Pick a project idea to run with

New Business

3. Come up with some broad user stories

Notes

- Team unanimously agreed to work on a study buddy “dating” application, suggested at last week’s meeting
- Technology stack:
  - Node JS
    - Most team members were more comfortable with JS than Python
    - Node JS has a vast library of modules to ease Rapid Application Development (RAD)
  - SQLite
  - GitHub
  - Trello & Slack for coordination
- Some names for the team and project were suggested (mostly just for fun):

  - Team name: DWAB (Don’t Worry About it)
  - Project name: Flipchat (?)
- Team discussed format of merge requests and branching structure in Git
  - After a team-member finishes a task, they’re to submit a pull request and have another team member take a look before merging. This will encourage proper coding practices and should minimize bugs.
  - During each iteration, we will have a development branch dedicated to it where changes will be merged. After we’ve verified that an iteration is stable, it will be merged into the master branch. This way, we should always have a working master branch and multiple iteration branches that may or may not be working at the time.

Next Meeting Agenda Items

- Work on project vision
- Elaborate on user stories and possibly start defining some tasks for iteration 0

Meeting 2

Wednesday, 01.13.2016

Attendees

Mathieu McMurray
Jorell Linsangan
Steven Bairos
Evan Hunzinger
Keane Kraus
Caleb Mueller

Agenda

Last Meeting Follow-up

1. More user stories
2. Project vision

New Business

1. Consider using meteor framework over Node.js
2. Dependency management
3. Workflow

Notes

- Jorell brought up the Meteor framework and showed off a simple demo application. Ultimately, the team decided to stick to its original plan to use the Node.js framework, since nobody was very familiar with Meteor and extra modules have to be installed to create a RESTful API
- We briefly discussed design patterns, but decided to leave that until the next iteration, when we actually start writing code.
- The team generated nearly a dozen new user stories:
    - "I want to know that potential matches have passed some sort of test (IQ test?)"

    - “I want to create/fill out a profile”
    - “I want to be able to view other people’s profiles (in lesser detail/pre-match)”
    - “I want to see a list of all my matches.”
    - “I want to see a list of people that I ‘liked’ but ‘disliked’ me.”
    - “I want to upload a profile picture.”
    - “I want to change my settings (themes, notifications, etc.)”
    - “I want to be able to delete my account.”
    - “I want to be able to form a study group consisting of my matches and their matches.”
    - “I want my transcript to be auto-parsed.”
- The team answered some questions relating to the project vision:
    - Who is it for?
        - Post-secondary students
        - Age range: 18 - 30 years old
        - People who don’t know many other students
    - Why is it valuable?
        - Finding people with the same desire to learn
        - Potentially aids in boosting user’s GPA
        - Helps create connections between users
    - Success criteria?
        - Be able to match students
        - Look nice/appealing to the eye
        - Have the service be intuitive

Next Meeting Agenda Items

- Discuss which SQLite package the team would like to use
- Review work done and work yet-to-be-done due for iteration 0
- Design patterns (?)
- Detail user stories