**CS673 Software Engineering**

**Team 6: Panda**

**Meeting Minutes**

**Meeting #11**

**Date and Time:** 11/10/2022 2pm

**Place**: Zoom

**Participants:** Siming, Dawei, Shawn, Alex，Lijian Yao(James),haiyang

**Minutes taker:** Shawn

**Purpose:** weekly meeting

**Meeting record:**

[**https://bostonu.zoom.us/rec/share/yxI5tgVBf71BQBJM18VZmREXpgWzHtJpoGW1BpgVCe76OgFghTwzbYxinK41p8Hv.ofd58T98YPSJVGcI**](https://bostonu.zoom.us/rec/share/yxI5tgVBf71BQBJM18VZmREXpgWzHtJpoGW1BpgVCe76OgFghTwzbYxinK41p8Hv.ofd58T98YPSJVGcI)

**Passcode: G?3!C9zu**

**Agenda:**

1. **Progress report**
2. **Pivotal Tracker**
3. **Assign Tasks**
4. **Algorithm discuss**

**Action Items: Progress report**

* Dawei:

1. Update the feature for student to answer the survey
2. Link the web application backend to our algorithm
3. Update the feature to run group distributor

* James:

1. Connect GUI to Django template

* Shawn:

1. Debug algorithm code with haiyang
2. Delete question and options
3. Lock survey feature

* Haiyang:

1. Debug algorithm code with Shawn
2. Wrote test case for some Django function
3. STD

* Alex:

1. Import student data feature. (cvs, json, etc)

* Siminig:

1. **For everyone in the team.** Keep in mind the existence of PivotalTracker and check your own progress box after it is done. If you don't see the one you have done or feel anything wrong about the job you have there such as description or detail stuff; let me know I will fix those.
2. Need to go through Django models and related stuff to implement model.py (DB) to store team distribution results.

Reference:

**Web page direction map:**

https://www.figma.com/file/sBjinmzpBsksR5cehXRMxL/IGroup?node-id=0%3A1&t=ZwMzx7iGNPnypVQy-0

**CS673 Software Engineering**

**Team 6: Panda**

**Meeting Minutes**

**Meeting #10**

**Date and Time:** 11/04/2022 5pm

**Place**: Zoom

**Participants: shawn,alex, haiyang,dawei,james**

**Purpose:** weekly meeting

**Meeting record:**

**https://bostonu.zoom.us/rec/share/oEgK4Py0KARqlCl9dpOrL3BDJl3ll\_vfTt1Ajgt0q04fLYO4tD725A6U1Wdh-URB.w0H\_M3Nz7Qlvu85b (密码: +@tn2d\*g)**

**Agenda:**

* Shawn
  + Goal of next:

1. Create delete feature
2. Create conform feature

* James
  + All the front end pages
  + Quick demo

Next:

Frontend changes

Help Connect backend

* Haiyang
  + Implementation :

Tests of questions, surveys, answers, answer-sheets

* + Goal of next meeting/iteration:

instance, score calculation, team tests

* Dawei
  + Web implementation:
    - Survey creation/update; Option adding/update
    - Instance creation
    - Services integration
* Alex
  + View:

Survey branch

* + Iteration 2
* All:
  + Each team member should create their own branch

**Discussion:**

* Click Conform
  + Lock /unlock

**Key Decisions**

* Click Conform
  + Lock
* Connect app to django web
* Create own branch

**Action Items:**



**CS673 Software Engineering**

**Team 6: Panda**

**Meeting Minutes**

**Meeting #9**

**Date and Time:** Oct 28 14:00

**Place**: mugur library

**Participants:** Shawn, dawei, siming, Alex Wang,haiyang

**Purpose:** Project Weekly Meeting

**Agenda:**

1. **Configuration consistency**
2. **Assign tasks for individual**
3. **Coding standards**
4. **Continue testing**

**Discussion:**

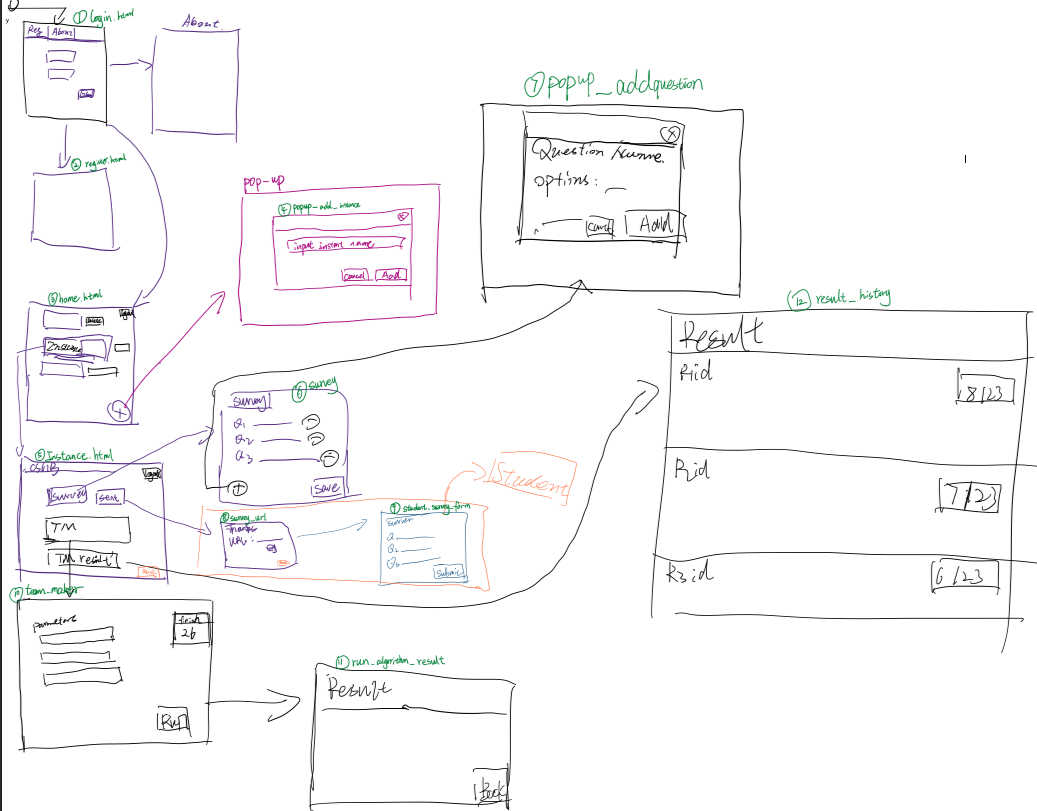
1. **Configuration:**

1. **Tasks:**
   1. **Django:**
      1. **URL schema**
      2. **Backend models**
      3. **View functionality**

**Decisions:**

**URl:**

1. **initial page:**
   1. **Redirect to login page @1(GET): .../login**
2. **Register page:**
   1. **Register page @2(GET): …/reg**
   2. **Register an account at register page (POST): …/reg**
3. **login page:**
   1. **Login page @1(GET): …/login**
   2. **Login an account at login page (POST): …/login**
4. **Home page(after login):**
   1. **List of instance @1(GET)@3: …/home**
   2. **To add instance @3(GET)@4: …/instance/add**
   3. **Add instance @4(POST)@3 …/instance/add.**
   4. **Get instance @3(GET)@5 …/instance/<instance id>**
5. **Instance page:**
   1. **To create survey page @5(GET)@6: …/instance/<instance id>/survey/create (\*\*require front end conditional statement)**
   2. **To add question page @6(GET)@7: …/instance/<instance id>/survey/create/question**
   3. **Create and add question @7(POST)@6: …/instance/<instance id>/survey/create/question/<question index>**
   4. **Save survey @6(POST)@5: …/instance/<instance id>/survey/create/save**
   5. **Sent survey @5(GET)@8: …/instance/<instance id>/survey/<survey id>/sent**
6. **Team-maker page:**
   1. **To team-maker page @5(GET)@10: …/instance/<instance id>/teammaker**
   2. **Run : @10(POST)@11: …/instance/<instance id>/teammaker/run**
   3. **Result page: …/instance/<instance id>/results/new**
   4. **Teammaker results page@5(POST)@12: …/instance/<instance id>/results**

****

**3. Coding standard:**

**url :** [**https://google.github.io/styleguide/pyguide.html**](https://google.github.io/styleguide/pyguide.html)

* **Use pylintrc to check**

**CS673 Software Engineering**

**Team 6: Panda**

**Meeting Minutes**

**Meeting #8**

**Date and Time:** Oct 22 14:00

**Place**: mugur library

**Participants:** all group members

**Purpose:** Project Weekly Meeting

**Agenda:**

1. **Coding convention**
2. **Django URL config**
3. **Django Model config**

**Dawei:**

1. **Model: survey, iGroup**

**Alex:**

1. **Model: user, group**

**Reference:**

[**https://docs.djangoproject.com/en/4.1/topics/db/models/**](https://docs.djangoproject.com/en/4.1/topics/db/models/) **(django model tutorial)**

**https://github.com/qiwsir/DjangoPracticeProject**

**CS673 Software Engineering**

**Team 6: Panda**

**Meeting Minutes**

**Meeting #7**

**Date and Time:** Oct 20 14:00

**Place**: zoom

**Participants:** Shawn,James ,Haiyang, Dawei, Alex

**Purpose:** Project Weekly Meeting

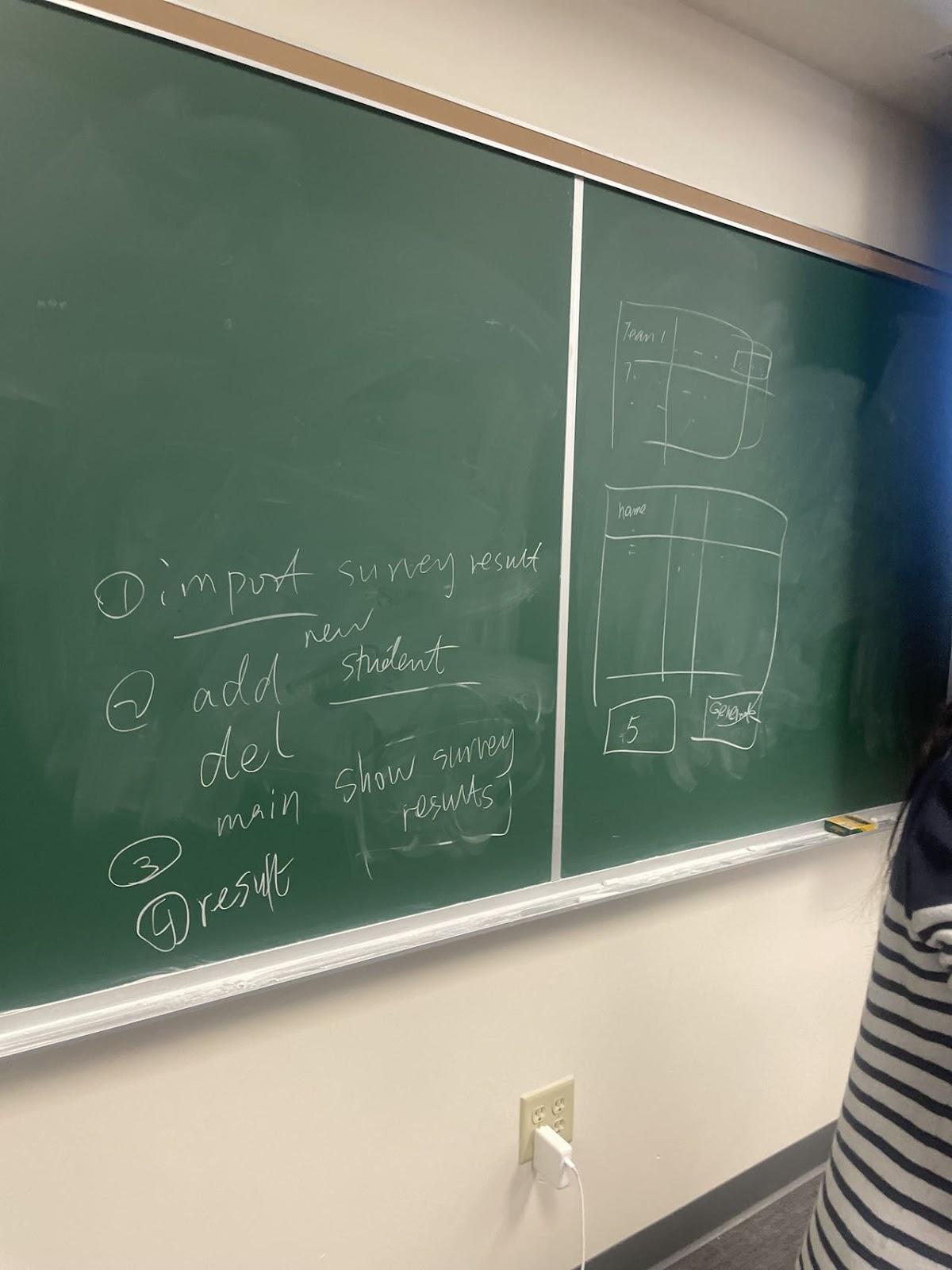
**Agenda:**

**Assign task:**

1. **QA test (haiyang)**

* “Sppp” add two more test (acceptance test, regression test)
* Make a task assignment list

1. **UI (James)**

****

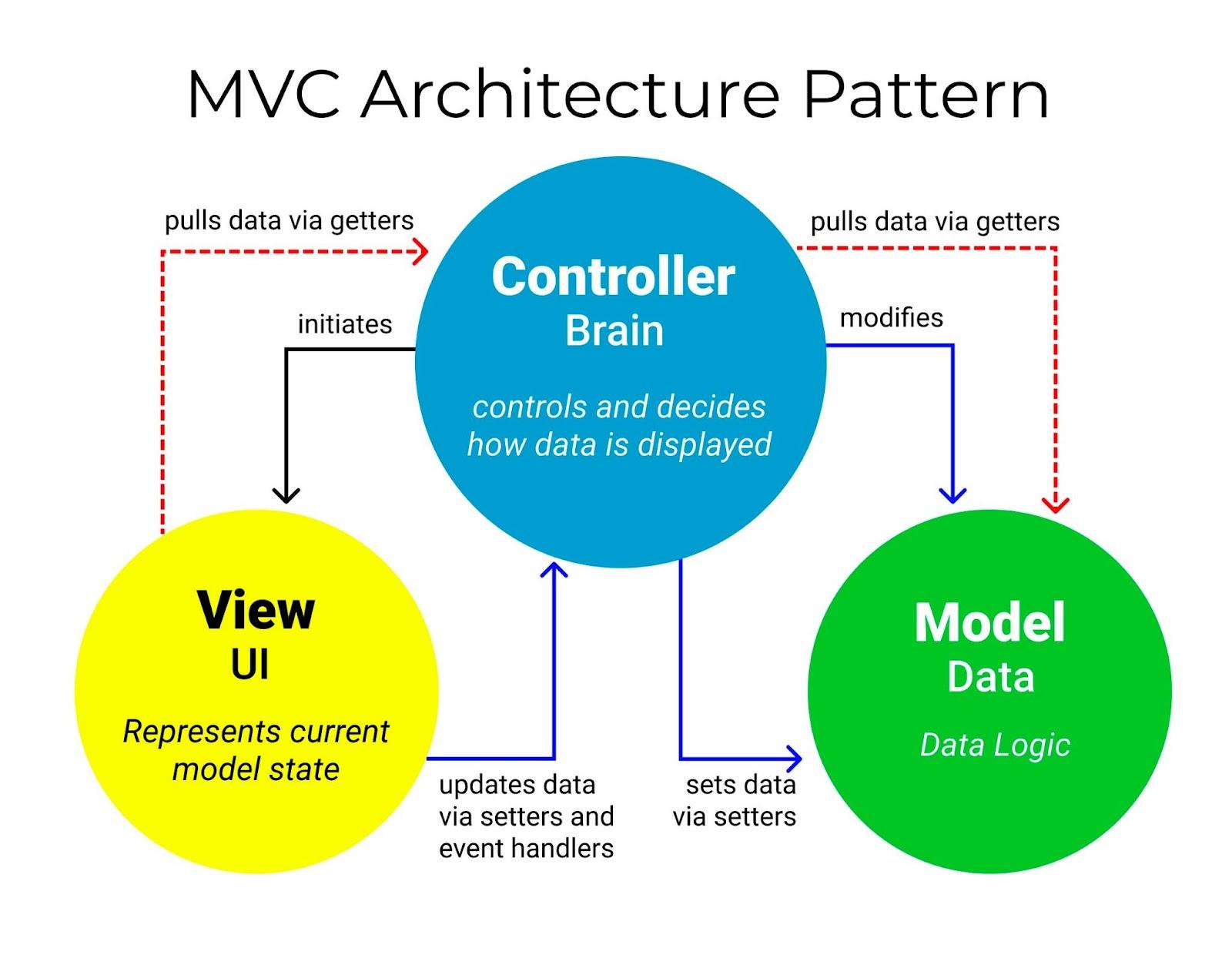
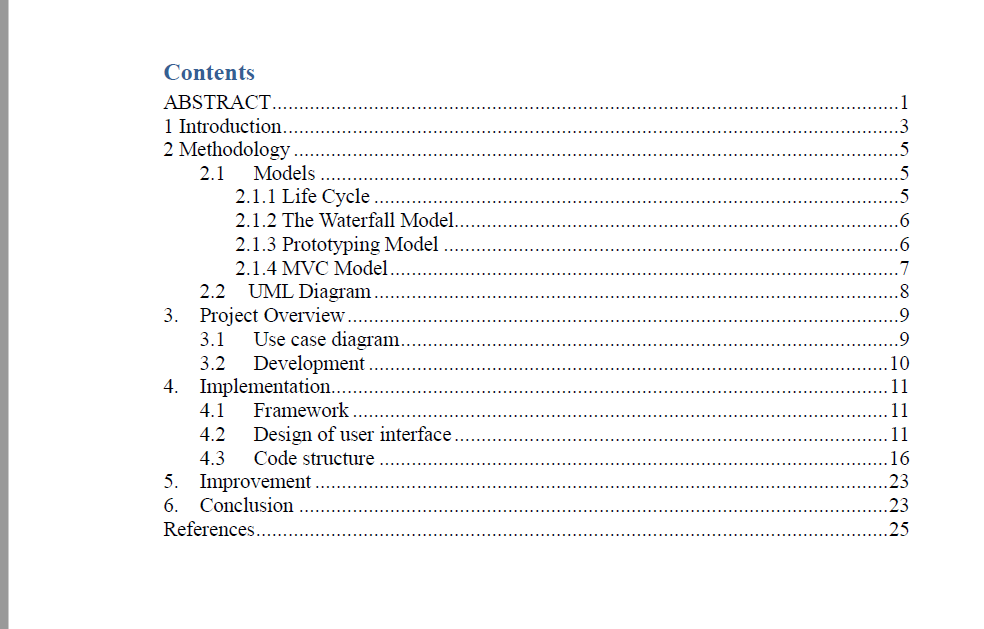
* UI require 7 pages (sddd)

1. Course selection
2. Setting survey
3. Import student survey data
4. Set group number
5. Get initial group result
6. Manually setting
7. Get final result
8. **Alex**

* SDDD

1.database design

2. Design Patterns (MVC)

****

1. **Dawei**

* sddd

# Key Algorithms （pseudo code)

1. **Shawn**

* Organize iteration 1 materials

**Next:**

1. **Group meet at Saturday 10/22 (Mugur)**
2. **Discuss about rest two algorithm**
3. **Decide database design**
4. **Learning Django**

**CS673 Software Engineering**

**Team 6: Panda**

**Meeting Minutes**

**Meeting #6**

**Date and Time:** Oct 14 14:00

**Place**: Mugur Library Basement

**Participants:** Shawn,James ,Haiyang,Siiming, Dawei, Alex

**Purpose:** Project Weekly Meeting

**Agenda:**

1. **UI (James)**

* SDDD form

1.UI Design (if applicable)

In this section, you can describe your UI design

**# UI ( student)**

* Sign in
* Login
* Take Survey
* Submit Successfully
* Rate page (用户打分页面）

**# UI ( instructor)**

* Sign in
* Login
* Set Survey
* Create a Link Successfully
* Get Result ( shows how many student complete survey)
* Manually setting groups
* Final group result

1. **QA test (Haiyang)**

* Sppp ( QA part)

1. Metric
2. Testing ( unit , Integration , system)

* Some test code

1. **Requirement (siming)**

* Sppp

Doc/ProjX\_userstories(generated)

* SDDD
* Summary additional things about iteration 1 , sent to shawn

# Security Design (Aflex)

1. System Design (Dawei)

6. Question(shawn):

which type

* SDDD

# Software Architecture

* 1. Implementation of survey system.
  2. Implementation of objective function for calculating the multiple choice answer score within a team.

1. Database design (option)

**CS673 Software Engineering**

**Team 6: Panda**

**Meeting Minutes**

**Meeting #5**

**Date and Time:** Oct 08 19:00 -21:00

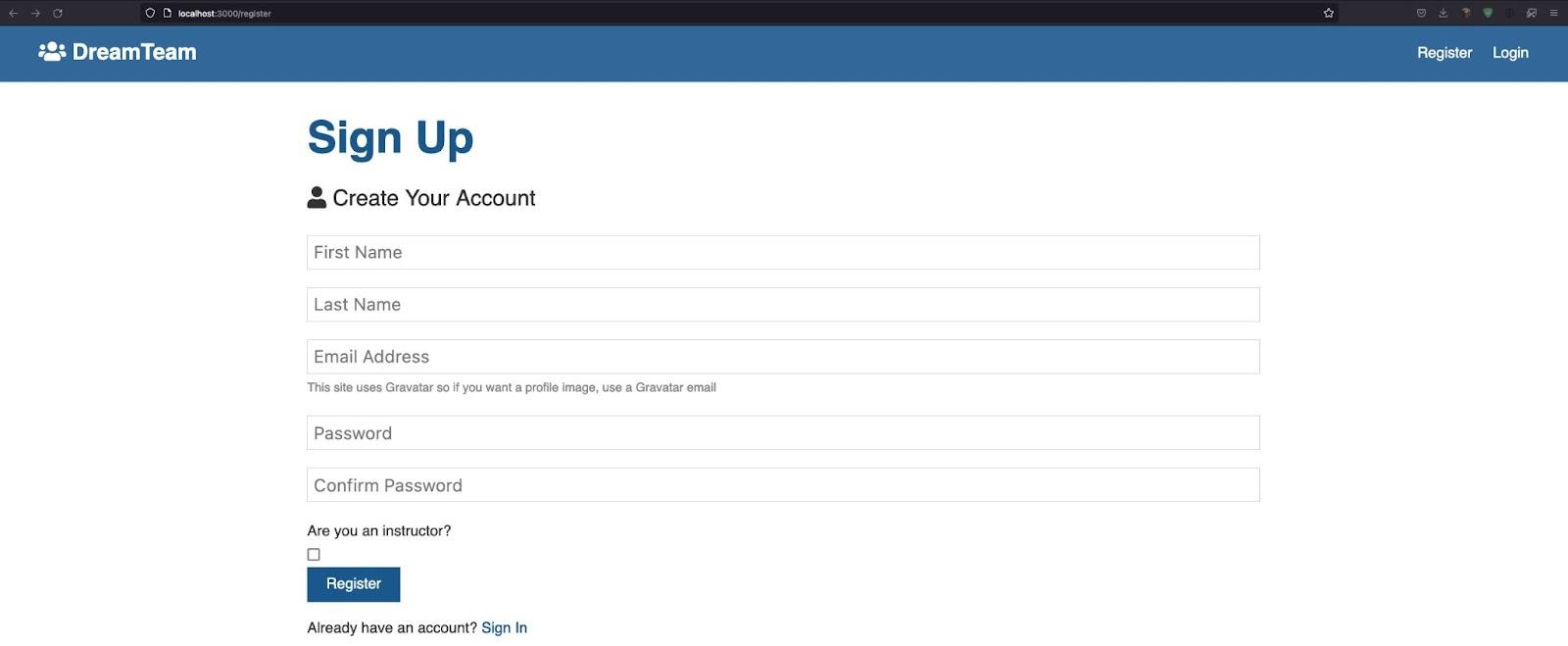
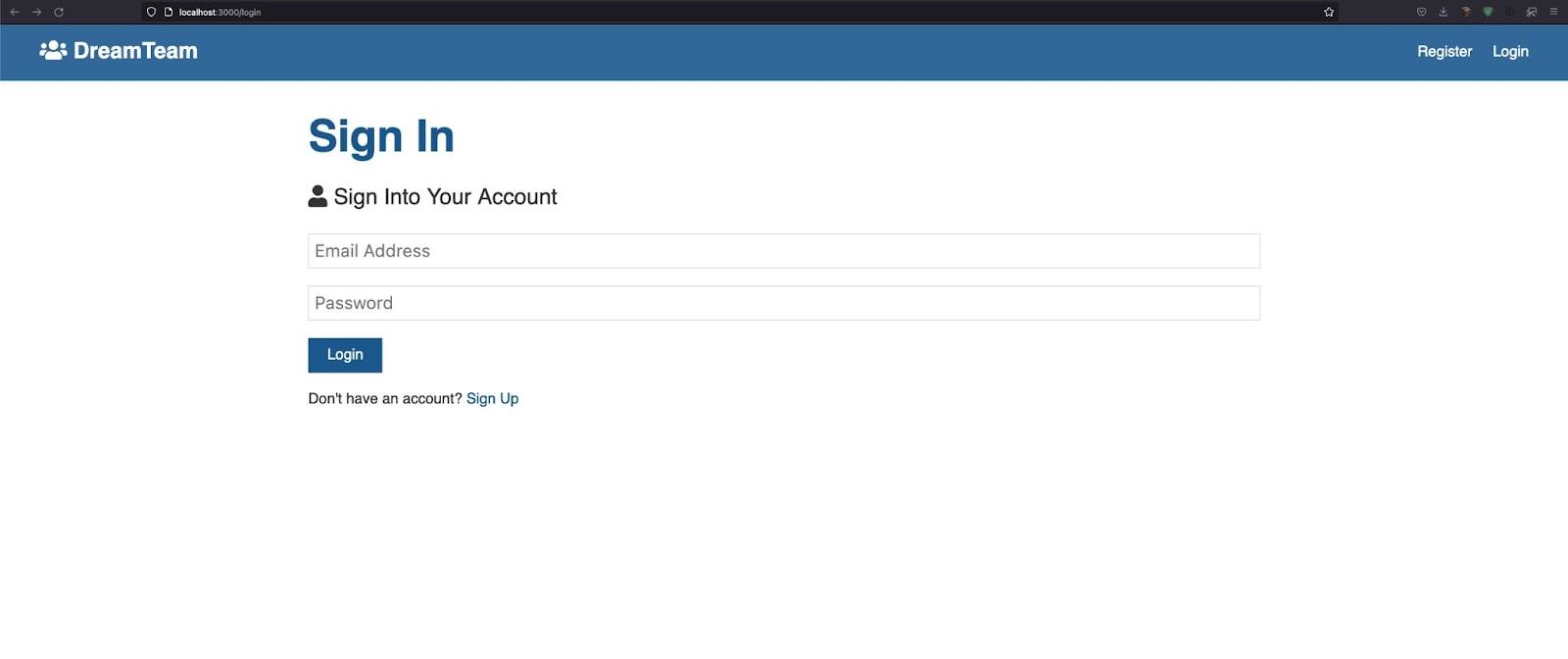
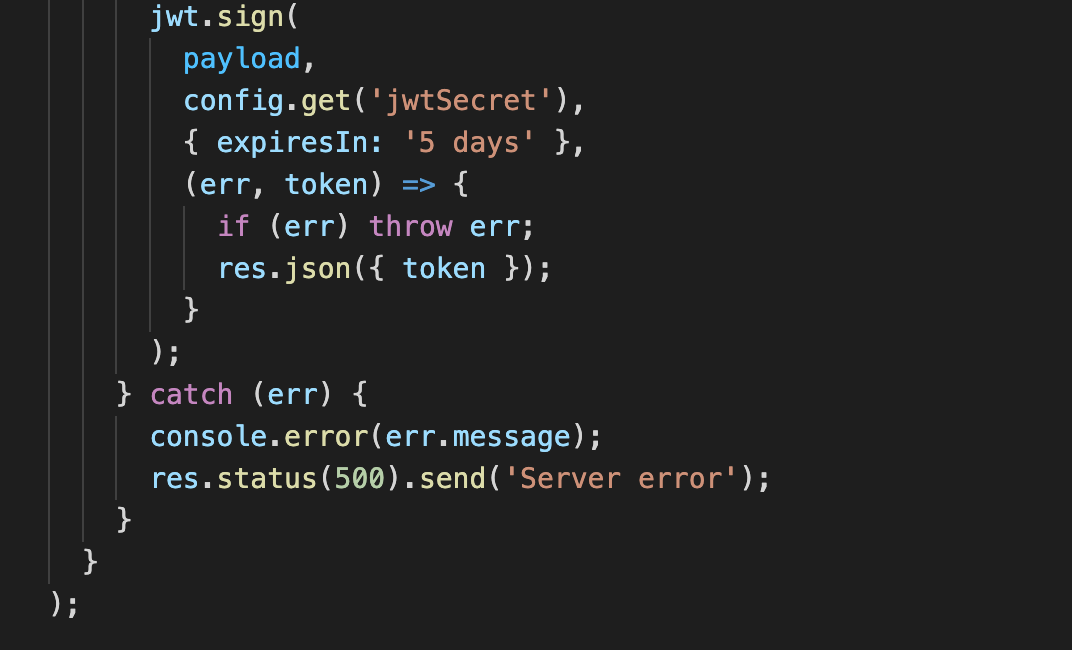
**Place**: Zoom

**Participants:** Shawn, Lijian Yao(James), Alex Wang, Siming Qian, Dawei Yin

**Purpose:** Project Weekly Meeting

**Agenda:**

1. **Past project explanation (James)**

* **Any frontend design idea**
* ****
* In the interface in the previous project, there were several components that I didn't know if there was a bug or there was a problem with my dependency, so I could only open these two pages.
* Very bad comment, basically none.
* Couldn’t see the algorithm from the previous project.
* Something we can refer to is the verification of JWT:
* 

1. **QA rules and discussion (Haiyang)**

* **Sppp**

1. **Security Requirements (Alex)**

* [CS673\_Module2\_reqanalysis](https://docs.google.com/presentation/d/1x_mByuEH1JNtiTdHE-Q1NP-vq-4GGS7B6V9IZumibEw/edit#slide=id.gd7c1b61fce_0_0)
  + **Use System Test (Recovery testing. A recovery test is a system test that forces the software to fail in various ways, therefore verifying that the recovery is performed properly. ...**
  + **Security testing. ...**
  + **Performance testing. ...**
  + **Regression testing. ...**
  + **Alpha testing. ...**
  + **Beta testing.) to design a plan for security**

**Test Plan**

* Test plan

Revision History

| **Name** | **Date** | **Reason For Changes** | **Version** |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |

# **1.** **Objectives**

## **1.1** **Modules under test**

<Identify the modules or system under test.>

## **1.2** **References**

<Give the file name of the document that is referred in this plan >

# **2.** **Testing levels and methods**

## **2.1** **Testing levels**

<Describe if this is unit testing, integration testing, system testing or regression testing>

## **2.2** **Testing methods**

<Describe the methods that are used in the testing: equivalence class testing, and/or boundary value testing and/or statement/branch/path testing, or load testing>

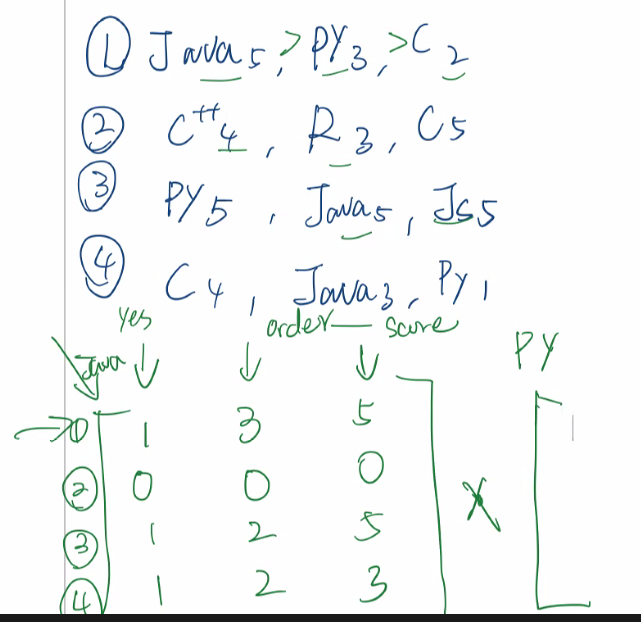
1. **Django learning( Alex)**

* **Use Django to build class and database**

1. **assign tasks for 4 score calculation function**

* [aee-vol02-issue01-p09.pdf](https://drive.google.com/file/d/150-YjmL0yQNJHHSe1QA4T2kNe9riVxE-/view?usp=sharing)

Alex’s conjecture for multichoice? Maybe a provement follows?



|A| only for n x n Matrix

So, maybe we should not use matrix and change it to vector calculus，it also can solve the problem that the nember of students in one group is uncertain.

If one group has x student it will have x vector, v1,v2,......vx

vx=[ order, score]

If we want to calculate the java, use code to select the student who choose the java (means the [must be1, order, score]), so we should need a table in UI

Like

|  | java | python | C/C++ | html | css | Js | Customization | Note：  Maybe  We need to list as many languages as possible that students will use in the software development process |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| yes/no |  |  |  |  |  |  |  |  |
| Order  [Number of language type,0] |  |  |  |  |  |  |  |  |
| Score  [0,5] |  |  |  |  |  |  |  |  |

We can get the data which will be stored in student class when students finish this table,

Thus, we just use the most simple way, vector add, to calculate.

Then, let the code choose all the students who choose yes from the java row

If we find a student 1 and 2 write the “yes” in Java row (when we get all students table, Sort algorithm can complete this)

its vector is v1=[order, score], v2= [order, score]

The best case is must be vector(java)= v1+v2=[8,5]+[8,5] = [16,10] (assume there are 8 language types)

Then calculate the sum of student1 and student2 in python row

If student 1 choose yes in python but student 2 choose no the case of them in python bar should be v1 + 0 x v2 (选no直接变零向量就好)= v1

Then， calculate the next language row of them， until last bar

We will get all vector(language)

When we sign them to the group, just keeping every group that student vector sum in is similar to the best case? Maybe can set the lowest sum of this, if the vector is smaller than lowest , swap students to another group.

I think it is easier to use code for implementation? The algorithm in paper and matrix way. I think the whole night but also can not find corresponding codes. It is just a swap code on paper.

1. **upload objective score functions to GitHub**

**Create objective function for calculating single choice scores.**

1. **Coding standard(convention)**

* **Input , output ( type, return)**
* **Comment specificly**
* **pycharm (format)**

1. **Iteration 1 prepare**
2. **Data structure**

* **Please refer to the github**

**CS673 Software Engineering**

**Team 6: Panda**

**Meeting Minutes**

**Meeting #4**

**Date and Time:** Sep/29/2022 14:00

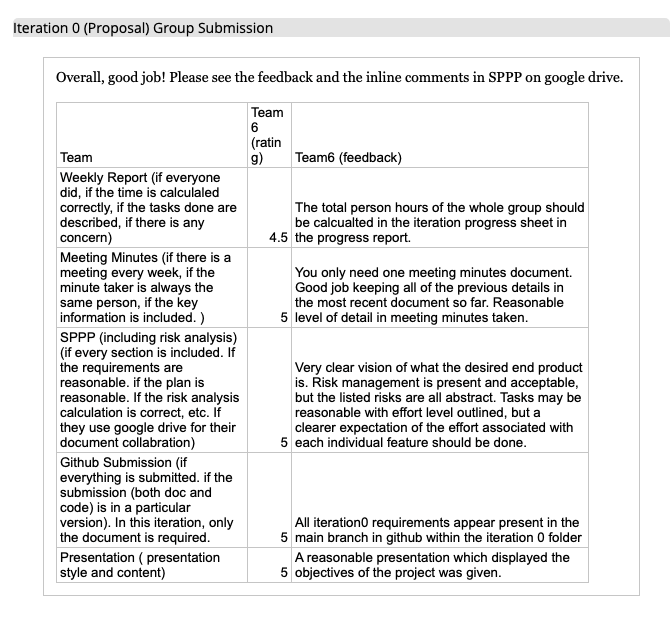
**Place**: Zoom

**Participants:** Shawn,Dawei,Lijian(James),Alex,Siming

**Purpose:** Project Weekly Meeting

**Agenda:**

1. **GO back on SPPP**

* ****
* Revise SPPP (detail)

1. **Assign the task**

* One for frontend (pass project )

\* James

<https://github.com/BUMETCS673/BUMETCS673A1F21P3>

* One for django

\* alex

* Design for Data Structure

\* Siming （Flowchart）

\* Dawei (Class diagram)

* QA test (Not yet)

\* Haiyang

1. Algorithm Discussion

* https://drive.google.com/drive/folders/1snlFwbQyd2IVzxyMKFb32hS9\_4ZHmOHR
* [aee-vol02-issue01-p09.pdf](https://drive.google.com/file/d/150-YjmL0yQNJHHSe1QA4T2kNe9riVxE-/view?usp=sharing)

1. Lab#2

**Discussions:**

1. **GO back on SPPP**

* Revise SPPP (detail)

1. **Assign the task**

* Algorithm and communicate

\* Shawn

* One for frontend (pass project )

\* James

<https://github.com/BUMETCS673/BUMETCS673A1F21P3>

* One for django

\* alex

* Design for Data Structure

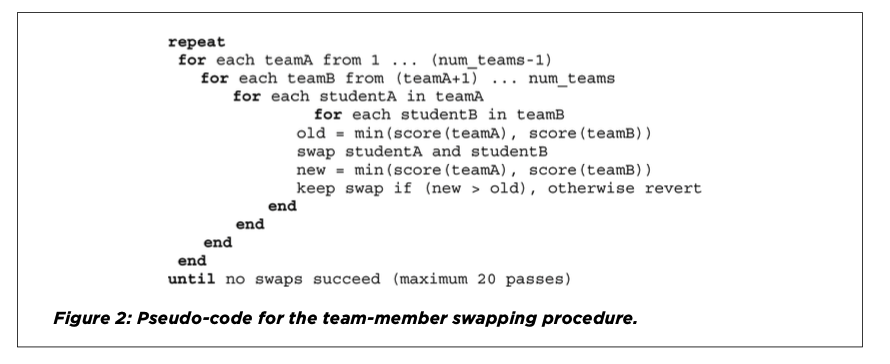
\* Siming （User Flowchart）

\* Dawei (Class diagram)

* QA test (Not yet)

\* Haiyang

1. Algorithm Discussion

* https://drive.google.com/drive/folders/1snlFwbQyd2IVzxyMKFb32hS9\_4ZHmOHR
* [aee-vol02-issue01-p09.pdf](https://drive.google.com/file/d/150-YjmL0yQNJHHSe1QA4T2kNe9riVxE-/view?usp=sharing)
* 

1. Lab#2 (10/5 due)

* Pivotaltracker (user stories)

https://www.pivotaltracker.com/n/projects/2599395

* UML (class diagram) (make sure one for everyone)

**Key Decisions:**

* Algorithm and communicate

\* Shawn

* One for frontend (pass project )

\* James

<https://github.com/BUMETCS673/BUMETCS673A1F21P3>

* One for django

\* alex

* Design for Data Structure

\* Siming （User Flowchart）

\* Dawei (Class diagram)(UML)

**CS673 Software Engineering**

**Team 6: Panda**

**Meeting Minutes**

**Meeting #3**

**Date and Time:** Sep/22/2022 14:00

**Place**: Zoom

**Participants:** Shawn, Dawei, Haiyang, Siming, Alex

**Purpose:** Project Weekly Meeting

**Agenda:**

Determine responsibilities of each leader role and

* Shawn([baymax@bu.edu](mailto:baymax@bu.edu)) - Team leader
* Dawei([davidyin@bu.edu](mailto:davidyin@bu.edu))- Design and Implementation leader
* Lijian([yao049@bu.edu](mailto:yao049@bu.edu)) - Configuration leader
* Haiyang Wang ([whaiyang@bu.edu](mailto:whaiyang@bu.edu))- QA leader
* Siming ([s1a1d1f1@bu.edu](mailto:s1a1d1f1@bu.edu)) - Requirement Leader
* Alex([alexrw@bu.edu](mailto:alexrw@bu.edu)) -Security leader

Assigning different task

* SPPP form for each different role
* All the forms

Determine Project Algorithm

Possible algorithms:

* CATME’s Team-Maker algorithm(rasdfghjklefer to the google drive)
* Design and Validation of a Web-Based System for Assigning Members to Teams Using Instructor-Specified Criteria(refer to the google drive)
* Stable matching algorithm(please google)
* Or somehow combine them?

Determining

* [CS673\_STD Team 6](https://docs.google.com/document/d/1oHVu9x4UpW2W6c9cb_XWMOwt-3vkoT8551xDtq2vPhU/edit)

Find and discuss related works

Discuss last project about team maker

* <https://github.com/BUMETCS673/BUMETCS673A1F21P3>
* haiyang
* Siming
* Dawei

Discuss risks

* Fail to design a valid algorithm
* Not meeting or part of user’s requirements
* Can't finish front end

**Discussions:**

1. Determine responsibilities of each leader role and

* Shawn([baymax@bu.edu](mailto:baymax@bu.edu)) - Team leader
* Dawei([davidyin@bu.edu](mailto:davidyin@bu.edu))- Design and Implementation leader
* Lijian([yao049@bu.edu](mailto:yao049@bu.edu)) - Configuration leader
* Haiyang Wang ([whaiyang@bu.edu](mailto:whaiyang@bu.edu))- QA leader
* Siming ([s1a1d1f1@bu.edu](mailto:s1a1d1f1@bu.edu)) - Requirement Leader
* Alex([alexrw@bu.edu](mailto:alexrw@bu.edu)) -Security leader

1. Assigning different task (prepare for Iteration 1)

* SPPP form for each different role
* All the forms

1. Determine Project Algorithm

Possible algorithms:

* CATME’s Team-Maker algorithm(refer to the google drive)
* Design and Validation of a Web-Based System for Assigning Members to Teams Using Instructor-Specified Criteria(refer to the google drive)
* Stable matching algorithm(please google)
* Or somehow combine them?

1. Determining

* [CS673\_STD Team 6](https://docs.google.com/document/d/1oHVu9x4UpW2W6c9cb_XWMOwt-3vkoT8551xDtq2vPhU/edit)

1. Find and discuss related works

Discuss last project about team maker

* <https://github.com/BUMETCS673/BUMETCS673A1F21P3>
* haiyang
* Siming
* Dawei

1. Discuss risks

* Fail to design a valid algorithm
* Not meeting or part of user’s requirements
* Can't finish front end

**Key Decisions:**

1. **Focus on algorithm design**
2. **Start to learn frontend skill**

**CS673 Software Engineering**

**Team 6: Panda**

**Meeting Minutes**

**Meeting #2**

**Date and Time:** Sep/15/2022 14:00 - 16:25

**Place**: Discord

**Participants:** Haiyang, Shawn,Lijian,Dawei, Siming

**Purpose:** Project Weekly Meeting

**Agenda:**

Decide group name

Determine Project name

Project ideas

Assign roles

Set the weekly meeting time

Find and discuss related works

Assigning task

Brainstorm requirements

Discuss risks

**Discussions:**

1. Determine group name

* Panda

1. Determine project name

* IGroup

1. Project ideas

* Team Match system

1. Assign roles

* Shawn([baymax@bu.edu](mailto:baymax@bu.edu)) - Team leader
* Siming ([s1a1d1f1@bu.edu](mailto:s1a1d1f1@bu.edu)) - Requirement Leader
* Dawei([davidyin@bu.edu](mailto:davidyin@bu.edu))- Design and Implementation leader
* Lijian([yao049@bu.edu](mailto:yao049@bu.edu)) - Configuration leader
* - QA leader
* Haiyang Wang ([whaiyang@bu.edu](mailto:whaiyang@bu.edu)) - Security leader

1. Provide effort hours so far

* Members will email hours spent so far to Shawn (due every weekly meeting)
* Need to decide start/end of week
  + - Week starts ends Thu 2 pm (online) , ends Wed after class (offsite)

1. Find and discuss related works

* Try to comprehensive the previous student project

<https://github.com/BUMETCS673/BUMETCS673A1F21P3>

* Survey

<https://docs.google.com/forms/d/e/1FAIpQLSfkskHnSJLweJSulYyeBenhNPSyzHkQTRW6wzakM_Ffb3gJFA/viewform>

* Team Sign

<https://docs.google.com/document/d/1PzFrv9GLrKaXZP46fl1S_kaAev4Gb6BcfAxBUfSvNSs/edit>

* Related
* https://cdn.discordapp.com/attachments/1019358936496881734/1020067567949316218/Forming\_More\_Effective\_Teams\_Using\_CATME\_TeamMaker\_and\_the\_Gale-Shapley\_Algorithm.pdf

1. Brainstorm requirements

* Algorithm Discussion

1. Discuss risks

* Not perfect algorithm

1. Project criteria

Priority -

1. Algorithm Implementation

2. Backend framework

3. Frontend framework

4. Learning

1. Separation of duties

* Frontend - Shawn, Siming
* Backend - Dawei,haiyang,Lijian

Key Decisions:

* Project name is IGroup
* Time tracking
  + Week start on Thursday
  + Week end on Wednesday
* Roles assigned:
  + Shawn([baymax@bu.edu](mailto:baymax@bu.edu)) - Team leader
  + Siming ([s1a1d1f1@bu.edu](mailto:s1a1d1f1@bu.edu)) - Requirement Leader
  + Dawei([davidyin@bu.edu](mailto:davidyin@bu.edu)) - Design and Implementation leader
  + Lijian([yao049@bu.edu](mailto:yao049@bu.edu)) - Configuration leader
  + - QA leader
  + Haiyang Wang ([whaiyang@bu.edu](mailto:whaiyang@bu.edu)) - Security leader
* Separation of duties
* Frontend - Shawn, Siming
* Backend - Dawei,haiyang,Lijian

**CS673 Software Engineering**

**Team 6: BOLTX**

**Meeting Minutes**

**Meeting 1**

**Date and Time:** Sep/14/2022 11:00 am -13:15pm

**Place**: Zoom & Discord

**Participants:** Dawei, Shawn, Siming, TSing, Lijian,Haiyang

**Purpose:** Project Kickoff Meeting

**Agenda:**

1. **Decide group name**
2. **Determine Project name**
3. Project ideas
4. **Provide effort hours so far**
5. **Finalize communication plan**
6. Find and discuss related works
7. **Brainstorm requirements**
8. **Discuss risks**
9. **Determine an approach/process to use**
   1. **Agile**
   2. **waterfall**
10. **Assign roles**

**Discussions:**

1. Determine group name
   * Is this the same as project name? Yes
2. Determine project name
   * BOLTX
3. Project ideas
4. Provide effort hours so far
   * Members will email hours spent so far to Shawn (due every monday meeting)
   * Need to decide start/end of week
     + Week starts Wed 5pm (offsite) , ends Mon 10pm (online)
5. Finalize communication plan
   * Google group - email distribution
   * Google Docs/Drive - upload and track all documents (including agenda, minutes, etc)
   * Discord/Zoom/Webex - Discussions/brainstorming/to-do and completed tasks
6. Version control
   * Git/Github
7. Find and discuss related works
8. Brainstorm requirements
   * Put in “Project idea”
9. Discuss risks
   * New tools - not understanding/knowing how to use tools
   * Schedules - work and home life
   * Keep it simple/limit scope creep
   * Originality - what differentiates us from others?
   * Multiple user functionality - may be too time consuming
   * Limited time for project as a whole
10. Project criteria
    * Usefulness -
    * Complexity -
    * Originality -
11. Determine an approach/process to use
    * Agile with feedback/iteration
    * Deliver/update project with smaller iterations
12. Assign roles
    * Shawn([baymax@bu.edu](mailto:baymax@bu.edu)) - Team leader
    * Siming ([s1a1d1f1@bu.edu](mailto:s1a1d1f1@bu.edu)) - Requirement Leader
    * Youqing ([yshaots@bu.edu](mailto:yshaots@bu.edu)) - Design and Implementation leader
    * Lijian([yao049@bu.edu](mailto:yao049@bu.edu)) - Configuration leader
    * Dawei([davidyin@bu.edu](mailto:davidyin@bu.edu)) - QA leader
    * Haiyang Wang ([whaiyang@bu.edu](mailto:whaiyang@bu.edu)) - Security leader

Key Decisions:

* Project name is BOLTX -
* Time tracking
  + Week start on Wednesday
  + Week end on Monday
  + Get time to Shawn by noon on Monday
* Version control
  + Git/GitHub
  + Labels - java,XX,XX
* Communication Plan
  + Use Google group for email communication
  + Use Google Docs for task tracking (to-do and complete)
  + Use Webex and Zoom for discussions/brainstorming
  + Use Git/GitHub for document and code repository, version control
* Approach/process to use
  + Agile with feedback/iteration
* Roles assigned:
  + Shawn([baymax@bu.edu](mailto:baymax@bu.edu)) - Team leader
  + Siming ([s1a1d1f1@bu.edu](mailto:s1a1d1f1@bu.edu)) - Requirement Leader
  + Youqing ([yshaots@bu.edu](mailto:yshaots@bu.edu)) - Design and Implementation leader
  + Lijian([yao049@bu.edu](mailto:yao049@bu.edu)) - Configuration leader
  + Dawei([davidyin@bu.edu](mailto:davidyin@bu.edu)) - QA leader
  + Haiyang Wang ([whaiyang@bu.edu](mailto:whaiyang@bu.edu)) - Security leader

**Action Items:**

* Submit time to Shawn by noon on Monday - Siming,Youqing,Lijian,Dawei,Haiyang

Below is an example from a previous project (You shall delete this part in your meeting minutes)

**Date and Time:** 1/26/12 7 - 8PM

**Place**: Group Phone Call

**Participants:** Dan Spuches, Grace Hopkins, Craig Cato

**Minutes taker:** Dan Spuches

**Time Keeper:** Craig Cato

**Purpose:** Project Kickoff Meeting

**Agenda:**

* Determine group name
* Determine project name
* Provide effort hours so far
* Finalize communication plan
  + Google group vs. Trello
* Find and discuss related works
* Google code
  + Create project site
  + File a test bug
  + Check in/out a test document
* Brainstorm requirements
* Discuss risks
* Determine an approach/process to use
* Assign roles

**Discussion:**

* Determine group name
  + Is this the same as project name? Yes
* Determine project name
  + Yet another weight tracker - taken
  + Yet another weight program - YAWP
    - Don't want to make YAWP noise when you stand on the scale
  + BodyStats
  + Yet another weight history program
  + Yet another weigh-in program
  + Yet another weight oriented program
* Provide effort hours so far
  + Members will email hours spent so far to Grace
  + Need to decide start/end of week
    - Week starts Saturday, ends Sunday
* Finalize communication plan
  + Google group - email distribution
  + Google code - upload and track all documents (including agenda, minutes, etc)
  + Trello - Discussions/brainstorming/to-do and completed tasks
* Find and discuss related works
  + http://download.cnet.com/Weight-Tracker/3000-2129\_4-10458217.html
  + weightchart.com
    - Web based
  + weightwatchers.com
    - Web based
  + Our project is standalone, not web based, open source (differentiator)
* Google code
  + Create project site
  + File a test bug
  + Check in/out a test document
  + SVN or GIT?
    - We will use SVN
    - Tortoise SVN for windows
  + What license will we use?
    - Apache 2.0
    - What are the terms?
    - Need to tag all works with the license text from http://www.apache.org/licenses/LICENSE-2.0
* Brainstorm requirements
  + Functional
  + Non-functional
  + Desktop java standalone client
  + Not networked
  + Single user per instance
    - Future - multiple users
  + Need to be able to enter weights
  + Calculate BMI
  + Charting over time
    - Export charts?
    - Daily weight change
    - Monthly weight loss
    - Trending of data
    - Projections
  + Target weight
  + Sounds?
    - Applause for loss
    - YAWP for gain
  + Computerize printed charts
  + Print charts/data
  + Export and save functions
  + Options
    - Configurable units
      * English vs metric
      * LBS vs KG vs Stones?
* Discuss risks
  + New tools - not understanding/knowing how to use tools
  + Schedules - work and home life
  + Keep it simple/limit scope creep
  + Originality - what differentiates us from others?
  + Multiple user functionality - may be too time consuming
  + Limited time for project as a whole
* Project criteria
  + Usefulness - nobody has yet found the best way to do it, there are a lot of other ones out there, none are right yet?
  + Complexity - will be sufficiently complex
  + Originality - it is original because Craig created the concept
* Determine an approach/process to use
  + Waterfall with feedback/iteration
    - Ability to revisit requirements and re-shuffle priorities
    - Need to build in the ability to respond to risks as they arise and difficult requirements
  + Possibly some agile concepts/aspects - prototype and test driven
  + JUnit testing - test driven development
* Assign roles
  + Grace - Leader and QA
  + Craig - Configuration Mgmt
  + Dan - Implementation

**Key Decisions**

* Project name is YAWP - yet another weight-tracking program
* Google code
  + https://code.google.com/p/yawp/
  + We will use SVN on Google code
  + Source code license - Apache License 2.0
  + Labels - health, academic, java
* Time tracking
  + Week start on Sunday
  + Week end on Saturday
  + Get time to Grace by noon on Sunday
* Communication Plan
  + Use Google group for email communication
  + Use Trello for task tracking (to-do and complete) and discussions/brainstorming
  + Use Google Code for document and code repository, version control
* Roles assigned:
  + Grace - Leader and QA
  + Craig - Configuration Mgmt
  + Dan - Implementation

**Action Items:**

* Review terms of Apache license - Dan, Craig, Grace
* Submit time to Grace by noon Sunday - Dan, Craig, Grace

**CS673 Software Engineering**

**Team 6: Panda**

**Meeting Minutes**

**Meeting #7**

**Date and Time:** Oct 20 14:00

**Place**: zoom

**Participants:** Shawn,James ,Haiyang, Dawei, Alex

**Purpose:** Project Weekly Meeting

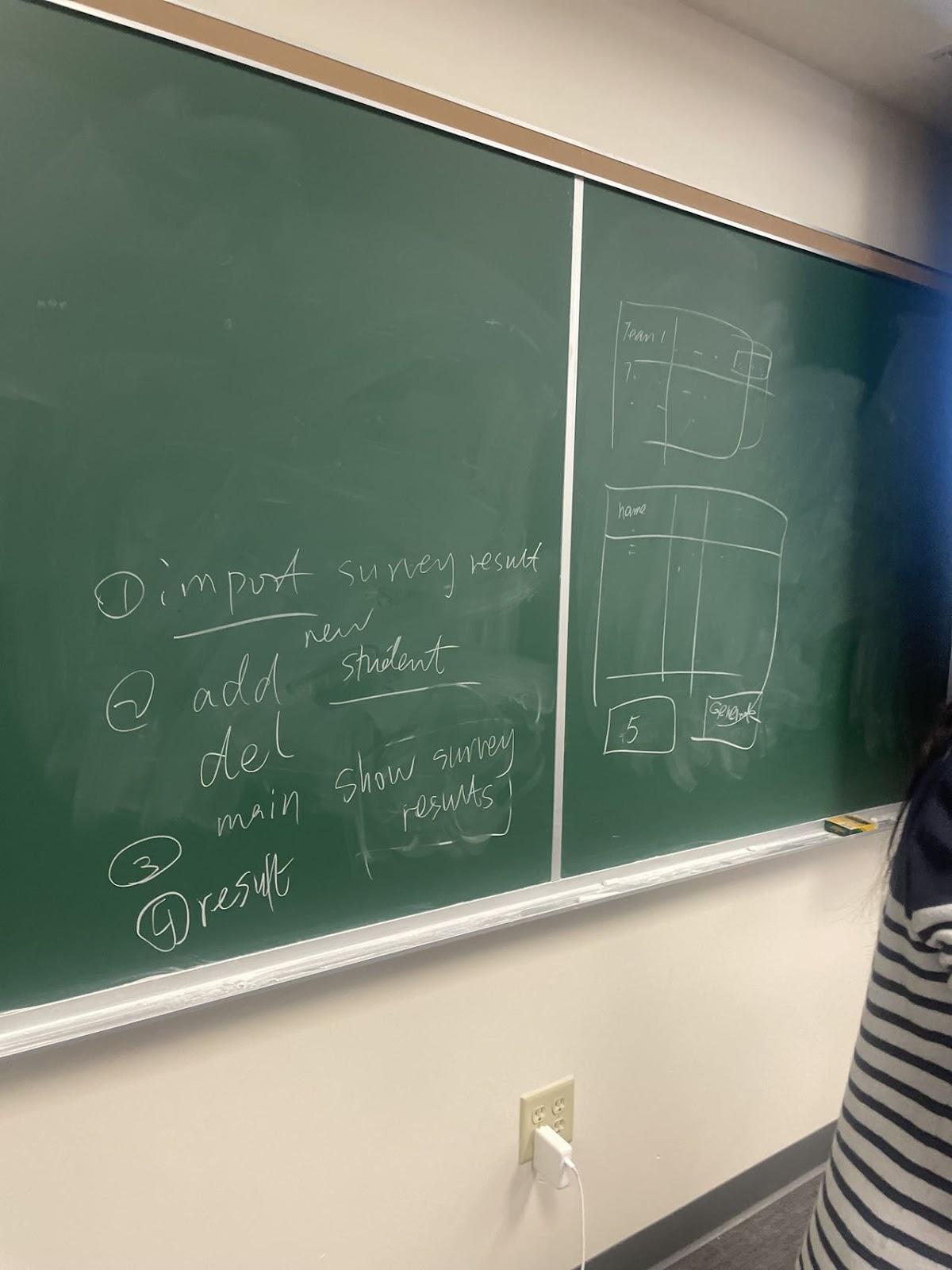
**Agenda:**

**Assign task:**

1. **QA test (haiyang)**

* “Sppp” add two more test (acceptance test, regression test)
* Make a task assignment list

1. **UI (James)**

****

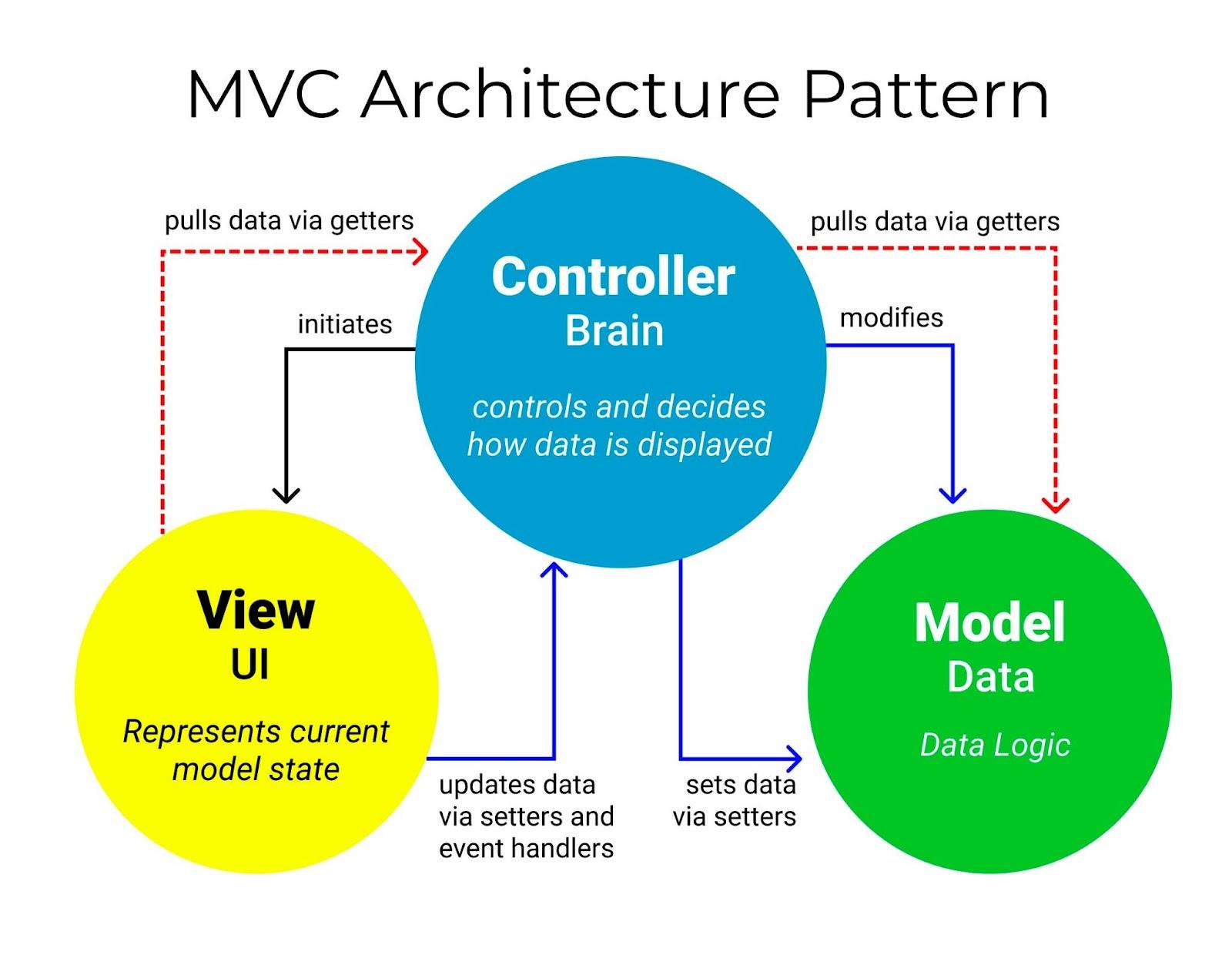
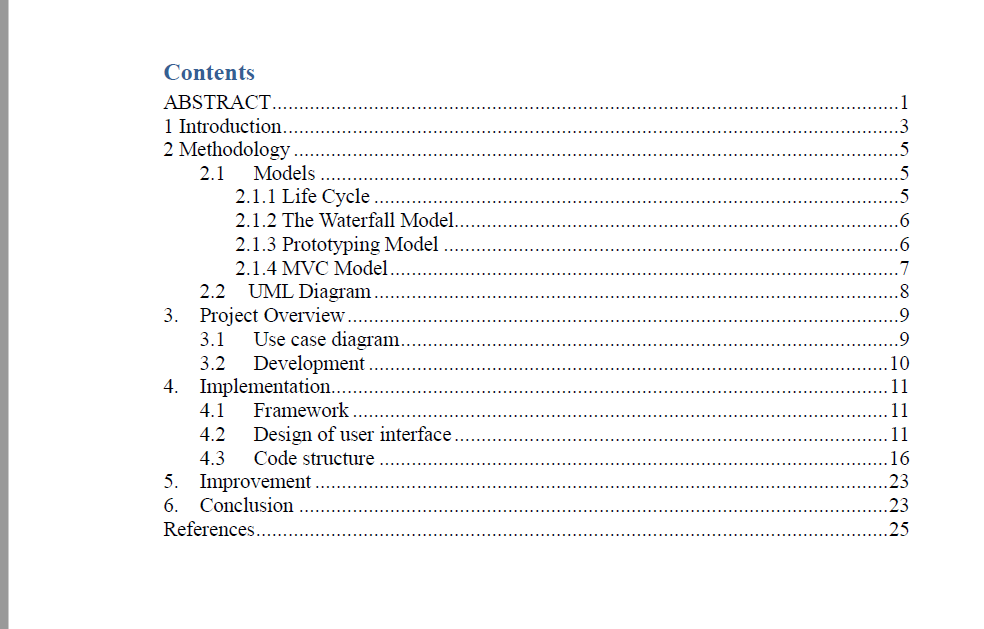
* UI require 7 pages (sddd)

1. Course selection
2. Setting survey
3. Import student survey data
4. Set group number
5. Get initial group result
6. Manually setting
7. Get final result
8. **Alex**

* SDDD

1.database design

2. Design Patterns (MVC)

****

1. **Dawei**

* sddd

# Key Algorithms （pseudo code)

1. **Shawn**

* Organize iteration 1 materials

**Next:**

1. **Group meet at Saturday 10/22 (Mugur)**
2. **Discuss about rest two algorithm**
3. **Decide database design**
4. **Learning Django**

**CS673 Software Engineering**

**Team 6: Panda**

**Meeting Minutes**

**Meeting #6**

**Date and Time:** Oct 14 14:00

**Place**: Mugur Library Basement

**Participants:** Shawn,James ,Haiyang,Siiming, Dawei, Alex

**Purpose:** Project Weekly Meeting

**Agenda:**

1. **UI (James)**

* SDDD form

1.UI Design (if applicable)

In this section, you can describe your UI design

**# UI ( student)**

* Sign in
* Login
* Take Survey
* Submit Successfully
* Rate page (用户打分页面）

**# UI ( instructor)**

* Sign in
* Login
* Set Survey
* Create a Link Successfully
* Get Result ( shows how many student complete survey)
* Manually setting groups
* Final group result

1. **QA test (Haiyang)**

* Sppp ( QA part)

1. Metric
2. Testing ( unit , Integration , system)

* Some test code

1. **Requirement (siming)**

* Sppp

Doc/ProjX\_userstories(generated)

* SDDD
* Summary additional things about iteration 1 , sent to shawn

# Security Design (Aflex)

1. System Design (Dawei)

6. Question(shawn):

which type

* SDDD

# Software Architecture

* 1. Implementation of survey system.
  2. Implementation of objective function for calculating the multiple choice answer score within a team.

1. Database design (option)

**CS673 Software Engineering**

**Team 6: Panda**

**Meeting Minutes**

**Meeting #5**

**Date and Time:** Oct 08 19:00 -21:00

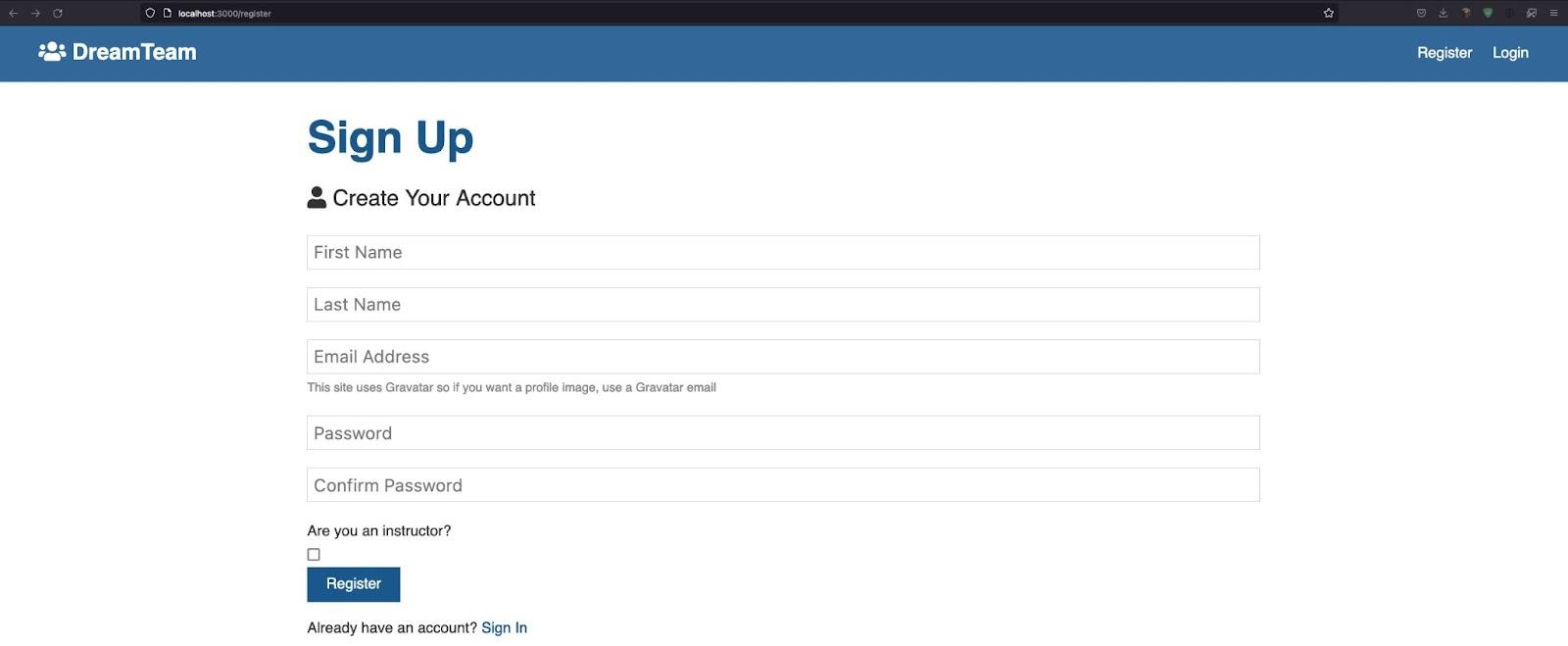
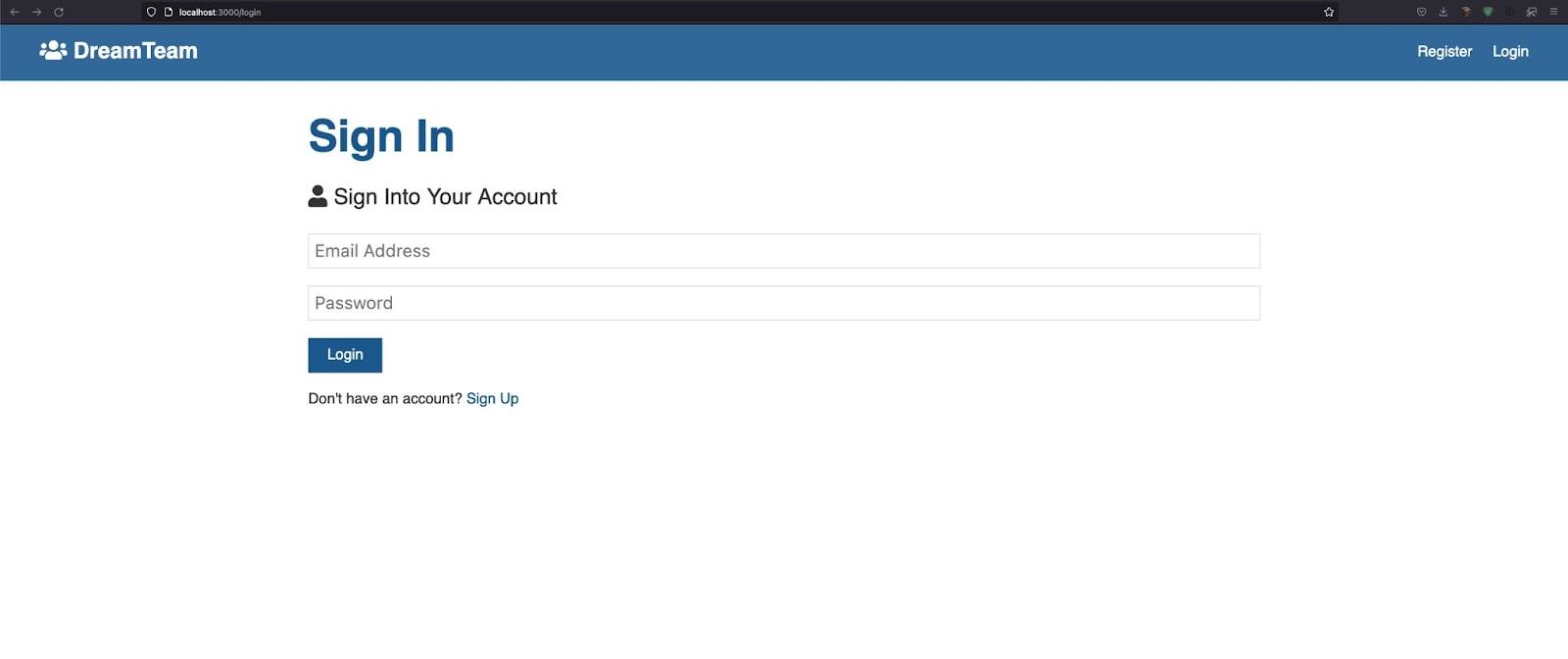
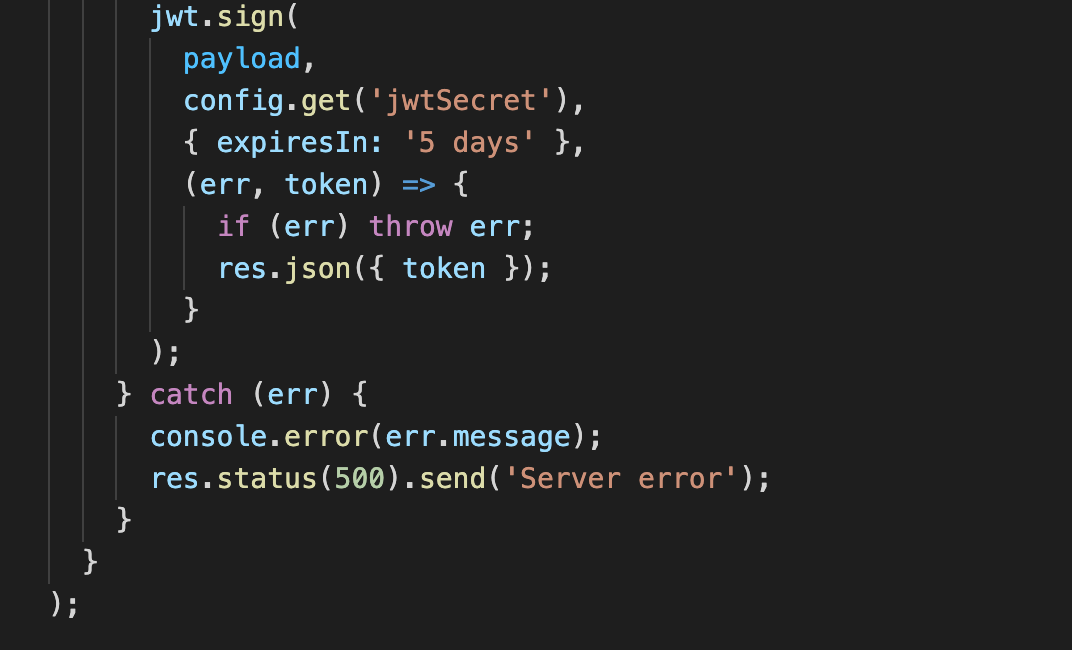
**Place**: Zoom

**Participants:** Shawn, Lijian Yao(James), Alex Wang, Siming Qian, Dawei Yin

**Purpose:** Project Weekly Meeting

**Agenda:**

1. **Past project explanation (James)**

* **Any frontend design idea**
* ****
* In the interface in the previous project, there were several components that I didn't know if there was a bug or there was a problem with my dependency, so I could only open these two pages.
* Very bad comment, basically none.
* Couldn’t see the algorithm from the previous project.
* Something we can refer to is the verification of JWT:
* 

1. **QA rules and discussion (Haiyang)**

* **Sppp**

1. **Security Requirements (Alex)**

* [CS673\_Module2\_reqanalysis](https://docs.google.com/presentation/d/1x_mByuEH1JNtiTdHE-Q1NP-vq-4GGS7B6V9IZumibEw/edit#slide=id.gd7c1b61fce_0_0)
  + **Use System Test (Recovery testing. A recovery test is a system test that forces the software to fail in various ways, therefore verifying that the recovery is performed properly. ...**
  + **Security testing. ...**
  + **Performance testing. ...**
  + **Regression testing. ...**
  + **Alpha testing. ...**
  + **Beta testing.) to design a plan for security**

**Test Plan**

* Test plan

Revision History

| **Name** | **Date** | **Reason For Changes** | **Version** |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |

# **1.** **Objectives**

## **1.1** **Modules under test**

<Identify the modules or system under test.>

## **1.2** **References**

<Give the file name of the document that is referred in this plan >

# **2.** **Testing levels and methods**

## **2.1** **Testing levels**

<Describe if this is unit testing, integration testing, system testing or regression testing>

## **2.2** **Testing methods**

<Describe the methods that are used in the testing: equivalence class testing, and/or boundary value testing and/or statement/branch/path testing, or load testing>

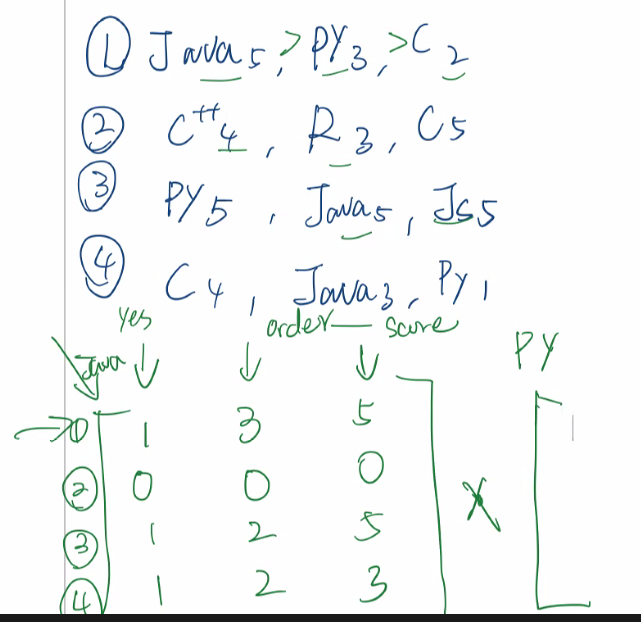
1. **Django learning( Alex)**

* **Use Django to build class and database**

1. **assign tasks for 4 score calculation function**

* [aee-vol02-issue01-p09.pdf](https://drive.google.com/file/d/150-YjmL0yQNJHHSe1QA4T2kNe9riVxE-/view?usp=sharing)

Alex’s conjecture for multichoice? Maybe a provement follows?



|A| only for n x n Matrix

So, maybe we should not use matrix and change it to vector calculus，it also can solve the problem that the nember of students in one group is uncertain.

If one group has x student it will have x vector, v1,v2,......vx

vx=[ order, score]

If we want to calculate the java, use code to select the student who choose the java (means the [must be1, order, score]), so we should need a table in UI

Like

|  | java | python | C/C++ | html | css | Js | Customization | Note：  Maybe  We need to list as many languages as possible that students will use in the software development process |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| yes/no |  |  |  |  |  |  |  |  |
| Order  [Number of language type,0] |  |  |  |  |  |  |  |  |
| Score  [0,5] |  |  |  |  |  |  |  |  |

We can get the data which will be stored in student class when students finish this table,

Thus, we just use the most simple way, vector add, to calculate.

Then, let the code choose all the students who choose yes from the java row

If we find a student 1 and 2 write the “yes” in Java row (when we get all students table, Sort algorithm can complete this)

its vector is v1=[order, score], v2= [order, score]

The best case is must be vector(java)= v1+v2=[8,5]+[8,5] = [16,10] (assume there are 8 language types)

Then calculate the sum of student1 and student2 in python row

If student 1 choose yes in python but student 2 choose no the case of them in python bar should be v1 + 0 x v2 (选no直接变零向量就好)= v1

Then， calculate the next language row of them， until last bar

We will get all vector(language)

When we sign them to the group, just keeping every group that student vector sum in is similar to the best case? Maybe can set the lowest sum of this, if the vector is smaller than lowest , swap students to another group.

I think it is easier to use code for implementation? The algorithm in paper and matrix way. I think the whole night but also can not find corresponding codes. It is just a swap code on paper.

1. **upload objective score functions to GitHub**

**Create objective function for calculating single choice scores.**

1. **Coding standard(convention)**

* **Input , output ( type, return)**
* **Comment specificly**
* **pycharm (format)**

1. **Iteration 1 prepare**
2. **Data structure**

* **Please refer to the github**

**CS673 Software Engineering**

**Team 6: Panda**

**Meeting Minutes**

**Meeting #4**

**Date and Time:** Sep/29/2022 14:00

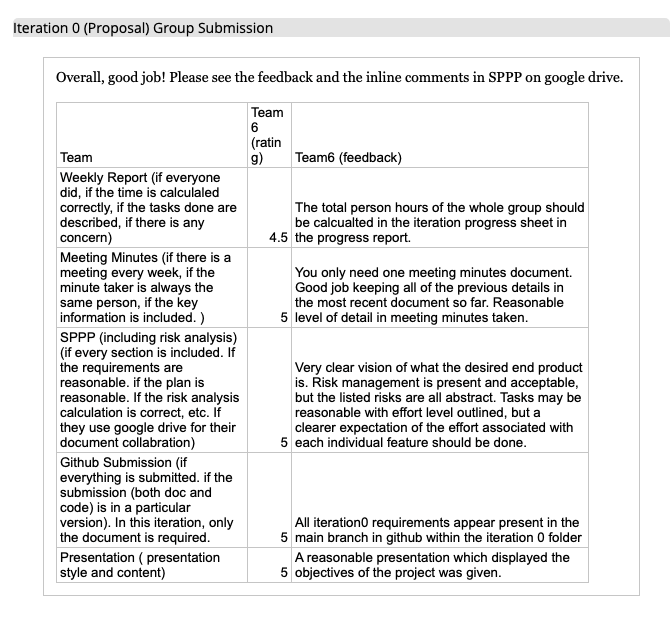
**Place**: Zoom

**Participants:** Shawn,Dawei,Lijian(James),Alex,Siming

**Purpose:** Project Weekly Meeting

**Agenda:**

1. **GO back on SPPP**

* ****
* Revise SPPP (detail)

1. **Assign the task**

* One for frontend (pass project )

\* James

<https://github.com/BUMETCS673/BUMETCS673A1F21P3>

* One for django

\* alex

* Design for Data Structure

\* Siming （Flowchart）

\* Dawei (Class diagram)

* QA test (Not yet)

\* Haiyang

1. Algorithm Discussion

* https://drive.google.com/drive/folders/1snlFwbQyd2IVzxyMKFb32hS9\_4ZHmOHR
* [aee-vol02-issue01-p09.pdf](https://drive.google.com/file/d/150-YjmL0yQNJHHSe1QA4T2kNe9riVxE-/view?usp=sharing)

1. Lab#2

**Discussions:**

1. **GO back on SPPP**

* Revise SPPP (detail)

1. **Assign the task**

* Algorithm and communicate

\* Shawn

* One for frontend (pass project )

\* James

<https://github.com/BUMETCS673/BUMETCS673A1F21P3>

* One for django

\* alex

* Design for Data Structure

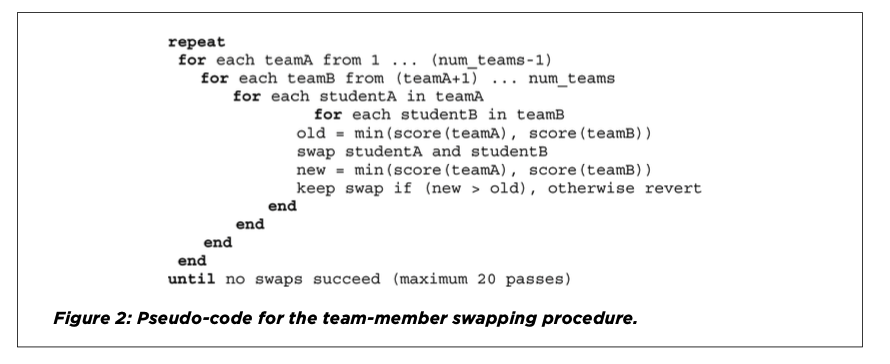
\* Siming （User Flowchart）

\* Dawei (Class diagram)

* QA test (Not yet)

\* Haiyang

1. Algorithm Discussion

* https://drive.google.com/drive/folders/1snlFwbQyd2IVzxyMKFb32hS9\_4ZHmOHR
* [aee-vol02-issue01-p09.pdf](https://drive.google.com/file/d/150-YjmL0yQNJHHSe1QA4T2kNe9riVxE-/view?usp=sharing)
* 

1. Lab#2 (10/5 due)

* Pivotaltracker (user stories)

https://www.pivotaltracker.com/n/projects/2599395

* UML (class diagram) (make sure one for everyone)

**Key Decisions:**

* Algorithm and communicate

\* Shawn

* One for frontend (pass project )

\* James

<https://github.com/BUMETCS673/BUMETCS673A1F21P3>

* One for django

\* alex

* Design for Data Structure

\* Siming （User Flowchart）

\* Dawei (Class diagram)(UML)

**CS673 Software Engineering**

**Team 6: Panda**

**Meeting Minutes**

**Meeting #3**

**Date and Time:** Sep/22/2022 14:00

**Place**: Zoom

**Participants:** Shawn, Dawei, Haiyang, Siming, Alex

**Purpose:** Project Weekly Meeting

**Agenda:**

Determine responsibilities of each leader role and

* Shawn([baymax@bu.edu](mailto:baymax@bu.edu)) - Team leader
* Dawei([davidyin@bu.edu](mailto:davidyin@bu.edu))- Design and Implementation leader
* Lijian([yao049@bu.edu](mailto:yao049@bu.edu)) - Configuration leader
* Haiyang Wang ([whaiyang@bu.edu](mailto:whaiyang@bu.edu))- QA leader
* Siming ([s1a1d1f1@bu.edu](mailto:s1a1d1f1@bu.edu)) - Requirement Leader
* Alex([alexrw@bu.edu](mailto:alexrw@bu.edu)) -Security leader

Assigning different task

* SPPP form for each different role
* All the forms

Determine Project Algorithm

Possible algorithms:

* CATME’s Team-Maker algorithm(rasdfghjklefer to the google drive)
* Design and Validation of a Web-Based System for Assigning Members to Teams Using Instructor-Specified Criteria(refer to the google drive)
* Stable matching algorithm(please google)
* Or somehow combine them?

Determining

* [CS673\_STD Team 6](https://docs.google.com/document/d/1oHVu9x4UpW2W6c9cb_XWMOwt-3vkoT8551xDtq2vPhU/edit)

Find and discuss related works

Discuss last project about team maker

* <https://github.com/BUMETCS673/BUMETCS673A1F21P3>
* haiyang
* Siming
* Dawei

Discuss risks

* Fail to design a valid algorithm
* Not meeting or part of user’s requirements
* Can't finish front end

**Discussions:**

1. Determine responsibilities of each leader role and

* Shawn([baymax@bu.edu](mailto:baymax@bu.edu)) - Team leader
* Dawei([davidyin@bu.edu](mailto:davidyin@bu.edu))- Design and Implementation leader
* Lijian([yao049@bu.edu](mailto:yao049@bu.edu)) - Configuration leader
* Haiyang Wang ([whaiyang@bu.edu](mailto:whaiyang@bu.edu))- QA leader
* Siming ([s1a1d1f1@bu.edu](mailto:s1a1d1f1@bu.edu)) - Requirement Leader
* Alex([alexrw@bu.edu](mailto:alexrw@bu.edu)) -Security leader

1. Assigning different task (prepare for Iteration 1)

* SPPP form for each different role
* All the forms

1. Determine Project Algorithm

Possible algorithms:

* CATME’s Team-Maker algorithm(refer to the google drive)
* Design and Validation of a Web-Based System for Assigning Members to Teams Using Instructor-Specified Criteria(refer to the google drive)
* Stable matching algorithm(please google)
* Or somehow combine them?

1. Determining

* [CS673\_STD Team 6](https://docs.google.com/document/d/1oHVu9x4UpW2W6c9cb_XWMOwt-3vkoT8551xDtq2vPhU/edit)

1. Find and discuss related works

Discuss last project about team maker

* <https://github.com/BUMETCS673/BUMETCS673A1F21P3>
* haiyang
* Siming
* Dawei

1. Discuss risks

* Fail to design a valid algorithm
* Not meeting or part of user’s requirements
* Can't finish front end

**Key Decisions:**

1. **Focus on algorithm design**
2. **Start to learn frontend skill**

**CS673 Software Engineering**

**Team 6: Panda**

**Meeting Minutes**

**Meeting #2**

**Date and Time:** Sep/15/2022 14:00 - 16:25

**Place**: Discord

**Participants:** Haiyang, Shawn,Lijian,Dawei, Siming

**Purpose:** Project Weekly Meeting

**Agenda:**

Decide group name

Determine Project name

Project ideas

Assign roles

Set the weekly meeting time

Find and discuss related works

Assigning task

Brainstorm requirements

Discuss risks

**Discussions:**

1. Determine group name

* Panda

1. Determine project name

* IGroup

1. Project ideas

* Team Match system

1. Assign roles

* Shawn([baymax@bu.edu](mailto:baymax@bu.edu)) - Team leader
* Siming ([s1a1d1f1@bu.edu](mailto:s1a1d1f1@bu.edu)) - Requirement Leader
* Dawei([davidyin@bu.edu](mailto:davidyin@bu.edu))- Design and Implementation leader
* Lijian([yao049@bu.edu](mailto:yao049@bu.edu)) - Configuration leader
* - QA leader
* Haiyang Wang ([whaiyang@bu.edu](mailto:whaiyang@bu.edu)) - Security leader

1. Provide effort hours so far

* Members will email hours spent so far to Shawn (due every weekly meeting)
* Need to decide start/end of week
  + - Week starts ends Thu 2 pm (online) , ends Wed after class (offsite)

1. Find and discuss related works

* Try to comprehensive the previous student project

<https://github.com/BUMETCS673/BUMETCS673A1F21P3>

* Survey

<https://docs.google.com/forms/d/e/1FAIpQLSfkskHnSJLweJSulYyeBenhNPSyzHkQTRW6wzakM_Ffb3gJFA/viewform>

* Team Sign

<https://docs.google.com/document/d/1PzFrv9GLrKaXZP46fl1S_kaAev4Gb6BcfAxBUfSvNSs/edit>

* Related
* https://cdn.discordapp.com/attachments/1019358936496881734/1020067567949316218/Forming\_More\_Effective\_Teams\_Using\_CATME\_TeamMaker\_and\_the\_Gale-Shapley\_Algorithm.pdf

1. Brainstorm requirements

* Algorithm Discussion

1. Discuss risks

* Not perfect algorithm

1. Project criteria

Priority -

1. Algorithm Implementation

2. Backend framework

3. Frontend framework

4. Learning

1. Separation of duties

* Frontend - Shawn, Siming
* Backend - Dawei,haiyang,Lijian

Key Decisions:

* Project name is IGroup
* Time tracking
  + Week start on Thursday
  + Week end on Wednesday
* Roles assigned:
  + Shawn([baymax@bu.edu](mailto:baymax@bu.edu)) - Team leader
  + Siming ([s1a1d1f1@bu.edu](mailto:s1a1d1f1@bu.edu)) - Requirement Leader
  + Dawei([davidyin@bu.edu](mailto:davidyin@bu.edu)) - Design and Implementation leader
  + Lijian([yao049@bu.edu](mailto:yao049@bu.edu)) - Configuration leader
  + - QA leader
  + Haiyang Wang ([whaiyang@bu.edu](mailto:whaiyang@bu.edu)) - Security leader
* Separation of duties
* Frontend - Shawn, Siming
* Backend - Dawei,haiyang,Lijian

**CS673 Software Engineering**

**Team 6: BOLTX**

**Meeting Minutes**

**Meeting 1**

**Date and Time:** Sep/14/2022 11:00 am -13:15pm

**Place**: Zoom & Discord

**Participants:** Dawei, Shawn, Siming, TSing, Lijian,Haiyang

**Purpose:** Project Kickoff Meeting

**Agenda:**

1. **Decide group name**
2. **Determine Project name**
3. Project ideas
4. **Provide effort hours so far**
5. **Finalize communication plan**
6. Find and discuss related works
7. **Brainstorm requirements**
8. **Discuss risks**
9. **Determine an approach/process to use**
   1. **Agile**
   2. **waterfall**
10. **Assign roles**

**Discussions:**

1. Determine group name
   * Is this the same as project name? Yes
2. Determine project name
   * BOLTX
3. Project ideas
4. Provide effort hours so far
   * Members will email hours spent so far to Shawn (due every monday meeting)
   * Need to decide start/end of week
     + Week starts Wed 5pm (offsite) , ends Mon 10pm (online)
5. Finalize communication plan
   * Google group - email distribution
   * Google Docs/Drive - upload and track all documents (including agenda, minutes, etc)
   * Discord/Zoom/Webex - Discussions/brainstorming/to-do and completed tasks
6. Version control
   * Git/Github
7. Find and discuss related works
8. Brainstorm requirements
   * Put in “Project idea”
9. Discuss risks
   * New tools - not understanding/knowing how to use tools
   * Schedules - work and home life
   * Keep it simple/limit scope creep
   * Originality - what differentiates us from others?
   * Multiple user functionality - may be too time consuming
   * Limited time for project as a whole
10. Project criteria
    * Usefulness -
    * Complexity -
    * Originality -
11. Determine an approach/process to use
    * Agile with feedback/iteration
    * Deliver/update project with smaller iterations
12. Assign roles
    * Shawn([baymax@bu.edu](mailto:baymax@bu.edu)) - Team leader
    * Siming ([s1a1d1f1@bu.edu](mailto:s1a1d1f1@bu.edu)) - Requirement Leader
    * Youqing ([yshaots@bu.edu](mailto:yshaots@bu.edu)) - Design and Implementation leader
    * Lijian([yao049@bu.edu](mailto:yao049@bu.edu)) - Configuration leader
    * Dawei([davidyin@bu.edu](mailto:davidyin@bu.edu)) - QA leader
    * Haiyang Wang ([whaiyang@bu.edu](mailto:whaiyang@bu.edu)) - Security leader

Key Decisions:

* Project name is BOLTX -
* Time tracking
  + Week start on Wednesday
  + Week end on Monday
  + Get time to Shawn by noon on Monday
* Version control
  + Git/GitHub
  + Labels - java,XX,XX
* Communication Plan
  + Use Google group for email communication
  + Use Google Docs for task tracking (to-do and complete)
  + Use Webex and Zoom for discussions/brainstorming
  + Use Git/GitHub for document and code repository, version control
* Approach/process to use
  + Agile with feedback/iteration
* Roles assigned:
  + Shawn([baymax@bu.edu](mailto:baymax@bu.edu)) - Team leader
  + Siming ([s1a1d1f1@bu.edu](mailto:s1a1d1f1@bu.edu)) - Requirement Leader
  + Youqing ([yshaots@bu.edu](mailto:yshaots@bu.edu)) - Design and Implementation leader
  + Lijian([yao049@bu.edu](mailto:yao049@bu.edu)) - Configuration leader
  + Dawei([davidyin@bu.edu](mailto:davidyin@bu.edu)) - QA leader
  + Haiyang Wang ([whaiyang@bu.edu](mailto:whaiyang@bu.edu)) - Security leader

**Action Items:**

* Submit time to Shawn by noon on Monday - Siming,Youqing,Lijian,Dawei,Haiyang

Below is an example from a previous project (You shall delete this part in your meeting minutes)

**Date and Time:** 1/26/12 7 - 8PM

**Place**: Group Phone Call

**Participants:** Dan Spuches, Grace Hopkins, Craig Cato

**Minutes taker:** Dan Spuches

**Time Keeper:** Craig Cato

**Purpose:** Project Kickoff Meeting

**Agenda:**

* Determine group name
* Determine project name
* Provide effort hours so far
* Finalize communication plan
  + Google group vs. Trello
* Find and discuss related works
* Google code
  + Create project site
  + File a test bug
  + Check in/out a test document
* Brainstorm requirements
* Discuss risks
* Determine an approach/process to use
* Assign roles

**Discussion:**

* Determine group name
  + Is this the same as project name? Yes
* Determine project name
  + Yet another weight tracker - taken
  + Yet another weight program - YAWP
    - Don't want to make YAWP noise when you stand on the scale
  + BodyStats
  + Yet another weight history program
  + Yet another weigh-in program
  + Yet another weight oriented program
* Provide effort hours so far
  + Members will email hours spent so far to Grace
  + Need to decide start/end of week
    - Week starts Saturday, ends Sunday
* Finalize communication plan
  + Google group - email distribution
  + Google code - upload and track all documents (including agenda, minutes, etc)
  + Trello - Discussions/brainstorming/to-do and completed tasks
* Find and discuss related works
  + http://download.cnet.com/Weight-Tracker/3000-2129\_4-10458217.html
  + weightchart.com
    - Web based
  + weightwatchers.com
    - Web based
  + Our project is standalone, not web based, open source (differentiator)
* Google code
  + Create project site
  + File a test bug
  + Check in/out a test document
  + SVN or GIT?
    - We will use SVN
    - Tortoise SVN for windows
  + What license will we use?
    - Apache 2.0
    - What are the terms?
    - Need to tag all works with the license text from http://www.apache.org/licenses/LICENSE-2.0
* Brainstorm requirements
  + Functional
  + Non-functional
  + Desktop java standalone client
  + Not networked
  + Single user per instance
    - Future - multiple users
  + Need to be able to enter weights
  + Calculate BMI
  + Charting over time
    - Export charts?
    - Daily weight change
    - Monthly weight loss
    - Trending of data
    - Projections
  + Target weight
  + Sounds?
    - Applause for loss
    - YAWP for gain
  + Computerize printed charts
  + Print charts/data
  + Export and save functions
  + Options
    - Configurable units
      * English vs metric
      * LBS vs KG vs Stones?
* Discuss risks
  + New tools - not understanding/knowing how to use tools
  + Schedules - work and home life
  + Keep it simple/limit scope creep
  + Originality - what differentiates us from others?
  + Multiple user functionality - may be too time consuming
  + Limited time for project as a whole
* Project criteria
  + Usefulness - nobody has yet found the best way to do it, there are a lot of other ones out there, none are right yet?
  + Complexity - will be sufficiently complex
  + Originality - it is original because Craig created the concept
* Determine an approach/process to use
  + Waterfall with feedback/iteration
    - Ability to revisit requirements and re-shuffle priorities
    - Need to build in the ability to respond to risks as they arise and difficult requirements
  + Possibly some agile concepts/aspects - prototype and test driven
  + JUnit testing - test driven development
* Assign roles
  + Grace - Leader and QA
  + Craig - Configuration Mgmt
  + Dan - Implementation

**Key Decisions**

* Project name is YAWP - yet another weight-tracking program
* Google code
  + https://code.google.com/p/yawp/
  + We will use SVN on Google code
  + Source code license - Apache License 2.0
  + Labels - health, academic, java
* Time tracking
  + Week start on Sunday
  + Week end on Saturday
  + Get time to Grace by noon on Sunday
* Communication Plan
  + Use Google group for email communication
  + Use Trello for task tracking (to-do and complete) and discussions/brainstorming
  + Use Google Code for document and code repository, version control
* Roles assigned:
  + Grace - Leader and QA
  + Craig - Configuration Mgmt
  + Dan - Implementation

**Action Items:**

* Review terms of Apache license - Dan, Craig, Grace
* Submit time to Grace by noon Sunday - Dan, Craig, Grace