**CS673 Software Engineering**

**Team 1 : Team 1**

**Meeting Minutes**

All meeting minutes are kept in this single document. The latest meeting minutes should be at the beginning of the document. For example, meeting 3 minutes is placed before meeting 2 in the document. The team leader should prepare a basic agenda for the meeting and team members should rotate to be the minutes taker. Each group should have at least one meeting per week, and you may have multiple meetings if needed.

**Meeting 6**

**Meeting 5**

**Meeting 4**

**Date and Time:** 9/13/2023 @7:00PM EST

**Place**: Discord/Zoom

**Participants:** Clyde Yeung, Brian Fenstermacher, Jisoo Lee, Mali Rivera, Sean Rawson, Sherif Zeyada, Chaitanya Saraogi

**Minutes taker:** Clyde Yeung

**Timekeeper:**

**Purpose:** Review demo and recording

**Agenda:**

* Final check up on the SPPP doc
* Final review presentation slide and recording

**Action Items**:

* Finalize the iteration0 submission

**Next Meeting:**

9/14/2023 after class

**Meeting 3**

**Date and Time:** 9/11/2023 @7:00PM EST

**Place**: Discord/Zoom

**Participants:** Clyde Yeung, Brian Fenstermacher, Jisoo Lee, Mali Rivera, Sean Rawson, Sherif Zeyada

**Minutes taker:** Mali Rivera

**Timekeeper:**

**Purpose:** Follow up

**Agenda:**

* Check-in on progress of SPPP doc
* Review Jira workflow
* Divide workload for week 2
* Demo

**Discussion:**

Jira Review:

* User login/authentication is lower priority than database/budget tracker functionality
* GUI and Database primary focus for now
* Input income will probably be easier to implement
* If we have time will work on user login
* Will login information be looped into user’s account -> username, user’s income, etc. or should login just be for username and password?
* Password will probably not be saved to DB
* First name, last name, monthly expense, income etc. will be saved in DB and that’s where data will be rendered back to the web
* Iteration 1: primarily focused on DB and login information

Priority Stories:

* Following SPPP stories (section 8)
* Priority 1
  + Ensure Requirements + Class Diagram all set
  + 1. Create Database - BackEnd
    - Enable trigger dev branch merge
    - Pipeline create container
* Priority 2
  + Implement features one by one
  + 1. Input monthly income
  + 2. User expenses
  + 3. Enable expense tracking
* Must haves:
  + username, password
  + Input for income and expenses
  + Then we have baseline features to create expense categories, create visuals, etc.

Potential split up:

Sherif

* Database script

Clyde

* Database container

Clyde/Sean

* LDAP

Brian/Mali

* Web UI

Sherif/Jisoo/Sean

* Engine

Demo

* Not everyone has to contribute to the first demo, since this is about the requirement Jisoo/Brian will take the lead on this one

**Action Items**

* Complete the demo for project 0 submission
* Review your slide in powerpoint for Demo, make any necessary adjustments

**Meeting 2**

**Date and Time:** 9/9/2023 @7:00PM EST

**Place**: Discord

**Participants:** Clyde Yeung, Brian Fenstermacher, Jisoo Lee, Mali Rivera, Sean Rawson, Sherif Zeyada

**Minutes taker:** Clyde Yeung

**Timekeeper:**

**Purpose:** Follow up

**Agenda:**

* Project details that have been set up/researched since yesterday
* Create a baseline set of features that we'd like to implement in the project
* Ensure that we all have the information we need to complete our portions of the SPPP

**Discussion:**

Reviewed project proposals: <https://docs.google.com/document/d/14l03Mj_ML6cio_Yx3NpVa63a3AQ5Y1jFXbzqMYHWL8I/edit>

General themes in proposals:

* Budgeting application to track monthly expenses
* Visual representation of expenses for user
* Financial recommendations for users

Competitor app/inspiration

* Mint <https://mint.intuit.com/?dd_pm=none&dd_pm_cat=finance_app>
* Mint functionality
  + Links up all accounts (credit, loans, investments)
  + Tracks bills, subscriptions, notifies users when subscription cost increases
  + Allows users to create budget categories
  + Provides suggestions for smarter spending
  + Over-budget, unusual spending, and bill reminder notifications
* Unlikely that we will be able to create an app that links up to bank accounts in 6 weeks
  + Will have the user manually enter income, expenses, etc.
* Want to include functionality for users to create budget categories and set $ limits
  + Visual representation for user:
  + Pie chart break down of user’s expenses
  + Bar tracking expenses -> green in budget, red over-budget
* Want to have a UI component, unsure whether that will be with React or through Java (JFrame)
* Would like to add a financial recommendation feature if we have time
  + For example, could take the user’s income and calculate the suggested amount that they should spend on rent, food, etc. from that
  + 30% rule for rent
  + Average monthly spending on groceries $250-$550
  + 20% towards savings
  + 50/30/20 rule, 20% for savings, 50% for necessities, 30% for discretion

Survey on the tool.

* Plan is for each team member to provide 1-2 user surveys for a budgeting application

Team name: TBD

Features on the budgeting application:

* Monthly income vs Monthly expenses
* Set category
* Set budget for each category
* Finance Recommendation (may be)
* Periodic reports

Reference - everydollar

**Action Items**

Basic features - Brian (ready by noon)

Jira story - Clyde

Decide DB - Sherif

**Next Meeting:**

9/10/2023 @7:00PM

**Meeting 1**

**Date and Time:** 9/8/2023 @6:30PM EST

**Place**: Discord

**Participants:** Clyde Yeung, Brian Fenstermacher, Jisoo Lee, Mali Rivera, Sean Rawson, Sherif Zeyada

**Minutes taker:** Jisoo Lee

**Timekeeper:**

**Purpose:** first meetup

**Agenda:**

Project Title and focus

Role assignment

Coding standard

Meeting schedule/deadlines

**Discussion:**

* Introductions
* Reviewed the documents that are due this week
* Went over role assignment and responsibilities
* Decided that the roles will be flexible - team effort
* Discussed project ideas
  + banking application, some team members work in finance
* project name ideas from chatGPT
  + BudgetWise
  + SpendSmart
  + MoneyMinder
  + ExpenseEase
  + MoneyMate
* Decided to split up the SPPP doc based off of our Roles
* created another shared google doc to brainstorm project ideas and proposals further

**Roles**

Team leader: Brian Fenstermacher

Requirements leader: Jisoo Lee

Design and Implementation leader: Sherif Zeyada

QA leader: Mali Rivera

Configuration and leader: Clyde Yeung

Security leader: Sean Rawson

**Key Decisions:**

Project idea: Banking application

Project name:

**Action Items:**

Coding Standard - Sean and Sherif

Schedule - Thursday after class/Sunday afternoon

Jira - Create Jira project

SPPP - work distribution by each roles

Project scheme work - GoogleDoc (online), meet up Saturday 7pm

Presentation 0 prep - Sunday / Monday evening

**Schedule:**

Thursdays: 8-9pm EST

Sundays: 7-8pm EST

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

**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