**CS673 Software Engineering**

**Team 6 : FitFusion**

**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 3**

**(ppt content integration; practice presentation; time/role allocation/….)**

**Meeting 2**

**Date and Time:** 2024.9.22 3:00PM-4:00PM

**Place**: Online

**Participants:** All team members

**Minutes taker: Haoran Zheng**

**Timekeeper: Haoran Zheng**

**Purpose: Talk about Iteration 0’s PPT**

**Agenda:**

* Design for Iteration 0’s PPT slides
* Plot outline of presentation
* PPT pages distribution
* Assign roles in presentation
* Choose PPT template
* File share

**Meeting 1**

**Date and Time:** 2024.9.20 3:00PM-5:00PM

**Place**: Online

**Participants:** All team members

**Minutes taker: Yunrui Huang**

**Timekeeper: Yunrui Huang**

**Purpose: To finish Iteration 0**

**Agenda:**

* Determine group name
* Determine project name
* Provide effort hours so far
* Assign roles for each team member
* Overview
* Find and discuss related work
* Brainstorm requirements
* Finish SPPP
* Review and discuss
* Discuss risks
* Assign remaining tasks for each members

**Discussions:**

* Determine group name
  + Is this the same as the project name? Yes
* Determine project name and what topics we should do
  + Project name? Yes
  + Topics? Yes
* Provide effort hours so far
  + Members will be assigned tasks to finish
  + Each two or three days we will hold a meetings
* Discuss risk management.

### Finalize Communication Plan

* **Discord: The team will use Discord for daily communication. It allows real-time collaboration among team members, making it easier to discuss progress and issues. Discord's voice and text features help us maintain efficient communication.**
* **GitHub: GitHub will be used as the platform for managing the project's code. The team can collaborate, control versions, and review code on GitHub, ensuring consistency and standardization of the codebase.**
* **Jira: We are using Jira for task management and progress tracking. Jira helps us set task priorities and monitor progress, ensuring the project proceeds according to plan.**

### Find and Discuss Related Works

**We studied several existing fitness tracking projects and referenced two key resources:**

1. **Git Branching Strategies  
   We referred to** [**Git Branching Strategies**](https://www.flagship.io/git-branching-strategies/) **to help design our code management process. This resource provides detailed guidance on managing and merging code efficiently.**
2. **MuscleBook.net  
   We also looked at** [**MuscleBook.net**](https://github.com/cfilipov/MuscleBook.net)**, an open-source fitness tracking system. Its structure and features inspired our own design, particularly in handling weight tracking and data visualization.**

**These references provided valuable insights that helped shape our project development.**

### Brainstorm Requirements

#### **i. Essential Features**

1. **User Registration and Login: Users need to register and log in to manage their fitness data.**
2. **Personalized Fitness Plan: Users can access personalized fitness plans to review past workouts and set new goals.**
3. **Exercise Tracking: Users can track their workout time and intensity to optimize their training schedule.**
4. **Push Notifications: The system will send workout reminders to help users stay on track with their fitness routines.**

#### **ii. Desirable Features**

1. **Video and Illustration Support: Provide video assistance to help users improve their performance.**
2. **Social Interaction: Users can share workout progress and connect with others.**

#### **iii. Optional Features**

1. **Integration with Wearable Devices: Support data transfer from devices like Fitbit and Apple Watch to prevent data loss.**
2. **Workout Suggestions: The system provides personalized workout suggestions based on the user's data.**
3. **Fitness Challenges and Competitions: Users can participate in fitness challenges to motivate themselves and improve their fitness levels.**

### Project Criteria

* **Usefulness: The project will provide users with a complete fitness tracking tool, including registration, login, personalized fitness plans, and push notifications. This helps users manage and monitor their fitness progress.**
* **Complexity: The project will integrate multiple modules, such as video support and wearable device data syncing, ensuring rich functionality and a good user experience.**
* **Originality: This project is designed as a standalone desktop application, ensuring user data privacy and security. Its combination of features, such as BMI calculation, fitness challenges, and social interaction, sets it apart from existing fitness tracking tools.**
* **User-friendliness: The interface will be simple and intuitive, allowing users to easily manage their fitness plans and track their progress.**
* **Scalability: The code will be modular, making it easy to extend in the future. The system will also support cloud integration (e.g., AWS) to accommodate a growing user base.**

### Determine an Approach/Process to Use

**We will adopt a flexible and adaptive development process to ensure the project stays on track and meets its goals. Each phase will have clear objectives, and the team will regularly review progress to ensure the development stays aligned with the project’s vision. Adjustments will be made as necessary to address any new challenges that arise.**

**The project emphasizes innovation and feedback, encouraging team members to bring new ideas forward at every stage to overcome potential challenges.**

### Assign Roles

* **Haoran Zheng - Team Leader**
* **Jiankun Dong - Design and Implementation Leader**
* **Chengqin Li - QA Leader**
* **Yuhan Pan - Requirement Leader**
* **Yunrui Huang - Configuration Leader**
* **Hangqi Wu - Security Leader**

**Each team member has a clear role to ensure the project progresses smoothly and maintains high quality.**

### Key Decisions

1. **Project Name: FitFusion**
   * **Decision: The project name is FitFusion.**
   * **Reason: The name reflects the project’s goal of helping users track and optimize their fitness plans. It is simple, memorable, and aligns with the core functions.**
   * **Impact: A clear, relatable name enhances branding and allows users to quickly grasp the app’s purpose, making promotion and user engagement easier.**
2. **Time Tracking Plan**
   * **Decision: The week starts on Sunday and ends on Saturday. All team members are required to submit time reports by noon on Sunday.**
   * **Reason: A unified time tracking plan ensures that the project progresses on schedule and allows the project leader to receive timely progress updates.**
   * **Impact: Ensures that team members submit progress reports on time, allowing the project leader to adjust task assignments based on the progress.**
3. **Communication and Task Management Plan**
   * **Decision: The team will use Discord for daily communication, GitHub for code management, and Jira for task tracking and project management.**
   * **Reason: Discord is a convenient tool for real-time communication, GitHub offers version control and collaboration for the codebase, and Jira is a powerful project management tool suited for task assignment and progress tracking.**
   * **Impact: This toolset allows for effective communication, streamlined collaboration, and real-time progress tracking, ensuring that the project stays on course.**
4. **Team Role Assignment**
   * **Decision: Assign specific roles to each team member:**
     + **Haoran Zheng - Team Leader**
     + **Jiankun Dong - Design and Implementation Leader**
     + **Chengqin Li - QA Leader**
     + **Yuhan Pan - Requirement Leader**
     + **Yunrui Huang - Configuration Leader**
     + **Hangqi Wu - Security Leader**
   * **Reason: Clear role assignments help increase work efficiency and ensure that each member has a defined responsibility.**
   * **Impact: Each team member is aware of their specific responsibilities and tasks, ensuring smooth project progress and maintaining high quality.**
5. **Development Tools and Languages**
   * **Decision: The project will use JavaScript/TypeScript for the front-end, Node.js for back-end development, Visual Studio Code as the primary development tool, and MongoDB for the database.**
   * **Reason: JavaScript/TypeScript is suitable for both front-end and back-end applications. Node.js provides non-blocking I/O, ideal for handling high concurrency, and MongoDB is well-suited for handling unstructured data.**
   * **Impact: This tech stack will improve development efficiency, ensure project scalability, and effectively manage and store user data.**

**Action Items:**

* **Team leader, Haoran Zhang, assigned tasks to each team member.**
* **Review some basic knowledge of Vue, Nodsjs, MongoDB and API.**
* **Come up with ideas about how to finish template SPPP.**

↓ keep the example for future reference.

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 Keep - 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