Cpts 322: Software Engineering Principles I

Dr. Haipeng Cai

Project Milestone 1

Coding Girls and a Manager

11514680	Madison Jane Holcomb
11611751	Kadupitige Ashi Chamarasinghe
11625004	Sydney Erin Gormley
11628495	Navroop Kaur
11689463	Sonam Yangtso
11701083	Jongyun Kim

February 6, 2020

1. Key Tools

a. Visual Studio Code

It is not necessary to choose a specific text editing tool for group project. Visual Studio Code (aka vscode) is the most popular text editing tool which supports cross-platform as running on various operating systems and includes numerous extensions to develop fast and easy. Vscode is a free and open source program, there is no restriction for use at any project in schools and companies. It supports a basic git interface and aid flexible and fast development through integrated terminals.

b. Chrome and Safari web browser

Web browsers are essentially required to view the outcomes of the web-based application. To check the different rendering results of HTML, we adopt two web browsers, Google Chrome and Apple Safari, with the highest market share.[1] Depending on the type of web engine, some web components may be displayed differently because HTML and CSS tags are interpreted and rendered differently.

c. Git and Gitlab

Git is a basic developing tool to keep tracking version and source code and helps collaborating with team members. We will use Centralized workflow as Git workflow which suit for small project and small group. The Centralized Workflow uses a central repository to serve as the single point-of-entry for all changes to the project. Instead of trunk, the default development branch is called master and all changes are committed into this branch.[2] So, each subgroup will develop a feature in individual branch, then it merges to master when feature completed to implement.

d. Webserver on old smartphone

To run our web application, we need a web server running 24 hours a day. The web server installed on a physical server computer which connected to external network so that it can always provide services to customers. In general, real products require to lease a virtual machine from cloud service firms such like AWS, Azure, GCP, etc. to provide stable and reliable services, but since this is a simulated project, we can use an old smartphone to create a web server that does not cost money but performs powerfully.

2. Technologies

a. JavaScript and React

JavaScript is today's most popular web application development language; it was previously used only to implement dynamic features within Html files. Due to the continuous development of the language, JavaScript has grown to become a major language for developing various kinds of applications. Among the many JavaScript-based front-end frameworks, we chose the most common framework, React[3]. React makes a request to the

[1] https://gs.statcounter.com/browser-market-share

[2] https://www.atlassian.com/git/tutorials/comparing-workflows#centralized-workflow

[3] https://2019.stateofjs.com/front-end-frameworks/

backend to retrieve the data and generates different html files automatically to display appropriate information based on each customer's request.

b. **SQLite**

Databases are essential for trainers, managers, and customers to communicate with each other through applications. The database must be built to manage customer membership information, check trainer schedules, and apply for lessons. SQLite is not as good at handling massive volumes as other large enterprise databases are, but it is integrated with web applications and works quickly and accurately.

3. Process Model *Waterfall process model*

a. **Reasons for choosing process model**:

Our project is a simple web-based gym tracking system, therefore we decided to choose the waterfall process model. Since we are a small team (6 members), and it is many of our first times working on the full software lifecycle, the simplicity and inclusivity of the waterfall model is easiest to follow along with. We will likely use a partially evolutionary process using the waterfall as our base model, since we may need to communicate to get on the same page, but not change anything about our planning or implemented models. Our deployment will also be milestones building up an incomplete project so we can’t use an iterative process. If we are not able to implement a non-linear waterfall, then it’s important to complete each step before moving to the next step for the process flow. Those steps are as follows:

i. Communicate with customers and gym managers to gather requirements for the web-gym tracking system such as different features they need on their web. Communication can be done through meeting them in person and giving them surveys to fill out.

ii. Once every requirement is gathered then start planning what type of a web-based gym tracking system we will be developing. In the planning process, it’s important determine, what type of tools, technologies we might need in order to design the web-design. Furthermore, drawing a flow chart will help to determine what needs to be done first.

iii. Based on planning, design the model for the gym tracking system and create necessary design and requirement documentation for our project. Could also be a small adjustment to design.

iv. Implement the tracking system and then do testing to quality ensure our code as well as quality ensure our project before deployment.

v. Once testing has been implemented, deploy the web-gym tracking system or deploy a partial version of the final web-gym tracking system.

Waterfall’s approach is highly methodical therefore each step has clean transfer of information. In order to get the most out of our project, using a simple model everyone can easily follow along with will be the most efficient.

b. Disadvantages of choosing process model

Since waterfall makes it hard to accommodate changes, one of the disadvantages of choosing waterfall process is that if our client decided to change one of the requirements, it’s hard to go back to update since If a change happens, we must first get through the process and start from the beginning. Biggest limitation is exactly that, all five steps of the cycle must be complete, and if you are working on something that you discover needs a major change to components of the model, then you have to wait for coding, testing, and deployment before going back to make edits to fix the feature that had been overlooked in planning for that iteration of the waterfall process. Another issue is that testing can’t be done until coding is finished, when there are some cases it is beneficial to write the test cases with the code, or even base your code around some test cases.

The other main disadvantage is that many process models have been made based off the waterfall process. So, by using a simple process model we could potentially miss out on details or have a project process that isn’t as efficient as it could be. There are models we considered, such as spiral, but having a short time frame and limited resources we decided that sticking to something we can always remember would work best for us as a group.