Group #4

Christopher Juncker

Justin Greever

Samantha Zeigler

Tori Anderson

Naya Mairena

Ian Guy

Dan Jang

Term Project: *ChocAn*

Project Report

Retrospective

Table of Contents

Project Report Introduction: **2**

Slideshow Video: **5**

Bonus Website: **6**

Chocoholics Anonymous Group Four Website 6

Project Report Introduction:

Our group project focused on key components of software development. Various elements of the modified waterfall software development model were used, including the drafting & submission of Requirements, Design, and Test Plan documents. Our group collaborated through a *GitHub* repository. Visual Studio Code was our primary IDE used, with a *GitHub* extension for useful & up-to-date updating and collaboration that was based on commits & differential branches for version & individual development environments.

For our most project highlights, firstly, we found that *GitHub* was wonderful in ensuring that everyone in our group was able to access & have live updates to the repository, thus, our software project. *GitHub*'s commit system was able to also ensure that edits to the same set of files were not constantly overwritten or any issues with disjunct-structured development since our Group mostly stayed remotely for a majority of our development.

Secondly, in regards to the remote nature of our development, by utilizing remote meeting platforms, our group was able to successfully perform development - jointly, whenever possible through scheduled team meetings. Furthermore, since we used *GitHub* for our development of the software, we were able to ensure, or at least avoid, collisions or overriding of written or to-be-written code, as aforementioned.

Thirdly, our Group's overall socially cohesive synergy was very positive & wholesome, even at the initial beginning of our software development journey. Being able to choose & match our fellow friends enabled us to skip the social icebreaking & get right on down to the nitty-gritty: Our software development.

In regards to some of the *not-so-highlights* of our software development process was the issue with asynchrony in general. Specifically (and firstly), we noticed at times, a difficulty in the overall niche development of some components of the software suite. Perhaps with issues with the individual development environments (e.g. *Visual Studio Code & PyCharm*), the configurations of these IDEs and development tools caused some initial confusion and subtle delays in our software development process.

Secondly, with regards to the overall design schema & functionalities of our software suite, niche but specific details, that were in development or development-to-begin, initially caused some sense of disarray, which may simply be due to the large nature of the Requirements & Design documentation in general.

Thirdly, we had a slight issue with the advent of using *Python* for our main project software coding language. Initially, we were in the debate about using perhaps *C++* or *Java* for our coding base. However, with *Python*'s overall simplicity in its syntax, compilation efficiency & efficacy (with no need to use, e.g. *g++*, as with *C++*), we ultimately decided to stick with *Python*. Thus, with *Python* - with a majority of members being unfamiliar or somewhat unease with programming in a new language.

Suggestions to address these issues would be first, to increase the level of online connectivity & overall communication. Such tools to help in these aspects would be to use instantaneous messaging and completely disallow email usage whatsoever; our Group did not use email at any time to communicate, where such communication methods would be a bottleneck for overall team responsivity as well as hinder the level of cohesive and intuitive synergetic development-flow of our Group. Overall importance to the fastest, most reliable, and time-efficient communication methods should be maintained & a must for software development teams, in our opinion.

Secondly, to use organizational tools such as *Trello* or *Notion* to help structure - concretely & accurately - any niche or structural details as well as key functionality aspects, such as how the database may communicate with specific tables or types of data within the database & its tables if SQL-based perhaps.

Thirdly, by utilizing more resource-gathering techniques for platforms and tools, such as for *GitHub*, *Git* overall, *Python* debugging, or perhaps quick-start guides for *Visual Studio Code* or other onboarding-brainstorming, much of initial confusion with the aforementioned software development tools, suites, and IDEs may have been prevented. This may also tie into an overall team-level of synergy that must be constantly maintained and promoted by each member of the Group.

Slideshow Video:

The ChocAn Healthcare Software Suite, v. 3.00 (Q4, 2021)

by Group 4 Software Solutions Inc.

https://www.youtube.com/watch?v=Zd1msfgt8WE

Bonus Website:

Chocoholics Anonymous

https://chocoholics.xyz