

CPSC 233 WINTER 2019

UCID: 30063099

Instructor: Sohaib Bajwa

PROJECT REFLECTION

By
HAOHU SHEN

April 7, 2019

CPSC Self Reflection

My own approach to learning

what worked

- I learnt basic knowledge of SQL syntax before taking advanced courses like [CPSC471](#) since there are lots of free learning materials available out there, such as [SQLite Tutorial](#) or [w3schools](#)
- I learnt that the unit test plays a really important role during the whole life cycle of the software development, thus how to construct a set of reasonable test samples that can cover all corner cases would also be a challenge for developers. In some cases, the time that developers spent on thinking about the samples would be more than 50% of the whole project.
- Once again the challenges I encountered in the project prove that "[No Silver Bullet](#)" in Software Engineering. In our project, we not only use SQL to store data and JavaFX for GUI design, but also use CSS to decorate the interfaces, regular expressions for validation. In conclusion, in some occasions we must combine different programming languages, frames or strategies to solve a specific problem.

what didn't work

- I tried to implement a function that simulates "Press any key to continue" in Java, after I checked the relative information in stackoverflow, I figured out the implementations are different in different operating systems. In Linux, for example, it is impossible to use pure Java to implement the function, instead, you need to invoke a Linux IO library in C or C++, and connect Java and C/C++ by [JNI](#). I failed in linking the shared library which is the last step and still do not know how to fix it.
- Testing GUI with Junit is different from that in text-based mode. A better method is using third-party libraries with JUnit, such as [FXTest](#), but I didn't use it because we had no enough time for it and we need to use Gradle to initialize it and learn its API.

My effectiveness in the team

What did I do that helped the team's success?

- Finished the implementations of APIs that connects our Java code and SQLite3.

- Finished almost all components of test-based and GUI.
- Finished Junit test samples of several important classes, such as Database.java and Validation.java.
- Kept track of the update of the repo, including merging and pushing code to github on time.

What could I have improved?

- Since someone mentioned SQL Injection in Discord, I checked some best practice about how to defend it, then I reconstructed the original API with parameterized SQL statements.
- I noticed in Form-based application, no matter in a client or in a browser, an algorithm will be used to assess the robustness of the user's passphrase, thus I tried to create a set of rules about it, which can be found in the javadoc in Utility.java.

What did I see others do that really helped my team?

I found I contributed too much on this project, like more than 80% of it, I think the problem is that I did not explain well enough about the SQL concepts to them at the first place, which made them cannot keep up the pace, or sort of lost motivation of discovering more of it. On the other hand, in the later stages of the project, most team members were busying other courses' exams or assignments, which actually shows a poor skill of time management. Although I tried to allocate some easy jobs to some team members, I need to spend time on re-checking their work. But Alex did well in the design of web spider, which obtained a collection of juicy data and made our project more practical.

Tech-stacks

What technical skills did I acquire?

- SQL Operations on simple transactions.
- Debugging on different background, locally or remotely.
- How to use Vim editor to quickly edit the source code and database files.
- How to construct regular expressions.
- How to reduce and optimize Java code by using lambda functions.
- How to use Java reflection and Run-Time Type Identification(RTTI), observers.
- How to organize code and update, including operations of Github.

How do you see yourself applying this in future endeavors?

- After this course, I would say I am confident to take CPSC471, maybe I have to use another relational database management, such as MySQL or MsSQL, which the concept of concurrency and the way to defend Ddos attack will be introduced, but the essence is still focusing on the interaction of data, and the

policies to validate the integrity of it.

- Next time, when I need to add a Java plugin to my project, I will use package managers, such as Gradle, instead of downloading it as a separated Jar package.