INF221 Project

Architecture-Based Development with Java 9

Java has recently introduced Java Platform Module System (JPMS) in its 9th version. Using Java module system, the developer explicitly specifies the system’s components (i.e., modules in Java) as well as the specific nature of their dependencies, in a file called *module-info*. In this project you first need to understand and recover the architecture of a pre-Java 9 system and then migrate it to Java 9 modular platform. This migration needs proper choice of packaging and modularization, defining precise dependencies among the modules. Modularizing Java systems would make easier for software architects to recover the system’s architectural styles. Hence, after the migration you need to manually recover the architectural styles of your migrated system and describe them in the project report. The detailed description of first phase of this project is as follows.

**1st phase: Due by Nov 19th**

* Choose a Java software system among the [list](https://docs.google.com/spreadsheets/d/1u0qicSoVwW6ueMe0wPwcms2Bpd7fCqjXyy5gs-tDsYM/edit?usp=sharing) of suggested Java systems on GitHub. You can also pick an open-source Java project of your own choice, but it has to conform with the following constraints:
	+ At least 100K LOC
	+ At least 50 classes
	+ Needs to have test cases to validate its key functionalities
	+ Not developed by you: was developed by others and is used by thousands of people. Publicly available, open source.
	+ Must be unique in class: each group has their own program — no two groups may share the same system.
	+ Must be recognizable, well-used, and popular project

In this case, first email negargh@uci.edu the link to the GitHub repository of your project and if the project is approved, you can proceed with the next steps.

* Enter your group members in the [Google sheet](https://docs.google.com/spreadsheets/d/1u0qicSoVwW6ueMe0wPwcms2Bpd7fCqjXyy5gs-tDsYM/edit?usp=sharing) in front of the corresponding project (or add your project and group members after the approval)

**Implementation**

* First, study the source code, comments, and any documentation available for the system to understand and recover its architecture as implemented
* Define appropriate modules according to the system’s architecture (for both the code and test cases)
* Migrate the system to Java 9
	+ Fix the possible problems
		- Explain the problem, and how you fixed it
	+ Compile and run the system with JDK 9
	+ Run the system’s test cases and report their result and coverage
	+ Make your implementation available on GitHub (ideally by forking the repository where the system is hosted)

**Analysis**

Write a report and

* Compare the system before and after your architectural modification
	+ Visualize/model the architecture of the system before and after the migration
	+ Briefly explain the modules (i.e., their role and functionality) and the reasons you defined them
	+ Describe any changes you made to the architecture of the system as a result of migration or possibly as a result of detecting issues with the original architecture of the system
	+ Manually recover and describe architectural styles of the system after migrating to Java 9
* Explain the test results before and after the migration, and any related challenges

Teams are highly encouraged to create a pull request for the project’s GitHub repository with your implementation.

**2nd Phase:**

The details of this phase will be announced later.