

**De La Salle University • College of Computer Studies**

**SystemScape**

**Platform and Language**

Name (last name first) : Angeles, Antonio Mariano

Coquilla, Philip Bryan

Cote, Christian Gabriel

Fernandez, Ryan Austin

Poblete, Clarisse Felicia

Quindoza, Rissa Marie Grace

Salceda, Juan Francesco

Tan, Shayane

Uy, Mervin Stewart

Velez, Gio Anton

Section : S19A

Date of Submission : February 9, 2015

**Platform and Language**

We, the developers of Team SystemScape, propose the use of the Java programming language and Java Virtual Machine (JVM) as platform for developing our project entitled “CAI-STA Purchasing and Inventory Records System”

The justification for such is as follows:

Java Virtual Machine platform has been around for a while and has been tested and deployed in many industries. There is a large community of companies and developers who are continually working to improve its features and implementation quality. There are many books / online resources / tutorials on how to develop on it. In addition to this, the client will be running the application on a computer running a Windows operating system.

Java language has a huge amount of documentation available. It is platform independent and has an incredibly rich standard library and plenty of third-party libraries as well. Automatic Memory Management implemented is by g[arbage collection](http://c2.com/cgi/wiki?GarbageCollection) and n[o explicit pointers.](http://c2.com/cgi/wiki?NoExplicitPointers) Also, the syntax is familiar to the myriad programmers that know any other C based language. Of course, there are plenty of Java IDEs available. Java version changes are released at a slower pace. It is also backwards compatible. Like C# and C++, Java is object-oriented.

Below is the proficiency level of each member of our team:

|  |  |  |
| --- | --- | --- |
| Team Member | Proficiency Level | Remarks (for the proficiency level) |
| Phillip Bryan Coquilla |  |  |
| Christian Gabriel Cote |  |  |
| Juan Francesco Salceda |  |  |
| Ryan Austin Fernandez |  |  |
| Shayane Tan |  |  |
| Rissa Marie Quindoza |  |  |
| Gio Anton Velez |  |  |
| Clarisse Felicia Poblete |  |  |
| Mervin Uy |  |  |
| Antonio Angeles |  |  |

**Training Plan**

In order to achieve a proficiency level of at least “moderate” for all team members to help us perform our project, we propose the following training plan:

Each member of the team will be given specially crafted mini-design challenges that will vary depending on each member’s role in the team. These design challenges will test everyone’s capabilities in Object Oriented Programming Concepts (OOP), Application Programming Interface (API) usage, and Graphical User Interface programming skills (GUI) in the Java language.

Developers will be tested on OOP and SW Design i.e. SOLID Principles to prepare themselves for heavy design work. Less focus will be given on graphical user interface design. In the model-view-controller architecture, more focus will be given to the model and controller modules.

Analysts are expected to be proficient in the API for the Java Swing elements. In the model-view-controller architecture, more focus will be given to the view module. As such, some appropriate design patterns might apply e.g. Builder, Factory.

Quality Assurance / Testers are expected to have sufficient knowledge in all aspects, for they will be responsible for testing each component of the system.

For members who aren’t proficient enough in the said field of expertise, training sessions outside of class hours will be given by the developers.

**References:**

Lam, M. (2007, April 17). Why choose Java? Retrieved February 8, 2015, from [https://weblogs](https://weblogs.java.net/blog/mlam/archive/2007/04/why_choose_java.html)

.[java.net/blog/mlam/archive/2007/04/why\_choose\_java.html](https://weblogs.java.net/blog/mlam/archive/2007/04/why_choose_java.html)

Java Pros And Cons. (n.d.). Retrieved February 8, 2015, from http://c2.com/cgi/wiki?JavaProsAndCons