Alejandro Reyes Bautista – Q13134663

Agile Development Presentation: SCRIPT

Slide 1:

This presentation is about how Version Control is being used in the management of Athletics club Demonstration Program.

The VCS or Version Control System used is GIT. The Version Control Repository is GitHub and we actually use two clients; on one hand we have SourceTree, by Atlassian, for track analysis, design modelling and associated documentation, and on the other hand we have Netbeans with a plugin for Git, for track each version of software and work moving between branches in an easier manner.

Slide 2:

But, why do we use a Version Control System? The advantages are huge, but in the context of the project we can highlight 5 points.

First, we can make a change to code, to realize there was a mistake and revert back.

Second, we can review the story of diagrams, use cases text, code, etc. and see the difference between two (or more) versions. GitHub highlight the changes, uses green to mark added contain and red for deleted.

Third, we let other teammates to work in our code. If I am working in a piece of code, and get struggle, or I am not able to find a bug, a teammate which more experience in that area only have to pull the project, have a look and fix the minor details, pushing the changes into the branch to finalize the process.

Fourth, Git allow us to track how much work is being done, and where, when and by whom.

And last, we can develop/experiment with a new feature without interfering with working code. The Product Owner has always available the last release while the team is working in the next Sprint.

Slide 3:

This is an example of review the story of the diagrams. We can see distinct versions of the Use Cases Diagram for Event. Each version has attached the commit message that the author has written explaining the changes introduced. So we can appreciate by whom, where and when has the new commit being done. In the same manner, at the bottom we can see the differences between the old versions of the diagram and the most recent.

Slide 4:

This is another example of review the story, now about use cases text. Same as before, we can see that the main advantage is to have the distinct versions of the same document, with the description within the commit about what have changed, the date and by whom. We keep only one file, not all the versions of the same file. The versions can be seen in the story, and you can revert back if the last updated was not accuracy.

Slide 5:

Next step is to explain the branches which compose our repository.

First we have the master, where the entire project is going to be at the end of the development. This branch is splitted up in two. Design analysis documentation is the branch where the use cases, robustness diagrams, sequence diagrams, database schemes, considerations, decisions and the rest of documentation is stored. Application, on the other hand, stores the source of the java project. This last branch is also splitted up in three. Training branch keep track of the java code implementing features related with trainings, event keep track of the java code for events, and memberships the same for memberships.

At the bottom we can see a picture with all the branches, being master the default branch and being able to initiate a new pull request just with a click.

Slide 6:

When a new feature is being developed, the code is only in the concrete branch, and only when the entire feature has been developed and tested, and not before, the branch is merged with the application branch. This procedure allow us to work in distinct features at the same time without alter the working code. If there are any problem, we revert back the application branch to a previous state while we work on solve the conflict. The SourceTree client allow us to see in an easy manner the dependencies and differences between the branches, the last update, the description and who made it.