SoftRUNNER - Weekly Progress Report

Since we are a small group and close to each other, we do not share the work, but do everything together. In this week, we have concentrated on the programming language we will use. But in order to do this, at least we should have an opinion about the features of our system. So we worked on the data description and the functions of the system. With the data collected about our system (they are not certain yet, only the main borders are fixed), we searched the internet for the programming languages; which will be more useful and easy to implement, etc. And also this will not only help us with the programming language, but with analysis report. The discussions about the properties, specifications and the features will also ease to form our scope in analysis report.

In addition to these technical issues, we have got a contact with a project manager in Siemens, and he accepted to be our advisor. And we have fixed an appointment time which is at the 1st of November.

The data description, functions and the results of our search are as follows:

Introduction to Data Descriptions

As I mentioned above the tables of our database are not certain yet but we think that the main tables will be as follows:

Tbl_User: All the information about all users subscribed to our system will be hold with this table. Besides personal information, this table contains username, system password, the projects he/she works on, etc.

Tbl_Resource: The resources will be hold with this table. These resources may be materials or people. If it is a person, that entry will be related with our user table Tbl_User. Also this table will hold Project_id which will enable us to relate with projects.

Tbl_Group: With this table the groups of a project will be defined. The tasks will be assigned to groups to reduce the redundancy, so for any combination of people which will work on same task, there should be a group defined. This table will contain project information to be related with projects. And also all groups will contain a manager which will be hold with him/her User_id in the table.

Tbl_Task: This table will hold task information such as beginning date, duration, name, group works on this task, etc. It will be related with groups so at the same time with the projects.

Tbl_Project: All the projects and their information such as project name, beginning date, duration, id, etc. will be hold with this table. Most of the other tables will be related with this project table with its id.

Fuctions

General functions of our system are add/delete/update options for all of our tables such as add/delete/update task, group, project, resource, etc. Also there will be Gantt Chart function which will draw the Gantt Chart of the project with the information of tasks. Of course there will be other specific functions, but they are not certain yet. All functions will be fixed with our design report.

In addition to these general functions, we will have system functions such as log-in, sign up a new user or sending the password via e-mail, etc. But the specifications of these features are not properly discussed yet.

Comparison Between Java and C#

We made a research through internet and tutorials to decide which language to use in our project, and compared C# with Java. The main differences and advantages of both languages are stated below.

Stating the obvious, Visual C# was designed to work with Windows. Therefore it takes advantage of the .Net platform and other Windows controls. However C# as a generic programming language is not dependent on Windows. It just so happens that Windows was the first platform to run on.

Because Java is platform independent, it relies on basic graphic user interfaces. Although a class library, called Swing was developed to compete with Windows controls, writing GUI interfaces in Java can be long and tedious. Not exactly ideal for Rapid Application Development type work, an area where Visual Basic excels very well.

Features like operator overloading, pointers, preprocessor directives, delegates and deterministic object cleanup make C# more expressive than Java in a number of cases. There

are also features that are missing in Java, such as boxing, enumerations and pass by reference in C#. On the other hand the lack of checked exceptions, inner classes, cross platform portability or the fact that a class is not the smallest unit of distribution of code makes the choice of C# over Java not a clearcut case of choosing more language features without having to make any compromises.

C# contains a simple mechanism for marking regions of code as unsafe. Within these unsafe regions, the safeguards put in place by Java and later C# to prevent programmers from directly altering memory locations and using point arithmetic are suspect. Dropping to the memory address level when working with a garbage-collected high-level language muddies the intentional object/memory separation .Bugs can slip in easily, debugging is a nightmare, and buffer overruns once again rear their ugly heads to open the security holes for which C and C++ are famous.

Java has better tool support even when integrated development environments (IDEs) are taken into account. Visual Studio .NET is a fine IDE. It represents years of work and is feature rich. However, the Eclipse IDE, which includes Java support, has pulled ahead of Visual Studio in stability, ease of use, and offered features.

GUI development in C# is a typical example of RAD, like Delphi. In Java the GUI is usually done with Swing. Even though Swing is quite large and complex, the fact that the architecture is well thought out and scalable combined with the availability of the source code allows for development of GUI of any complexity.

ASP.NET has some minor advantages when it comes to creating small scale web sites .However these advantages separate web-applications into a distinct segment of software. Many developers specialize in either web development or GUI development. In Java web-applications can easily be created without the use of special technologies like JSP, JSTL or Struts. All you have to say is extends HttpServlet.

If you search the internet for comparisons of Java against other languages, you will see many advantages Java has. Cross platform portability, ease of development, garbage collection, a strong security model, and much more. But the most important feature is that Java is by far the fastest computer language ever invented. Java is hundreds of times faster than any other language. This is hugely simple to demonstrate, with a test anyone can perform on any

machine. In any language, simply time how long it takes to run a simple empty loop for a very large number of iterations. You can't get a simpler test than that! Here are the results for a few languages, using a loop iteration count of 10 000 billion iterations:

Language	Time taken to run a simple empty loop test for 10 000 billion iterations
Java	Under one second
Perl	approx. 1 month
C/C++	approx. 1 month
C#	approx. 1 month
Assembler	approx. 1 month
Fortran	approx. 1 month
Ada	approx. 1 month
Basic (e.g. Visual Basic)	approx. 1 month
You name it	approx. 1 month or greater
Other languages	approx. 1 month or greater

Main advantages of both languages are:

JAVA

- Cross platform
- Maturity
- Better IDE's
- Established Practices

.NET

- ASP.NET

- The power and flexibility offered by the the advanced features and functionality of ASP.NET is much better then anything on the Java side of the world.

- C# is a better Java

- C# took many of the ideas and concepts of Java and made them better.

- More innovation
- Language neutrality

- The .NET Framework allows us to write our .NET applications in whatever language we choose.

- Web Services

To sum up both of the languages have advantages, we are still not definite about which language to choose for our project implementation. It seems java to be more convenient especially because it is platform independent but our members are a bit more experienced in programming with C#. In analysis and design phases our choice will be more accurate. We will make more researches on other programming languages and come to a decision.