Project title:

Distributed System for ‘MSc Properties’.

Aim of the project:

For this project I aim to develop a distributed system for ‘MSc Properties’, to enable them to create and manage their properties, customer accounts, tenancies, rent accounts and employee accounts, and enable them to report on this data stored in a database. I will also develop a web site that allows ‘MSc Properties’ to advertise their property portfolio to the public and allow customers to register themselves and submit their interest for a property or submit a repair request as an existing tenant. As an addition there could be a facility on the website for tenants to pay their rent or deposit for a property.

Introduction to the project:

‘MSc Properties’ is an estate agent with a number of sites nationwide across England. Due to recent legislation changes resulting in local councils being able to house homeless families outside of the local borough and welfare caps in benefits meaning families are having to move out of their local borough due to not being able to afford local rents, ‘MSc Properties’ has required the need to be able to transfer customers from one site to another site, meaning the transfer of data across sites that could be 100’s of miles apart. For my MSc dissertation I am going to develop a distributed system that allows employees of ‘MSc Properties’ to create and manage the property portfolio and the customer accounts for the business, as well as creating and managing both tenancies and rent accounts for ‘MSc Properties’, meaning that data can be stored on a server or locally and all the different sites of ‘MSc Properties’ will be able to access this data. The system will also allow managers of the ‘MSc Properties’ sites to manage their employee accounts from their own site, and will have a login facility, which will work across all the different sites of ‘MSc Properties’.

Word Count = 200

Methods or methodology of the project:

To assist me in designing the distributed system, I’m going to use the unified modelling language (UML) to develop diagrams which will allow me to virtualize the design of the structure, behaviour and interaction of the distributed system. When designing the distributed system I’m going to use programming techniques to ensure my program has high cohesion, strong encapsulation, and low coupling. This means the components in my program will be designed to match a well-defined idea, the related components in my program will be kept together in one place and the encapsulated component will be insulated from the outside world using techniques such as interfaces and abstract classes, and lastly components will be as less dependent on each other as possible.

For the implementation of the distributed system, I will use more programming techniques such as sockets or remote method invocation (RMI), this will allow me to develop a distributed system that can communicate between hosts and a server on the same or

different networks. I will also use Java GUI frameworks such as AWT and Swing to develop the Java interface. I will also use JDBC to provide a common interface (API) to the MySQL database, allowing the distributed system to interact with the database. During the development I will use a concurrent version system to track the evolution of files in the development, allowing me to keep track of updates made and a rollback facility to go back to a previous working version if necessary.

For the testing of my program source code I am going to use a number of testing strategies to ensure my program developed has no faults or errors at runtime and the program works as set out by the requirements. I am going to do unit testing using the white box testing strategy as this will allow me to test individual components as a single to ensure each of these work alone and find defects with my program source code as early as possible in the development. White box testing will mean the test script developers know how the source code is written so will mean they can write strict tests that the units must pass using boundary values to test the system and get the best results. I am also going to do integration testing and system testing using the black box testing strategy as this will allow me to take the individual components of my program tested in the unit tests and test them as a group to ensure the components of my program work together as intended. Black box testing will mean the test script developers do not know how the source code is written so test scripts will be developed on the basis of the project requirements and therefore if the programmer has got the logic incorrect but the system works, these tests will identify if the system conforms to the actual requirements. To allow me to control the execution of my tests and compare actual outcomes against predicted outcomes, I am going to use a bug tracking and testing tool.

Word count = 512

Objectives of the project:

178 words left for this section

Bibliography

http://en.wikipedia.org/wiki/Unified_Modeling_Language