**Name: Susmitha D/O Appalanaidu**

**Admin Number: 1505544E**

**Subject: Software Engineering**

**Definition and phases of software development life cycle (SDLC)**

Software development life cycle is basically what goes on from the start of project to deployment of software system. It is a process used by the software industry to design, develop and test high quality softwares. The aim of this cycle is to make sure high-quality softwares are produced which meets or exceeds customer expectations and reaches completion within a given time and a cost estimate. There are a few phases of a SDLC. The phases are, requirements engineering, analysis and design, implementation which involves coding, testing and deployment.

The requirements engineering phase is where information such as what the client/customer wants the software to do will be gathered. This is the most important and fundamental stage in SDLC as the information gathered in this stage will be used to plan the basic project approach. This is the phase where Mr Wang will state what are the tasks he require the hotel management system to carry out. A pitfall of this phase is it is difficult to specify the information precisely as the clients/customers tend to change their minds and tend to have different perceptions. Different stakeholders might have different requirements.

The analysis phase is where texts and diagrammatic forms are combined and used to depict requirements. Through this, the clients/customers will get a clearer idea of what the end product will be like. The design phase is where a representation of an entity that will be built later is produced. A pitfall of this phase is the representation produced might not meet the requirements. To avoid that, requirements should be noted down properly, the designers must meet up with the clients/customers clarify with them, create a prototype and also get the clients/customers to sign on a document that clearly states the requirements.

The implementation phase is where detailed designs are converted into instructions written in the programming language. Different high level programing languages such as C, C++, Java and PHP will be used for coding. The programming language is chosen with respect to the type of software that is being developed. A pitfall of this phase is different developers will be involved therefore, there will be a need to integrate different codes.

And lastly, the testing phase is where the software is tested to ensure that it is reliable and meets users’ needs and the deployment phase is where the application is distributed among a group of selected clients/customers prior to official release and later delivered to clients/customers. This is where Mr Wang and users try out some parts of the hotel management system and give some feedbacks. A pitfall of the testing phase is there might be insufficient testing done and a pitfall of the deployment phase is there might be a mismatch between development and the production environment.

**Models**

* **Prototyping**

A prototype is a model developed based on requirements that are known at the period of time. A prototype is developed instead of a full-fledged system. Prototypes are usually used to clarify requirements. A throwaway prototype is the initial specifications for the software which does not meet the requirements properly. After knowing what can be improved, the throwaway prototype is thrown away and an improved prototype is built. An advantage of this model is that users are actively involved in the development and a disadvantage is that this method might increase the complexity of the system as the scope may expand beyond the requirements that was planned originally.

* **Unified process**

A unified process is an iterative and incremental software development process framework. It is not a single process model. Each phase consists of one or more iterations and each iteration consists of analysis, design, coding and testing. An advantage of this model is the development time required is less due to reuse of components and a disadvantage is the developers need to be expert in their field to develop a software under this methodology.

* **Agile**

An agile development model is an incremental and iterative model. Software is developed in incremental, rapid cycles which results in small incremental releases with each release building on previous functionality. Each release is thoroughly tested to make sure software quality is maintained. An advantage of this model is there will be regular adaptation to changing circumstances such as requirements and a disadvantage is the project can be easily taken off track if the client is not clear of what is the final outcome they want.

**References**

<https://www.tutorialspoint.com/sdlc/sdlc_overview.htm>

<http://istqbexamcertification.com/what-is-prototype-model-advantages-disadvantages-and-when-to-use-it/>

<http://www.my-project-management-expert.com/the-advantages-and-disadvantages-of-rup-software-development.html>

<http://istqbexamcertification.com/what-is-agile-model-advantages-disadvantages-and-when-to-use-it/>