Thien Yun Kai

1502868A

**Question 1**

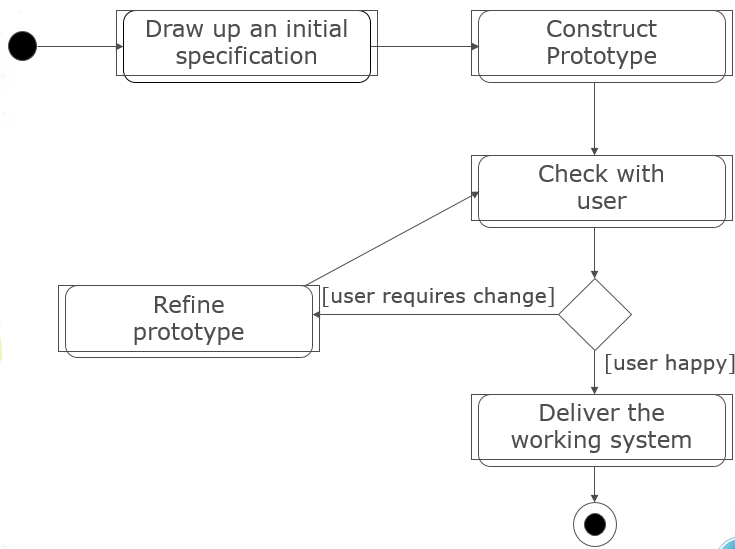
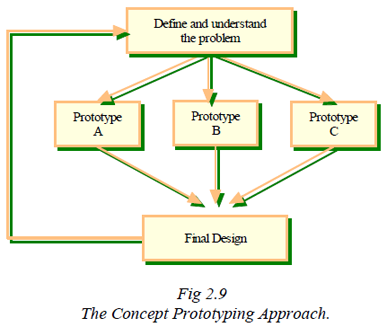
Software Development Life Cycle (SDLC) is used to describe a process for 5 phases, namely Requirements engineering, Analysis & Design, Implementation, Testing, and Deployment. It is essentially a series of phases that provides a model for the development and the life cycle management of a software from the start to its release.

* Requirements Engineering is to find out what the client or customer wants the software to do and achieve. It must be specific and cover every part of the software.
* Analysis is to analyse and use a combination of text and diagrammatic forms to depict requirements specifically. Design is to produce a representation of an entity that will be developed and built on, such as a form of layout. This includes architectural design, user interface design, and database design for example.
* Implementation refers to detailed designs converted into instructions written in the programming language such as Java to make a working software. This means the real code and backend of a software will be written here, along with the designs and interfaces of it.
* Testing is a phase in which the team will test out the software to ensure that it is functioning, reliable, and meet the client’s needs and requirements.
* Deployment refers to the application being distributed among a group of selected customers before release (alpha, beta versions) or other customers. This is the final stage of the SDLC.

Software development model is a plan of what steps to take to develop a project in making a software. It can be used to predict what will be done as well as analyse the current development process and make adjustments.



The first software development model is the traditional waterfall approach, which is a sequential design process. It is when progress is seen as a waterfall, flowing steadily downwards through the phases which are Requirements, Analysis/ Design, Coding, Testing, and Development. It can be used to divide complex tasks into smaller and hence more manageable tasks since they are being broken down. In addition, it helps the team identify the deliverables for each task and is easy to control and monitor as it deals with one activity at a time. However, one can only see the product at the end for the client as there is no opportunity to validate user requirements at the early stage of development. Also, a problem might not be able to be solved if it is discovered at an early stage as it can only be done at a later stage. Last but not least, it does not stress the need for anticipating changes.



Throwaway Prototyping Evolutionary Prototyping

The second software development model is Prototyping, where instead of developing a full-fledged system immediately, the software engineers start with a prototype which is a simplified version of a software. Its purpose is to clarify the requirements and can be used to design user interfaces, demonstrate feasibility, and verify that the new technology will work for the project. There are 2 types of Prototyping, the throwaway prototyping and the Evolutionary Prototyping. Throwaway prototyping is to verify the requirements that are unclear so that both the clients and the software engineers will know exactly what the requirements are. Evolutionary prototyping is to uncover unknown requirements to be built and evolved on and it is especially useful when requirements are hard to specify.



The third software development model is the Unified Process (UP) which is a framework for deriving a specific process model for the project. UP uses an iterative layout for different phases such as Business modelling, Requirements, Analysis & Design, Implementation, Test, Deployment. Each iteration will consist of these and it will continue its iteration until the software is completed.

The phases of the SDLC can be used in the project so that we can have a clearer understanding of which stage we are at as well as predict what will be done as well as analyse the current development process and make adjustments throughout the project when necessary.

The project will also use Evolutionary Prototyping so we can constantly improve and work on the website and improve it based on customers and user’s opinion. This also helps us have a better understanding throughout the project and save the business cost as problems are being solved throughout the stages as the website is being developed.

**References**

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

<https://en.wikipedia.org/wiki/Waterfall_model>

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

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

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