ENPM808X Rohith Jayarajan
Week 3 Student ID: 115458437

Software Engineering

**5.1 How are software changes classified by their purpose? What is the most common purpose of the change?**

Software changes are classified by their purpose in the following categories:

- *Perfective changes:* Perfective changes are the most common changes in the list of software changes categorized by purpose. These changes are used to introduce new functionality and to increase the value of the software.
- *Adaptive changes:* These are the changes made to help adapt software to new circumstances within which the software operates.
- *Corrective changes:* Corrective changes are used to correct software bugs and malfunctions.
- *Protective changes:* Protective changes are those changes that are invisible to the user and they also shield the software and its value in a proactive way.

From the point of view of software changes by purpose, ***perfective changes*** are the most common.

**5.3 When is it permissible to do quick-fix changes?**

The only acceptable circumstances to do a quick fix change during the evolution stage is in the situation of an emergency where human life or a substantial value is at stake. Therefore the fix has to be done quickly and the speed of the fix outweighs every other consideration.

During servicing stage, quick fix is a common strategy of change as the fixes just keep it afloat.

**5.5 What is a product backlog?**

Usually there is a whole set of requirements that need to be managed by programmers, and this set of requirements is stored in what is called a product backlog which is also known as a requirements database and sometimes called as a project wish as it lists desired future product properties and functions.

A shared vision of the project stakeholders for the future of the product is described in the product backlog. The product backlog is created and incremented in a process of requirements elicitations. The stakeholders contribute some requirements by their own initiative.

**6.6 Describe a situation when a grep search fails. What would you do if this happened to you?**

Rohith Jayarajan
Student ID: 115458437

grep often fails in a search for implicit concepts; their names usually don’t appear in the code as there is no code, identifier, or comment that indicates the presence of the concept extension. In such cases where it fails, using other concept location techniques is recommended.