Lab 1: Choosing Software Process Models

Akshar Panchani ID- 202101522

IT314 Software Engineering

7/31/23

Lab 1: Choosing Software Process Models

The answers are as follow.

a) A simple data processing project.

- **Waterfall Model**. As we can clearly understand that the requirements are already known, further the data processing project doesn't need to be changed and is of short duration. So we can depict this.

b) A data entry system for office staff who have never used computers before. The user interface and user-friendliness are extremely important.

- **Prototyping model.** It is mentioned that the UI is very much important for user. It needs to be dynamic in order to have new features and new technologies to make this user friendly, so having this model is a suitable fit for this software.

c) A spreadsheet system that has some basic features and many other desirable features that use these basic features.

- **Prototyping model.** Here we can see that the spreadsheet need to be updated with new features and should be dynamic, also with this the old features should be as it is, hence we can use this model.

d) A web-based system for a new business where requirements are changing fast and where an in-house development team is available for all aspects of the project.

- **Incremental Model.** Web-based system is new and its requirements are changing fast moreover we do have a team to work on it this would be very helpful. Also as it is emerging business it may have not cleared requirements.

e) A Web-site for an on-line store which has a long list of desired features it wants to add, and it wants a new release with new features to be done very frequently.

-**Incremental Model.** This requires good planning with many builds constantly. Also it requires a well defined model for this and should be enabled with new technology.

f) A system to control anti-lock braking in a car.

- **Waterfall Model.** We are given a specific use of this software which is in the anti-locking so brakes and further this does not to be changed later as the main function remains the same whatever be the technology.

g) A virtual reality system to support software maintenance

- **Incremental Model.** We notice that the virtual reality is an emerging field and need to be change frequently with this the software maintenance would cost the customer a lot and also need a vey good planning.

h) A university accounting system that replaces an existing system

- **Prototyping Model.** The requirements of the university would be different. It need to be change, also this cost a lot while the product should have a good feedback from the customer itself so this should be open for them too. This is flexible and could be reusable.

i) An interactive system that allows railway passenger to find train times from terminals installed in stations.

- **Iterative/Prototyping Model.** Here the most important thing comes up is the regular feedback from all the user so that the model can be fine tuned for the cases, also the large input from the train data would be difficult to handle this many things.

j) Company has asked you to develop software for missile guidance system that can identify a target accurately.

- **Spiral Model.** The most important thing we want is the highest accuracy and the least risk factor, time would be also crucial and critical. This model is the best fit for this software development.

k) When emergency changes have to be made to systems, the system software may have to be modified before changes to the requirements have been approved. Choose a process model for making these modifications that ensures that the requirements documents and the system implementation do not become inconsistent.

- **Iterative Model.** Here it is mentioned that it needs faster changes in the software, this sometimes would be implemented without an approval too in some cases. So this implies that the time is also critical and risk factor is also not consistent. So this model would be a better fit.

l) Software for ECG machine.

- **Waterfall Model.** This is quite simple and have a particular requirement , also this doesn't need a rapid change with that it doesn't need the dynamic system to work, time is also not that critical. So this model can be implemented in a easier way.

m) A small scale well understood project (no changes in requirement will be there once decided).

- **Waterfall Model.** Here a clear understanding with no changes in requirement with the small scale, it will also need a short duration. As it doesn't need any change and is not to be dynamic we can easily implement this model to get a better and faster productivity.