LAB:-01

★ *Choosing Software Process Models*

- ● *IT 314 - Software Engineering*
- ● *Name: Vimal Sonagara*
- ● *ID : 202101490*
- ● *Group : 6*

(a) A simple data processing project.
Ans- Waterfall model. Since the requirements are predefined or 'frozen' and it
doesn't require changing it, the Waterfall Model is the best suited.

(b) A data entry system for office staff who have never used computers before. The user interface and user-friendliness are extremely important.
Ans- Prototyping models. Since the users are novice and they need a user-
friendly software- which means UI is important, prototyping models are best suited.

(c) A spreadsheet system that has some basic features and many other desirable features that use these basic features.
Ans- Evolutionary Prototyping Model. Since the software requirements need to
be changed i.e. in this case, added as well as the software shouldn't lose the old
features too that means it doesn't need to be discarded, it would require
Evolutionary Prototyping 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.
Ans- Simple Incremental Model. Since the requirements are changing fast ie.
unclear beforehand and there is just one team for every aspect of the project, a
a simple incremental model should work out.

(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.
Ans- Incremental Waterfall Model. Since there are multiple builds to be released
frequently with desired changes, Incremental Waterfall Model is the perfect
model to choose.

(f) A system to control anti-lock braking in a car.
Ans- Waterfall Model. Since the ABS in car has specific requirements and
functionality known beforehand, Waterfall model can represent it perfectly.

(g) A virtual reality system to support software maintenance.
Ans- Incremental Prototyping Model. Since VR is a new concept development
and user requirements might evolve in the future, this model is the best suited.

(h) A university accounting system that replaces an existing system.
Ans- Throw-away Prototyping Model. Since the software needs to be completely
replaced after the new build, throw-away prototyping model is perfectly suited.

(i) An interactive system that allows railway passengers to find train times from terminals installed in stations.
Ans- Iterative Models. Since it's an interactive system, time is important and it
should have a quick response to the user's query, Iterative model should be used.

(j) Company has asked you to develop software for missile guidance system that can identify a target accurately.
Ans- Spiral Model. Since it is a big project, time is very critical and risk cannot
To be taken, a Spiral model needs to be used. Moreover, the types of target i.e.
requirements may change.

(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.
Ans- Iterative Model. Since it requires rapid changes done without even being
approved- that implies that it might need to be undone. Moreover, time is very
critical and risk of software becoming inconsistent shouldn't have any scope. So
The Iterative Model is perfect.

(l) Software for ECG machine.
Ans- Waterfall Model. All the requirements are known beforehand and it has
specific functionality. So the Waterfall model is the best suited.

(m) A small scale well understood project (no changes in requirements will be there once decided).
Ans- Prototyping Models. Since it's a project, even though it is well understood,
the possibility that the requirements may change while the final product is still
under construction shouldn't be ignored. However, it's specified that the later
changes won't be allowed. So it is perfectly represented by Prototyping models.