ANOTODE

The Web Annotator .

SDLC Model Revision 2.0

CS Group 1

Authors:
Avi (201451070)

Reviewers:
Manohar (201451024)
Saurabh (201451017)

Revision Table

Revision	Author	Reviewer	Revision Date	Revision Tracking Notes
1	Avi	Manohar	25-9-16	Written document with V-Shape as our SDLC Model
2	Avi	Saurabh	28-9-16	Switched to Incremental Model

Contents

1 Introduction

We are using the Incremental model for our Software Engineering project. The incremental build model is a method of software development where the product is designed, implemented and tested incrementally (a little more is added each time) until the product is finished. It involves both development and maintenance. The product is defined as finished when it satisfies all of its requirements. This model helps us to get a working application ready early in the project development phase and will enable us to invite beta testers for testing our application. Also we hope this help us keep our team motivated as we will be able to see real progress.

To pick our SDLC model, we had to go through all the SDLC models and we keenly examined the pros and cons of each model. Here is a short account of those examinations.

2 Waterfall model

The waterfall Model is a linear sequential flow in which progress is seen as flowing steadily downwards (like a waterfall) through the phases of software implementation. This means that any phase in the development process begins only if the previous phase is complete. The waterfall approach does not define the process to go back to the previous phase to handle changes in requirement.

Why choose Incremental over it ?

Waterfall only gives a product release after all the requirements have been satisfied. So it can take long time to show us the results.

3 Big Bang Model

The Big Bang model is SDLC model where we do not follow any specific process. The development just starts with the required money and efforts as the input, and the output is the software developed which may or may not be as per customer requirement. There is no planning and the requirements are decided on the fly in the development process. This model is best suited for small/personal projects.

Why choose Incremental over it ?

Our project is quite complex with Android, Web app and browser extension. For it, Big Bang model is not suited as without planning, project has high chances of failure.

4 V Shaped Model

This model is very much like Waterfall but involves testing after each and every phase. There are separate tests for each phase. For this reason, it is also known as Verification and Validation model. For V Shaped model to work, the requirements should be very clearly defined from the start and there should be no ambiguity.

Why choose Incremental over it ?

Same like Waterfall model, V shaped can take even more time to give us results. Also we may have to add certain features (in Phase 2) in our project so Incremental felt the best for this.

5 Prototyping Model

It involves activity of creating prototypes of software applications before actually developing the final product. This is done to make sure that the project matches the requirements and the customer expectations. To do this, the prototype is created again and again until the customer requirements is satisfied.

Why choose Incremental over it ?

We didn't have the time that prototyping model requires. The majority of time in Prototyping model goes in creating various versions of prototypes and we simply didn't have that much time.

6 Spiral Model

This model of development combines the features of the prototyping model and the waterfall model. Various iterations of product is created and the product is contiously being improved following customer's suggestions. Spiral model is said as a "meta" model because it is not a real model. When a software development project lasts only once cycle, it resembles Waterfall model.

Why choose Incremental over it ?

Lot of time is required to reach the final product aslike Prototyping model. We didn't have that much time.

7 Agile

It is based on iterative and incremental development, where requirements and solutions evolve through collaboration between cross-functional teams. This method requires a really experienced team and high input hours per day. It

requires continuous inputs from customer and there is no room for guesswork.

Why choose Incremental over it ?

Most of the team members have little software development experience in our team so Agile methodology is not for us. Also we will not be able to take out the time for continuous interaction between teams that the Agile method requires due to classes and college timings.