TERM PAPER ON AGILE SOFTWARE DEVELOPMENT PROCESS

By: Vanja Vivek Vardhan
Roll no: 190101097
vanjavivek@iitg.ac.in
B.Tech, C.S.E.

Introduction

There are many SDLC models that are proposed. But major companies are not using the old methods and models to do the projects. AGILE model is being used currently by most of the companies.

Because with the increase in the rate of technology, there are so many features coming to software in a short amount of time. AGILE life cycle is very short (**couple of weeks to couple of months**). With the help of AGILE model, we can build software or add features to an existing application quickly in a short span of time.

Now, comparing classic waterfall model with AGILE. In waterfall model, we first fix the requirements. And then, we go on design and coding the software. After that we will test it and deploy. Now, if the client wants to add a new requirement at this stage, it would be impossible. Because, to add even a single extra requirement, we have to go back to requirement gathering stage and start all over again. This is the problem of waterfall model.

While building software, if we follow waterfall model, **it would be impossible to change something in between.** But in AGILE model, due to its fast-paced iteration, we can able to add the requirements. Here the requirements, design, testing would be done throughout the project lifecycle[1].

The rest of the document is organized as follows, "*History",* telling briefly, how AGILE is born and what lead it to become more popular among recent times. "*Process Outline*", briefly explains the steps that are involved in the AGILE method. "*Explanation of steps*", will explain the steps of AGILE in detail. And then followed by Popular AGILE methods used and references.

History

Before 2001, software companies used to follow waterfall model approach. It's a sequential approach where phases of development needs to be completed[2]. But often clients would be dissatisfied, because projects would take more time to complete and would cost more than the estimated budget. **Only 9% to 16% of the projects are delivered in-time and within the budget[3]**. Even when the project is completed, client would be unsatisfied because of changing needs over time.

So, in February 2001, 17 software developers came together for a 2-day summit for discussing alternative models for software development. They identified the common problems, like, lengthy documentation, and aimed to support lightweight methods. Over the 2-day summit, they finalized with AGILE. That means, the software development approaches are now be able to include/react to changes from time to time quickly.

They formed "*agile alliance*" and introduced "*agile manifesto",* which consists of 4 core values and 12 Principles.

Initially, AGILE is developed for software development, but many believed that, with some changes in the method, it can be applied to any business. So, this lead to "*Declaration Of Interdependence*" in 2005.

The main goal of declaration of interdependence is to encourage businesses to adopt AGILE beyond software development. It included an "*opening statement*" and "*6 principles".*

Process outline

AGILE refers to a software development approach based on iterative development.

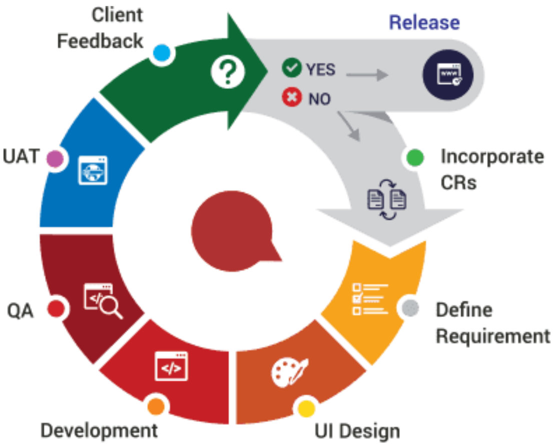

Figure 1 source:https://www.intelegain.com/wp-content/uploads/2015/04/agile_development_model.png

AGILE breaks the entire project into small parts and make iterations over these tasks. So that, if you need to add something, we can add it in the next iteration.

We call this divided tasks/parts as **product backlogs**. Product backlogs contains "*user stories*", some technical specifications. User stories specifies the requirements which are needed for client.

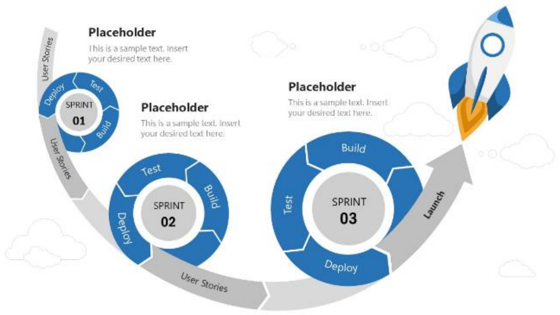
Placeholder
This is a sample text. Insert your desired text here.
Placeholder
This is a sample text. Insert your desired text here.
Placeholder
This is a sample text. Insert your desired text here.
User Stories
Deploy
Test
SPRINT 01
Build
User Stories
Test
Build
SPRINT 02
Deploy
User Stories
Build
Test
SPRINT 03
Deploy
Launch

Figure 2 source: https://cdn.slidemodel.com/wp-content/uploads/20998-01-agile-development-process-powerpoint-diagram-1.jpg

The brief process for any AGILE method is as follows:

1. **Modularity:** Dividing the entire module into smaller modules called user stories
2. **Iterative:** Small tasks are taken, then are developed and iterated to make any changes.
3. **Time Bound:** Setting a time for an iteration. (Typically 1-6 weeks)
4. **Adaptive:** We may face issues while implementing, so, based on it, solutions and next steps should be taken.
5. **Incremental:** In each iteration, we'll add the extra requirements/features to the modules. Thus, incrementing the scope of the project.

The steps involved in the AGILE methodology are:

1. **Project Planning**
2. **Roadmap Creation**
3. **Release Planning**
4. **Sprint Planning**
5. **Daily Meetings**
6. **Sprint Review and Retrospective**

Explanation of steps

1. Project Planning

A sprint is a task that takes about 1-3 weeks to complete. In a sprint, some set of user stories are selected based on priority and are developed.

The planning mainly consists of:

1. What should be built?
2. How the team should build it?

1.What should be built?

The product owner will decide what to build. The team leader(SCRUM master) and product owner, then divide the project into chunks known as product backlogs. Then, the product owner sets the priority of those product backlogs based on requirements. So, highest priority ones are selected to work upon in that sprint.

2.How the team should build it?

Here, the team tries to figure out how to build what they have decided. This can vary from team to team. Some may divide the product backlogs into smaller arts and work upon them. Breaking backlogs into tasks helps in maintaining the right amount of work.

By the end of planning stage, the team will get a idea of what they are doing and how they are going to do it.

2. Roadmap Creation

A roadmap is a schedule of tasks over the project period, that tells the team what they should be doing in each sprint.

A roadmap contains context of everyday tasks, so that, the teams knows what to do on a daily basis.

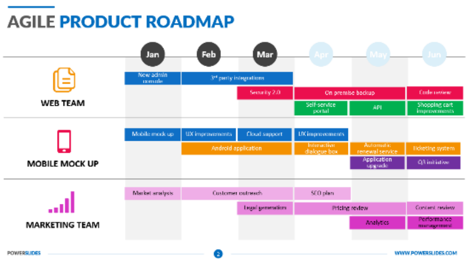

Figure 3

Different teams may share different roadmap based on the tasks they got allocated.

3. Release Planning

In release planning, we will plan the incremental releases of the product.

After every sprint, there would be a change/addition of features to a product. But, we won’t release product after every sprint.

With the help of release planning, the team will know when to release a new version of product and after how many sprints.

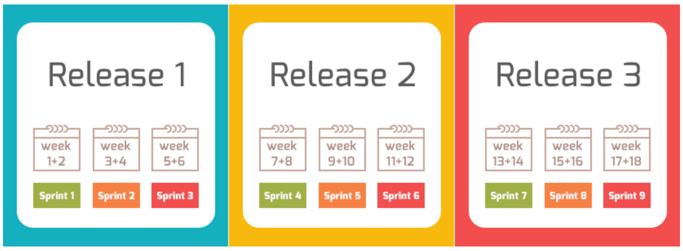

Figure 4 source: https://guide.quickscrum.com/wp-content/uploads/2018/09/planning-the-releases.png

Here, in the above diagram, we can see that, “the product is not released after every sprint”. New release happened after every 3 sprints. The number of sprints may vary based on requirements.

4. Sprint Planning

In sprint planning, every team member will be assigned their work before a sprint starts, so that they can find ways to work on it.

In this stage, the product owners decide what is to be done in this sprint, i.e, a sprint backlog, and then the work is assigned to the team.

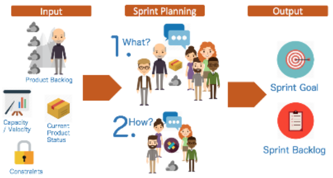

Figure 5

5. Daily Meetings

Team members would have a daily discussion on progress of their work. The team members would share their previous day work and discuss their today’s work.

And they will share their issues, can suggest improvements on their part.

The daily meetings help’s in smooth progression of the sprint and project.

6. Sprint Review and Retrospective

In sprint review, after every sprint, the team see what they have done and improves the product backlog.

Sprint review focusses on product improvement.

Sprint retrospective takes place after sprint review and before next sprint planning. In retrospective, team finds ways to improve their workflow.

Sprint retrospective focuses on process improvement that is involved while developing the project.

Popular agile methods

Since AGILE got popular due to its fast-paced nature, many organization’s began to adapt AGILE methodology. And the popular AGILE methods that are used widely are:

1. Scrum

Many companies around the world uses Scrum. Scrum is widely used in the **IT sector and software development sector**.

Big Companies such as GOOGLE, APPLE, FACEBOOK, SPOTIFY, ADOBE, etc uses Scrum[4].

The other popular AGILE methods are: Extreme Programming (XP), LEAN, Dynamic Systems Development Method (DSDM), Kanban, Feature Driven Development (FDD), Crystal, etc.

References

[1],[2],[3] An introduction to agile software development by Victor Szalvay, found at: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.451.9579&rep=rep1&type=pdf

[4] source: https://www.professionaldevelopment.ie/who-uses-scrum

Figure1 source: https://www.intelegain.com/wp-content/uploads/2015/04/agile_development_model.png

Figure 2 source: https://cdn.slidemodel.com/wp-content/uploads/20998-01-agile-development-process-powerpoint-diagram-1.jpg

Figure 3 source: https://www.google.co.in/url?sa=i&url=https%3A%2F%2Fpowerslides.com%2Fpowerpoint-business%2Fbusiness-models%2Fagile-product-roadmap%2F&psig=AOvVaw25Kjv86CC_CqXkFQbz_5Jh&ust=1651493724842000&source=images&cd=vfe&ved=0CAwQjRxqFwoTCOCA5umjvvcCFQAAAAAdAAAAABAa

Figure 4 source: https://guide.quickscrum.com/wp-content/uploads/2018/09/planning-the-releases.png

Figure 5 source: https://pm-powerconsulting.com/wp-content/uploads/2018/03/Sprint-Planning.png