Noroff
School of technology and digital media

Technical Report

CA – Project Methodology

Benedikte Sejersen

Word count

Summary: 100 | Main text: 954

Table of Contents

1. Summary

In this report I’m explaining why Intellectual Property is something to consider when developing. This applies for all in a team, every role should have knowledge. The primary objective in this report goes around explaining process and showing my knowledge about what I’ve learned in this course.
The Gantt-chart I’ve created is explained properly, with justifications. My chart is build up from the facts and statements in the LinkedIn videos.

Most of the sources is taken from the lessons, as well as the LinkedIn Learning videos. Considering little knowledge about project methodology from the beginning, these courses has been interesting.

2. Body

Introduction

In this course I'm going to introduce you to my Gantt-chart. My choices and knowledge about the main purpose of the chart are also included.

To show you my recently learned knowledge within project methodology, I have mentioned the Agile framework (and workflows), SDLC, IP and how to communicate across the team, and with the costumers.

Main section of report

SDLC
System Development Life Cycle is very important to think about when planning a development project. To start, requirements, design, construction, and implementation (create milestones) need to be clear for every member. This is usually the Project Manager's main task to inform the development team.

It is in this system the user stories comes in. It's like series of steps to get a wider perspective about the requirements. My Gantt-chart is build upon this System Development Life Cycle.

My Gantt chart
When I started to create my Gantt chart, I wasn't sure how I would create it. But when I started thinking of what I've learned in lessons, and LinkedIn, simplicity and a neat structured chart is the best. Colours helps to define and indicate the time the different roles on a team have. To recognizing/explaining which role are the selected colours, you will find these elements on the top of the chart. The faint colors are slacked, so it can be like flexible time. Be aware of I didn't have the daily standup meetings in the chart, because these meetings only lasts less than 15 minutes pe day.

I chose 'planning' as the first phase (finding possible problems, create schedules etc) on this project where all in the team and the costumers are assembled to ensure good communication/information. In the analysis phase (studying, requirements, backlog) the photgrapher is not included. The others will start to analyse and studying together/alone. The design phase is one of the most comprehensive and important phase, therefore it's necessary to include the team the whole time. In this phase the presentation for stakeholders and costumers will be to get confirmation and feedbacks. Coding and testing is two phases where the front-end developers has the main role, while after this the presentation for stakeholders and costumers will be. Implementation phase is to document, both for system and user documentation. At the end the final presentation is presented, and if necessary the phase maintainence where studying and reporting problems that did overcome in this project.
And to be clear, the meetings is where all are assembled.

I was thinking along the way, maybe it would take longer time to complete the tasks, but whether the weeks or days, I just prioritzed to show you my knowledge about how to understand, and create a Gantt chart. I have knowledge that it would take longer time to complete the various tasks, but how long time each role and in what order to complete them, is shown in my chart. I did also decide to add keyword to the tasks to remind the team the milestones.

How to communicate with the team and costumers
One of the best way to communicate when working in a team is by creating Gantt-chart, but also have meetings. Like the Agile Framework, Scrum and Kanban is a workflow methodology to have meetings (sprints) of what to do, requirements, issues, and what's done. Like daily standup meetings, planning and sync-ups meetings, softwares for chat, weekly reports etc. are examples of communication tools

These workflows are self-organized. There will be safe to use VCS to keep update your development team, as well as backlogs and documentation to inform such as costumers and the team, generally.

Gantt helps to keep in track the timing of the various sections, and are useful for the team, but also the project manager and the costumers. I prefer to use Excel because of it's ease and simplicities. Presentations and contuniously meetings with the costumers helps to avoid misunderstanding. When the meetings with the costumer's overcomes, IP is something to take into considering.

IP (Intellectual Property)
IP (or Intellectual Property) helps you to protect your and others trademark. By that, I mean it helps businesses to confidential information between employer-employee, that the business have copyrights or not, database rights, designs (considered ''new'' or not), patent of the products rights and trademarking (such like create a website). All in all, IP is important for companies if competitors have similar content etc. An of course, stopping others selling, making, using, or importing it without any permission.

As a Front-End Developer, having IP in the rear makes us a better developer, but also the knowledgement when creating new ideas and examination processes. Basically, IP is a must for the team when developing, and to protect your IP it's important to have knowledge about all these. As well as, domain names and a contract.

Conclusion

Bascially this course was quite short, but when digging deeper in each section it did have a wider scope. To learn about Gantt, and how effective and easy it's for developers to use, is something I'm going to think about when starting to developing by myself.

When I created the Gantt chart, my brain like crashed completely. Somehow I didn't know how to build an neat and easily readable chart, for everyone to read. And at the same time, keep short, but descriptive keywords of tasks for the central roles.

I used Git Bash to upload my chart. This was a confusing process, and did take long time to understand because of uocoming errors underway the process.

After all, I feel like I’ve learned how to work with these methodes, and how much benefits it comes within. These workflows and methods of working/communicate in teams, is definitely something to bring along.

3. References

Lessons:
Lesson 1.1 – Front-end Practice and Technical Report Writing
Lesson 1.2 – SDLC
Lesson 1.3 – Agile and Scrum
Lesson 1.4 – Kanban and user stories
Lesson 2.2 – Version control
Lesson 2.3 – Git and Git Commands

LinkedIn Learning:
Learning Gantt Charts
Scrum: The Basics
Version control for everyone

Internet:
http://oer.nios.ac.in/wiki/index.php/Phases_of_System_Development_Life_Cycle

My Repository:
https://github.com/BenedikteSejersen/Gantt.git

4. Acknowledgements

Asking a question to Michael about the Gantt-chart.

5. Appendices