Project plan – SCK CEN – Jetze Luyten

Contents

Acknowledgements:

This project plan was part of my internship at SCK CEN. I am very satisfied with how far I got to get to this point. Thanks to all the people who helped me make this internship possible:

- My parents, for continuously believing in me.
- Jitse, for guiding me so well and to help explain some small parts in his code.
- Jeroen, for, even though he was my mentor for a short time, contacting me in a timely manner before the internship began and for being a good internship mentor.
- Evert, for the good action of taking over the internship title from Jeroen and making my internship even more extensive.
- Sidney and Pieter Jan, for giving me an explanation about Alexandria and always assist with answering my questions relating to this.
- Arne, for explaining the differences between .NET CORE and .NET Framework. I also want to thank him for explaining more about git in Azure DevOps.
- Eric, for explaining Telerik to me.
- Maarten, for showing me what a datacenter looks like and keeping Alexandria up to date.
- The ICT-sector of SCK CEN, for welcoming me so well and for giving me the opportunity to take an internship there.
- Finally, SCK CEN itself, for giving me the opportunity to do my internship within it and for the kindness of the staff.

Introduction

I am going to explain more about what the problems in SCK CEN are to lead to an internship opportunity. Next, I am going to explain what my project entails. After this, I will explain the purpose of my project. Then I formulate the planning of my project. Finally, I will give my conclusion around the project/internship.

Problem statement

My internship company has a plethora of documents because of all the research. To solve this, they have had Alexandria for +/-12 years. Alexandria is a library where all the documents within a company reside. This program has security that ensures that the right people can get to the right documents.

The problem with Alexandria is that it is difficult to know which documents someone needs. Because of this, a tool was created that downloads all the documents from a particular "node" and places it locally. This tool is called CDST (Critical Documents Synchronization Tool).

The problem with this tool is that all documents are downloaded over and over again, and documents deleted on the server remain. There are also some bugs that are in CDST:

- In CDST, you can filter for attributes in the node. That filter has a 'mandatory' checkbox that is off by default, this should be on.
- Crash with unclear error message when syncing shortcut without permissions on original document
- Bad (less visible) error handling
- Tool fails on nodes with a question mark or special character in the name (e.g., 'Documents (How?)')
- Stopping an automatic sync happens immediately

Project description

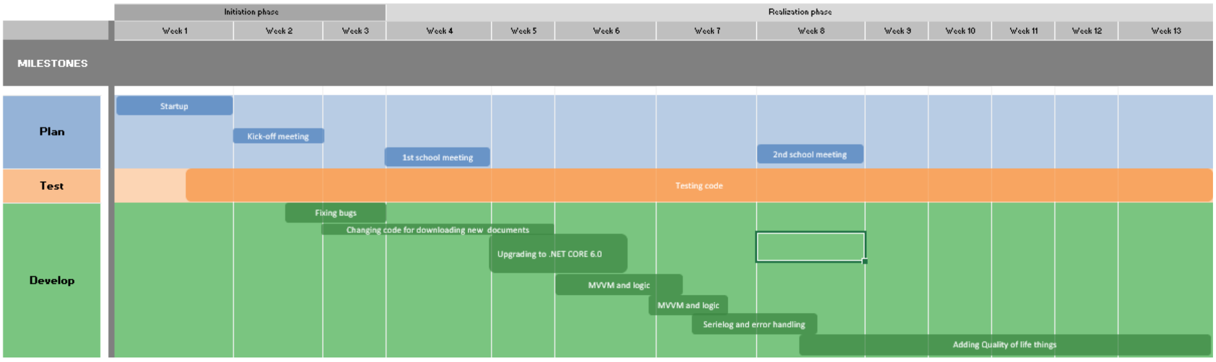
In my project I need to make sure that these bugs and problems are fixed, I am going to do this by:

- Turning on the 'mandatory' checkbox by default so that node filtering is easier
- Clearer error handling so the user can more easily report the problem to ICT.
- Showing a messagebox that asks the user if they are sure or if they want to stop the automatic sync.
- Using Regex to remove illegal special characters from the name.
- Create a new table (FileHistory) in sqlite that stores all files being downloaded in a table.
    - If the table is empty or if no nodeId can be found in the table but can be found in Alexandria, it will be placed in the table.
    - If the documents are updated on the server, then the documents are updated locally as well.
    - Delete documents that are no longer on the server locally (this is still in the FileHistory table).
    - Documents that were moved on the server (or folder is renamed), those are also moved locally.
    - Update documents from which a particular version was downloaded, but whose creation date/version number combination no longer matches.

Objective

The objective of my project is to upgrade CDST so that the user less/no problem has with the tool. More specifically: The documents need to be downloaded, not copied.

Project planning



Week(s)	What I did
1	Getting to know Alexandria and SCK CEN.
2	Meeting with supervisor to check if everything started fine.
2-3	Start fixing some bugs like: ○ Tool fails on node with illegal characters ○ Default yes for 'mandatory' attributes
3-5	Changing the original code so that the tool downloads the documents, instead of copying it.
5-6	Upgrading the project from .NET Framework v4.7 to .NET CORE v6.0.
6-7	Adding MVVM—logic and code logic to the code for reading the code easier.
7-8	Changing the way, the logfile is created. Using Serilog, we can send errors to a person through mail. Serilog also adds more structure in the logfile.
8-13	Adding Quality of Life improvements so that the user has easier understanding about what happened. One of the QoL improvements is showing the user if the downloaded ended successfully or not.

Conclusion

I have drawn some conclusions from the project:

- Understanding code from someone is harder than expected.
- SCK CEN has sliding hours
- Staff members are friendly and helpful.
- I learned more from developing code than looking up my problems.
- There are vague problems during programming that can happen at any time.