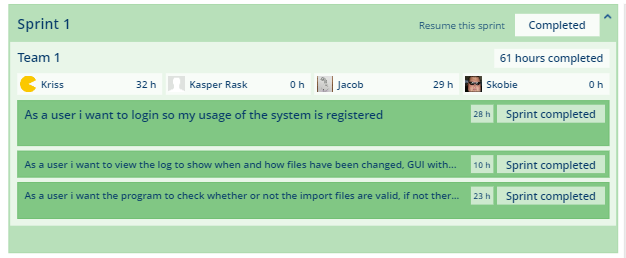
**Sprint 1**

3.1.    Sprint planning

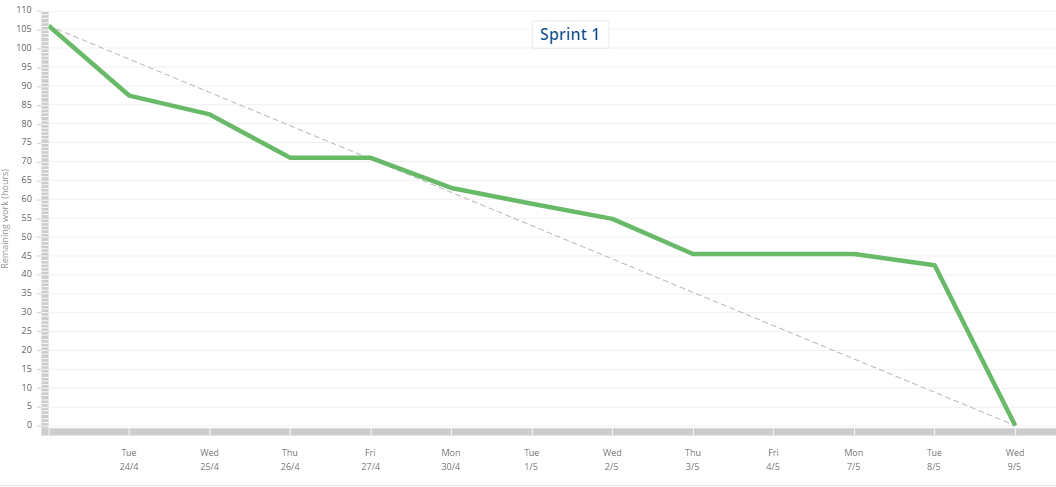
After the first sprint meeting we created the initial backlog for our project, we added some tasks to each user Stories in SCRUMWISE ( www.scrumwise.com ), and estimated the time for each task by using pointing poker (www.pointingpoker.com ), so we could see how must time each user stories take, In each stories description we wrote how to demo the user story. So before the story is done, it should pass the demonstration. We planned how much time our team had until the next sprint meeting with the client.

**Backlog**



**Figure x.x** Backlog of Sprint 1

This is the completed backlog items, but that few of the things we did in this sprint, because we had a huge user story, so this user story should be split into smaller user stories, because that Story was almost what the entire program should do. So we removed that story from this sprint after the sprint ended, so that could be break into smaller user stories.



**Figure x.x** **Burndown*.*** *Sprint 1*

This is the burndown form sprint one, as you can see in the beginning we follow the line, but after some time, we begin working on the huge user story and could not finish it because there was too many task we need to do for completing this user story.

3.2.    Daily meetings

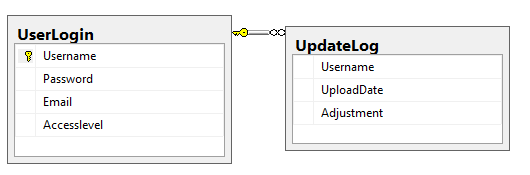
In our standup meeting, we talk about three thing. We did the standup meeting as the first thing them we meet.

* What we have done since last time.
* How we did it.
* What we should do today.

3.3.    GUI

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

3.4.    Data model



**Figure x.x** **Database Table*.***

This is the table we have in SQL database. As you can see we have “UserLogin” that is the table of user for the application. “UpdateLog” is the table for what people does in the program. So one user can have multiply data in “UpdateLog”. The requirement for this program was that we should save, who did it, then it was done and what was done, so we made a table with this three columns.

3.5.    Implementation

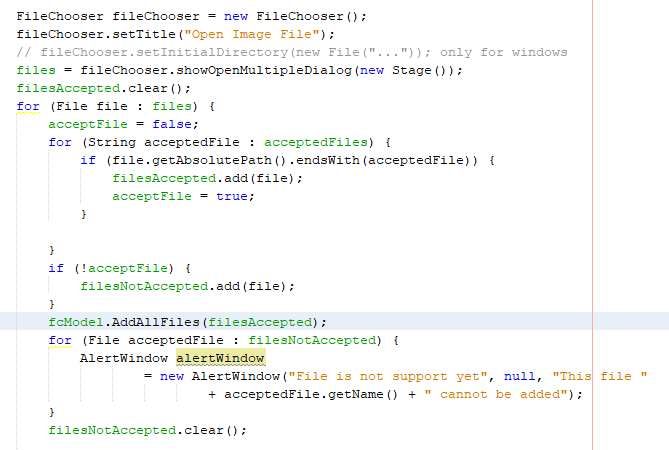
Then this sprint started the thing that was start working on was the most important part for the client, and that was Login. So the costumer will be able to track who did something in this program. The second part was the huge task we need to break to smaller stories. Right know the program is able to convert multi xlsx files, without threading, and able to log in as a specific user. Check if the program is able to convert this file type yet, and able to log what file is converted.

**Login 3 layers**



**Figure x.x** *Login to the program.*

So the first view you will come to in this program is the login. The first thing that happens, if you click login, is that it set the two text fields value to the BE class called UserLogin attributes after that it send the class UserLogin to BLL layer to check if that UserLogin attributes match. So after that the BLL just parse, the function to the database, there is find the Username in database, and check if the password match the password from userLogin. It will return a boolean so it is either true or false.



**Figure x.x**. *Import files.*

So this is the method for importing files. So the first thing there happens is that it opens a files chooser in a new stage, that are able to import more files, and that save it to a list of “Files”. After that it check if the files is accept yet, so it check what kind of file type it is, and if it match the files types the program can convert it yet it add it to a new list and save that list to the model so we can use it in the export view. It also add all the non-accepted files to a list and open alert windows, their say what files there was something wrong with.

3.6.    Sprint Review

The first sprint review meeting with the client, he told us that, he did not want the remember me function because of security, and the rest of it was more like the thing we already have planned to next sprint, like be able to add user, and track what each person does. He also told us that it would be nice if he could select a folder, instead of files. The most important thing was Customize JSON format, He told us that he did not need to have pause, stop converting files, but that was a requirement from the curriculum.

3.7.    Sprint Retrospective

|  |
| --- |
| **Good**   * Test every thing before we push it. * Work sharing, we was good to help each other. * Prepared for meeting. |
| **Could have been better**   * User stories, they was to big. * Work efficient. Some of the time was used on something we did not need. |
| **Improvement**   * Implement before Façade pattern. * Create Unit test before instead of after. |