**3.       Sprint 2**

3.1.    Sprint planning

After the second meeting, we split the huge, user story to smaller stories. And after that we calculated how must time we had this sprint and added the most important user stories first. We could exactly add all to this sprint if we have estimated the time right.

**Backlog**



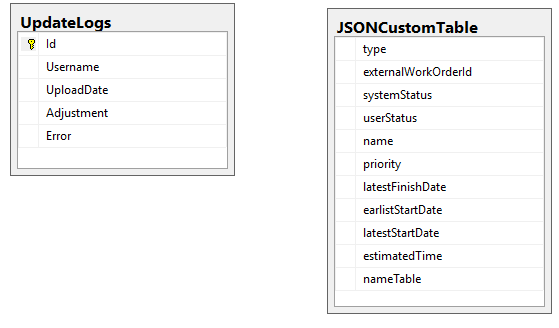
**Figure x.x.** *Backlog of sprint 2*

This is the completed backlog items In sprint 2. In this one we still have a huge user story, the first one but it is less than before, but I should still have been break to smaller stories.

3.3.    GUI

!!!!!!!!!!!!!!!!111

3.4.    Data model



**Figure x.x** **Database table update*.***

So as you can see from last time, the “UpdateLogs attributes has be change, now it have Error and ID, the reason for Error is added, is because then that row easily can be highlight, so you can see the errors. ID is added, so it could add one by one data to the table view, instead of you have to wait few seconds.

The “JSONCustomTable” was added, so the user could add the customized format that they want, so now they can made their own customization and see their customization.

3.5.    Implementation

In this sprint multithreading, connection pool, façade pattern for database.

In this sprint Is added :

* Multithreading
* Connection pool
* Facade pattern to BLL layer
* Singleton to model
* Convert csv
* Customize how the json file should look like
* Start, stop and pause threading
* Made a log for what each user does in the program
* Moderator able to add user to the program

3.5.1. Code examples



**Figure x.x Thread**

So for the conversionProgress method , it start with create a CompletableFutere, it is use because if one or more thread are trying to complete or cancel a CompletableFutere, only one of the them will succeeds, and that will be a good idea, in this case so we do not make a task two times, or override it.

runAsync means that it will return a new CompletableFutere, so it will continuous run to there is no more files, in the list. But if we run something with view change in this method, it will freeze.

In runAsync we have the method convertfile(File), that is the file their convert the files to json files.

So that why we use Platform.runLater, is fore the JavaFX application thread. It will run this Thread at some point, it will post the event to an event queue so it now what to do, then this thread run.

So in this Platform.runLater, it will remove the file from the model, so you can see how many files you still need to convert. There is also a progress bar so you can see how far you are with convention files.

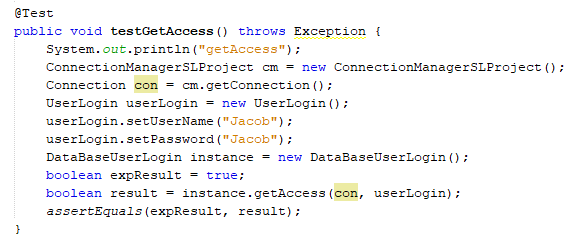
3.5.2. Design Patterns

In this sprint was implemented Connection pool, the reason for that is, because then it keep the connection to the database for an specific amount of time, so we do not have to connect to it each time we will get some information. So that will optimize the program a lot, that is used for the update log.

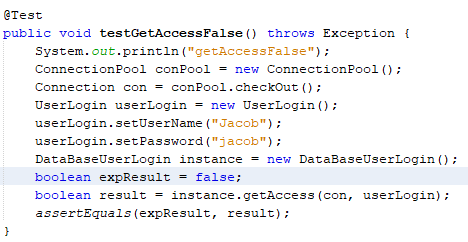
In the models, there is used singleton so, there will only be created one model, so there will be no null pointer exception if you try to create a new model. Because you only can create one instance of this class.

3.5.3. Unit test

In this application, there is not must their made sense to run unit test on. So the unit test was done on some of the logic from the database



This was one of the test, there was done, ass you can see, we need to create a connection to the database so we can get the data form the SQL database and send a userLogin with the right password and username to the DataBasesUserLogin, and the logic in that class will return true if that match the userLogin in the SQL database table.



The second test in this class in the opposite, their in BE Class UserLogin send the wrong password to the current username, so we expect it will be false, assertEquals will fail the test if expResult and result not is the same.

3.6.    Sprint Review

This was the second sprit review meeting with the client, and he told us that.

* Buttums should be easier to see that they are bottom.
* Missing array in the JSON file .
* Naming of the button should be more precise.
* Add information so it is easy to fix the bug for the customer.
* See the users.

3.7.    Sprint Retrospective

|  |
| --- |
| **Good**   * We did what we want to do. * Made the requirements for this project. |
| **Could have been better**   * Better working together * More meetings. |
| **Improvement**   * Write better notes of what we did. * Discus how it should look like in program so we do not waste to must time |