Product Backlog

0 Database stores all the info
1 Server interacts with DB and client
2 Client acts as a window for communication
3 Admin App has access to manage all the data that stored in the DB
4 Authenticator verifies that the user information matches that in the DB

Sprint Backlog

Time: Mar 28 - Apr 11

0 Database

1. Design the structure (30 min)
2. Set up the MySQL (20 min)
3. Learn and set up JDBC connection, make it able to talk to the server (2 h)

1+2 Server/Client Model App

1. Set up connection between server and client using socket (1 h)
2. Make sure client and server hear from each other, response to each other (1 h)
3. Set up the mechanism that can interact with the DB and the current methods (8 h)
4. Update the current view (4h)

3 Stand-Alone Admin App

1. Set up connection between admin app and DB (1 h)
2. Set up the mechanism that can read/update/delete from the DB (3 h)

4 Authenticator

1. Set up algorithm that encrypts the password (1 h)
2. Establish a method within the Authenticator to compare the encrypted data retrieved from the database with the submitted credentials to determine if a user can successfully log in. (3 h)

Daily Scrum

3.29:
First meeting with the customer. Gained an initial understanding of the specific requirements for Evol2
Team meeting for product backlog

3.30:
Break

3.31:
Break

4.1:
Meeting with the customer. Continue to confirm details
Team meeting for both product backlog and sprint backlog/ Tasks break

4.2:
Break

4.3
Set up server
Will work on socket

4.4
Set up socket
Problem: Client cannot talk to the IO handler from Evol 1

4.5
Socket problem fixed. Now Client and Server can interact
Will update view for the separate users(Student/ Professor)

4.6
Set up a subproject for the new admin app
Will work on DB setup and CSV parser

4.7
DB design finished and got set up
Trying to use the MySQL built-in import method to implement the CSV parser.
Authenticator finished

4.8
Gitlab issue board setup
Sample input file created

CSV parser based on MySQL import implemented
Problem:
Discovered that the built-in import functionality of MySQL is quite inflexible, and have decided to abandon the current approach

4.9
New Student Professor views finished
New parser based on MySQL insert implemented

Problem: The database cannot store and output the hashkey and salt correctly

4.10
Fixed the hash/salt store problem
Admin App finished
Server logic finished

Will integrate everything tomorrow

4.11
Integration and debugging.

Sprint Retrospective

Overall, we believe that everything went well, although the timeline was somewhat tight. Some team members were juggling multiple course projects, others were preparing for defenses, and some were preparing for job interviews. This made the process for this Evolution quite rushed, but ultimately, we completed it successfully. One area for improvement is our task distribution, the dependencies among different tasks meant that sometimes we had to wait for all components to be completed before we could conduct testing. We all agree that in the next phase, we should use our time more wisely and distribute tasks more rationally to optimize efficiency.