



Attendance Automation 2 - Compulsory Assignment #2

GitHub Repository:

<https://github.com/dpankov91/AttendanceFinal>

Date: 17/04-2020

Handed-in by:

- 1. Christian Hansen**
- 2. Dmitri Pankov**
- 3. Mario Ampudia**
- 4. Nebojsa Gutic**
- 5. Tienesh Kanagarasan**

Place of Education:

Erhvervsakademi Sydvest Esbjerg

Pre-Face:

As to strengthen our knowledge about management systems for future creation of larger scale projects in the same field of work, we are required to make a report on the progress throughout our compulsory assignment, **Attendance Automation 2**.

Within this project, we have included different concepts and patterns, implementations, project organization and architecture of our project.

Constructing this project report showed us that we have to enhance our knowledge in regards to the system's capabilities and showed us the errors of our ways. We learnt the importance of scrum, sprints and teamwork in project work.

Table of Contents

INTRODUCTION	3
1.1. Background	3
1.2. Problem Statement	3
1.3. PRODUCT VISION	3
1.4. STRATEGIC ANALYSIS	4
PRE-GAME (SPRINT 0)	4
2.1. PROJECT ORGANIZATION	4
2.2. OVERALL PROJECT SCHEDULE	5
2.3. INITIAL PRODUCT BACKLOG	5
2.4. ARCHITECTURE	5
SPRINT 1	6
3.1. SPRINT PLANNING	6
3.2. GUI	6
3.3. DATA MODEL	6
3.4. IMPLEMENTATIONS	7
3.5. CODE EXAMPLES	7
3.6. DESIGN PATTERNS/PRINCIPLES	8
3.7. SPRINT REVIEW	8
3.8. SPRINT RETROSPECTIVE	8
SPRINT 2	9
4.1. SPRINT PLANNING	9
4.2. GUI	9
4.3. Implementations	9
4.4. Design Patterns/Principles	9
4.5. Sprint Review	9
4.6. Sprint Retrospective	9
CONCLUSION	9
REFERENCES	10
APPENDICES	10
4.1. Appendix A: Overall Schedule	10

1. Introduction

1.1. Background

Whether it be used in educational facilities like schools or for workplaces like restaurants, attendance systems are extremely important. Unlike the capabilities of humans who are questionable, attendance automation systems guarantee precise time records and, unlike the usage of manual data entry, reduces the chances of errors occurring.

While systems reduce errors, they also increase productivity and are more efficient and are less time-consuming than the outdated versions of attendance. In the past, sheets had to be printed out and the payroll had to be made manually, time-consuming tasks after time-consuming tasks. That's not an issue with the attendance automations as they free up administration time with an automated system that keeps track of hours and the individual's attendances.

1.2. Problem Statement

How to develop a working program that solves the problems described, using SCRUM for the process and GitHub for version control and incorporate common software design patterns and a layered architecture in the software, including test.

Document the process planning and execution and software design and implementation in this report. Utilizing SCRUM for working in close collaboration with the end users.

1.3. Product vision

Our idea was to create an app where teachers have to generate a key and tell it to students. Students should insert it, that's how attendance is confirmed. Also 2 overview windows, for teachers and for students.

In student overview window, students are able to see a list of the days when they are IN school and next to it an Apple Chart with general data, how many days IN school, how many OUT.

For teachers overview window our idea was to create a table with a list of students and days IN school and next to it when you click to one of the students appears data about the selected student.



1.4. Strategic Analysis

- Strengths:**
- Good idea how to confirm attendance
 - Also good and fully finished GUI part
 - Support from tutors
- Weaknesses:**
- No experienced programmers in a team
 - Time wasting
 - Weak communication
- Opportunities:**
- Opportunity to have proper scrum meetings
 - Opportunity to learn new techniques
 - Opportunity to try design patterns
- Threats:**
- To lose opportunity for live meetings(pandemic)
 - Threat not to be ready on time

2. Pre-Game (Sprint 0)

2.1. Project Organization

Roles	Duties
Product Owner: Jeppe Moritz Led	<ul style="list-style-type: none">• Offering useful feedback• Sets times for sprint• Declare primary and ongoing project demands to the team.
Stakeholders: Trine Graungaard H. Thomsen Peter Gaarsmand Stegger Nielsen Stig Salskov Iversen Søren Spangsberg Jørgensen	<ul style="list-style-type: none">• Arranging Sprint Review meetings with the team to check on process so together they can inspect the results.
Scrum Master: Dmitri Pankov	<ul style="list-style-type: none">• Makes sure that the everything goes at a steady pace, in accordance to Scrum.• Ensures that everything relevant to the task is understood.
Team Member(s): Christian Hansen	<ul style="list-style-type: none">• Together, sprint goals are set up.

Dmitri Pankov Mario Ampudia Nebojsa Gutic Tienesh Kanagarasan	<ul style="list-style-type: none">• Approximate necessary demands for the project and provide that the demands are met through regular work efforts.• Sprint goals are being daily achieved.• Backlogs are constructed together.• Participate in scrum meetings• Attend Sprint Review meetings with stakeholders to inspect the results at the end of each sprint.
--	--

2.2. Overall Project Schedule

Overall project schedule can be found in the Appendices, Appendix A.

2.3. Initial Product Backlog

During the initial with the costumer, the team found these essential tasks:

1. The solution can be either centralized or decentralized
2. Must be easy for students and teachers alike to change the state or rectify errors, while also preventing the students from cheating.
3. The students must also be capable of seeing an overview of their attendance, displaying the total percentage of both present and absent.
4. The teachers must be able to see the following:
 - a. An overview of class attendance.
 - b. A summarized attendance, with most absent student at the top.
 - c. A summarized attendance on every student, in regards to which day in the week, he or she is most absent and total absence.

The initial backlog was centered around the most important functions of the application. With these different points, the team took these into account to create user stories. Those same user stories will be used as the startline for the team on what needs their focus.

2.4. Architecture

Taking experience and inspiration from previous assignments and projects, the team decided to rely on the Three-Layer-Architecture. The team in its entirety have learnt and worked with this in the first semester and continued to do so in the second semester. The Three-Layer-Architecture has various benefits that proved useful.

By utilizing the Three-Layer-Architecture, we are able to improve performance and availability. And three layers, each layer can be constructed by different members on the team.

To create a bridge between the two layers, DAL and BLL, and GUI the group decided to continue using the facade pattern. This pattern allows the higher levels of adaptability and it also allows the team the capability to easily field-test any functions that have been made in short amounts of time. While it is an incredible feature to use, the group did not find use for it.

3. Sprint 1

3.1. *Sprint Planning*

Following the mandatory tasks from the initial backlog, the team of five decided to delegates tasks of simple elements during the first sprint. Through lengthy discussions between team members, tasks were estimated to be achieved in shorts amounts of time. To perform the delegated tasks, no one in the team lacked the necessary information or knowledge to complete them. However, if the need for aid rose, the other team members were able to provide the necessary information to help out the struggling members. When additional tasks rose, additional discussions between members were made.

3.2. *GUI*

GUI was decided to be made with the expertise of Dmitri, Mario and Nebojsa. Once in agreement, they all agreed that the GUI should be easy to understand and visually appealing to the users for its functionality, in regards that it is a system dedicated to educational use. In the student overview window, there is an area for an graph to display the total percentage of the student's absence and present, which is pulled from the database.

3.3. Data Model

During 1st sprint we were working with Mock Data.

3.4. Implementations

As the work process through Sprint 1 went steady, the team began to work on the system's range of capabilities. This was done via the creation of the Business Logic Layer and the Data Access Layer. This was done in order to connect to the database, otherwise the functionality of the program would have gone downhill.

3.6. Design Patterns/Principles

For this program, we have implemented the facade pattern in the BLL and DAL. Facade pattern provides faster and more efficient access to the different layers. By using the facade pattern, the team has established a pathway for the data in the database to pass through the various numbers of layers.

3.7. Sprint Review

When the sprint review meeting came, the team showed the results of Sprint 1 to the stakeholders with their progress displayed in a burndown chart. However the progress of this sprint was not approached well, resulting in not so great results.

3.8. Sprint Retrospective

After the sprint review meeting, the group commenced a long discussion, debating identified problems that had to be tackled, either fixing them in some way the team knows of or how to improve them with their current knowledge.

Our main problems during this sprint was the lack of experience with using Scrumwise and giving tasks that were not easy to accomplish. Due to this lack of experience with the program, Scrumwise, the members would forget the need to put in their used time on the task.

The second problem of granting tasks that proved very difficult for the team to take care proved also to be very troublesome. As previously mentioned, this resulted in the burndown chart not looking too good at the sprint review meeting with the stakeholders.

4. Sprint 2

4.1. Sprint Planning

Our first sprint review gave us more clarity about what was the correct path to follow in order to get things to work. Now we use 'As a user' technique. And split backlogs for few smaller tasks. Also we created database during this Sprint.

4.2. GUI

During this sprint we didnt work a lot on gui, we spent more time on functionality. In a Gui part we add Help in MenuBar and add today's date to label in TeacherGenerateKeyWindow.

4.3. Database

In the beginning, we started with a database with a single table, user. A user table with the names of the students, their usernames, passwords and whether or not they were teachers. But as we progressed, we made additional tables, one for tracking attendance and the other for key holding.

4.3. Implementations

We created LogIn function. We passed inserted username and password to database and get back if its correct, one of the windows for teacher or for student, it recognise who is who by boolean isTeacher.

Also we created generateKey method. That method will save generated key to database.(In Sprint 3 we created method what will allow to create key only once per day).

4.4 Code Examples

Insert Todays Key:

```
private void insertTodaysKey() throws IOException {  
    String todaysKey = txtNewKey.getText().trim();  
    LocalDate dateNow = LocalDate.now();  
    model.insertKey(todaysKey, dateNow);  
    txtNewKey.clear();  
    setUpAlert("Success", "Todays Key is generated");  
    goBack();  
}
```


Label what shows today's date:

```
public void showDate() {  
    Date date = new Date();  
    DateFormat dateFormat = new SimpleDateFormat("dd/MM/yyyy");  
    lblTodaysDate.setText(dateFormat.format(date));  
}
```

Database connection method for login:

```
public User getUser(String username, String password) throws SQLServerException, SQLException  
{  
    String sql = "SELECT * FROM [dbo].[User] WHERE username = ? AND password = ? ";  
  
    Connection con = connector.getConnection();  
    PreparedStatement pstmt = con.prepareStatement(sql);  
    pstmt.setString(1, username);  
    pstmt.setString(2, password);  
    ResultSet rs = pstmt.executeQuery();  
  
    if(!rs.next()){  
        return null;  
    }else{  
        int id = rs.getInt("id");  
        String firstName = rs.getString("Fname");  
        String lastName = rs.getString("Lname");  
        boolean isTeacher = rs.getBoolean("teacher");  
  
        User us = new User(id, firstName, lastName, isTeacher);  
  
        return us;  
    }  
}
```

4.5. Design Patterns/Principles

In program, we first time used singleton pattern for model class. It was easy to implement and very useful we will use it in the future as well.

4.6. Sprint Review

Our first sprint didn't go as planned, so this was our chance to improve. In this sprint we covered more complex tasks, therefore sometimes it took us more time. On SCRUM meeting we get positive review from our stakeholders.

4.7. Sprint Retrospective

After the sprint review meeting, the team concluded that the sprint was a success, we finished the tasks in time, within our estimations. We also held a discussion about what could be improved.

5. Conclusion

As this was the first time the team had to work with Scrumwise and implement the Scrum method into project work, there was room for improvements due to the plethora of problems that were encountered.

While this project did progress at a steady pace, all team members did struggle with balancing this project and get used to online classes due to the quarantine that followed the COVID-19 outbreak. Some moved back to stay with their family during these tough times, either in other cities or in their home country. With the Corona-Virus outbreak, times had toughened and focus on school and educational elements wavered, making the members decrease the work on task.

A lot of time had been wasted and fear and worry spread as a side effect of the virus outbreak.

Because of the outbreak's mandatory quarantine rules, communications were strictly limited to online calls or chats via social media, occasionally Zoom meetings would be brought up if the situation called for it.

This resulted in various features not being included in the project's construction.

6. References

3-tier application architecture, by Margaret Rouse, posted on online website:

<https://searchsoftwarequality.techtarget.com/definition/3-tier-application>

7. Appendices

4.1. Appendix A: Overall Schedule

Week 11	<ul style="list-style-type: none">● Kick-Off meeting:<ul style="list-style-type: none">○ Project presentation○ Overall Schedule○ How to write a report● Planning:
---------	--

CS 2st semester Compulsory Assignment
Attendance Automation 2 -
Compulsory assignment #2



Hand-in: Christian, Dmitri, Mario, Nebojsa, Tienesh

	<ul style="list-style-type: none"> ○ Product Definition ○ Product Vision ○ Initial Backlog ○ Initial User Stories
Week 12	<ul style="list-style-type: none"> ● Sprint 1 start, Monday 16th of March ● Delegating tasks in Sprint 1 ● Start coding. ● Sprint 1 Review meeting (Friday, 9 AM) ● Sprint Retrospective ● Backlog Stories Updating ● Task and User stories updating
Week 13:	<ul style="list-style-type: none"> ● Sprint 2 Start, Monday 23d of March ● Start coding on tasks and user stories.
Week 14:	<ul style="list-style-type: none"> ● Continue coding on delegated tasks and user stories. ● Finish Sprint 2 ● Sprint 2 Review meeting, Friday 3rd of April. ● Sprint Retrospective ● Update Backlog Stories ● Update Task and User stories
Week 15:	<ul style="list-style-type: none"> ● Easter Holidays, with work. ● Sprint 3 Start, Saturday 28th of April ● Coding on delegated tasks and user stories.
Week 16:	<ul style="list-style-type: none"> ● Final Adjustments ● Finish writing report ● Submitting, Friday 17th of April, 02:00 PM. DK: 09:00