Attendance Automation 2 - Compulsory Assignment #2

**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:**

Table of Contents

[1. Introduction 2](#_Toc37840233)

[1.1. Background 2](#_Toc37840234)

[1.2. Problem Statement 3](#_Toc37840235)

[1.3. Product vision 3](#_Toc37840236)

[1.4. Strategic Analysis 3](#_Toc37840237)

[2. Pre-Game (Sprint 0) 3](#_Toc37840238)

[2.1. Project Organization 3](#_Toc37840239)

[2.2. Overall Project Schedule 3](#_Toc37840240)

[2.3. Initial Product Backlog 3](#_Toc37840241)

[2.4. Architecture 4](#_Toc37840242)

[2.5. Preliminary Usability Test 4](#_Toc37840243)

[3. Sprint 1 4](#_Toc37840244)

[3.1. Sprint Planning 4](#_Toc37840245)

[3.2. GUI 4](#_Toc37840246)

[3.2.1. UI-Design Patterns 4](#_Toc37840247)

[3.3. Data Model 4](#_Toc37840248)

[3.4. Implementations 4](#_Toc37840249)

[3.5. Code Examples 4](#_Toc37840250)

[3.6. Design Patterns/Principles 4](#_Toc37840251)

[3.7. Unit Test 4](#_Toc37840252)

[3.8. Sprint Review 5](#_Toc37840253)

[3.9. Sprint Retrospective 5](#_Toc37840254)

[4. Sprint 2 5](#_Toc37840255)

[4.1. Sprint Planning 5](#_Toc37840256)

[5. Conclusion 5](#_Toc37840257)

[6. References 5](#_Toc37840258)

[7. Appendices 5](#_Toc37840259)

# Introduction

## Background

Whether it be used in educational facilities like schools or for workplaces like restaurants, attendance systems are extremely importance. 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reduce errors, they are also increase productivity and are more efficient and are less time-consuming than that the outdated versions of attendance. In the past, sheets had 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.

## 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

Elevator pitch of the project idea.

## 1.4. Strategic Analysis

# Pre-Game (Sprint 0)

## 2.1. Project Organization

|  |  |
| --- | --- |
| **Roles** | **Duties** |
| Product Owner:  Jeppe Moritz Led | * 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 | * Arranging Sprint Review meetings with the team to check on process so together they can inspect the results. |
| Scrum Master: | * 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  Dmitri Pankov  Mario Ampudia  Nebojsa Gutic  Tienesh Kanagarasan | * Together, sprint goals are set up. * 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

## 2.3. Initial Product Backlog

## 2.4. Architecture

## 2.5. Preliminary Usability Test

# Sprint 1

## 3.1. Sprint Planning

## 3.2. GUI

### 3.2.1. UI-Design Patterns

## 3.3. Data Model

## 3.4. Implementations

## 3.5. Code Examples

## 3.6. Design Patterns/Principles

## 3.7. Unit Test

## 3.8. Sprint Review

## 3.9. Sprint Retrospective

# Sprint 2

## 4.1. Sprint Planning

# Conclusion

# References

# Appendices