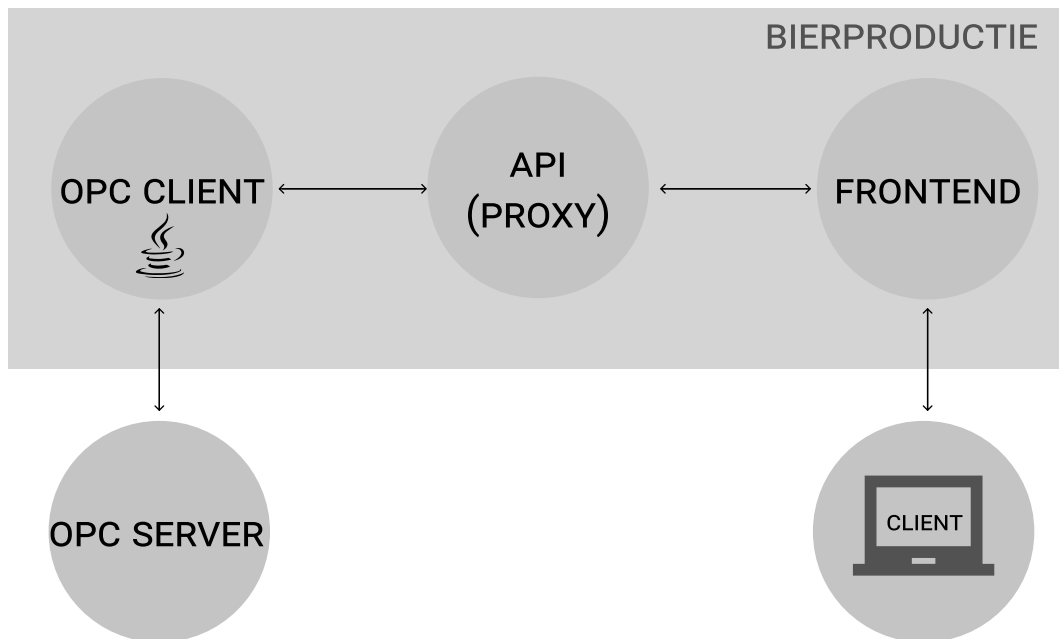


# Bierproductie

A management system for brewing machines



Bachelor of Engineering, Software Technology

Semesterproject 3. semester, ST3-PRO

**Project Period:** 31.08.2020 - 19.12.2020

**Hand in date:** 19.12.2020

## Group 06:

Jakob Rasmussen, jakra19@student.sdu.dk

Kenneth M. Christiansen kechr19@student.sdu.dk

Kevin K. M. Petersen, kepet19@student.sdu.dk

Kristian N. Jakobsen, kjako19@student.sdu.dk

Simon Jørgensen, sijo819@student.sdu.dk

**Supervisor:** Parisa Niloofar, parni@mmmi.sdu.dk

University of Southern Denmark  
The Faculty of Engineering  
The Mærsk Mc-Kinney Møller Institute  
Campusvej 55, 5230 Odense M

**Title:** Bierproductie

**Institution:** University of Southern Denmark  
The Faculty of Engineering, The Mærsk Mc-Kinney Møller Institute  
Campusvej 55, 5230 Odense M

**Education:** Bachelor of Engineering, Software Technology

**Semester:** 3. Semester

**Course Title:** Industrial 4.0 cyber-physical software systems


**Internal Course Code:** ST3-PRO

**Project Period:** 31.08.2020 - 19.12.2020

**ECTS:** 10 ECTS

**Supervisor:** Parisa Niloofar

**Project group:** 06



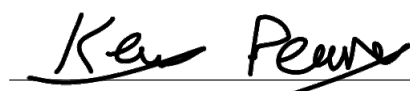
---

Jakob Rasmussen, jakra19@student.sdu.dk



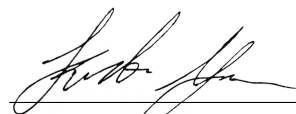
---

Kenneth M. Christiansen, kechr19@student.sdu.dk



---

Kevin K. M. Petersen, kepet19@student.sdu.dk



---

Kristian N. Jakobsen, kjako19@student.sdu.dk



---

Simon Jørgensen, sijo819@student.sdu.dk

Pages: 10

Appendix: 0

By signing this document, each group member confirms that everyone have participated equally to this project, and everyone is thus collectively responsible for the content of the report.

# I   Summary

# II Table of Contents

# III Editorial

**IV    List of Figures**

# 1 Introduction

## 2 Background



### 3 Problem analysis

# 4 Theory & Methods

## **5 Requirements**

### **5.1 Overall Requirements Specification**

### **5.2 Selected Detailed Requirements**

#### **5.2.1 Functional & Non-Functional Requirements**

#### **5.2.2 The Physical Setup (The Brewery Machine)**

#### **5.2.3 The Simulator**

### **5.3 Use Cases**

#### **5.3.1 Actor List**

#### **5.3.2 Detailed Use Cases**

*From project description*

#### **5.3.3 Use Case Diagram**

## 6 Analysis

### 6.1 Use Case analysis

#### 6.1.1 Class Candidates

#### 6.1.2 Description of Classes

#### 6.1.3 UML Analysis Diagram

### 6.2 Use Case Realisation

#### 6.2.1 Sequence Diagrams

#### 6.2.2 Operation Contracts

Operation contracts is used to specify what state the program is in, before it can run the operation. Operation contracts also specify what happens after the operation is run.

<b>start</b>	
<b>System operation</b>	start
<b>Cross References</b>	Use case: Start machine see table ??
<b>Responsibility</b>	Starting the beer machine if the pre-conditions is met. If the pre-conditions is not met, the beer machine will not start
<b>Output</b>	The beer machine started the production
<b>Pre-conditions</b>	The beer production machine needs to be in ready mode, that is, not producing beer.
<b>Post-conditions</b>	The beer machine started brewing

**Table 1:** Operation Contracts start

<b>stopProduction</b>	
<b>System operation</b>	stopProduction
<b>Cross References</b>	Use case: Stop the beer Machine see table ??
<b>Responsibility</b>	Stop's the beer machine if the pre-conditions is met. If the pre-conditions is not met, the beer machine will not do anything
<b>Output</b>	The beer machine is stopped
<b>Pre-conditions</b>	The beer machine needs to be running
<b>Post-conditions</b>	The beer machine is stopped

**Table 2:** Operation Contracts stopProduction

<b>reset</b>	
<b>System operation</b>	reset
<b>Cross References</b>	Use case: reset see table ??
<b>Responsibility</b>	It is responsible for resetting the beer machine.
<b>Output</b>	reset the beer machine.
<b>Pre-conditions</b>	The beer production machine needs to be in ready mode, that is, not producing beer.
<b>Post-conditions</b>	The beer production machine has been reset.

**Table 3:** Operation Contracts reset

<b>clear</b>	
<b>System operation</b>	clear
<b>Cross References</b>	Use case: clear see table ??
<b>Responsibility</b>	It is responsible for clearing the beer machine.
<b>Output</b>	The beer machine has been cleared.
<b>Pre-conditions</b>	The beer production machine needs to be in ready mode, that is, not producing beer.
<b>Post-conditions</b>	The beer production machine has been cleared.

**Table 4:** Operation Contracts clear

<b>display live data</b>	
<b>System operation</b>	displayLiveData
<b>Cross References</b>	Use case: displayLiveData see table ??
<b>Responsibility</b>	It is responsible for posting data to the client.
<b>Output</b>	Post data to the client.
<b>Pre-conditions</b>	The beer production machine needs to be on and producing beer.
<b>Post-conditions</b>	Live data has been displayed for the user.

**Table 5:** Operation Contracts monitorAndDisplayData

<b>batchReport</b>	
<b>System operation</b>	batchReport
<b>Cross References</b>	Use case: batchReport see table ??
<b>Responsibility</b>	Make a report after the pre-conditions is met and adds the report to the database.
<b>Output</b>	Produces a batch report and display it for the user.
<b>Pre-conditions</b>	The beer Machine needs to have produced a batch.
<b>Post-conditions</b>	A batch report has been displayed for the user.

**Table 6:** Operation Contracts produceBatchReport

### 6.2.3 Updated UML Class Diagram

## 7 Architecture

## 8 Design

## 9 Implementation



## 10 Verification & Validation

## 11 Evaluation

## 12 conclusion