

# PU5

TTM4100 - COMMUNICATION SERVICES AND NETWORKS

TDT4140 - SOFTWARE ENGINEERING

TDT4145 - DATA MODELING, DATABASES AND DATABASE MANAGEMENT  
SYSTEMS

TDT4180 - HUMAN-COMPUTER INTERACTION

## **Gruppe 6:**

Espen Albert, Finn Inderhaug, Kristoffer Andreas Dalby  
Christoffer B. Nysæter, Andreas Wien, Jonas André Dalseth

21. mars 2014

# Innhold

<b>1</b>	<b>Time</b>	<b>1</b>
1.1	Unexpected tasks . . . . .	2
1.2	Java FX . . . . .	2
1.3	Loss of manpower . . . . .	2
<b>2</b>	<b>Systemtesting</b>	<b>3</b>
2.1	Results . . . . .	3
2.2	Changes . . . . .	4
<b>3</b>	<b>Experiences</b>	<b>4</b>
3.1	Negative experiences . . . . .	4
3.1.1	Bad communication . . . . .	4
3.1.2	Meeting in time . . . . .	4
3.2	Positive experiences . . . . .	4
3.2.1	New friends . . . . .	4
3.2.2	New knowledge . . . . .	4
3.3	What could we do better? . . . . .	4
3.3.1	Scrum meetings . . . . .	4
3.3.2	Group leader . . . . .	5
3.3.3	Deadlines . . . . .	5

## 1 Time

Task	Estimated hours	Registered hours
Create ER-diagram	10	12
Logic database schema	10	10
Setup the database server	8	7
Create SQL schema	16	17
Implement JDBC	8	9
Implement query methods	16	20
Planning server to client	6	4
Planning client to server	6	4
Server to clients, Master	10	7
Server and clients, Threads sending JSON objects to logged in clients	16	12
Clients to server, Master	10	9
Clients to server, Check for inconsistency	10	8
Clients to server, multiple Threads	10	7
Creating login functionality	7	7
Parsing json objects to model	8	9
Define model and API structure	12	11
Create json objects based on the model	8	9
Creating vital functions and objects	16	17
Optimize the notify function	15	12

Discuss how we want the user interface	12	10
Designing the user interface using paper models	12	14
Show papermodel to studass and another group	6	8
Fix papermodel after feedback	3	5
Conceptual model	6	4
Screen design	12	12
Construction design	12	12
Make login screen	4	5
Make appointment view	16	14
Make week view	16	18
Other functionality	16	16
Administration	0	10
PU1	0	41
PU2	0	41
User interface discuss	0	4
Paper model	0	26
Show papermodel	0	10
PU1-3 eval	0	5
Java helperclasses	0	2
MMI Report	0	14
Java FX screen design	0	81
PU5	0	11
D3	0	24
Testing and debugging	0	7

### 1.1 Unexpected tasks

As you can see in the table, the last rows are task we did not think of as of the planning in the PU1 assignment. Some of those task took us by surprise and took very long to finish. This had a major impact on the projects work breakdown schedule.

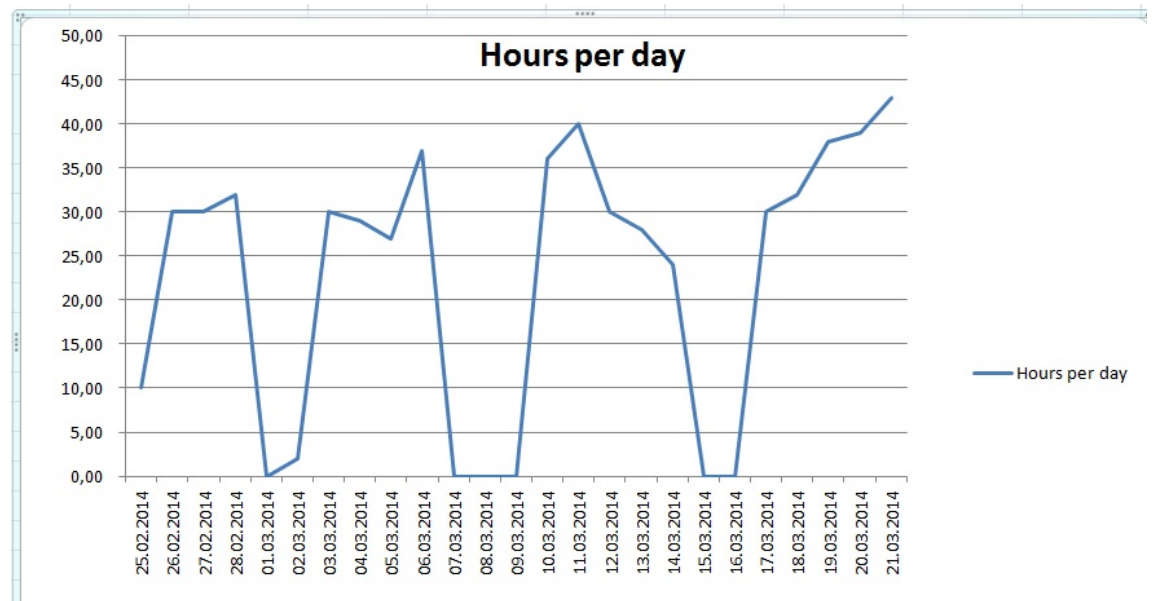
### 1.2 Java FX

We tried to use Java FX because it was the newer, more modern UI framework. This was not that easy as we though and unexpected problems ended up taking up a lot of time, which affected the product and made testing hard.

### 1.3 Loss of manpower

Throughout the project we had a problem with one of the group members being sick, over most of the project period. This resulted in loosing 14 days worth of

work. We have also had team members that had other meetings to go to which also eat away from our work hours. The occasional sickness also targeted us by taking out a member for 4 days. To make up for the lost work power the members that was able to meet worked longer days at the end. We worked approximately 32 hours a day, and at the end we worked up to 43 hours a day. Our plan was originally to work around 36 hours a day.



Figur 1: Work hours per day

## 2 Systemtesting

### 2.1 Results

Test	Number	Success
Log in	# 1	Yes
Add appointment	# 2	Yes
Respond	# 3	-
Edit appointment	# 4	Yes
Delete appointment	# 5	-
Book room	# 6	-
View	# 7	Yes
Check status	# 8	-
Decline invitation	# 9	-
Book room	# 10	-
View	# 11	Yes
Track invite	# 12	-
View other calendars	# 13	Yes
Alarm	# 14	-

## **2.2 Out of time**

We did not have time to finish all the functionality of the program, most of the requirements are nearly done, but we did not have time to finish the implementation of all of them. And therefor there was not any time to test it.

## **2.3 Changes**

# **3 Experiences**

## **3.1 Negative experiences**

### **3.1.1 Bad communication**

A negative experience from this project has been bad communication between the members in the group, especially with the members that has been sick or unavailable a lot. This has resulted in many a lot of working hours that has just been used for catching up and updating them unnecessary. Between the group members that have been meeting regularly the problem has been mainly that they have not spoken of what they are doing and how far they have come. This has resulted in a little unbalance workload between some parts of the program. Another problem has also been that people has not asked for help instead of hitting the wall with a problem over and over.

### **3.1.2 Meeting in time**

A problem for most members in the group is to meet at time, or meeting at all. Very many members are deep sleepers, and sleep right through their alarm. This has resulted in many meetings not starting as late as one hour after the agreed meeting time.

## **3.2 Positive experiences**

### **3.2.1 New friends**

The members of the group has come very well along right from the start. And we would all say that a very positive experience is that we have been able to make friends across the different studies we are enrolled in. A very good thing about this is that we have been able to work together closely every day in a longer time periode without wanting to kill one another.

### **3.2.2 New knowledge**

From the first meeting we had we discussed the different roles that coding the project would include and who was interested in what. Every member quickly selected a main area they wanted to focus on and everyone got to work with what they wanted. By choosing an assignment that a member would like to work on, the productivity was rather good. At the point of writing this, every member is really satisfied with all the new knowledge they have gained. The practice experience in the area of databases, graphical interfaces and networking is very good for the future.

### **3.3 What could we do better?**

#### **3.3.1 Scrum meetings**

Our group have not been very good with regular scrum meetings, this is definitely something we could do better. We see that the negative experiences from bad communication would probably have been eliminated if regular scrum meetings was something we did every day. We ended up with trying to catch up with everything when we had meeting, which resulted in long inefficient meetings.

#### **3.3.2 Group leader**

Every group did get a group leader which was assigned by the teaching staff. The group in general did not use this person as the leader, instead we all took responsibility, which in the start worked very well. But at the end of the project it was hard for everyone to keep up with all the loose ends and what need to be done in what order. Since we had no one that had the overview over the big picture it was really hard to assign the different issues that still was around to the people that did not have anything to do. When we look at this problem in retrospective it is easy to see that the group leader role should be assigned and given specific task to do as we went through the project.

#### **3.3.3 Deadlines**

Another thing we should have done is to create and work with deadlines throughout the project. We did not have any deadlines except the due dates for every assignment, which actually was a problem when we started to code and did not have regular due dates.