Timetable for Software Engineering

Christian Foyer, 315200
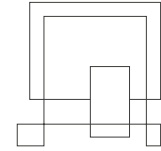
Martin Rosendahl, 315201

Nina Wrona, 315202

Robert Barta, 315242

Software Technology Engineering
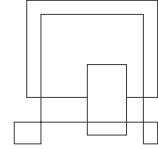
Semester 1Y

6th of October, 2021

Table of content

1. Background Description

At VIA University College in Horsens, Bob oversees making the timetable for the courses software engineering students must attend. As this is one of the most populous areas of study at the school, it can be a very stressful task.

First, Bob receives a document from the head of department, which includes classes with the students enrolled, teachers, courses, and the ECTS value of each course. Different types of students must be added or removed to individual courses based on the information in their transcripts. Then, he manually plans a timetable that collects and implements wishes sent by the teachers and combining them with the students' best interest.

Subsequently, he assigns courses to rooms, but must check for overlaps by hand. Bob then sends out the test week he has created to be reviewed by the teachers. Once Bob has incorporated the suggestions of the teachers, he publishes the timetable as a spreadsheet on Studienet and removes the holidays. However, during the semester, updates must be made, which takes plenty of time.

In the current method, all teachers and students can view all of the timetables without logging in. With the intricacies of how the timetable is made now, it is not possible to find a replacement if Bob is not able to do it. Due to the shortfalls of the current method, Bob is not able to easily and efficiently create and publish a timetable for both teachers and students to view. Finally, students can access individual timetables.

To add to the challenges that stand in the way, the university college just moved to a new campus in the center of town with limited rooms; each room is now precious. In a review conducted in 2014, a group of students at the Worcester Polytechnic Institute analyzed and optimized the scheduling of operating rooms to increase their usage and efficiency. They implemented a new scheduling software that used linear programming. Consequently, the time the operating rooms were in use increased. "According to our liaison at UMMMC, Dr. Tze Chiam, Ph.D., the external benchmark for OR utilization is between 70 and 80%; UMMMC is currently operating at approximately 65% utilization
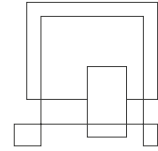
for their operating rooms" (Carrol, Juers, and Kent, 2014). Similarly, the current method is underutilizing the limited rooms at the new campus.

There are current market solutions to help with scheduling, but they are short of ideal. Bob's schedule is special because some rooms can open into each other, and some courses have multiple teachers assigned. Coupled with the inability to account for those features of the timetable, many current tools available have outdated user interfaces that directly make using and reading the timetable difficult (Top 10 Free Scheduling Software For Schools And Colleges, 2021). As these older user interfaces make the software difficult to use, the lack of flexibility in older software compounds the issues.

In a systematic review of timetabling available for different higher education institutions, Vrielink and other researchers found that many timetable tools used today do not effectively cater to the flexibility needed for higher education institutions (Oude Vrielink, Jansen, Hans and van Hillegersberg, 2017).

Summarizing, Bob's current method requires too much manual data entry and creates obstacles to an efficient timetable.

## 2. Problem Statement

Main problem
Bob does not have an effective tool to make and publish a timetable for software engineering students and teachers at VIA University College, causing underutilization of rooms and overbooking.

1. What information is needed to avoid double booking?
2. What information do we need to take into consideration while creating a timetable?
3. What information do we need to update the timetable for Bob?
4. What should students and teachers be able to see?
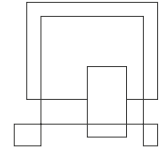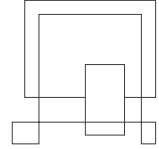
3. Definition of purpose

The purpose of this project is to develop a system that will assist with Bob's scheduling of software engineering courses and publish an optimal timetable that meets the needs of students and teachers.
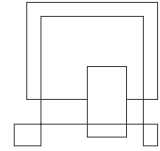
4. Delimitation

1. We will not create individual timetables.
2. We will not schedule holidays automatically.

5. Methodology

During the first semester, students are limited to using the waterfall methodology: a primitive and iterative development cycle that depends on the previous step to complete the next task. The process consists of a rigid sequence of analysis, design, implementation, and testing. During the final weeks of the project, the waterfall methodology will not be strictly implemented.
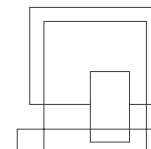
6. Time schedule

Total hours: 1100 hours split between 4 group members (275 per person)

Week	Date	Number of Hours (per member)	Tasks
1 (41)	11/10 – 15/10	23	Project description revision
	12/10 6PM		Project Description 1st Draft
	13/10 6PM		Partner Group Feedback + Mid Sem Evaluation
	15/10 4PM		Project Description Final
2 (43)	25/10 – 29/10	23	Analysis
3 (44)	1/11 – 5/11	23	Analysis
4 (45)	8/11 – 12/11	23	Analysis
5 (46)	15/11 – 19/11	23	Designing
6 (47)	22/11 – 26/11	23	Implementation
7 (48)	29/11 – 3/12	45	Java implementation/Documentation
8 (49)	6/12 – 10/12	45	Java/Website design/Documentation
9 (50)	13/12 – 17/12	45	Documentation
	17/12 1PM		Project Deadline
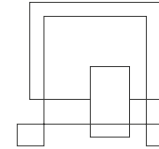
7. Risk assessment

Risks	Likelihood Scale: 1-5 5 = high risk	Severity Scale: 1-5 5 = high risk	Product of likelihood and severity	Risk mitigation e.g. Preventive- & Responsive actions	Identifiers	Responsible
Unreadable timetable	2	3	6	Agree on the color scheme. Asking random person to read the timetable.	Unreadable text, mixing days with months and weeks etc.	Robert
Invalid document received from the Head Department	2	4	8	Rewrite the document, convert it into a different doc type or rewrite code to match the doc type.	Java doesn't read the document properly.	Chris
GUI doesn't function properly	3	4	12	Work on the GUI in a timely manner so there is plenty of time to debug/fix errors if needed

Get Help if needed | Whether or not the GUI displays properly | Nina |
| Java doesn't compile due to syntax errors | 3 | 5 | 15 | Work on the code in a timely manner so there is plenty of time to debug/fix errors if needed

Get Help if needed | Error is shown in IDE that does not allow code to compile | Martin |
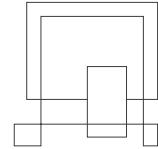
8. Sources of Information

Carroll, C., Juers, M. and Kent, S., 2014. Improving the Scheduling of Operating Rooms at UMass Memorial Medical Center. [ebook] Worcester, MA: Worcester Polytechnic Institute. Available at: <https://web.wpi.edu/Pubs/E-project/Available/E-project-030614-144337/unrestricted/Final_OR_Paper.pdf> [Accessed 5 October 2021].

Gaddis, T., 2015. Starting Out with Java: Early Objects. 5th ed. Boston [etc.]: Pearson.

Oude Vrielink, R., Jansen, E., Hans, E. and van Hillegersberg, J., 2017. Practices in timetabling in higher education institutions: a systematic review. Annals of Operations Research, [online] 275(1), pp.145-160. Available at: <https://www.researchgate.net/publication/320675938_Practices_in_timetabling_in_higher_education_institutions_a_systematic_review> [Accessed 5 October 2021].

Vagueware.com. 2021. Top 10 Free Scheduling Software For Schools And Colleges. [online] Available at: <http://www.vagueware.com/free-scheduling-software-for-schools-and-colleges/> [Accessed 4 October 2021].
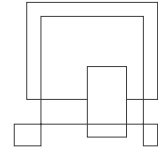
9. Appendices

GROUP CONTRACT

Project Name:	SEP PROJECT 1
Due Date:	17 dec, 6PM

Group 1:

1.1.1.1 **Name**	1.1.1.2 **Surname**
Nina Anna	Wrona
Martin	Rosendahl
Robert	Barta
Christian "Chris"	Foyer

Weekly meeting's rules

- We accept 3 absences.
- Discussing previous meeting minutes at the beginning of every meeting.
- Meeting minutes will be taken.
- Group members will be respectful to each other.
- We accept 15 minutes window to arrive.
- Meetings take place at Campus.
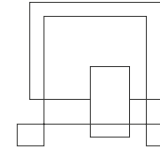
- Communicating delays and absences in a timely manner.

Expected schedule

- Tuition period: 1 times a week (obligatory) – Wednesday afternoon.
- Project period: 9.00-12.00 and 13.00-16.00 from Monday to Friday.

Supervisor meetings

- All members attend supervisor's meetings.

Additionally:

- We agree to help each other with our knowledge gaps.

Consequences:

1. Reading the contract as a group + warning.
2. Round discussing the issue.
3. Evaluation of self and peers regarding issue.
4. Supervisor meeting.

All people signed down agree to all the rules written down in this document.

Nina Wrona

Christian Foyer

Martin Rosendahl

Robert Barta