Project Report Clara Tschamon

Table of Contents

Background/Purpose

The task was to develop a hotel management system with the following functionalities:
- Room reservation
- Processing of guest arrivals
- Automatic billing
- Processing of guest departures
- Contingent management for travel agencies
- Contract partner management (travel agencies, companies)
- Possibility of control of all performed actions in the computer system
- Creation of various evaluations and statistics

It was clear from the beginning that we would not be able to implement all of them, therefore we defined the so-called “Minimal Viable Product”.
A minimal viable product (MVP) is a product that has the minimum set of features necessary.
Our MVP consisted of these features:

- Walk-In
- Check-In
- Check-Out
- Make a reservation through a webform
- Reservation Overview Table
- Booking Overview Table
- Create a simple invoice
- Simulate sending an email after making a reservation
- Import payment information from accounting system

In total we had 10 weeks for planning and deploying.
At the beginning we were 5 people (Dominik Aigner, Samuel Jäger, Nina Hartmann, Ida Mazinger and Clara Tschamon).
After 6 weeks Ida could not continue working with us, so we were half a teammember less. (Even though she was on sick leaf, she managed to help us sometimes.)

The purpose of the project was to learn the agile working method scrum and to expand our programming skills.

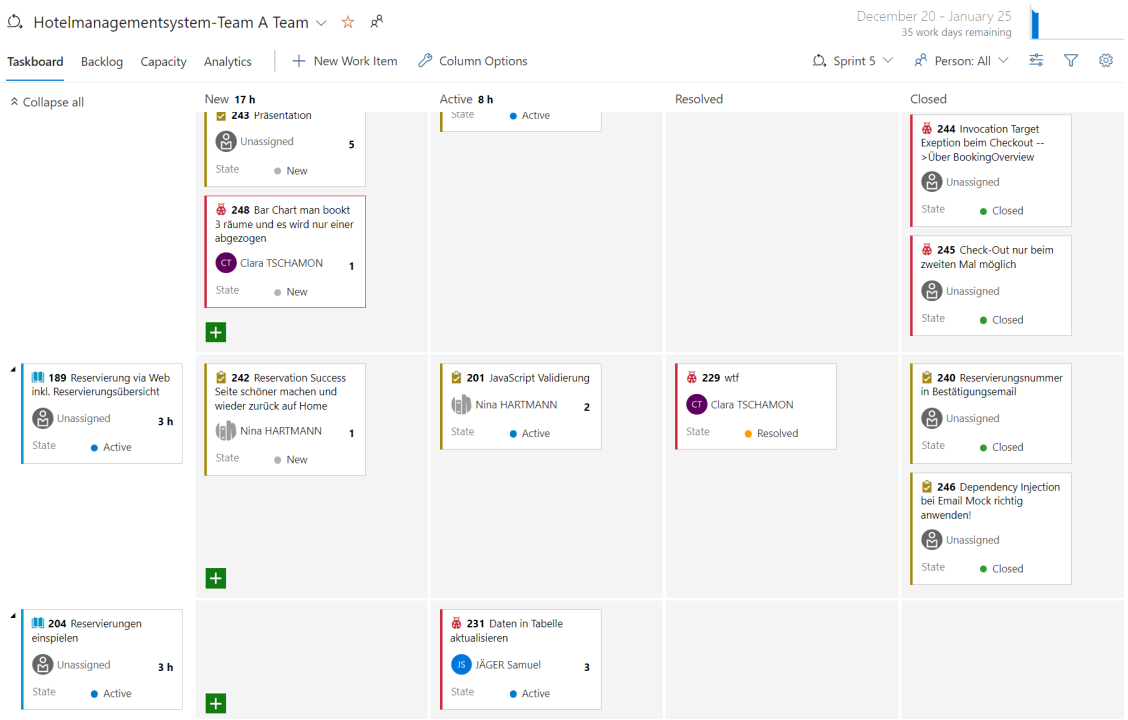
200 words

Methods

The working method we used is called “Scrum”.
The origin of the term scrum is the sports rugby. The term scrum was chosen because it implies teamwork.

In Scrum, small teams work together to complete a series of defined tasks, called "sprints," in a set amount of time of typically two to four weeks. Each sprint begins with a planning meeting, during which the team determines which tasks it will work on during the sprint. The team then works on these tasks until the end of the sprint where it holds a review meeting to demonstrate the completed work to stakeholders and discuss in the retrospective what went well and what could be improved in the next sprint. On that day the taskboard has to be filled with the user stories which should be completed by the end of the next sprint. The team has to split-up the user stories in smaller tasks and estimate for each task, how long it is going to take for completing it.

Here is an example of how a taskboard looks:
On the very left we see the user stories. On the right handside of the user stories we see the tasks. They have the states: New, Active, Resolved and Closed.
When a task is done, it is resolved. To drag it to closed, another teammember has to confirm that the task is completed by looking at the implementation.



Our sprint-duration was 2 weeks. After those 2 weeks we had to present our new feature(s) to the class and to the lecturers.

We always had a “Scrum Master” in the team. The Scrum Master was responsible for scheduling meetings and holding the meetings.

The meetings in scrum are called “daily standups”. A daily standup is always maximum 15 minutes long and should be held every day. Our team was very consequent with that and held at least 5 daily standups a week. Some of them were held online.

At the beginning of the project our team did a teambuilding day. We met at Ida's, ate pizza together and played a game in which we had to solve a murder case.

Our team agreed to meet every Thursday to do programming together. This "CodeThursday" helped to keep us motivated and it lead to a good cohesion within the team.
In addition to the „CodeThursday", we helped each other a lot over Discord which is a online communication platform. No one had to wait more than three hours for help.

Some technologies that we used:

- Figma for prototyping
- Intellij as development environment
- Java as programming language
- GitHub as code-sharing platform
- Tomcat as web container
- SceneBuilder for designing the User Interface

450 words

Results/Evaluation

Our result is a very well working software with a great design. We are all completely happy with the outcome.

There are a few features which we would have wished to implement but simply didn't have the time to do so.
An example is that it is not possible with our system to edit bookings and invoices. Also it is not possible to edit a reservation except while doing a check-in.

Belonging the design we don't have more wishes. In the lecture 'Software Engineering' we had the chance to let a professional User Interface designer test our software. We got a lot of constructive feedback. We took the chance and designed our system even more user-friendly.

What the project tought me about teamwork is that I still have to learn not to do everything myself. I spent between 15 and 20 hours a week on the project. That was too much. Only once I took a Sunday off. After Ida was not able to work anymore, I was constantly frustrated because I felt like I have to do almost everything myself and if I were doing as little work as the others and spend as little time as the others, nothing would have moved forward regarding the tasks. I told the other team members that I don't feel well because I am working that hard but the answer that I got was that I simply should stop working that hard. I think that I have to learn to talk more directly to them. What I should have said to them is that they should work more so that I don't have to work that hard.

270 words