Use Cases:

I. Sign Up – Alexander Ramharter

II. Login - Alexander Ramharter

III. Look at and search for films/screenings - Oliver Schweiger

IV. Buy ticket – Utz Nisslmüller

V. Look at, update (employees only) and cancel bought tickets - Utz Nisslmüller

VI. Manage films - Oliver Schweiger

VII. Manage screenings – Oliver Schweiger

VIII. Employee administration - Yasin Ergüven

IX. Hall administration – Yasin Ergüven

X. User administration - Alexander Ramharter

XI. Role Based Access Control - Alexander Ramharter

More detailed description of work distribution:

Alexander Ramharter: I was very interested in how Login and Sign Up actually works so I chose these two use cases. It took me a little to get back into PHP, but after a while I got pretty confident with using it. After seeing the implemented use cases we thought it would be nice that different users have different abilities on the webpage. And since I already used global session variables for log in I implemented that use case as well. In the end we decided to add additional operations for the user such as changing the password or delete the account.

Oliver Schweiger: Again, I was responsible for the Use Cases regarding the interactions with our movies and screenings (III, VI, VII). I also oversaw setting up the connection to MongoDB within PHP. For this purpose, I chose the newest version of the MongoDB Driver 1.5.3 that fully supports the latest PHP version. Like all team members, I also helped with the creation of the migration system from MySQL to MongoDB via Java.

Yasin Ergüven: For MS2 & MS3, I was responsible for the realization of the use cases “Employee administration” and “Hall administration”. I have added the CRUD operations for these two use cases and tried to demonstrate all information as simple as possible. The great challenge was the realization of the unary relationship for employees inside PHP and the representation of the generated data in an understandable way to provide a functionality for an efficient administration of information.

Utz Nisslmüller: For MS3, I was responsible for transforming the ticket.php site to be able to work with our newly implemented MongoDB structure. The functionality is equal to the one in MS2, meaning that the user can’t tell whether he is using the SQL or MongoDB version of our project. I also had to rework the parts of user.php which were responsible for ticket display as well as the ability to cancel an existing reservation. Finally, I was able to adapt all CRUD operations to our new data model seamlessly. Apart from this, I also expanded our Java program’s functionality, and it is now able to handle both the SQL and MongoDB databases.