TEAM 36: UCLH PEACH and Microsoft - Cancer Team

Date: 24.03.2017

Team Members: Ovidiu-Horatiu Ilie, Georgiana Birjovanu, Berat Baran Cevik

Project Overview

In the past two weeks, we prepared a presentation explaining the stage that our project components are currently at along with the tasks that we have achieved, the requirements that we have met and our plan until the deadline. We also demoed our prototypes to our TA and got some useful feedback in return.

We contacted the front-end team to help us with the front-end of our integrated website. Authentication system has been made available for a basic use online on an Azure VM. For collaborative document editing MongoDB has been researched as an alternative to SQL. For internal messaging system, we considered switching to an auxiliary solution to Rocket.chat in order to create a chat platform which meets requirements and which is deployable to Azure. Using this new solution is also going to help future developers to easily modify and improve the project.

We also planned our final project website and decided what information to add to the current version and what information to edit. We discussed and improved our plan to integrate our components without any issues. We finally decided exactly what to do in the next weeks and how our platform should look like at the end of the project.

Meetings Summary

Meeting #1: 14.03.2017**,** 1.21 MPEB

Description: We met in order to check the latest work, analyse what is still missing and how all our work can fall together into one piece easily. We also created a progress check presentation for Yun Fu, which we sent to him later.

Meeting #2: 15.03.2017, Cruciform Hub

Description: The meeting took place in order for us to check the progress and work together as a team, providing feedback on what each of us had been doing individually. Finally, we have created an abstract for the project, which was later sent to the department and which will be used for the final system showcase.

Meeting #3: 17.03.2017, 1.05 MPEB

Description: During lab hours, we presented what we have achieved and what are the struggles we are having. We discussed with other teams in order to present how the system works so far and decided what to focus on until the end of the term.

Meeting #4: 20.03.2017, 66-72 Gower Street

Description: We met with our supervisor, Yun Fu, in order to show him the progress we have made so far. He gave us feedback and told us what to improve until the final presentation in April. We clearly analised all the components, with clear explanations and reasons to implement certain features before the deadline.

Meeting #5: 23.03.2017, 1.21 MPEB

Description: This meeting was held in order for us to establish what we should improve on our website and how we should split the work for improving it between us. Moreover, we worked on an integration between our components and made some progress in combining everything.

Meeting #6: 24.03.2017, 1.05 MPEB

Description: We met Yun Fu and our client, Dr Navin, for the last time on this term. We evaluated the project so far and what it has to be done in the next period. We also worked for our integration of the components.

Tasks completed and project progress

- Integrate the 3 components within one website
- Make authentication available as a web service for other teams
- Got in touch with the Front End Super team in order to integrate the three components
- Researched how to implement MongoDB instead of mySQL
- Explored new solutions to replace Rocket.chat.
- Started development with a new solution to ensure it will be ready and will meet requirements when the project is delivered.
- **The project is running on time**

Problems to be resolved

- Find an easier way to integrate the messaging and editing systems with the Keycloak server
- Find a way to make Keycloak available as a web service to the other teams, instead of OAuth
- Decide if our team should implement MongoDB for the collaborative document editing and deploy it on Azure
- Integrate the three components in the same common front end framework
- Speed up the development to compensate the time spent on Rocket.chat.
- Deploy an initial version of the app to ensure its deployability.

Plan for the next weeks

In the next weeks, until the final deadline, we plan to finish our project and to make it fully available to our client. This means that we plan to integrate our components together, while also trying to provide them externally to other teams where needed. We will make sure that bugs are reduced to a minimum by testing the solution and we will deploy a working version on Azure. We will also write some guidelines about using and deploying our components, while also providing the code to the client on GitLab.

Regarding each component, we plan to create ways to make the auth available as a web service and easy to integrate. The document editing will be fully integrated with a database, while the messaging system will be customized to our client's needs.

We plan to implement all the requirements from our MoSCoW analysis and to put everything together into one web application.We also plan to update our website with all the information needed and to prepare our video and presentation before the deadline.
Finally, we will keep in touch with the client, in order to present him a final version that he can start testing and we will help other teams in understanding and using our project when needed.

Individual contribution

Ovidiu-Horatiu Ilie

During the last two weeks, I was responsible for the development of our authentication solution. Working with Keycloak, I managed to find a way to deploy an OAuth solution for others to use. Moreover, I deployed everything on a Windows Server VM on Azure, which still was not made public to others. Moreover, I managed to find a way to integrate our two other components under the Keycloak auth by creating realms. What I still need to do is to customize the theme of the component and to find a way to provide the authentication as a web service. Apart from this, I was discussing a lot with the client, keeping him updated with the project status. Moreover, I talked to students within the project which had experience with Azure VMs or Docker and who helped me understand how to deploy easily. Finally, I helped in creating and updating this report, the progress check presentation and the project website.

Georgiana Birjovanu

In the last two weeks I have been focusing on finding a common way of gathering the three components all together. After setting up the meeting with our module supervisor, Mr. Yun Fu, we decided to take into consideration his useful advice for the next phase of our app development. Therefore, we agreed to implement the same type of database for the three components for the moment. This means that the Collaborative Document Editing will use the MongoDB database instead of the mySQL one. I have started to do more research on this topic and the next step is to deploy the component on Azure in order to test its final functionality. Moreover, on the next lab session we will get in touch with the representative of the UI Super team, who will help us update our components' front-end with their standard UI classes. The last functionality to add to the Collaborative Document Editing would be the button which can be clicked in order to save the input. At the moment, this option is active but can be accessed via the menu of the collaborative pad.

Berat Baran Cevik

In the past two weeks, I explored alternative solutions to Rocket.chat since it had become very hard to customise due to lack of commenting and documentation. Also, I could not make any progress with deploying Rocket.chat on Azure because of its level complexity. In order to deliver a working internal messaging system which meets “Must Have” and “Should Have” requirements I found an auxiliary solution which will lead to a good chat platform. Moreover, it will be much better documented, much easier to develop further by the programmers who are going to work on the project, much easier to modify and much easier to deploy. Since I have gained experience and knowledge in Node.js, Express.js and MongoDB, the project should be delivered with meeting all “Must Have” and “Could Have” requirements.