StudentWise

Minutes of meeting

Denis Nagayuk, Evgeniy Terziev, Jonasz Kądziela, Karina Kozarova

**Table of contents**

[**Details**](#_xxq1jrcwbd5r) **2**

[**Agenda and notes**](#_fql6ihmnanij) **2**

# 

# Details

|  |  |
| --- | --- |
| **Project** | StudentWise |
| **Topic** | A showcase of the project’s current state and evaluation of progress |
| **Date** | 10.12.2019 |
| **Time** | 14:40-15:05 |
| **Location** | Building R1 Room 2.22, Fontys University of Applied Sciences, Eindhoven |
| **Written by** | Karina Kozarova |
| **Present** | Yuzhong Lin, Mentor  Denis Nagayuk, Group member  Evgeniy Terziev, Group member  Jonasz Kądziela, Group member  Karina Kozarova, Group member |

# Agenda and notes

|  |  |
| --- | --- |
| **Topic** | **Notes** |
| **Show progress on API** | We showed the extra endpoints we have on the API and explained why we will need the groups table and votes. Also, we explained why we added event participants and why events have different types. |
| **Explain how will the votes system work** | We explained how we took care fo this in the database, explained the business logic behind it. Explain why we choose not to be able to delete finished events. |
| **Git repository for the API** | We added a pipeline - an automatic job will be run on every commit by running the rspecs (tests). MR can be merged only if they are passing. We also added automatic deploy. |
| **Backend C#** | We have multiple objects that interact with the server. We explained the Server class, which provides authentication. Explain why the UserSession also contains info of the user now.  Explain the Enumeration of users that returns a list of the users. We now have a class Event which provides properties with all the information for the events and static methods to create, modify and delete events. Also explained how we can update the properties. Specifically explained how the participants’ properties work. |
| **User interface** | Explain the events tab. Explain why we can’t delete events that we have not created. Explain how the delete works and that we also delete them from the server and from the GUI. Explained why we can’t vote yet (no groups). Explained why karma is an extra feature. Explained why we will add filters - for filtration and pagination purposes. Explain what the default implemented behaviour will be. Explain how the expenses tab will work.  We explained why we create an component for the events. |
| **Voting API** | We explained in details how the voting part of the API works and all the conditions that it has to go through. We also explained why at most 3 ppl will have to vote. We also explained what happens if someone disagrees that the event is finished. |
| **Mentor’s feedback** | Mentor feedback to be able to add the time of the event and to design the event type so that it is easier for the user to see the difference. Also, we should think of renaming it and to not be just events. We need to think where to put the vote buttons. |
| **Explain what’s missing** | * We explained out progress - ⅔ of the API are already done * We explained why some of the features until now were more complicated than we expected so it took more time than we estimated |
| **Planning** | We explained how we are going to schedule our remaining tasks. |