

## Report Week #2

In the 2nd week, we reviewed our project again, and came to a single decision that we should start another project, since we decided to use bootstrap and templates in order not to complicate our task. and in this project we will use Mysql database and ORM to filter the data. So we decided to take the ToDo2 project. And in the individual plan so:

Utesh Aman: he imports the gorilla/mux and logrus package since we need the HTTP route and login to be convenient. Then he wrote on our project about the packages that need to be imported and the `t_alive()` function, which will respond `{"test": true}` to the client every time it calls.

Tolegenov Meiirzhan: He wrote the `init()` function to configure our Logrus Logger settings. `init()` will be executed when the program first starts. And he initialized our Gorilla/Mux HTTP router with a walrus operator. Since he directed `/t_aliveHTTP` to receive requests for the `t_alive()` function. Our router will listen on port 8000. And write `func index()`.

Yermurat Nurmukhanov: He primarily set the content type: `application/JSON` so that the client software understands the correct response. And then he connected our Mysql database which will use Docker to launch the container and expose the port 3306 to the localhost. And then install the GORM libraries and dialects.

Imangali Zholdas: He connected our database through the code and then initialized our MySQL connection to our database using GORM. It supplies GORM with a neat automatic database feature where it will create a database table. Defines the `Todoitem` model. Next, it starts automation for our MySQL database