International Information Technologies University



**Web Technologies**

Topic: IITU Schedule

Midterm Report

Schedule IITU

Submitted by:

Nursultan Alfaraby

Muldashev Turar

Group: CSSE-1801 SWD-1

Submitted to: Mukazhanov N.K.

2020, Almaty

Content:

1. Introduction
2. Main Part
3. Conclusion
4. References

**INTRODUCTION**

This is our Mid-Term project report on creating templates for our “Schedule IITU”. We will go through the two sides of our “Schedule”, and planned future functionality of the web app, and how it is present in the interface design. In creating this template pages we used HTML, and Bootstrap 4 as a solid base.

**MAIN PART**

First, we decided to divide our web app functionality into two parts:  
User part  
Administration Part

In terms of functionality

Basic User Privileges

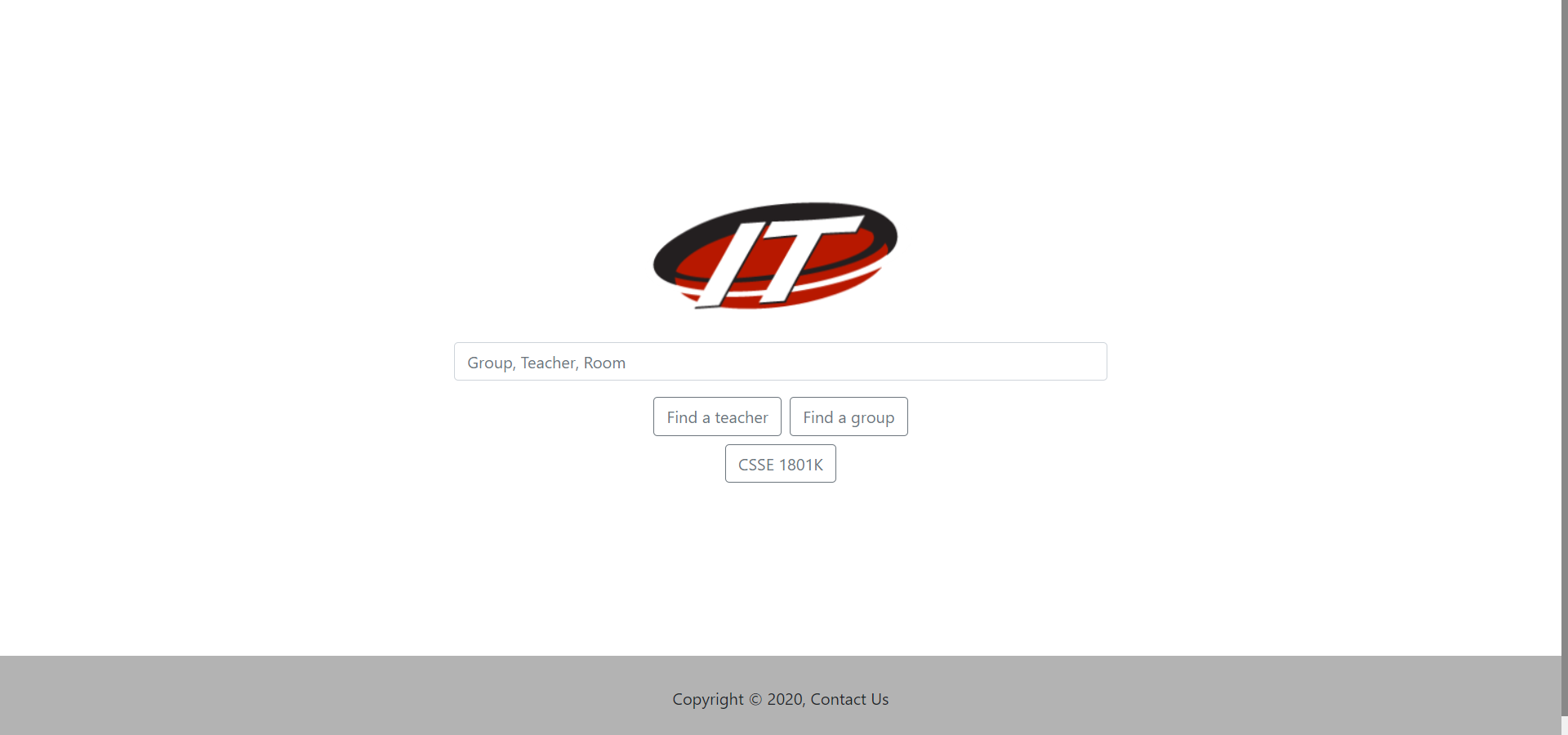
* Search Group Schedule
* Search Teacher Schedule
* See the recently searched schedule
* Leave a message to Dev. Team

Administration Part

* Adding records about new courses, departments, teachers, groups.
* Based on entered information, constructing "Schedule"
* Project, based on Admin inputs, will dynamically show applicable options in terms of weekday, hours and cabinet
* The Input constraints will be based on chosen Department, Teacher, Group(s), Cabinet or Cabinet Type, Weekday and Hour.

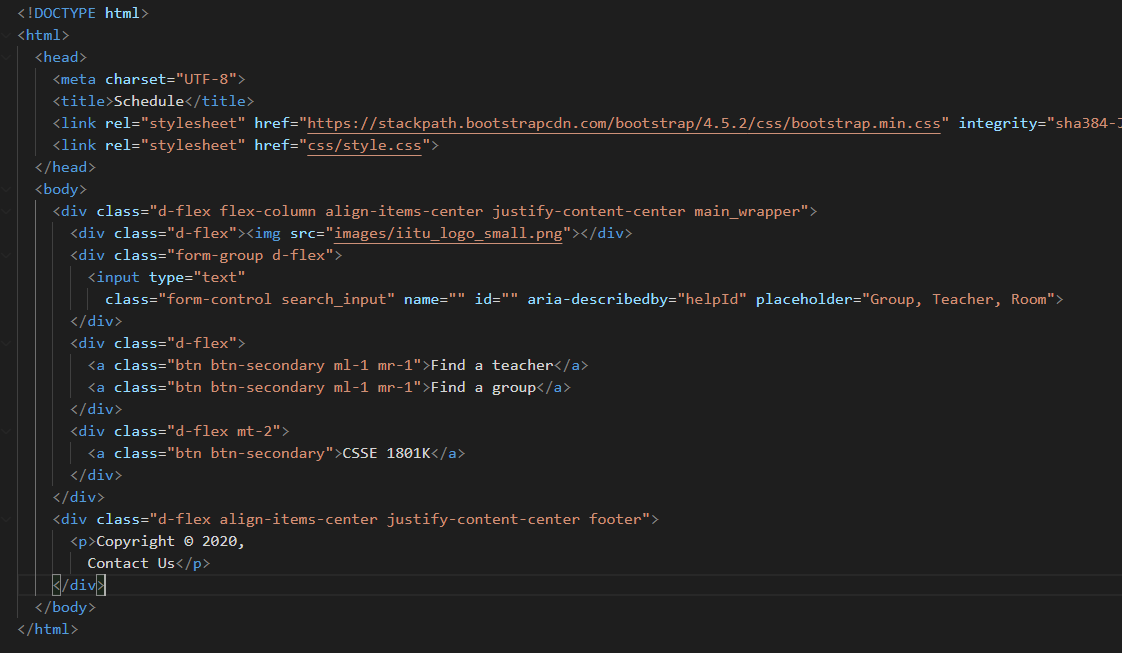
**Templates for Basic User:**

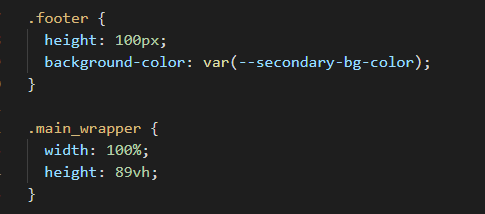
**Welcome page:**



Here User can search some groups or teachers, and go to the last searched schedule that he viewed, if such is present.

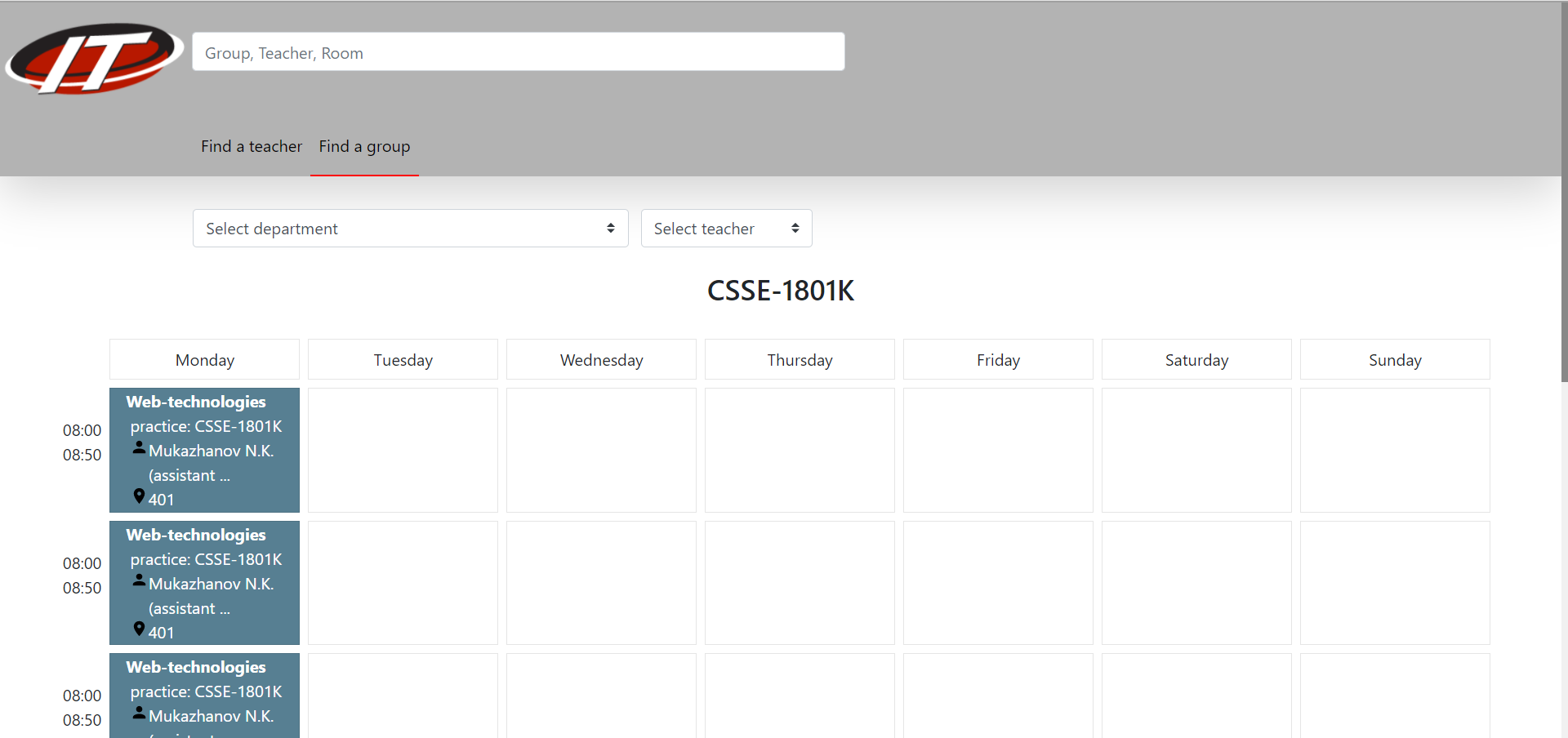
In this page not so many elements, basically we created flex-box elements which sizes are almost the whole page, and elements are centered using flex properties. Also, bootstrap has already defined classes to work with flex-boxes. Some of them are: d-flex, flex-column, flex-row, align-items, justify-content. They can be used separately or combined together.





Because of bootstrap, the css files are not large anymore.

**Schedule Page:**



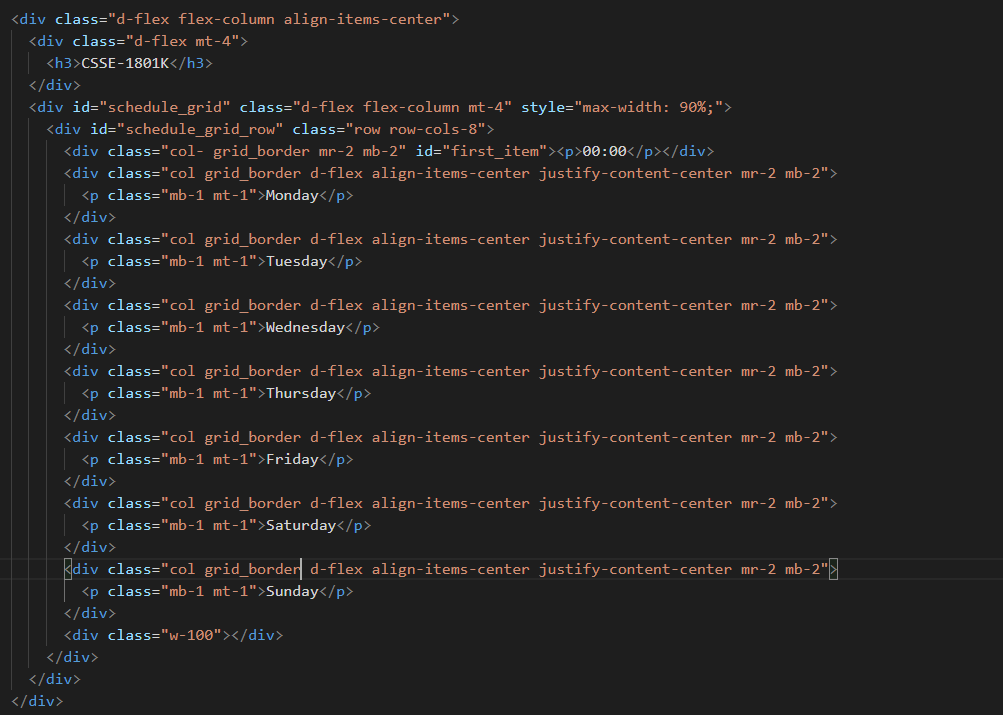


The schedule page consists of 3 parts. Header, schedule-grid, and footer. Footer is the same as in the previous page, the header is also created using flex-box(items aligned across the start of the flex-box). Bootstrap has already predefined classes for grid. The time cell is actually part of the grid, however without a border.

This is header part of html:



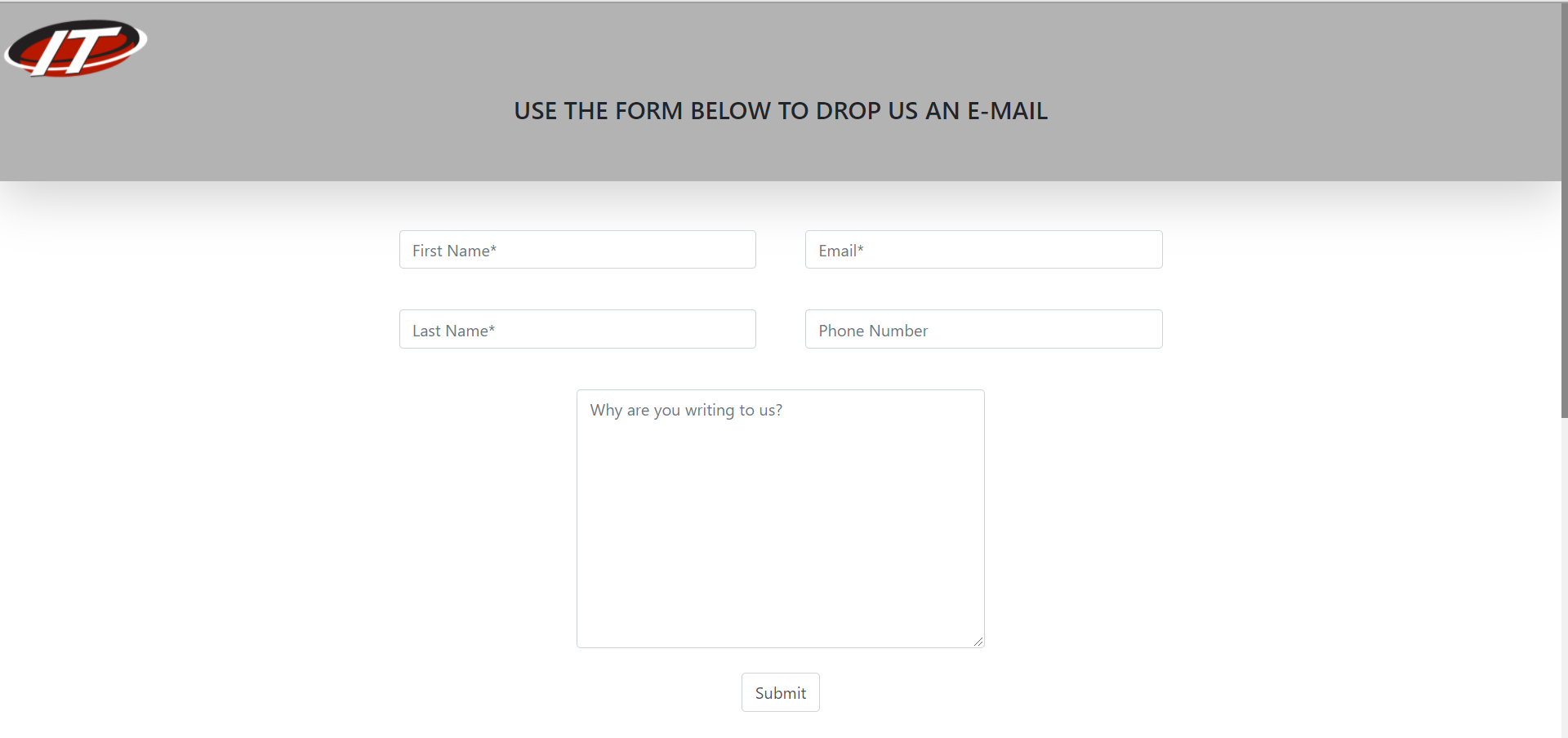
This is actual grid:

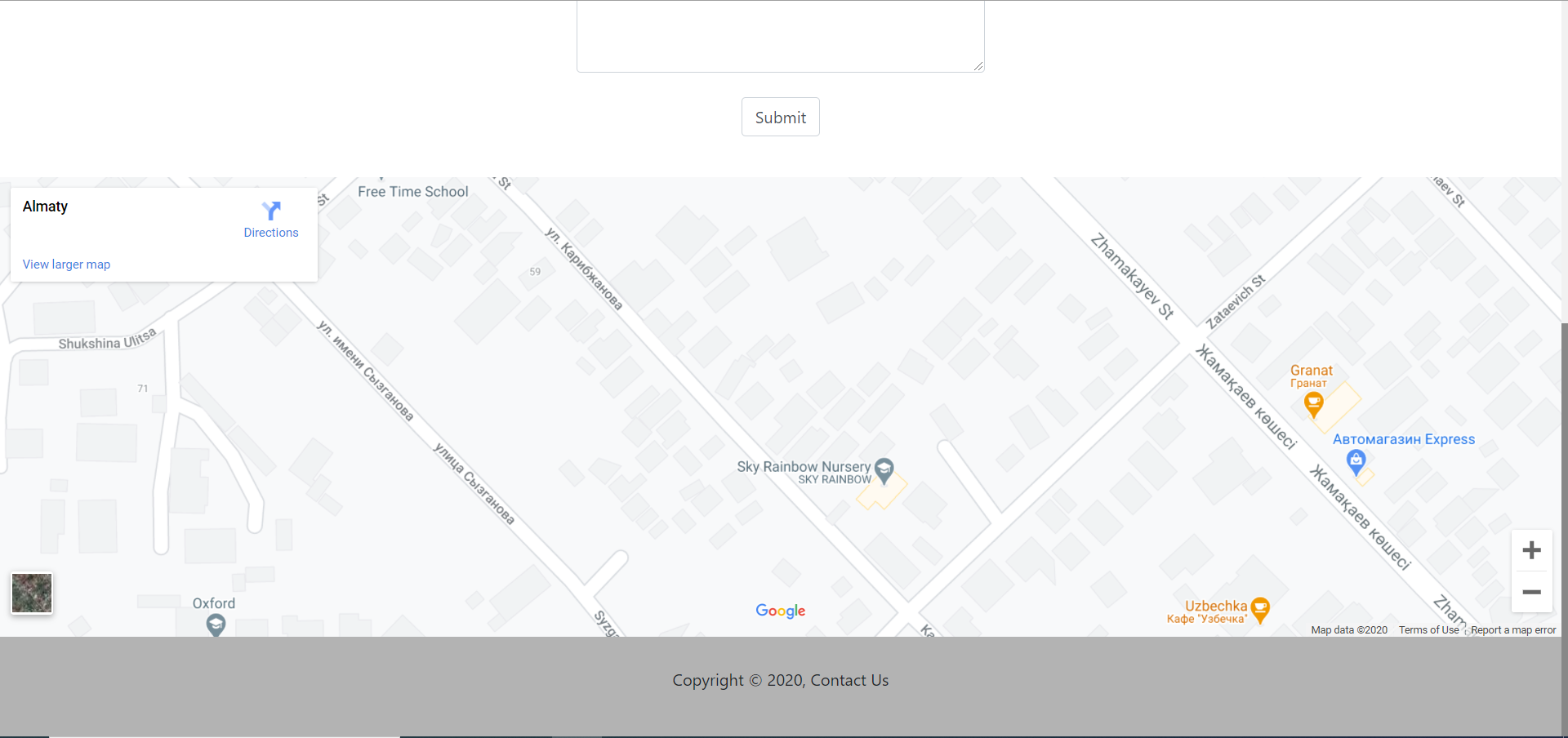


The rest of the grid generated by the JavaScript, which is following:



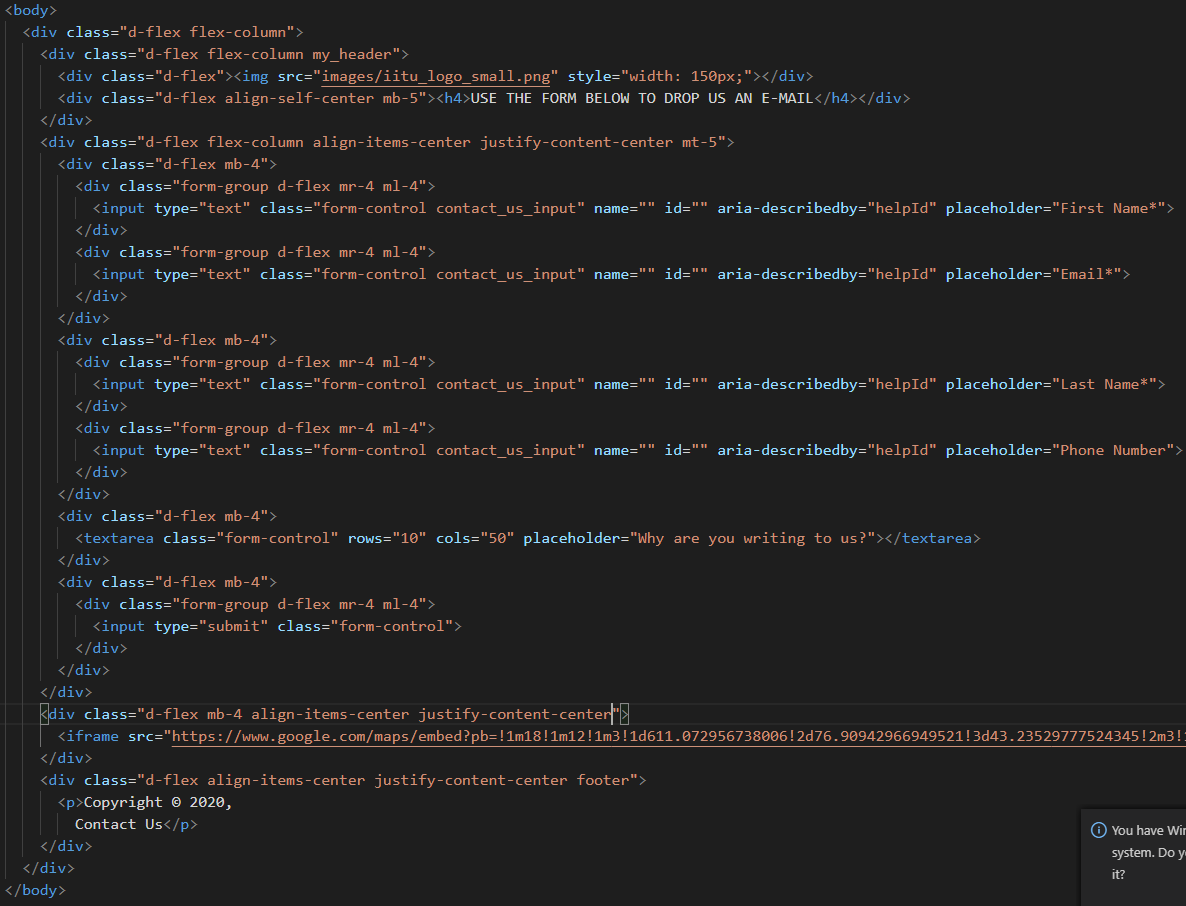
**Contact us Page:**





This page consists of 3 main parts: header, form where you can submit your question with a map where you can see geolocation, and the footer.

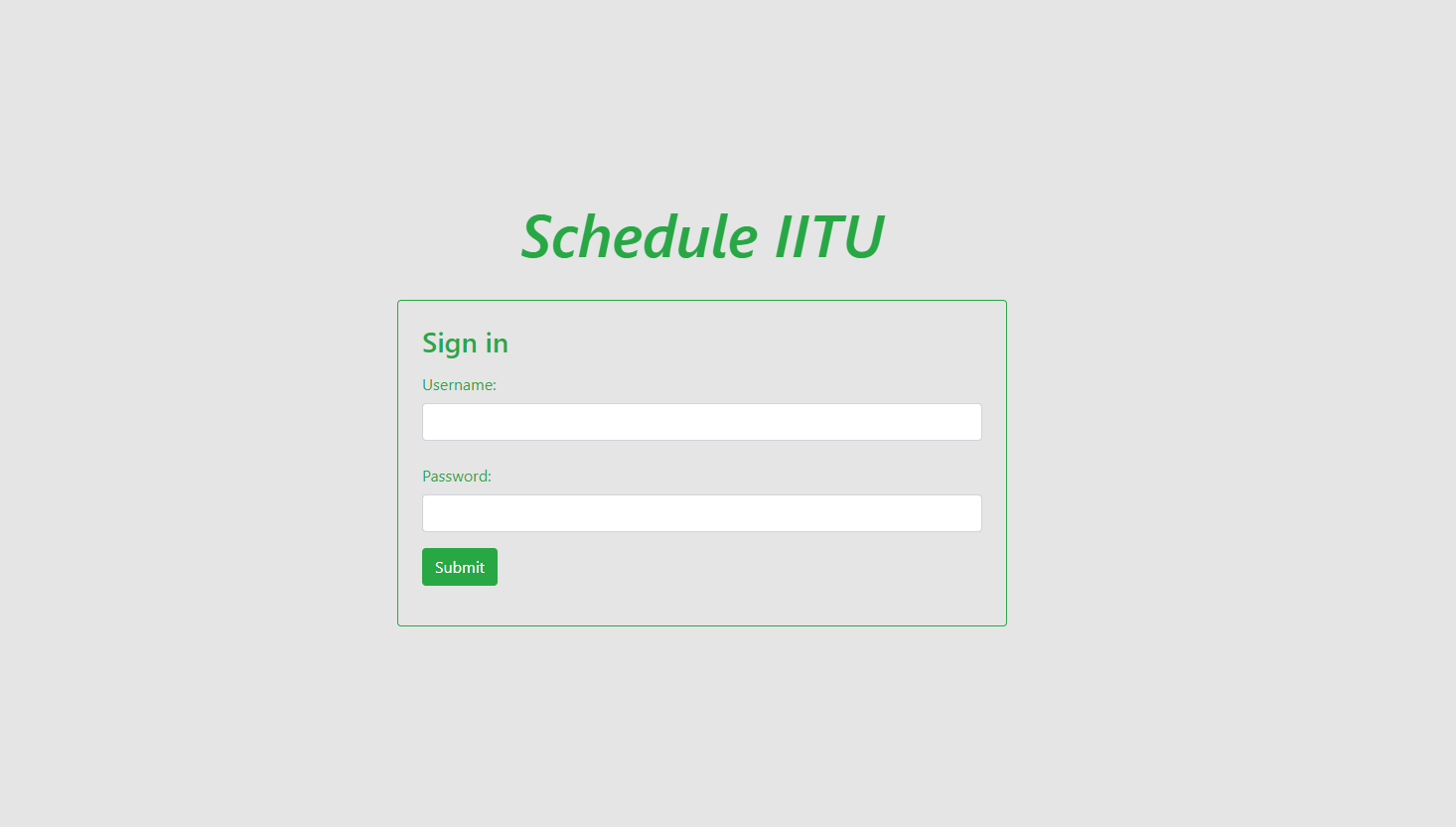
Html:



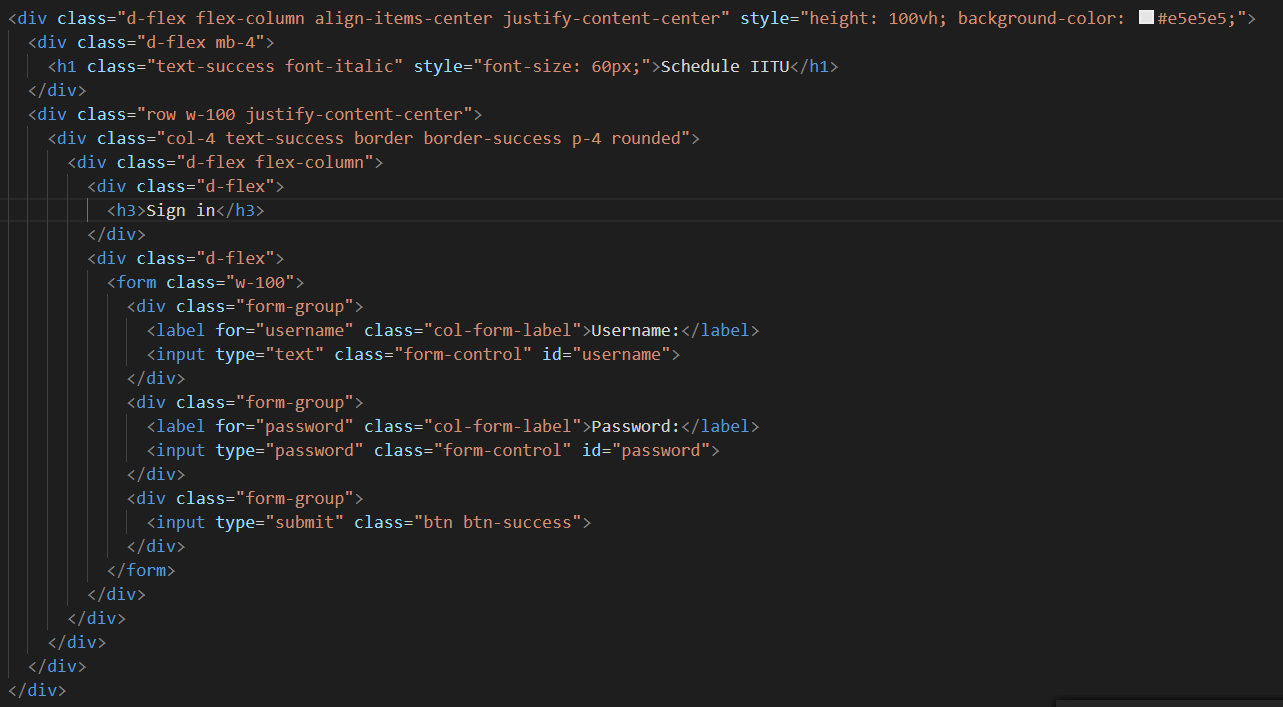
The css code was reused from previous pages, and bootstrap.

**Templates for Administration:**

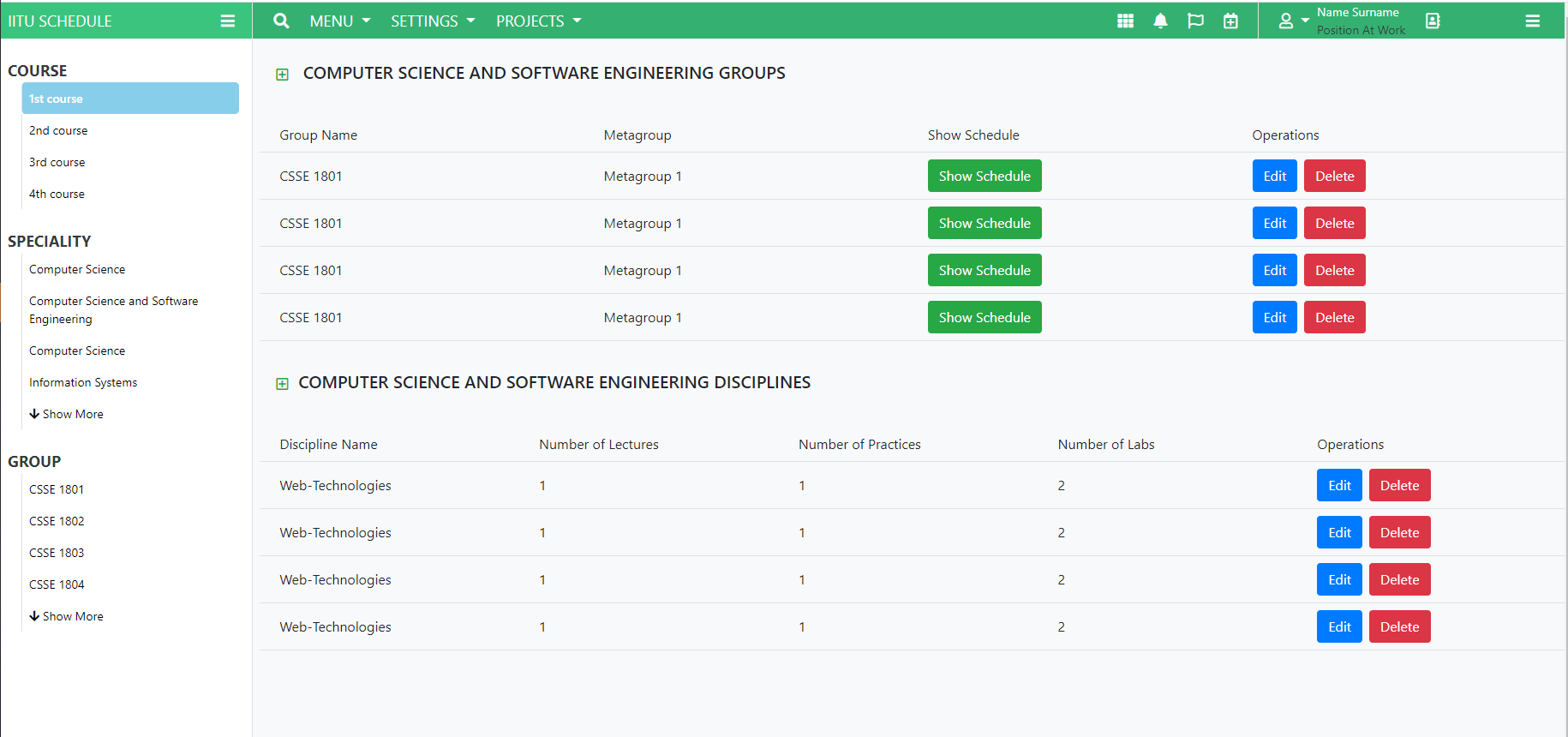
**Sign In Page:**

This is a page with a simple form, that will block unauthorized users from accessing the control panel where the main motion will be happening.

Here how it looks behind the scenes:



**Main Administration Page:**

This is a page where all our big things will be happening. The concept is not final, and can go through many changes, but the basic idea stays the same. Here, after authorization, users will be able to add new records about courses, departments, teachers, groups and cabinets. After initial data will be inputted, the full access to schedule creation will be given.

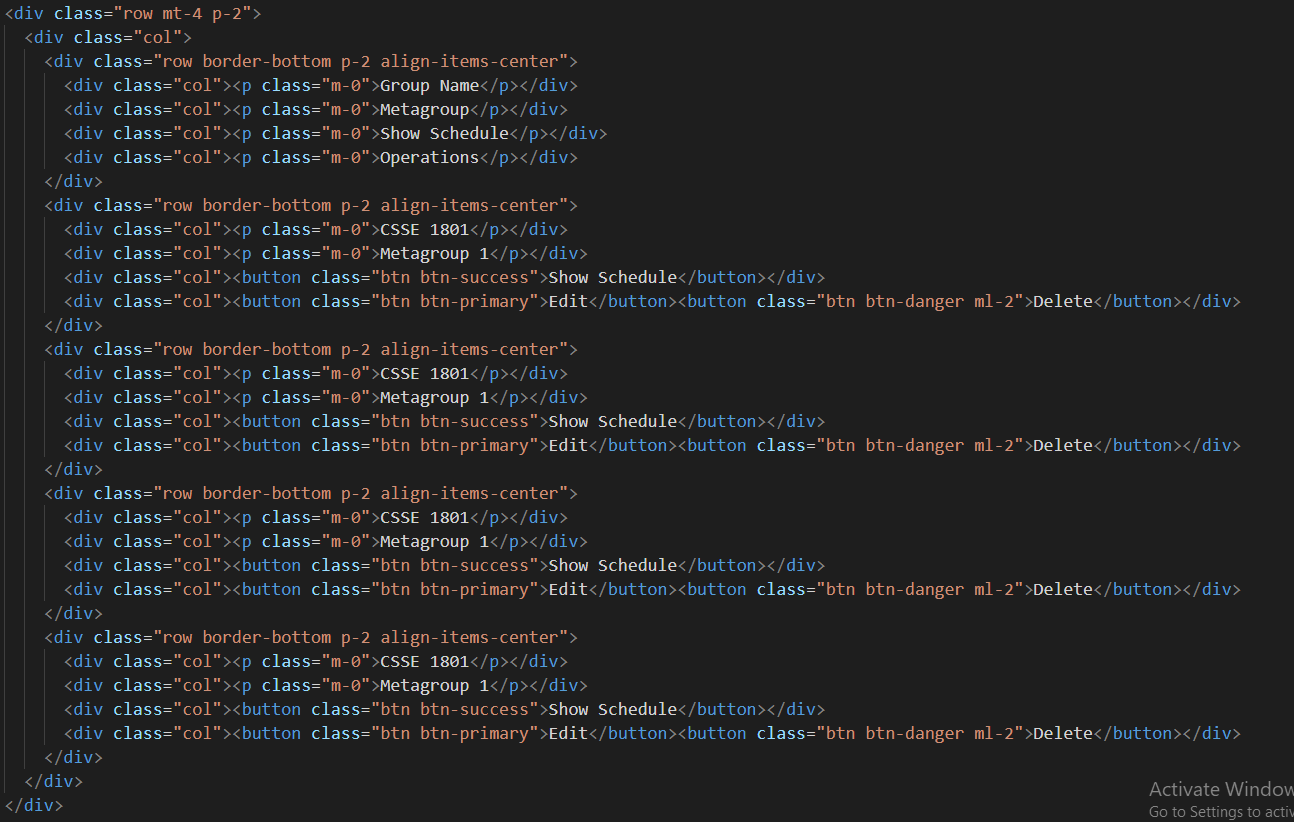
Users, based on chosen Department, Teacher, Group(s), Cabinet or Cabinet Type, Weekday and Hour, will be able to check if Given Cabinet in the Given time will be available. After that users can create a bond, and add a record to Schedule.

Also, User will be able to see other relevant info based on selected configurations:

Here how it looks behind the scenes:







**CONCLUSION**

In this laboratory work, we created two versions of one template with and without css framework. After that experience, we can confidently say that using frameworks is going to speed up your process of development. However, you can not fully ignore native css code, because sometimes you need something more specific. Also, you can use css frameworks as efficiently as possible only if you already have experience with pure css.

**REFERENCES:**

HTML, CSS, JS “handbook”:

<https://www.w3schools.com/>

The mostly used documentation:

<https://developer.mozilla.org/en-US/docs/Web/Reference>

Bootstrap 4.5:

<https://getbootstrap.com/docs/4.5/getting-started/contents/>

Some Icons Used from Font Awesome:

<https://fontawesome.com/>

Inspiration source for admin page: <https://colorlib.com/wp/free-bootstrap-admin-dashboard-templates/>