Philipps-Universität Marburg
Fachbereich Mathematik und Informatik
AG Programmiersprachen und -werkzeuge

Philipps Universität Marburg

A Web-Based Interactive Schedule Planner for Lectures and Exams

Motivation

Each semester the appointments for lectures, exercise classes and exams need to be planned. This requires checking that no appointments of courses overlap, which are mandatory or likely to be followed together. For the big basic courses, the appointment are fixed for years and only change under very rare circumstances, but compulsory elective courses can change their schedule. And since they sometimes are offered irregularly, the schedule of their last run may not be compatible with the current semester. Creating an optimal schedule automatically is very difficult, because several boundary conditions exist for scheduling, which are not easy to formalize. Therefore, a graphical tool is needed that visualizes the current schedule, allows to change it by dragging events to a new place and highlights possible conflicts.

Assignment

For this assignment an interactive web page should be implemented with React and the Material UI, which is fully browser-based, i.e., without a server component. The web app should use the Node Package Manager for the build configuration and be hosted as a Gitlab Page. It must be possible to upload an initial schedule based on the schedule of previous semesters or a previously saved schedule. And it must be possible to select which lectures to include in the schedule.
Two planning modes must be available: Firstly, lectures and exercise classes are scheduled for the whole lecture period and each week has the same schedule, with the extension that some appointments may be bi-weekly and some courses may be block events; block events may take place outside the normal lecture period. One course can have multiple lecture and exercise classes per week. Secondly, exams are spread over a period of three or two weeks and fall into two examination periods. One class can only have one exam per exam period; courses with oral exams may or may not be included in the planning.
Information relevant for the planning are the lecturer, kind of event, the room and the expected number of participants. It must be possible to specify and edit this information, and some of this information may be optional. Furthermore, it should be possible to consider conflicts between lectures. For this purpose, as part of the assignment, the suggested study plans which are part of the exam regulations of all study programs should be analyzed and dependencies should be derived. This is, courses suggested to be studied in the same semester of any of the programs are in strong conflict and courses in adjacent study year have a weak conflict. Based on these information, the planner should check for conflicts in the planning and mark them; also other kinds of conflicts should be considered like overlapping lectures by the same instructor or in the same room. Conflicts should also consider patterns like bi-weekly events and block events; in particular block lectures can potentially also overlap with the exam periods, which should be detected.
It should be possible to move appointments in a convenient way such as using drag-and-drop. The current planning should be stored in the *HTML 5 local storage* such that the status is not lost if the web page is closed accidentally and planning can be continued when the page is opened again. Furthermore, it should be possible to download the current planning and to upload it again, for example, to revert to a previous planning status. It should also be possible to save the current planning in the form of a human readable list that can be used as instructions for booking the appointments in the course catalog and reserving corresponding

Info

Teaching Organization Tools

React, Node, MUI

Typescript (or Javascript)

FoPra or Project Work

3–5 Persons

Theory
Practice

Contact

Prof. Dr. Christoph Bockisch

Interested?

Info Meeting: April 18, 2024 at 16:00h

Participate in the non-binding info meeting! For further information on the meeting and short term changes, please join the ILIAS course.
`https://uni-marburg.de/j0kmMT`

ILIAS

rooms. In this list, it should be visible, which appointments have been changed compared to the initial planning.

Requirements

Core Requirements

The core requirements are described above.

Optional Requirements

- When dragging an event, the planner may highlight possible target time slots, where no conflict would be detected.
- The planner should have a read-only mode, in which a schedule is only displayed but cannot be edited; in this mode, the schedule information may be read from a file that is (manually) uploaded in the Git repository of the Gitlab Page project.
- It should be possible to apply filters to the view of the schedule planner, e.g., by instructor or by study program.
- The schedule (possibly with applied filters) should be exported to a PDF.
- The planner should offer possible choices of rooms for events based on the expected number of participants and room-planning for overlapping events.
- It should also be possible to include a list of possible rooms instead of just one.
- When the appointments of basic courses are moved, a warning should be displayed, because these appointments should usually not change.