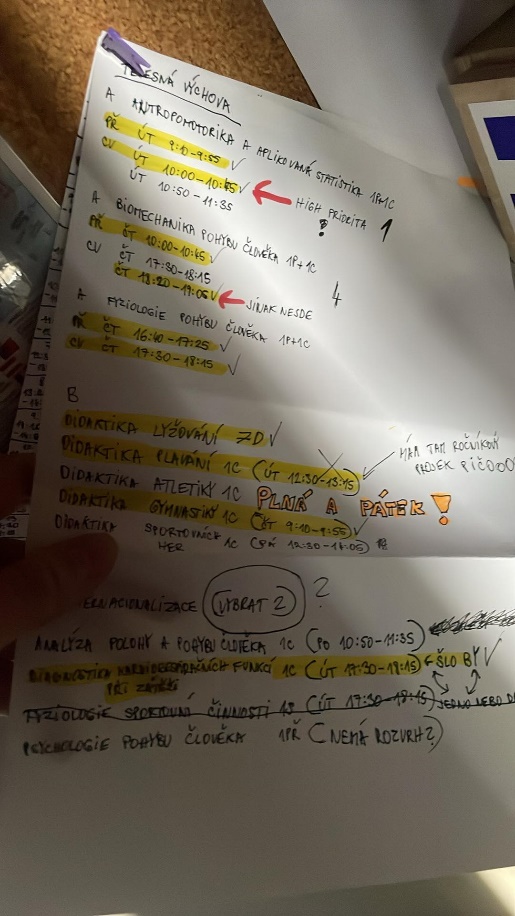
# **Name:** **Timetable Maker**

# **Vision:**

Current situation:

Students have to create their own schedules before the official timetable selection day in order to choose the best version of the timetable for their needs (more free time, less school days, part-time jobs, lunch breaks, sports, hobbies, etc.)

That might look like this:

Obsah obrázku text, Obdélník, Paralelní, měřicí tyč

Popis byl vytvořen automaticky

Students have numerous choices to select from, but the OSU portal doesn't offer the option to display them all at once, making it challenging to create a schedule.

First-year students, in particular, struggle with scheduling and often end up waiting in long queues at the Academic Affairs Department for help.

Future:

The project encourages students to take an active role in shaping their class schedules before the official selection day.

Students have the flexibility to customize their schedules to suit their academic and personal needs.

Making the scheduling process more efficient and less stressful for both students and administrators.

# **User roles:**

students

OSU students A. Kračmarová and V. Děcká, in addition to the team members, will actively use the system to test its features, functionalities, and provide feedback.

# **Functional requirements (What the system will do):**

System will:

* verify student logins.
* load students’ subjects from the Portal.
* generate a PDF file for printing once the timetable is completed.
* provide user-friendly interface for students to create and customize their timetables.
* dynamically present or display lessons in accordance with their respective durations.
* let users choose colours for the lessons.
* store users‘ progress.
* display lessons that occur only in even/odd weeks differently from other lessons.
* enable users to select courses, specify preferences (e.g., preferred class times, avoiding specific time slots), and add breaks.
* take into the account the time it takes to get from one building to another.
* display course details, such as course name, instructor, and location.
* automatically detect and notify users of any scheduling conflicts, such as overlapping class times.

-----------------------------------------------------------------------------------------------------------------

* automatically generate multiple timetable options based on the selected courses and preferences.
* allow users to add their own times when they are busy (e.g., hobbies, voluntary work, etc.), which the system will consider when generating timetables.
* enable users to compare multiple generated timetable options side by side to help them make informed decisions.

# **Non-functional requirements (How the system will do this):**

* User data, including login credentials and course preferences, must be stored and transmitted securely using encryption protocols.
* The system must implement proper authentication and authorization mechanisms to ensure that only authorized users can access and modify timetables.
* The system should provide clear and user-friendly error messages to assist users in resolving issues.
* The system should be designed to scale easily to accommodate potential future increases in user load or additional features without major architectural changes.
* The system should have a high level of availability, aiming for 99.9% uptime, with scheduled maintenance communicated to users in advance.
* The system should be compatible with a variety of web browsers (e.g., Chrome, Firefox, Safari) and devices (desktop, tablet, mobile) to ensure a seamless user experience.
* The system should be able to handle concurrent users during peak times, ensuring that it can support at least 1000 simultaneous users without significant performance degradation.

# **Risk list:**

1.

Risk: Unexpected system outages or downtime during peak usage periods, affecting students' timetable planning.

Mitigation: Develop a robust infrastructure with failover mechanisms and conduct regular load testing to identify and address bottlenecks.

2.

Risk: Low adoption rates among students and administrators due to usability issues or lack of awareness.

Mitigation: Create a user-friendly design, conduct user testing, implement a guide book, show the product to UO Counselling and Career Centre.

3.

Risk: Difficulty scaling the system to accommodate a growing number of users and course offerings.

Mitigation: Design the system with scalability in mind and monitor performance regularly to identify and address scaling challenges.

4.

Risk: Insufficient budget or resources to complete the project on time and within scope.

Mitigation: Develop a detailed project plan, secure adequate resources, and monitor progress closely to avoid scope creep.

5.

Risk: Inefficient handling of user feedback and feature requests, leading to dissatisfaction.

Mitigation: Establish a structured feedback process and prioritize enhancements based on user input.

6.

Risk: The system may struggle to generate diverse timetables for students with complex preferences.

Mitigation: Continuously improve the scheduling algorithm to offer a wider range of timetable options.

7.

Risk: It might not be possible to get data from student’s portal.

Mitigation: The student will be adding all the possible lessons manually (excel).