

Warsaw University of Technology

FACULTY OF
MATHEMATICS AND INFORMATION SCIENCE



Bachelor's diploma thesis

in the field of study Computer Science and Information Systems

Platform for hybrid learning

Kiryl Volkau

student record book number 30076

Illia Manzhela

student record book number 302514

thesis supervisor

Dr Eng., Janusz Oleniacz

consultation

Professor Władysław Homenda

WARSAW 2022

Abstract

Platform for hybrid learning

The platform for hybrid learning intends to demonstrate, how the design of educational software could be done using a non-object-oriented approach alongside applying principles of cloud computing. As students, during the pandemic of 2020-2021, we have seen how the educational system was struggling to handle such a change. Meanwhile, we have also discovered the advantages of studying online - it gave us a great level of flexibility and the possibility to re-access materials (in particular). Keeping in mind, that teachers would also benefit from the re-design of the current approach to knowledge transfer, we decided to try and implement a platform, that covers the interests of both groups. We did that using Rust programming language for our backend system, Elm programming language for the frontend, and Azure as a main hosting solution. This paper addresses the obstacles to implementing such a platform and how it differs from already existing solutions. We used our professional knowledge as acting software engineers and students to identify and solve arising issues. It is worth adding, that we do not focus on the software development pipeline here, as it would differ vastly from the real-world development team. Nevertheless, we address usability, extendability, supportability, and other important software traits, since they are crucial to the success of the design itself.

Keywords: hybrid learning, massive open online courses, functional programming, cloud computing, education

History of changes

Table of changes		
Author	Date	Change
Kiryl Volkau	20.10.2021	add: Abstract, Introduction
Kiryl Volkau	20.10.2021	add: Functional requirements
Illia Manzhela	20.10.2021	add: Non-functional requirements
Illia Manzhela	20.10.2021	add: SWOT analysis, schedule

Contents

Vocabulary	11
Introduction	12
Responsibilities and Schedule	13
Functional Requirements	14
Non-Functional Requirements	16
SWOT Analysis	17

Vocabulary

1. **Massive Open Online Courses (MOOC)** platform - platform with educational content including videos, text content, discussion forums.
2. **Application Programming Interface (API)** - set of definitions for building and integrating application software.
3. **Minimal Viable Product (MVP)** - version of product that has enough functionalities to be used by early customers.
4. **Infrastructure as a Service (IaaS)** - pay-as-you-go service where a third party provides you with infrastructure services, like storage and virtualization, as you need them, via a cloud, through the internet. <https://www.redhat.com/en/topics/cloud-computing/iaas-vs-paas-vs-saas>
5. **Platform as a Service (PaaS)** - on-premise infrastructure management where a provider hosts the hardware and software on its own infrastructure and delivers this platform to the user as an integrated solution, solution stack, or service through an internet connection. <https://www.redhat.com/en/topics/cloud-computing/iaas-vs-paas-vs-saas>

Introduction

What is the thesis about? What is the content of it? What is the Author's contribution to it?

WARNING! In a diploma thesis which is a team project: Description of the work division in the team, including the scope of each co-author's contribution to the practical part (Team Programming Project) and the descriptive part of the diploma thesis.

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumyeirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diamvoluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumyeirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diamvoluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

Responsibilities and Schedule

Full Name	Responsibility
Kiryl Volkau	Frontend (Elm) & DevOps (Azure)
Illia Manzhella	Backend (Rust) & DevOps (GitHub Actions)

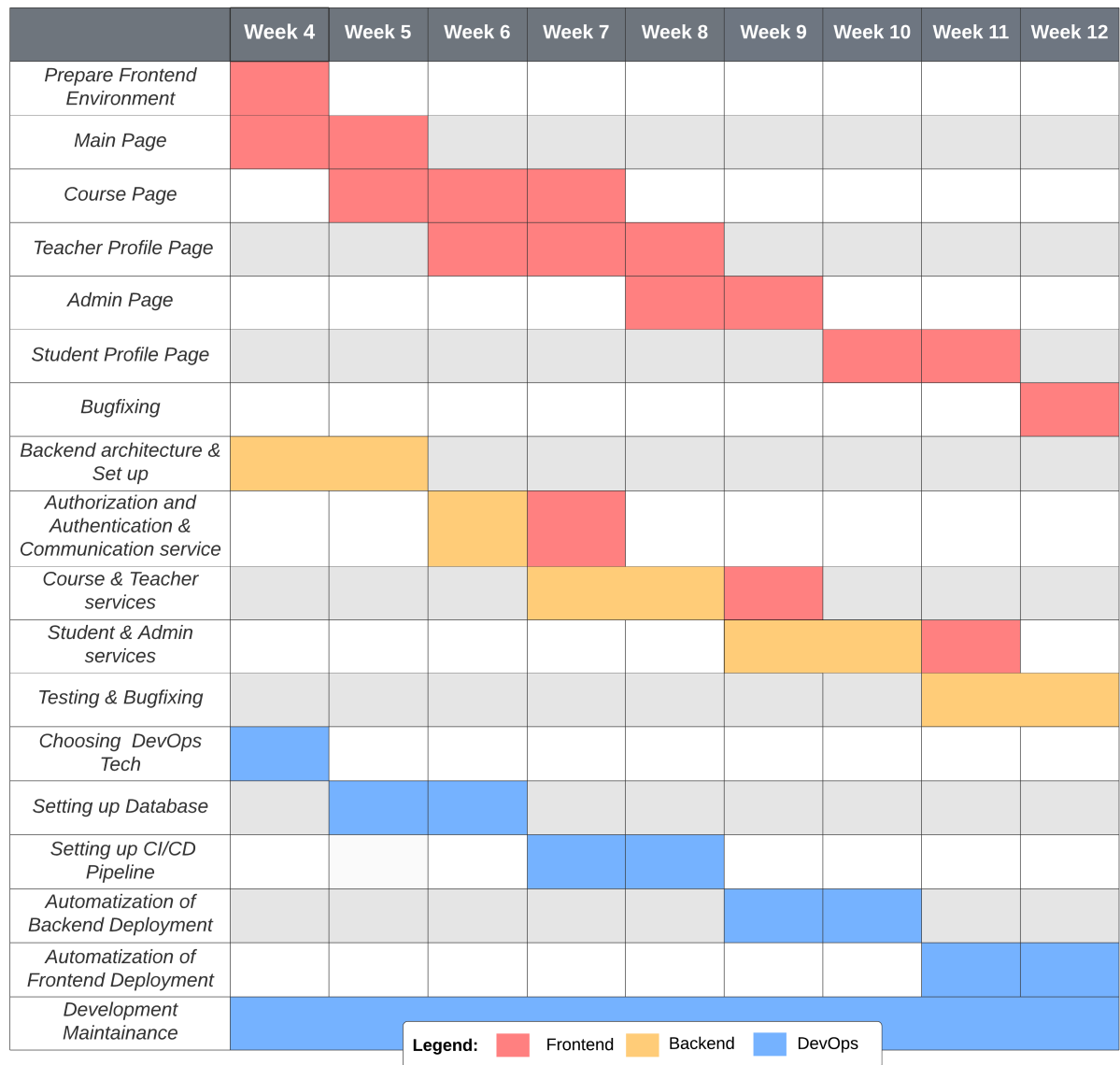


Figure 0.1: Gantt chart by task and person

Functional Requirements

We have decided to describe functionality of an application from the users' roles perspective, allowing us to represent functional requirements as a user stories. Moreover, it allows us to specify granularity of given access and permissions.

The roles are as follows:

- Everyone (combines all the groups below)
- Anonymous User (Unauthorised)
- Everyone Authorised (Teacher, User, Administrator)
- Teacher
- Student
- Administrator

1. **Everyone** can:

- View universities that has been made public
- View university courses (and its contents) that has been made public
- Search for courses by name, university, tags

2. **Everyone Authorised** can:

- Log out
- Change personal details
- See their University details (if they are hidden from anonymous users)

3. **Anonymous User** can :

- Create account as student
- Create account as teacher
- Login to the existing account

4. **Authenticated User** can :

- See their events calendar

- Leave messages on university / courses boards

5. **Teacher** can :

- Submit own courses
- Change own course information
- Delete own course
- See students' comments on his/her course
- Add additional materials to the course

6. **University Administrator** can :

- Manage university staff and students
- Review submitted university courses
- Accept university submitted courses
- Reject university submitted courses
- Delete university courses (with possibility to recover)
- See university students' and teachers' details
- Make university courses open

There is also a super-admin role foreseen, which should be used for the

Non-Functional Requirements

1. Usability

Level of basic user expertise assumed. User interface standards will be used.

2. Reliability

Platform will be available 24/7 with the possibility of restarting the system in case of emergency situations (using PaaS / IaaS functionality or kubernetes). Recoverability will be maintained by an engineer who will recover system from a shut-down failure in case of several unsuccessful attempts to restart). It will accessible from the most popular browsers like Google Chrome (and Chromium-based), Safari, Firefox, and Opera.

3. Performance

System response time is supposed to not exceed 2000ms, automatic recovery and start-up time is expected to take up to 30 minutes.

4. Supportability

System should be testable, maintainable and easily scalable for the needs of higher throughput of media streaming for large number of students. This includes possibility of refactoring the architecture to be more decoupled.

5. Extendibility

System should be easy to extend - meaning it should be possible to add more features to the existing architecture without significant need for the refactoring of previously implemented system.

SWOT Analysis

STRENGTHS	WEAKNESSES
<ol style="list-style-type: none">1. Good understanding of the chosen technologies2. Team of skilled well-coordinated developers3. Milestones are well-defined and understood	<ol style="list-style-type: none">1. Lack of experience with large projects2. Budget of the project3. Lack of marketing experience
OPPORTUNITIES	THREATS
<ol style="list-style-type: none">1. Growing interest in educational products gives more confidence in the success of the product2. Modernization of the market of educational websites	<ol style="list-style-type: none">1. Big competition on the market2. Not enough time to complete the project3. Absence of team members due to some reasons

Threats

1. *Big competition on the market* - as was mentioned in the Introduction, there are already several global players, being Udemy, Udacity, Coursera, etc. Moreover, there are local solutions - like <https://navoica.pl> for Poland offering similar functionality, having well-established processes and budget. Combined with listed above weaknesses - as lack of marketing experience - we may find ourselves in position, when the product is ready, but no one learned about it.
2. *Not enough time to complete the project* - there is a possibility that such a big project our team will not be able to finish on time. In such scenario, our team is planning to extend the time span for this project for one more semester.
3. *Absence of team members due to some reasons* - in case of absence of any of the team members for some reason, all the responsibilities of that person will be taken by the other members of the team.