MepML - ML/AI Model Evaluation Platform

Diogo Magalhães - 102470
Departamento de Electrónica, Telecomunicações e Informática
Universidade de Aveiro
Aveiro, Portugal
d.magalhaes@ua.pt

Rafael Gonçalves - 102534
Departamento de Electrónica, Telecomunicações e Informática
Universidade de Aveiro
Aveiro, Portugal
rfg@ua.pt

Leonardo Almeida - 102536
Departamento de Electrónica, Telecomunicações e Informática
Universidade de Aveiro
Aveiro, Portugal
leonardoalmeida7@ua.pt

Emanuel Marques - 102565
Departamento de Electrónica, Telecomunicações e Informática
Universidade de Aveiro
Aveiro, Portugal
emanuel.gmarques@ua.pt

Pedro Rodrigues - 102778
Departamento de Electrónica, Telecomunicações e Informática
Universidade de Aveiro
Aveiro, Portugal
pedrofrodrigues4@ua.pt

***Abstract*—The need to handle large amounts of data and advances in processing technologies have led to the mass development of intelligent systems. Machine Learning algorithms are often applied to optimise various real-life scenarios, leading to cost savings and increased productivity. Therefore, it is imperative to promote better education in this field of study. This project aims to provide an intuitive and collaborative platform, where students can improve their skills, by solving exercises created by teachers and uploading the results of their models for automatic evaluation.**

***Index Terms*—Machine Learning, Artificial Intelligence, Data Science, Education, Web Application, RESTful API, Model Evaluation, Performance Metrics**

I. Introduction

Analysing the state of the art, we found that there are some platforms aimed at publishing Machine Learning exercises. However, they have some aspects that limit their applicability in the university environment, namely they do not support the assignment of exercises to a restricted group of students, the import of classes from a spreadsheet and the addition and further reuse of new metrics for model evaluation. In addition, there is no clear distinction between the roles of student and teacher as actors in the systems. We took these shortcomings as an opportunity to create a valuable alternative, targeted at the teachers and students of the University of Aveiro who have to deal with this kind of problems. Hence, we developed a multi-layered web application, where teachers can create and manage exercises, import and manage their classes, track the progress of the assignees, and add new evaluation metrics, while students can view or solve exercises, get quick feedback on their models and rank themselves among their peers.

II. Proposed Method

The users interact with the system through a user-friendly interface, built with React (Javascript). This presentation layer is connected to a RESTful API, that is implemented with Django (Python) and contains the business logic. Finally, the data persistence is handled by a MySQL database. The whole system resides in a virtual machine provided by Google Cloud, to ensure scalability, availability and security. Besides, the components are all containerised with Docker, gathering all the dependencies in single packages. Regarding the students engagement, we used another modern Python framework called FastAPI to implement a bot that notifies them of new assignments on a customisable Slack workspace. For our core service, the model performance assessment, we relied on some Data Science libraries: Scikit-learn, for setting up the built-in metrics, Pandas, for data manipulation, and NumPy, for numerical computing. We emphasise that the models submitted by students are previously trained by them. Then, they simply upload the resultant prediction dataset to our system, which performs a quick statistical analysis and returns the metric scores.

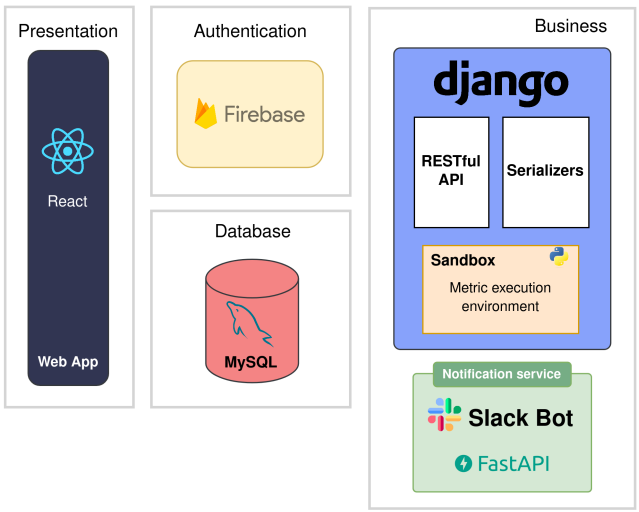

Fig. 1. Simplified Architecture Diagram

In order to assess the effectiveness, efficiency, and overall user experience of the developed platform we performed usability tests. The primary purpose of performing usability tests was to evaluate how easily and intuitively users interact with the system, identify any usability issues or bottlenecks, and gather valuable feedback to enhance its design.

With the collaboration of Professor Petia Georgieva, we had the opportunity to perform usability tests in one of her "Complementos de Aprendizagem Automática" classes. This course is part of the Master's degree in Informatics Engineering and Data Science. The usability tests were performed by 9 users, where 8 of them were students and 1 was the professor herself. As our system has two different types of users - students and professors - we decided to divide the users into two groups as follows:

- **Group 1**: 4 elements that will test the system with the role of students.
- **Group 2**: 5 elements that will test the system with the role of professors.

After performing the usability tests, the feedback received from the users was very positive, as they considered the system easy to use and intuitive. Nevertheless, the users also gave us valuable suggestions that could be implemented in future versions of the system, which would improve the user experience and the usability.

In the release stage of the project we could deliver an intuitive web interface to our users.

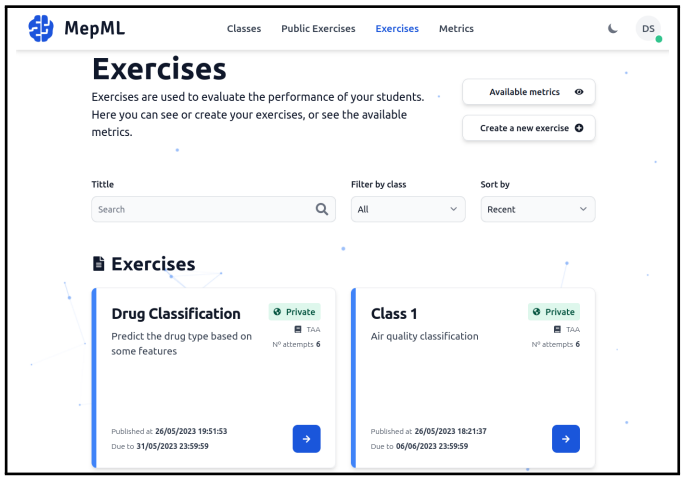

Fig. 2. Teacher Exercises page

III. Conclusions

The project demanded much of the knowledge acquired during the bachelor's degree, enforcing an iterative and incremental development approach, and the adoption of good software engineering practices, such as quality assurance testing. We believe that the proposed software solution will be a good asset for the university, contributing to the effectiveness of part of the learning process in Computer Science, Mathematics, and other courses.

References

[1] Serban, A., van der Blom, K., Hoos, H., & Visser, J. (2020). "Adoption and Effects of Software Engineering Best Practices in Machine Learning", Proceedings of the 14th ACM / IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM).

[2] Kathrin Blagec, Georg Dorffner, Milad Moradi, and Matthias Samwalda. A critical analysis of metrics used for measuring progress in artificial intelligence. pages 4–12, August 2020.

[3] Robert L. Mack Jakob Nielsen. Usability Inspection Methods. Wiley, 1994.

[4] Mark Richards. Software Architecture Patterns. O'Reilly Media, Inc., Second edition, 2015.