# Designing a Bot to Connect Galgotias University LMS with Telegram

Rishabh Raghav, Preeti Kumari

*(Scholar) Galgotias University, UP, India*

*Abstract:* **This service links a Telegram bot with the Moodle learning system to make it easier for students to get the information they need. Instead of logging into Moodle every time, students can use Telegram to quickly get answers to common questions and see important deadlines for assignments and tests. The bot takes this information directly from the Moodle course calendar and sends it as messages in Telegram. This way, students can stay updated and manage their tasks more easily, using an app they already use every day. It makes learning and communication simpler and more accessible.**

*Key words: integration, PHP, Telegram, Moodle, bot, Laravel.*
*Language: English*

## INTRODUCTION

This work continues the series of articles [1-3], describing the development of the IT infrastructure of SPbPU according to the Digital University model and on the way to the third generation university [4].

In the electronic educational environment of universities, various LMS systems are used to train students in electronic courses, the most popular of which are EDX and Moodle. Currently, SPbPU uses Moodle [5] installed on the traditional LAMP/LEMP infrastructure (Linux, Apache/Nginx, MySQL, PHP/Perl/Python) [6]. The architecture of distance learning portals is distributed. There is a portal for general disciplines (taught by several areas of study - for example, physics, mathematics), portals of institutes for special disciplines (taught by one or two areas of study), a server for olympiads, etc. As a rule, for each discipline for each teacher for each type of classes (lecture, practical exercises, laboratory), each semester creates its own course in the corresponding LMS. Using Telegram [7] as a convenient tool for accessing information about e-courses helps to improve the user experience and reduce the time spent searching for the necessary information.

Adding Telegram as an additional communication channel will increase the accessibility of learning, as many students already actively use Telegram in their daily lives. This will allow them to receive important information and communicate with the educational platform with minimum effort and maximum comfort. The advantages of such a bot are that users can quickly get answers to frequently asked questions, necessary materials and deadline dates without leaving the Telegram application.

Existing problems and their solution

Let's consider what problems the integration of the Telegram messenger into the Moodle learning management platform can solve.

Problem 1.

The problem of students failing to meet deadlines for assignments in Moodle, due to the fact that it is difficult for them to follow them due to the large number of portals and courses. This can have a serious impact on the effectiveness of the educational process.

Modern learners are often faced with a huge amount of information and responsibilities. In modern educational institutions and organizations providing online learning, students are provided with access to various courses and portals on the Moodle platform. Each of these courses may have their own due dates, deadlines, and requirements. Because of this variety of information and commitments, students may find it difficult to effectively manage their time and priorities. They may inadvertently miss or forget assignment deadlines as they constantly need to switch between different courses and portals to keep track of updates and deadlines.

The result of this problem can be a deterioration in the academic performance of students, stress and anxiety due to unfulfilled obligations, as well as general disorganization in the educational process.

To solve this problem, integrating the Telegram messenger into the Moodle learning management

platform can represent an effective solution. Through Telegram, students can receive information about the nearest deadlines, notifications from the course (see Fig. 1). This will allow them to be more organized in their studies, keep track of deadlines and properly distribute time between different tasks and courses. Such communication through a platform that students actively use in everyday communication (Telegram) can significantly improve communication and help students cope with the problem of missing deadlines due to fragmentation.

Problem 2.

The problem of frequently asked questions related to courses in Moodle can have a significant impact on the effectiveness of the educational process, both for students and teachers. This problem manifests itself in the fact that the same questions on courses arise from different students, and this can lead to a number of negative consequences. repetitive explanations, which can reduce their effectiveness in other duties. Students are wasting time - students asking the same questions are also wasting their time waiting for an answer or looking for the information they need. This can reduce their motivation and learning productivity. Decreased quality of feedback - repetitive questions can lead educators to focus more on providing general answers instead of addressing the individual needs of each student.

To solve this problem, the Moodle platform can implement various measures, such as:Creating a plugin that adds the username field to the user information section for the any of authorization in Telegram, using the Moodle REST API functionality extension. With its help, the Telegram user and the Moodle user are identified.Create a FAQ where r can educator can provide detailed answers to the most frequently asked questions. Where there is also an implemented generated link, by clicking on which you can log in to the Telegram bot.

Creation of a Telegram bot that will provide students with the opportunity to receive quick answers to the most frequent requests (see Fig. 2). This bot can be customized based on the FAQ section of the course and provide links to resources to help students find more information.

Repetitive workload for educators - when educators constantly receive the same questions from different students, this can take a significant amount of time

and energy. As a result, teachers spend more effort on To solve this problem, the Moodle platform can implement various measures, such as:
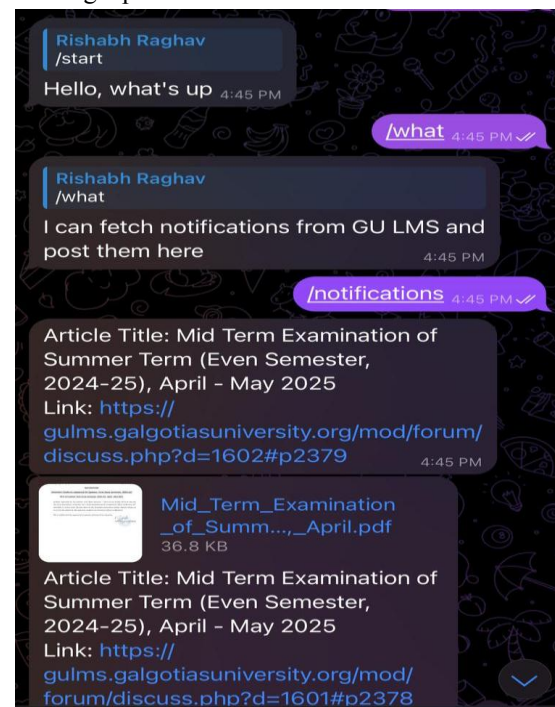Creating a pl



Fig. 2. Telegram interface, FAQ request.

System description
We decided to develop a Telegram bot in PHP [8]. Writing a Telegram bot in PHP has several advantages:

1. Widespread: PHP is one of the most popular programming languages for web development. It has a large developer community, extensive documentation, and many ready-made libraries and frameworks. This makes the development and support of the Telegram bot in PHP more accessible and convenient.

2. Integration with existing systems: many web applications and platforms, including Moodle, often use PHP in their development. Writing a Telegram bot in PHP makes it easy to integrate it with existing systems and interact with them through APIs or other mechanisms.

As a framework, we chose Laravel [9] due to the following reasons:

1. MVC architecture: Laravel is based on the MVC (Model-View-Controller) design pattern, which makes it easy to separate business logic, presentation, and data management.

2. Routing: Laravel provides handy tools for defining application routes. You can define routes for

various HTTP requests and associate them with the appropriate controllers and actions.

We also used the Telegram Bot SDK for Laravel [10]. Some features and capabilities of the Telegram Bot SDK for Laravel include:

1. Handling incoming messages: The SDK provides a mechanism for handling incoming messages from bot users. You can define handlers for different types of messages (text messages, images, audio, etc.) and react accordingly.

Database interaction: Telegram Bot SDK integrates with Laravel's database functionality, allowing you to store and retrieve data related to the bot and users.

Fig. 3 shows the application diagram and Fig. 4 shows the administration panel. In essence, the application is a microservice and interacts with Moodle API and Telegaram API.
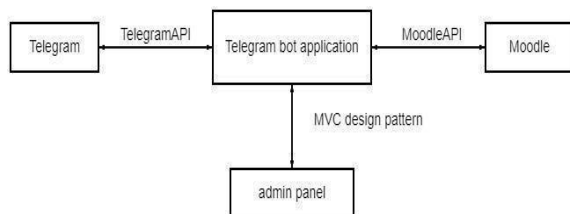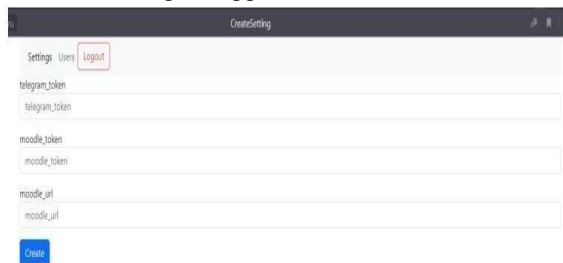


Fig. 3. Application schema.



Fig. 4. Administration panel.

Telegram Bot SDK allows you to create multiple bots that can be configured to work with individual Moodle portals. Also, with the help of a single database of authorized users, you do not need to reauthorize to work with another online course, which makes it easier to use.

CONCLUSION

As a result of the development of the bot, users received a convenient tool that allows them to receive answers to frequently asked questions and information about deadline dates in a convenient form. Thanks to this, users can effectively organize their training, complete tasks on time and not miss important deadlines. Thus, the bot significantly improves the experience and quality of learning for users of the Moodle platform. The developed bot also provides opportunities for further improvement and functionality expansion. For example, we plan to introduce additional features such as new content notifications or interaction with other learning-related tools

REFERENCES

[1] Kozhevnikov, V. A., & Tolpygin, S. S. (2022). Video transcription system. *ISJ Theoretical & Applied Science,* 09 (113), pp. 89-92. DOI: 10.15863/TAS.2022.09.113.17

[2] Kozhevnikov, V. A., & Tolpygin, S. S. (2022). Transform with migration of the university learning management system to cloud services. *ISJ Theoretical & Applied Science*, 10 (114), 53- 57. DOI: 10.15863/TAS.2022.10.114.10 3.

[3] Kozhevnikov, V. A., & Tolpygin, S. S. (2023). CRM system for online learning. *ISJ Theoretical & Applied Science*, 04 (120), 285-290. DOI: 10.15863/TAS.2023.04.120.55

[4] Wissema, J.G. (2009). *Towards the Third Generation University: Managing the University in Transition.* Edward Elgar.

[5] (n.d.). *Moodle*. Retrieved 20.08.2023 from https://moodle.org/

[6] (n.d.). *LAMP (software bundle).* Retrieved 20.08.2023 from https://en.wikipedia.org/wiki/LAMP_(software _bundle)

[7] (n.d.) *Telegram*. Retrieved 20.08.2023 from https://telegram.org/

[8] (n.d.) *PHP*. Retrieved 20.08.2023 from https://www.php.net/docs.php

[9] (n.d.) *Laravel*. Retrieved 20.08.2023 from https://laravel.com/

[10] (n.d.) *Telegram Bot SDK for PHP*. Retrieved 20.08.2023 from https://telegram-bot-sdk.com/docs/