Software Requirements Specification
For
Adaptive Learning Software, KI B^3 Project

Version 2.0

November 08, 2024

Table of Contents

1. Introduction

Artificial Intelligence (AI) will permanently change the economic value chain across industries and sizes as a central disruptive technological innovation. The specialists and managers required for this transformation process must be trained early. The joint project KI B^3 (*Künstliche Intelligenz in die Berufliche Bildung bringen*) addresses the previously missing qualification opportunities for data analysis and machine learning skills in vocational and further training. It is initially based on regional legal provisions that will be rolled out across Baden-Württemberg and combined to create national statutory ordinances.

The newly formed innovation cluster, which is composed of three Baden-Württemberg IHKs (Karlsruhe, Region Stuttgart, and Reutlingen) and the Technical University of Munich, Stuttgart, and Ludwig Maximilian University of Munich (LMU), has been developing a modular format that provides a cross-industry additional qualification to connect trainees with advanced training qualifications.

This is related to a deeper reflection on the creditability within vocational training and the exchange with academic education in the sense of a "currency." Since much of the AI content is currently being taught primarily in academic education, the specific skills should get into vocational training via adapted or revised digital modules. This software requires specification details regarding the software developed to support teaching content and the heterogeneous group of learners.

The project is supported by an extensive network that includes three Baden-Württemberg IHKs and various companies, unions, state ministries, a vocational school, the DIHK, and the DIHK Bildungs-GmbH, employment agencies, and scientific experts.

2. Purpose

This learning software is built upon the Moodle platform, enhanced with custom features to address diverse student needs by ensuring equal access to personalized educational content and adaptive support. **Integrating an AI Chatbot** as a digital tutor, guiding students with tailored feedback, adaptive assistance, and individualized tasks. Additionally, it integrates a Jupyter programming environment directly into Moodle, enhancing students' programming skills. Designed for hybrid learning environments, it combines Moodle's foundational structure with effective didactic strategies and added collaborative features such as discussion forums, badges, notifications, and real-time communication. Integrated functionalities like Python programming and machine learning further allow students to engage in hands-on learning within the platform. This comprehensive design supports both students and educators in managing, tracking, and engaging effectively with the learning process.

3. Document Convention

This document is created based on the IEEE template [1] for System Requirement Specification Documents. Furthermore, the requirements language criteria are based on ISO 29148 [2], i.e., this is a harmonized standard of IEEE 830 [3].

4. Intended Audience and Suggestions

The intended audience is cross-industry, to which the digital modules and adaptive learning software shall be deployed in three phases:

1. In Baden-Württemberg: trials in three Industrie- und Handelskammer (IHK) model regions (IHK districts Karlsruhe, Reutlingen, Region Stuttgart), then roll out the modules and software to the remaining 9 IHK districts.

2. And nationwide perspective roll-out, i.e., creating a national ordinance based on tried and tested regional regulations.

5. Project Scope

This adaptive learning software is developed as part of a larger educational project for IHK, leveraging the Moodle platform with custom enhancements to serve as a comprehensive "learning assistant." It uses AI-driven methods to assess and persoassess and promote each student's knowledge level and skills. By integrating an AI chatbot, Jupyter programming environment, and additional collaborative tools, this platform offers adaptive assistance and feedback tailored to individual learning needs. This approach ensures each student receives targeted guidance and resources to foster growth within a supportive hybrid learning environment.

6. Selection of the Base Learning Platform

An analysis of several existing learning platforms was conducted, into which the aimed adaptive learning software could be integrated to meet the requirements (see Section 7). The criteria used to select Moodle as the base platform for designing and developing the adaptive learning software.

A literature review was conducted to find the best learning platforms for further customization of the KIB3's needs according to the elicited software requirments. Among all three e-learning platforms that were narrowed down for further analysis are Moodle, ILIAS, and Chamilo. Closed-source e-learning platforms were excluded since one of the goals of the KIB3 project is to follow the open-source paradigm.

The three selected e-learning platforms were compared based on their features and capabilities, technical aspects, and the coverage of the KIB3 project's functional and non-functional requirements. Furthermore, available studies in the literature and deployment experience reports were also considered during the selection process.

Since the beginning of COVID-19, a bulk of new research has been published, specifically in rating virtual learning platforms to replace or be the nearest replacement for face-to-face classes. Most of such publications recommend the use of Moodle. The following lists a subset of the reasons why they recommend using Moodle.

- It is an open-source project based on the GNU license [1, 2, 3, 4, 5, 10, 14].
- It is a Content Management System and Virtual Learning Environment that provides its users with learner tools such as communication, productivity, and student involvement. Moreover, it provides support tools such as administration, course management, and content development [2, 6, 10, 11, 14].
- Moodle is easy to set up and can run on almost all servers that support PHP. Moreover, it is easy to upgrade from one version to another [7, 4, 8, 10, 12, 13].
- The key to Moodle's success in academia is its development, which considers both pedagogy and technology by providing a solid grounding in social constructionist pedagogy and good education [5, 10, 11, 12, 14, 15, 16, 17].
- It has many valuable features that can promote it as an alternative to face-to-face classes, such as easy installation, customization of options and settings, good support, and proper educational tools. Moreover, it has excellent documentation and strong support for security and administration [9, 10, 12, 14, 15, 16].

Additionally, other advantages were not discussed in previous studies.

Moodle is a plugin-based platform and provides a plugin marketplace (repository) with hundreds of already developed plugins, such as plugins for adding interactivity, communication tools, computer programming environments, and administration tools.

Moreover, Moodle has object-oriented and modular source-code configurations [5], making it easier to maintain and develop on top of it, which is one of the essential issues in the e-learning platforms ILIAS and Chamilio.

Finally, Moodle has a massive developer community with continuous releases of newer versions, which guarantees long-term support of the project and security assurance due to security updates in newer releases.

In addition to the aforementioned publications that describe the qualities of Moodle, it is also crucial to consider some statistics about Moodle. Moodle currently has (as of 05.08.2021) about 187K registered sites with 277M users and over 1.5B enrollments running in 246 countries [5]. To this date, Germany ranks fourth top Moodle user globally with 10135 registered sites [5].

Additionally, the vocational training school of Balingen has already been using Moodle. This training school is an important KIB3 stakeholder, which works closely with this project to support us in achieving the project's goals by providing us with information about the vocational training school’s needs.

In summary, we selected the learning platform Moodle as the base platform for KIB3's adaptive learning software based on (i) the aforementioned reasons derived from recent studies, (ii) Moodle’s functionalities that cover most of the basic requirements for this project,

(iii) its architecture allowing us to develop the adaptive learning software on top of it, and (iv) the frequency of use worldwide and in Germany, which is much higher than ILIAS and Chamilio.

7. System Architecture

An overview of the system architecture designed to fulfill the core requirements and functionalities of the adaptive learning software is provided in Figure 1. It depicts the overall architecture, showcasing a modular design integrating built-in features with custom-developed plug-ins to create a flexible, interactive learning environment.

Components shown in grey represent built-in functionalities provided by the selected base platform or existing plugins that can deliver the required capabilities upon installation. These include foundational elements like content management, user administration, exercises, and basic functionalities for tracking learning progress. In contrast, components in blue denote functionalities developed specifically for this project. Some of these custom plug-ins, such as a book search tool, an ice cream game, Jupyter Lab integration for programming tasks, and an AI-powered chatbot, are shown in green. They enhance the platform's interactivity, adaptivity, and user engagement, aligning with the software's educational objectives.

The adaptivity and interactivity layer supports personalized learning by tailoring exercises and monitoring individual progress. Communication and interaction are facilitated through integrated tools, enabling real-time engagement among users. This architecture ensures a comprehensive hybrid learning experience by combining the robustness of the base platform with custom plug-and-play extensions, making the system adaptable, scalable, and responsive to diverse learning needs.

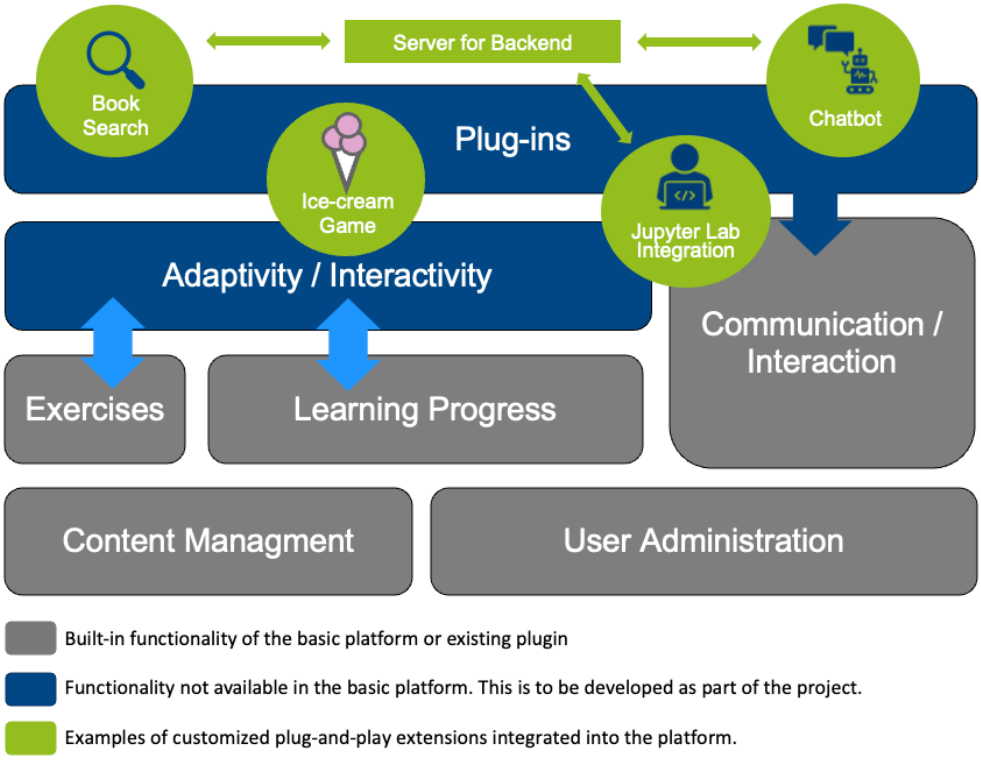

Figure 1: Overall Architecture

8. Software Requirements

This section outlines the requirements elicitation process, led by the Chair of Software Engineering at the Technical University of Munich, detailing the collaborative steps taken to gather and refine project requirements. It concludes with the final list of requirements, organized into functional and non-functional categories, forming the foundation for the software's development.

8.1 Requirements Elicitation Process

The requirements elicitation process for this project, led by the Chair of Software Engineering at the Technical University of Munich, began during the COVID-19 pandemic when physical meetings were not possible. Consequently, all elicitation activities were conducted through online meetings with stakeholders. This remote collaboration included representatives from a range of organizations, such as IHK Reutlingen (IHK RT), IHK Karlsruhe (IHK KA), IHK Region Stuttgart (IHK RS), the Institute for Natural Language Processing at the University of Stuttgart, and Ludwig Maximilian University of Munich (LMU), as well as several school teachers, ensuring a comprehensive blend of expertise.

The process began with a literature review conducted by software engineers at the Chair of Software Engineering, identifying general categories of functionalities, such as basic functions and chatbot tutoring, based on existing AI-supported learning platforms. Before moving into breakout sessions, stakeholders were invited to suggest any additional categories they felt should be included.

Participants were then divided into breakout sessions according to their expertise, where they discussed and brainstormed ideas using mind maps. These mind maps were subsequently consolidated and transformed into requirements compliant with ISO 29148 standards. A follow-up validation meeting allowed stakeholders to review the drafted requirements, remove unfeasible ones, and finalize an agreed-upon list. This final list was classified into functional and non-functional categories, establishing a structured foundation for the software's development.

8.2 Functional Requirements

Below are the essential functional requirements elicited in collaboration with the stakeholders to learn the platform.

ID	Functionality	Requirement Description	Status
FR01	Basic Functionality	The software ***shall*** provide features to browse and filter users, bulk user actions, adding/removing users, and uploading/editing user information.	DONE

ID	Functionality	Requirement Description	Status
FR02	Basic Functionality	The software ***shall*** support roles and permissions to be set for the individual users.	DONE
FR03	Basic Functionality	The software ***shall*** have authentication support for its users.	DONE
FR04	Basic Functionality	The software ***shall*** provide the following functionalities to manage its contents: organize courses in categories and sub-categories, create and manage courses, and set up meta-courses.	DONE
FR05	Basic Functionality	The software system ***shall*** provide internal enrollment, database-driven enrollment (LDAP, external databases, flat files, and IMS Enterprise files) functionalities to enroll users' to courses, support user requests to join a course.	DONE
FR06	Basic Functionality	The software ***shall*** be able to track users' activity to prepare reports to tell strengths and weaknesses, do statistics to ensure individuals' learning progress.	DONE
FR07	Basic Functionality	Push notifications feature ***shall*** be integrated within the software system. A notification system must be generated on the following events: when an assignment is assigned, forum posts and learning material is changed.	DONE
FR08	Basic Functionality	A Python programming environment ***shall*** be integrated with the learning platform.	DONE
FR09	Basic Functionality	The Python programming environment ***shall*** run in a safe manner, i.e., users shall not be able to execute malicious code.	DONE
FR10	Basic Functionality	The system ***shall*** provide a search mechanism to find materials related to keywords / concepts / definitions.	DONE
FR11	Basic Functionality	The system ***should*** also support teachers by displaying course progress.	DONE
FR12	Basic Functionality	The learning platform ***should*** provide a mechanism to train ML models.	DONE

ID	Functionality	Requirement Description	Status
FR13	Interaction Between Users	The learning software ***shall*** provide discussion forum, email, announcement, and chat functionalities to support interaction between its users.	DONE
FR14	Self-Evaluation	The exercises ***shall*** be automatically evaluated and reported to both teachers and students themselves.	DONE
FR15	Chatbot Tutor	The chatbot ***shall*** be able to interface with the content search mechanism/index to provide students with pointers to materials.	DONE
FR16	Chatbot Tutor	The chatbot **shall** be capable of acting as a quiz master.	DONE
FR17	Chatbot Tutor	The chatbot **shall** motivate students.	DONE
FR18	Chatbot Tutor	The chatbot ***shall*** provide its services in real time.	DONE
FR19	Chatbot Tutor	The chatbot ***shall*** be able to deliver certain answers/content from the curriculum in three formats, i.e., text, video and pictures.	DONE
FR20	Adaptation Functionality	The system ***should*** combine user-centered gamification (i.e., rewards, reminders, or notification) strategies with the adaptive modules to motivate students.	DONE
FR21	Basic Functionality	The system shall provide a glossary functionality to map relevant terms to their definitions.	DONE
FR22	Basic Functionality	The system shall autocomplete related activities.	DONE

8.3 Non-functional Requirements

Below are the essential non-functional requirements elicited in collaboration with the stakeholders to learn the platform.

ID	Functionality	Requirement Description	Status
NF01	Basic Functionality	The learning platform should be responsive to support different screen sizes, i.e., computers, mobile phones, and tablets.	DONE

ID	Functionality	Requirement Description	Status
NF02	Basic Functionality	Integrating the English language beside German only for the software interface is a nice functionality to have.	DONE
NF03	Basic Functionality	A mechanism will be provided to report malfunctions in the software.	DONE
NF04	Chatbot Tutor	Chatbot should avoid dangerous / offensive conversations / discrimination	DONE
NF05	Security and Privacy	Users' data privacy and protection must be guaranteed under GDPR.	DONE
NF06	Self-Evaluation	The following types of questions must be supported by the test module of the learning platform: Changing the order of the question types, Standard question types, Calculated, calculated multi-choice, calculated simple, Drag and drop into text, Drag and drop markers, Drag and drop onto image, Description, Essay, Matching, Embedded Answers (Cloze Test / Gap Fill), Multiple Choice, Short Answer, Numerical and Random short-answer matching.	DONE
NF07	Course Content Preparations and Design	Course contents must be designed in a modular fashion.	DONE
NF08	Course Content Preparations and Design	The content delivery system must support video, audio, image, and text data types. Besides the following types should be supported: Word (*.doc, *.docx, *.rtf), PDF (*.pdf), Image (*.gif, *.jpg, *.jpeg, *.png), *.svg, *.tiff), Video (*.mp4, *.flv, *.mov, *.avi), Audio (*.mp3, *.ogg, *.wav, *.aac, *.wma), Other (text box, type in comma-separated)	DONE
NF09	Course Content Preparations and Design	The software system must perform repetition of the exercise and significant parts of the course.	DONE
NF10	Course Content Preparations and Design	A unified template to deliver the course contents must be specified and used. This template must include the relevant logos.	DONE

ID	Functionality	Requirement Description	Status
NF11	Course Content	The software shall provide an implementation of a simulation game. The game lets users guess a number. The correct number is calculated by a prespecified formula. The game can be run a maximum number of trials for each user, where the maximum number can be set individually for groups of users.	DONE

9. Custom Plugin Development Completion and Overview

To meet the project's elicited requirements, a series of custom plug-ins have been developed and integrated to enhance and customize Moodle for KIB3's goals and objectives. These plug-ins extend the base platform's functionality, improving both student support and system services. Two types of plug-ins were developed: the first category provides front-end support for students, while the second category operates in the backend to handle tasks and logic based on user interactions. The table below provides an overview of both plug-in categories and their purposes.

Table 1. Overview of custom plug-ins

Category	Plug-in Name	Intended Use	Download
Front-end	Chatbot	For project use (Chat interface)	[18], From here
	Ice Cream Game	Educational game for learning	[19], From here
	Book Search	Search functionality for students	[20], From here
	Jupyter Hub Integration	Supports programming assignments	[21], From here
	Student Report Generation	Collects Data, used for research	[22], From here
Backend	Jupyter Hub Grade Service	Runs in background	[23], From here
	Chatbot Backend	Runs in background	[24], From here
	Auto Complete Activities	Automates activity completion	[25], From here
	Gen Book	Assists in uploading books easily	[26] From here

Further, the development effort for these custom plug-ins involved a total of 30,435 lines of code. The majority of the codebase is written in PHP (63.4%) and Python (25.1%), which support the core functionalities and interactive features. This is followed by JavaScript (4.6%) and Mustache (2.4%), which enhance the frontend interactivity and templating. A smaller portion of the code is comprised of Shell scripting (2.3%) and CSS (1.2%) for specific system tasks and styling. Minor contributions come from Dockerfiles (0.8%), Gherkin (0.1%), and Makefile (0.1%), ensuring a robust and flexible platform tailored to KIB3's educational needs. A further breakdown of these details is provided in Figure 2.

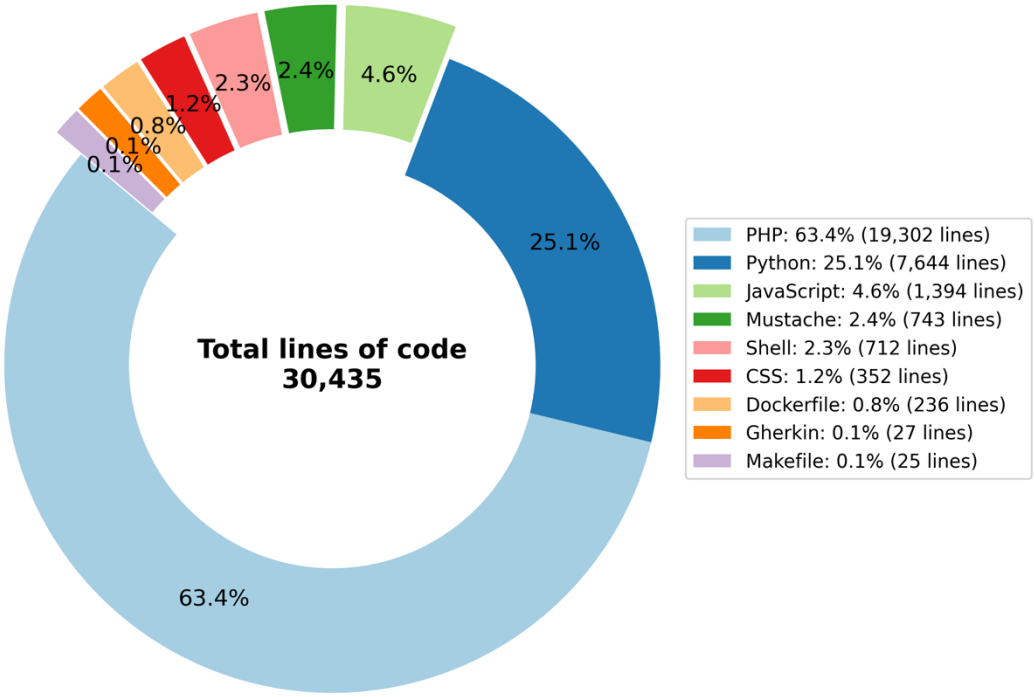

Figure 2: Overview of total Lines of Code and Distribution by Programming Language for custom plug-ins

Each plug-in is designed with a specific intended use, ranging from interactive educational tools to background processes that improve system performance and adaptability. A detailed description of each plug-in is provided in the following section. The development of these plug-ins ensures a robust, flexible platform that meets KIB3's educational needs through tailored, interactive, and efficient functionality.

This modular design, combining student-facing and backend plug-ins, allows the platform to support a dynamic and adaptive learning experience effectively.

9.1 Chatbot Plug-in and Chatbot Backend

The KIB3 Chatbot, Moodle Assistant, is a custom-built chatbot designed to enhance the learning experience within Moodle courses. Acting as a digital tutor, the assistant provides real-time support, helping students navigate course content, find resources, and receive assistance on demand. Integrated directly into Moodle, the chatbot appears on course pages, offering interactive guidance, answering questions, and assisting students in searching through learning materials content using the Book Search plug-in (see Section 9.3) or accessing supplementary tools like the Ice-Cream Game.

The KIB3 Moodle Assistant adapts to each course's needs by leveraging AI, making learning more accessible and engaging. This assistant provides personalized support for students and streamlines the educational process by automating responses and offering relevant resources

within the Moodle environment. Its interaction flow and underlying architecture are depicted in Figure 3.

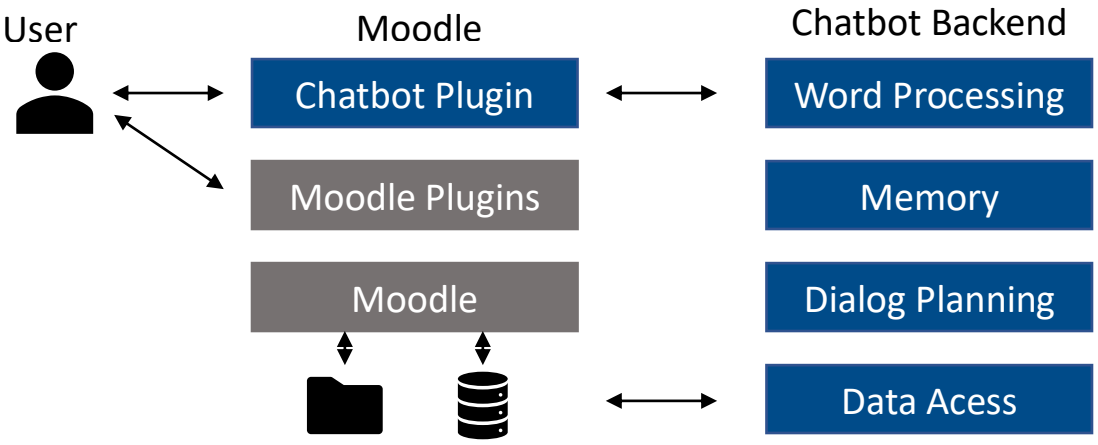

Figure 3: Chatbot architecture and possible interactions

9.2 Ice Cream Game

The Icecream Game is a Moodle plugin designed to introduce students to the basics of machine learning through an engaging, interactive guessing game. In this activity, learners are asked to predict ice cream sales, with each participant selecting from three groups that represent different machine learning strategies. The game begins with an introductory video, after which students choose a group. This choice is irreversible and stored in the Moodle database and the browser cache.

Each group receives information about expected sales in varying formats. Using the information provided, students make predictions for a specific day, with the option to configure the number of guesses. Upon completion, learners receive feedback on their results. This activity introduces students to machine learning concepts and fosters critical thinking and data interpretation skills.

9.3 Book Search

The Book Search plugin is a tool designed to enable text search within Moodle books that contain image-based chapters. It provides a convenient search field that allows users to search for specific text within PDF files corresponding to book chapters. For the search functionality to work, a PDF file with (almost) the same name as the Moodle book must be present in the course, with each PDF page aligning one-to-one with the book pages.

Search results appear as clickable links, directing users to the relevant page within the book. This plugin can be configured as a block within Moodle, or accessed via a web service for additional flexibility. By enabling streamlined navigation through course materials, the Book Search plugin enhances the accessibility of learning resources within Moodle, making it easier for students to find and reference key information. The plugin's interaction flow and usage are further illustrated in Figure 4.

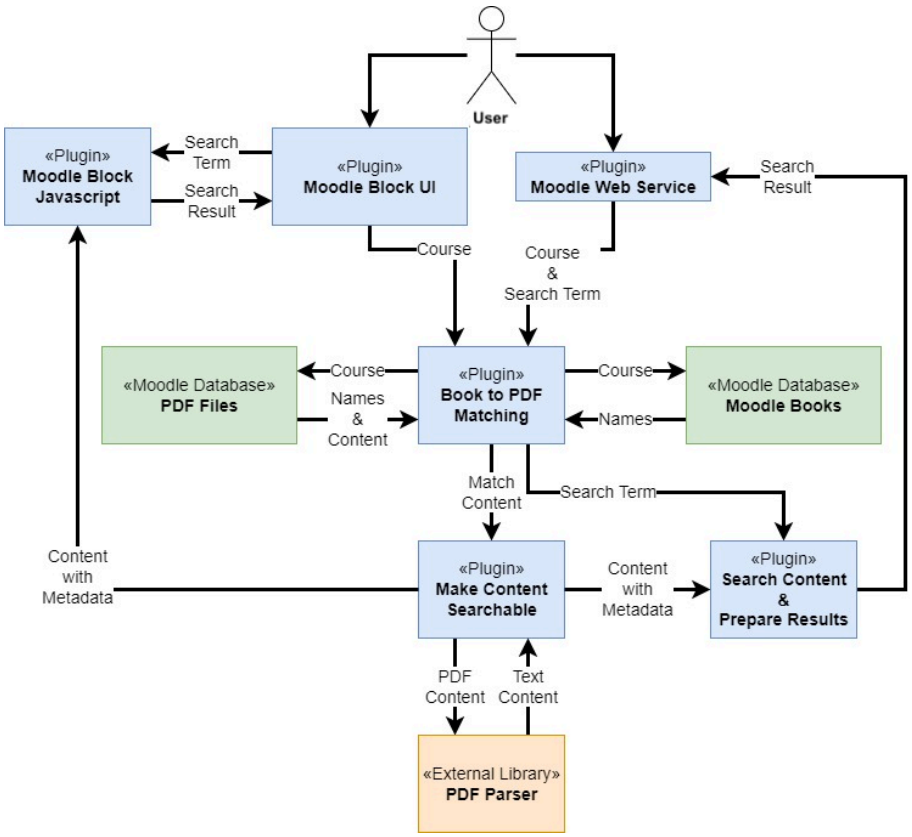

Figure 4: Book Search architecture and possible interactions

9.4 Jupyter Hub Integration and Grade Service

The Jupyter Moodle Plugin integrates Jupyter Notebooks into Moodle, providing students with a virtual programming environment directly within the platform. By connecting to a JupyterHub server, this plugin authenticates Moodle users, allowing them seamless access to Jupyter notebooks from within Moodle. The plugin also supports automated grading through Otter-Grader, enabling instructors to assess assignments efficiently and consistently.

With the Jupyter Moodle Plugin, educators can create programming assignments that students can complete within embedded notebooks, fostering hands-on learning experiences in fields such as Python programming and data science. The integration streamlines coding exercises by offering a centralized platform for coding, testing, and grading, all within the familiar Moodle interface. Figure 5 provides an overview of the plugin's interaction and architecture.

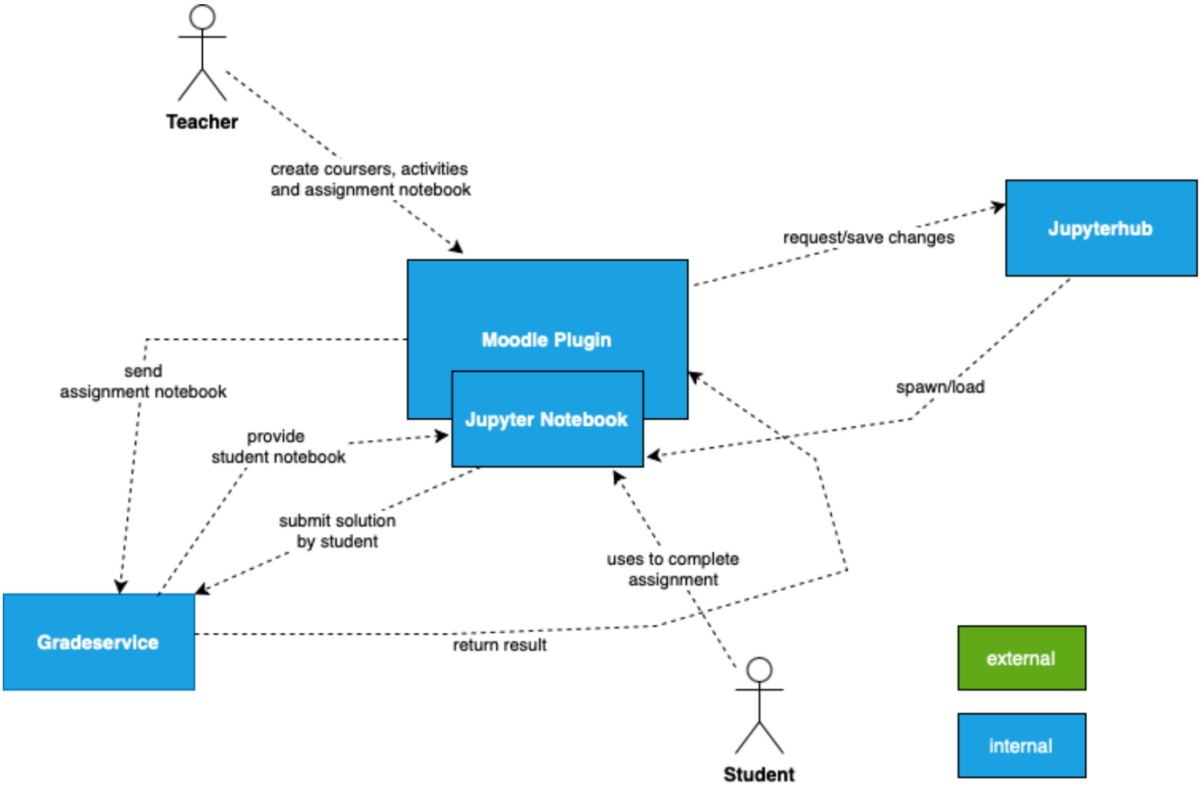

Figure 5: Jupyter Integration architecture and possible interactions

9.5 Student Report Generation

The Student Report Generation (SRG) plugin is designed to support course evaluation by providing learners with anonymized access to a portion of their Moodle log data. Through this feature, students can view and download their log data in an anonymized format, ensuring privacy by removing all personal identifiers, including user IDs.

When students access this plugin, they see a configurable description along with two buttons: one to preview the available anonymized data and another to download it as a file with a .kib3 extension. For users interested in inspecting the contents of a .kib3 file, it can be converted to .zip format for extraction. This plugin helps enhance transparency and engagement in course evaluation by giving students insight into their own learning data while protecting their privacy. Figure 6 provides an overview of the plugin's interaction and architecture.

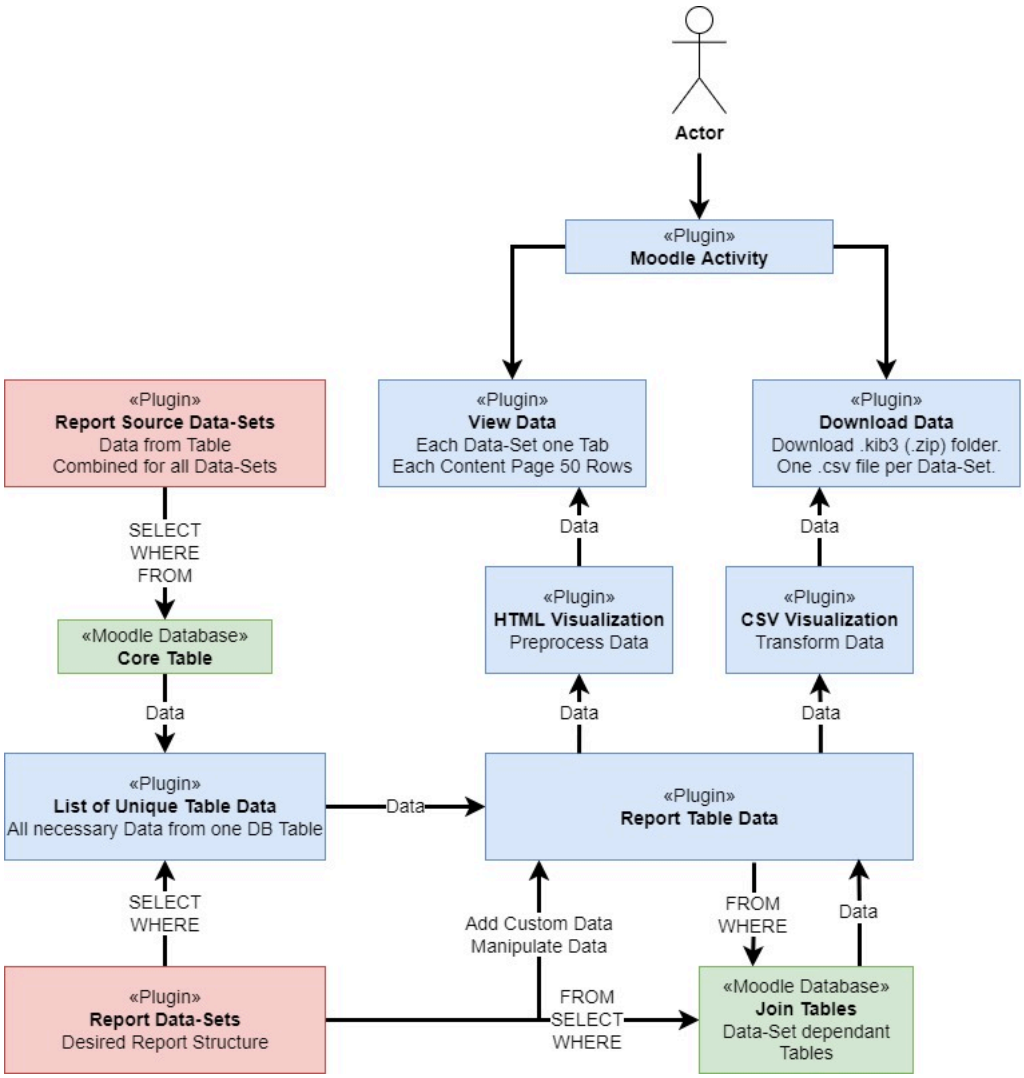

Figure 6: Students Report Generation architecture and possible interactions

9.6 Auto-Complete Activities

The Autocomplete Related Activities plugin streamlines activity completion tracking in Moodle courses that provide the same content in multiple formats, such as an interactive book, PDF, or video link. When a student completes any of these alternative formats, the plugin automatically marks the other formats in the same section as complete, reducing redundant tasks for learners.

The plugin uses a configurable regular expression to identify alternative materials within a section, marking all as complete once any one format is completed. Additionally, it supports an optional "I know this already" label, allowing students to bypass a section entirely if they feel confident with the material. This label is automatically marked complete when any of the alternatives are completed, although marking the label alone does not affect the status of the alternatives.

This plugin activates only in courses configured by the administrator, enhancing the course experience by allowing flexible learning paths and tracking student progress across various content formats.

9.7 Gen Book

The Genbook plugin enables the generation of Moodle books directly from image files, providing a streamlined way to create book resources within courses. Developed specifically to assist content developers in the KIB3 project, Genbook is designed for a specialized use case and may have limited applicability outside this project context.

Using the plugin, content creators can easily compile a series of images into a Moodle book by selecting a course section, specifying a book title, and uploading images as individual chapters. Each chapter is defined by a configuration file that links images to chapter names. This tool simplifies the process of creating structured learning materials, making it easier for developers to organize visual content as structured book resources. Figure 7 provides an overview of the plugin's interaction and architecture.

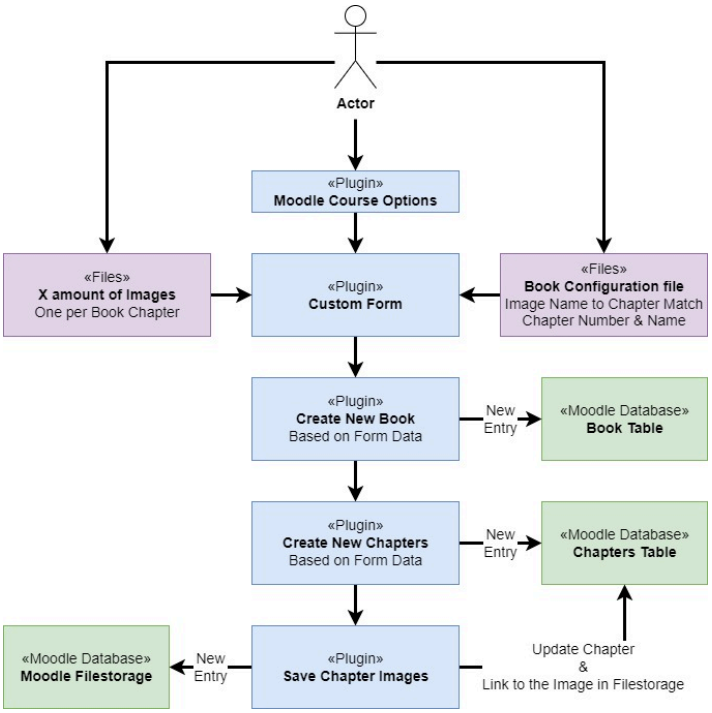

Figure 7: Students Report Generation architecture and possible interactions

10. Alternative Packaging and Delivery with Moodle Docker

To make the installation and configuration of Moodle with multiple custom plugins and course materials easier, Docker provides a reliable solution for consistent deployments. The Moodle Docker plugin offers a one-click setup for deploying Moodle, complete with all necessary plugins and teaching materials tailored specifically for KIB3 courses. Currently supporting the "Zusatzqualifikation" module (with DQR5 support coming soon), this Docker-based package automates the installation process, streamlining course deployment for educators and administrators.

With this setup, Moodle is delivered as a fully configured, ready-to-use environment. The plugin enables rapid deployment by packaging everything needed for a complete Moodle instance, including course backups, plugin installations, and necessary configurations. By utilizing Docker, users can easily manage server settings, adjust resource allocations, and enable SSL, creating a robust and consistent Moodle environment with minimal effort. This tool simplifies course setup, making it an ideal solution for delivering standardized learning environments across KIB3's educational initiatives. The Docker package for Moodle Docker is available for download here: https://github.com/SE-Stuttgart/kib3_moodle_docker

11. Licensing of Custom KIB3 Moodle Plugins

All custom KIB3 Moodle plugins are distributed under the GNU General Public License v3.0, ensuring they remain free and open-source. This license grants users the freedom to use, modify, and distribute the plugins while also requiring that any derivative works remain under the same license. This approach fosters collaboration and encourages further development within the Moodle community, aligning with the educational goals of the KIB3 project.

12. References

[1] G. Tortora, M. Sebillo, G. Vitiello, and P. D'Ambrosio, 'A multilevel learning management system', in Proceedings of the 14th international conference on Software engineering and knowledge engineering, 2002, pp. 541–547.

[2] M. Dougiamas. Moodle. 2008, www.moodle.org.

[3] M. Zenha-Rela and R. Carvalho. Work in Progress: Self Evaluation Through Monitored Peer Review Using the Moodle Platform. in Frontiers in Education Conference, 36th Annual. 2006. San Diego, CA: IEEE.

[4] A. Chavan and S. Pavri, Open Source Learning Management in Moodle. linux journal, 2004, 1(2): p. 78-97.

[5] S. Shearer, Open Source Software in Education. 2003, The Compton School: London. {20}

[6] M. Berry, An investigation of the effectiveness of Moodle in primary education, in Deputy Head. 2005, Haslemere.
[7] M. Dougiamas, Moodle: Virtual learning environment for the rest of us. TESL-EJ, 2004. 8(2): p. 1-8.
[8] J. Itmazi, Flexible Learning Management System To Support Learning In The Traditional And Open Universities, 2005, Granada University, Spain.
[9] Cheng-chao. Su. An Open Source Platform for Educators, in Proceedings of the Fifth IEEE Advanced Learning Technologies. 2005: IEEE Computer Society.
[10] A. Al-Ajlan and H. Zedan, 'Why moodle', in 2008 12th IEEE International Workshop on Future Trends of Distributed Computing Systems, 2008, pp. 58–64.
[11] M. Campo, A. Amandi, and J. C. Biset, 'A software architecture perspective about Moodle flexibility for supporting empirical research of teaching theories', Education and Information Technologies, vol. 26, no. 1, pp. 817–842, 2021.
[12] A. Kika, L. Leka, S. Maxhelaku, and A. Ktona, 'Using data mining techniques on Moodle data for classification of student? S learning styles', in Proceedings of International Academic Conferences, 2019, no. 9211567.
[13] I. Suartama, P. Setyosari, S. Sulthoni, and S. Ulfa, Development of Ubiquitous Learning Environment Based on Moodle Learning Management System. International Association of Online Engineering, 2020, pp. 182–204. Accessed: Aug. 05, 2021. [Online]. Available: https://www.learntechlib.org/p/217822/
[14] I. Suartama, P. Setyosari, S. Sulthoni, and S. Ulfa, Development of Ubiquitous Learning Environment Based on Moodle Learning Management System. International Association of Online Engineering, 2020, pp. 182–204. Accessed: Aug. 05, 2021. [Online]. Available: https://www.learntechlib.org/p/217822/
[15] S. Zelinskiy, 'Analysis of the Possibilities of the MOODLE Learning Management System for Organization of Distance Learning in the Conditions of the University', Social Science Research Network, Rochester, NY, SSRN Scholarly Paper ID 3742272, Sep. 2020. Accessed: Aug. 05, 2021. [Online]. Available: https://papers.ssrn.com/abstract=3742272
[16] G. Akçapinar and A. Bayazit, 'Moodleminer: Data Mining Analysis Tool For Moodle Learning Management System', MoodleMiner: Moodle öğrenme yönetim sistemi için veri madenciliği analiz Aracı, 2019, doi: 10.17051/ilkonline.2019.527645.
[17] M. Chang et al., Foundations and Trends in Smart Learning: Proceedings of 2019 International Conference on Smart Learning Environments. Springer, 2019.
[18] Chatbot Plug-in: https://github.com/SE-Stuttgart/kib3_moodle_chatbot_frontend
[19] Ice-cream Game Plug-in: https://github.com/SE-Stuttgart/kib3_moodleplugin_icecreamgame
[20] Book Search Plug-in: https://github.com/SE-Stuttgart/moodle-block_booksearch
[21] Jupyter Hub Integration Plug-in: https://github.com/SE-Stuttgart/moodle-mod_jupyter
[22] Student Report Generation Plug-in: https://github.com/SE-Stuttgart/moodle-mod_srg
[23] Jupyter Hub Backend: https://github.com/SE-Stuttgart/jupyterhub-gradeservice
[24] Chatbot Backend: https://github.com/SE-Stuttgart/jupyterhub-gradeservice

[25] Autocomplet Activities Plug-in: https://github.com/SE-Stuttgart/moodle-local_autocompleteactivities
[26] Gen Book Plug-in: https://github.com/SE-Stuttgart/kib3_moodle_genbook