CBIE 2023
CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO
Uma escola para o futuro: Tecnologia e conectividade a serviço da educação

Educational Software and Security Vulnerabilities: an experimental study

Diego Rossi, Lucas Bressan, Fernanda Campos, André Oliveira, Victor Ströele

Programa de Pós-graduação em Ciência da Computação – Universidade Federal de Juiz de Fora (UFJF) – Juiz de Fora, MG – Brazil

{diego.rossi,lucas.bressan,andre.oliveira,victor.stroele}@ice.ufjf.br, fernanda.campos.ufjf@gmail.com

Abstract. *The educational domain comprises fragmented solutions with different services, tools, and plugins. As a complex system, it raises several security and threat prevention concerns. We conducted an exploratory study to characterize security vulnerabilities and their impacts on Learning Management Systems. We focus on an intelligent educational solution using the risk assessment methodology HEAVENS 2.0. We identify vulnerabilities in architectural elements and detail two of them. Risks are not acceptable, and security measures must be adopted to avoid damage to students, tutors, and teachers. The results highlight the security vulnerabilities and the consequences of threats to users, hoping to motivate future research*

1. Introduction

Since the COVID-19 pandemic, the number of higher education enrollments in Brazil has increased, both in Distance Learning Education and online classes in face-to-face courses. In this context, the use of Learning Management Systems (LMS) has grown in education and training. An educational process rich in paradigms and pedagogical goals requires adopting technologies such as Artificial Intelligence (AI) and a high level of complexity (Rossi et al., 2022). New challenges are posed considering: technologies, evasion, distancing, students' and teachers' training, and student monitoring, among others.

The educational domain is characterized by fragmented and specific solutions, where different organizations develop different services, tools, and plugins. Among the challenges of organizations that provide LMS are innovation, complexity, and security (Martin et al., 2018) (Rossi et al., 2021). The difficulty in maintaining LMSs with such a diversity of resources makes up an interesting scenario for Software Ecosystem (ECOS) perspective (Abdalla et al., 2018). According to Mesa et al. (2018), the popularity of ecosystems that support plugin-based development is largely due to the number of customization options available as community-contributed plugins. Identifying Virtual Learning Environments as safety-critical systems is important, as any computing failure may lead to catastrophic consequences.

Most extensions that complement software applications with new custom-tailored behaviors are based on plugin solutions. Using plugins to extend web systems with additional and custom behaviors has both positive and negative implications for developers. We can highlight the possibility of quickly building web applications that are custom-tailored to the users' needs. However, these web applications can also be

conceived as a complex composition of multiple plugins from different sources, raising several concerns about plugin security (Mesa et al., 2018).

Vulnerability represents an internal weakness of a component and may be associated with a component's internal fault. Attacks can be identified into different categories as the attacker uses a fake identity to gain access to the system; the attacker can identify as someone else to gain access or higher privileges to the system; the attack to purposely make a component or service unavailable; and attack based on trial and error to gain access to the system. At last, attacks may also rise due to a threat, which can be either unauthorized access to service, an unauthorized modification of service or its data; or unauthorized denial of service, i.e., the attacker disables the system (Bressan et al., 2021). Software security techniques are applied during software development. Cybersecurity is focused on protecting internet-based systems from digital threats. The goal is to ensure applications and devices are secure and, in worse cases, remain functioning under a malicious attack (Coe, 2021).

The impact of digital threats on LMS ecosystem actors (students, teachers, and tutors), may change not only their daily tasks (teachers and tutors work overload), and for students a wrong intervention can lead to demotivation and class dropout.

We are a research group that has developed educational AI-based applications and sentiment analysis (Rossi et al., 2021a), (Rossi et al., 2022), (Rossi et al., 2021b), (Bobó et al., 2022), (Bobó et al., 2018) (Neves et al., 2021) (Nery et al., 2018). Our focus is the automatic detection of students with a great possibility of dropping out of school and students that need special attention in MOOC interactions. Actually, we are worried about the vulnerability of these plugins, considering threats to users' security (Bressan et al., 2021) (Bressan et al., 2022).

The main research question is *how to identify security threats posed by vulnerabilities of Learning Management Systems plugins*. We conducted an exploratory study to characterize security vulnerabilities as well as their impact caused by plugins in educational web-based systems. We focus on a smart education solution as our example is an AI-based application (Rossi et al., 2022). The remarks can help steer future research on web-based vulnerability detection and prevention in LMSs. Our exploratory study makes the following contributions: highlights the occurrence of security vulnerabilities caused by plugin dependability from Learning Management Systems and the consequences of threats to users' security. It also motivat**es** future research on web-based vulnerability detection and prevention in educational software.

Besides our previous experience in safety and security analysis with CRITVAR-ML (Bressan et al., 2021) (Bressan et al., 2022), the evaluation methodology includes the assessment model HEAVENS (Lautenbach, Almgren & Olovsson, 2021), with minor calibrations of the parameters. The goal of threat analysis and risk assessment is to identify and rate potential threats in order to determine which ones need to be cybersecurity mitigated and what level of mitigation is required.

The remainder of this paper is organized as follows: the second section presents the theoretical background and the related work. Section 3 describes the experimental study, its goal, methodology, case study, evaluation, and final analysis and results. Finally, in section four we have the final remarks and future works.

2. Theorical Background

In order to lay the foundation for the rest of our paper, in this section, we provide some background on plugins to LMSs and give an overview of safety vulnerabilities in web applications, highlighting what has already been investigated in this area.

A VLE combines a set of web plugins for distributing educational content, knowledge evaluation, interaction, tracking of student activities, quizzes, tutor monitoring, email, wikis, blogs, chats, forums, and even content search. One of the characteristics of the e-Learning domain is the large number of platforms, its heterogeneity, and the possibility of reuse, sharing, and interoperability of resources (Martins et al., 2018) (Abdalla et al., 2018).

According to Mesa et al. (2018), a plugin-based development is a promising engineering methodology for building complex web applications that may present several variabilities and must be easily customizable. Due to these benefits, to integrate smart learning environments, innovative uses, and new pedagogical approaches, the LMSs need to be composed of plugins, dependent on software, databases, and external actors.

2.1 Security Vulnerabilities

The complexity of web applications has been increasing rapidly, accompanied by the disclosure of many more security vulnerabilities (Mesa et al., 2018). Security requirements have become an increasing recognition of the importance of avoiding vulnerabilities. Vulnerabilities also plague plugin-based web systems, and it is unrealistic to anticipate all of them.

Dependability is an integrating concept encompassing reliability, the capability of providing the correct service for a specified period of time; safety, the absence of catastrophic consequences of a failure on the user(s) and the environment; availability, readiness for correct service; integrity, it is the absence of improper system modifications (made by nonauthorized entity or person); maintainability, the ability to undergo modifications and repairs; and security, which encompasses confidentiality, integrity, and availability (Sommerville, 2015).

Security challenges have emerged, requiring engineering approaches and methods to deal with threats, risk management, secure design, and cybersecurity measures over the whole life cycle (Macher et al., 2020). Safety standards provide requirements and guidance for analyzing and demonstrating safety properties at different levels.

CRITVAR-ML (Bressan et al., 2021) (Bressan et al., 2022) is a modeling language, method, and tool to support the realization of domain variability into safety and security analysis artifacts and variability resolution into re-configurable MOF-compliant system models enriched with dependability information. It is a variability realization modeling language built upon the CVL standard to support engineers specifying mappings between domain problem-space features and finer-grained safety and security information stated as annotations (elements and property values) into MOF-compliant system models (e.g., SysML, AADL) in the solution space. The CRITVAR Method relies on the integration between CHESS, Papyrus UML, and CRITVAR and is based on the activities prescribed by the ISO 26262 and ISO/SAE 21434 standards.

The concept of Digital Dependability Identity (DDI) (Schneider et al., 2015) of a component or a system contains all the information that uniquely describes the

dependability characteristics of a system or components. The DDI concept can be used for integrating components into systems during development as well as for the dynamic integration of systems to systems of systems in the field.

CRITIVAR (Bressan et al., 2021) (Bressan et al., 2022) main concepts will be used to evaluate the case study as well as the assessment model HEAVENS (Lautenbach, Almgren & Olovsson, 2021), described in the next section.

2.2 Related Work

Almeida & Gomes (2021) present indicators to evaluate educational software suggesting its acceptance or not for use in the classroom, depending on the learning objectives identified by the teacher. The authors defined 36 attributes for technical, pedagogical, and educational context aspects. Despite the article highlighting the technical aspects of educational software quality, unlike our work, it does not select security attributes.

Mesa et al. (2018) conducted an exploratory study to characterize vulnerabilities caused by plugins in web-based systems. The study is about WordPress vulnerability bulletins cataloged by the National Vulnerability Database, as well as associated patches maintained by the WordPress plugins repository, identifying the most common security-related topics discussed among developers. Our work does not use LMSs vulnerabilities database, as we describe our own plugin possible security treads.

According to Allodi et al. (2020) assessing the risks of software vulnerabilities is a key process of software development and security management. It depends on the assessor's knowledge and skills. They report an experiment to compare how accurately students with different technical education and security professionals are able to assess the severity of software vulnerabilities with an industry methodology. Our work does not focus on people's expertise in software security assessments accuracy, but both works highlight that security practices should be carried out as normal management practices.

3. STUDY DESIGN

We conducted an exploratory study to characterize plugin-related vulnerabilities in educational web systems. In particular, we investigated the occurrence of security vulnerabilities caused by plugin dependability from Learning Management Systems.

3.1 Goal and Research Questions

Our goal is to shed some light on the plugin-related security vulnerabilities found in educational software, mainly the Learning Management Systems and their plugins. We used the organization proposed by the Goal/Question/Metric (GQM) (Basili & Rombach, 1988) to define the goal of our study. The scope of our study is summarized as:

Analyze LMS plugin-related security vulnerabilities **with the purpose of** characterization and prevention **with respect to** a vulnerability item definition, threat analysis, and risk assessment, and cybersecurity claims and goals **from the point of view of** final users (students, teachers, and tutors) **in the context of** a smart education solution, an AI-based application (Rossi et al., 2022).

In addition, based on our goal, we came up with the research questions (RQ): How to identify security threats posed by vulnerabilities of Learning Management Systems plugins?

3.2 Evaluation Methodology

Considering that we can not evaluate all the Security Capabilities and Controls, we analyze some security threats and their risks using the risk assessment methodology HEAVENS 2.0 (Lautenbach, Almgren & Olovsson, 2021), with minor calibrations of the parameters. The HEAVENS model workflow includes:

a) Item definition - includes all parts required to start work on the item, such as the definition of the item boundary, the item function, the preliminary architecture, as well as the operational environment.

b) Threat analysis and risk assessment - assets, threat scenarios, damage scenarios, and attack paths are identified in the initial phases. Once accomplished, the attack paths are rated for attack feasibility, and the damage scenarios are rated for impact. When the attack feasibility rating and the impact rating have been estimated, the overall risk for the threat scenarios can be determined, after which a treatment decision has to be made.

b.1) Asset, threat scenario, and damage scenario identification: an asset is namely by either manual inspection of the item definition or by automated identification of assets in a data flow diagram; threat scenarios describe a set of actions that lead to one or more damage scenarios, where damage scenarios specify the adverse consequences of an attack; in other words, they specify the result of an attack.

b.2) Attack path analysis: it aims to identify the possible attack paths which might realize the threat scenario in question. The use of attack trees, versatile and well-established, should be considered.

b.3) Attack feasibility rating: each attack path should receive an attack feasibility rating. The proposed mapping to the rating levels is Very Low, Low, Medium, and High.

b.4) Impact rating: it can be estimated as soon as the damage scenarios have been identified. The impact rating must be Negligible, Moderate, Major, or Severe.

b.5) Risk determination: the last step focuses on determining the risk value. Using a risk matrix (Table 1) can be deterministic for final results.

Table 1: HEAVENS 2.0 – Risk matrix

		Impact rating			
		Negl.	Mod.	Maj.	Sev.
Attack feasibility rating	Very low	1	1	2	3
	Low	1	2	3	4
	Medium	2	3	4	5
	High	2	4	5	5

b.6) Risk treatment decision: once a threat scenario receives its risk value, a risk treatment decision has to be made, specifically if the risk should be avoided, shared or transferred, accepted, or reduced.

c) Cybersecurity claims and cybersecurity goals - this step complains addition of cybersecurity claims for accepted or transferred risks and cybersecurity goals as an outcome of a planned risk reduction. Cybersecurity claims are statements about why one risk is acceptable and under which circumstances. Software security solutions help ensure data is protected while in transit and at rest and can also help protect against system vulnerabilities like malware and ransomware attacks.

3.3 Case Study

Predicting Pedagogical Intervention (PRED-INTER) (Rossi et al., 2022) plugin is the case study. It merges Semantic Analysis (SA) and Natural Language Processing (NLP) techniques applied to student's forum posts, combined with Machine Learning (ML) based classification techniques. The proposal is to support teachers, tutors, and students in Virtual Learning Environments, seeking to identify, through implicit attributes in students' messages, those who need help. It is necessary to extract subjective information, comprehend how the student feels, and then choose the most appropriate pedagogical intervention for that student's educational moment.

According to (CAPUANO; CABALLÉ, 2019), natural language processing and predictive models can detect various attributes in post messages, like Sentiment, Post Type, Urgency, and Confusion. This automatic detection of implicit attributes in forum posts becomes fundamental for precise analysis and discovering how the student is feeling. Therefore, this information contributes to the tutors' performance, helping them to moderate and plan interventions and, consequently, cooperate with the student's learning process. To make tutoring more agile and efficient, even with automating some tasks, it is necessary to identify which students need specific help.

The PRED-INTER approach identifies the semantic patterns in student posts using ML-based classification techniques, improving earlier approaches (CAPUANO; CABALLÉ, 2019; BÓBÓ et al., 2019; Neves et al., 2019). Our solution stores the students' posts and their attributes in an ontology, making it possible to make inferences to detect the necessary pedagogical intervention. Attributes allow the system to define which action will be performed and then apply specific dialog patterns.

PRED-INTER layers and workflow architecture carry out the pedagogical intervention model. We detail the functionality of each one, and how they will illustrate the main security vulnerability points. As it depends on LMS ecosystem, the main repositories and dataset as also represented in the architecture (Figure 1).

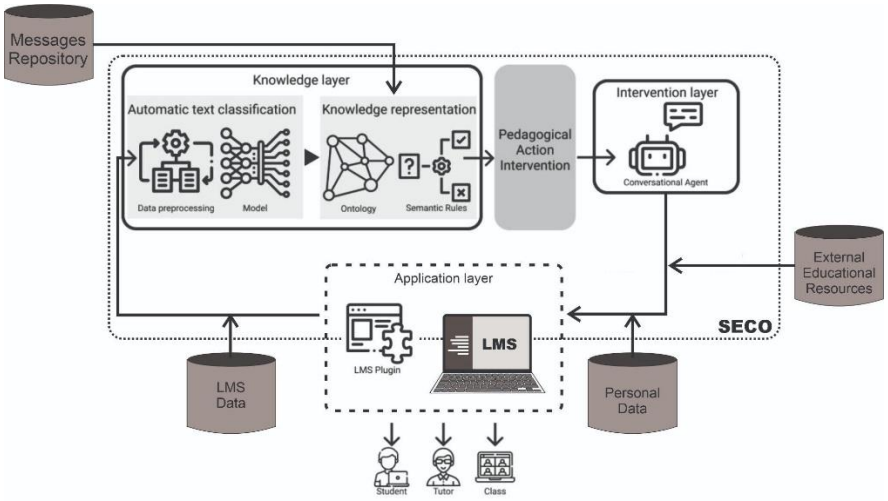

Figure 1: PRED-INTER layers and workflow architecture.

Application layer

The application layer is responsible for interacting with LMS and capturing students' messages. These messages can come from chats, forums, or any other form of textual

interaction and are stored in LMS Data. For the proposed architecture, it is also essential to identify the author (Personal Data) in addition to the posted message The interactions are related to course navigation events, video events (when the video was played, speed, etc.), and exercise logs (attempts, scores, tips used, etc.). PRED-INTER uses LMS data to predict behavior and outcomes.

Knowledge layer

The Knowledge layer is divided into two modules. The first is responsible for automatic textual classification, where the attributes of sentiment, confusion, and urgency are automatically identified in messages from pre-trained models. It starts with text pre-processing, where unwanted text parts of messages are removed (Messages Repository). Then, textual classification is responsible for identifying the attributes implicit in the messages, where pre-trained models are used to identify these attributes.

The textual categorization component labels messages through predictive models, considering three attributes: sentiment, urgency, and confusion. The models must be trained with vast data volume to make good predictions. Another essential step in contributing to the models' performance is the pre-processing, which is responsible for discarding unwanted data that does not add semantic information to the texts. The attributes are loaded in the ontology, keeping subjective information related to the student's message, allowing the execution of semantic rules that automatically identify the pedagogical intervention necessary to meet the student's educational moment.

The rules are responsible for analyzing the attributes and selecting the appropriate pedagogical intervention for the posted message, which will meet the student's needs at that moment, allowing him to receive a message instantly. These rules can be changed and adapted to the course or class, depending on the number of students per tutor, participation or not of colleagues in discussions and contributions of educational resources, availability of educational resources previously selected by teachers from the External Educational Resources repository and level of required interaction.

Intervention layer

Some cases of students' needs are identified as more critical; for example, when the post sentiment is negative, students need more attention. In these cases, our solution alerts the tutors so they can carry out an individualized follow-up. On the other hand, less critical cases, defined according to the attributes, can be monitored automatically through motivational, informative, or thank-you messages or by automatic interaction. These pedagogical interventions will prevent the student from creating the feeling of abandonment and help the tutor avoid task overload.

The intervention layer is composed of an autonomous conversational agent. The agent assists the student and the tutor, sending messages to carry out the pedagogical intervention. It also includes sending messages to the class. The mediation content of the message depends on the identified attributes, which are previously stored in the ontology, as well as the type of intervention, being also pre-defined and may vary according to the class or course. It may be an automatic message to the student or a message asking the class to help the student. When a post in a discussion forum indicates that the student needs more than a message for pedagogical intervention, the agent may start and conduct an automatic interference (help) or ask the tutor for help. For automatic help, the subject in the message is recognized, and a pre-selected educational resource from the External

Educational Resources repository is recommended to help the student with the content (ROSSI et al., 2021).

When applying semantic rules SWRL, the message is linked to a pre-defined intervention style, which will support an intelligent conversational agent. The rules were defined by humans with long experience in distance education and tutoring discussion forums. They considered a learning scenario where the messages could express parameters of education such as types of questions (about the execution of a task or a class content), topics (motivational, social, or collaboration), and expressions (compliments, frustrations, or personal difficulties).

Implementation

Considering that the PRED-INTER is a plugin of an LMS, it has to deal with different actors and organizations and reused and shared software. Its main function is communication, accomplishing student and tutor interaction, and monitoring. As a plugin, it may be used on different platforms. This work explored two deep learning approaches; the first used the pre-trained BERT model (DEVLIN et al., 2018), and the second used Hereditary Tree-LSTM (GOMES et al., 2022).

Several languages, libraries, and tools were used to develop the architecture, considering reproducibility and performance. According to (RASCHKA; PATTERSON; NOLET, 2020), the Python language has seen tremendous growth in the scientific computing community, which has led to innovations in machine learning and deep learning libraries. Taking into account these statements and the agility in reading data from semi-structured sources, the Python language was prioritized to develop the ontology. Some libraries assisted in this task: TensorFlow, Scikit-learn, and transformers.

3.4 Security assessment analysis

We applied the HEAVENS 2.0 method to the PRED-INTER plugin. First, we identify many vulnerabilities in architectural elements and workflow diagrams, categorized in Tables 1, as input, output, or internal according to their origin. The examples are important to contextualize the risks.

For simplicity, we detail only two risks, considered security vulnerabilities. The selected items illustrate the risk assessment in Table 2.

In the first case the predictive model will be trained with incorrect messages, for example, the message has negative sentiment and the model defines it as positive, so the agent intervention will be incorrect and adulterated. In the second case, the model works fine but the intervention comes manipulated. For example, offensive phrases, and resources that do not interest the student or are not adequate to the class content will be sent by the agent. In both cases, the cybersecurity claims and cybersecurity goals must consider that the cited risks are not acceptable and security measures must be adopted to avoid them. In case of these attacks, the plugin must be alerted.

Most of the risks can be avoided by adopting some measures that may include: security requirements definition, secure coding practices, static code analysis, testing, and severe access control.

Table 1: PRED-INTER examples of failure categories, risks, and consequences

Failure Category	Risk Description	The consequence to students, teachers, and tutors
Input	Use of LMS dataset to label messages through predictive models.	If the dataset is invaded, the students may receive the wrong intervention from the agent or tutor.
	The messages can come from any form of textual interaction.	If the messages have their authorship or content violated, the students may receive the wrong intervention.
	It is essential to identify the author's message.	If the message has its authorship changed, the student may receive the wrong answer.
	The performance of predictive models is influenced by the data quality used in the training process.	If the training process fails because of a dataset crash, the semantic rules may impact teachers' and tutors' work and students' daily interaction with the chat or other tools.
	The models must be trained with vast data volume to make good predictions.	If the training process fails because of the reduction of the dataset volume, the intervention may not have a high prediction assertiveness impacting the student's daily interactions and motivation.
	The semantic rules can be adapted to the course or class, number of students per tutor, participation or not of colleagues, and educational resources.	If the semantic rules do not reflect the class, students, teachers, and tutors will receive wrong answers from the agent.
	Availability of educational resources previously selected by teachers.	Students may receive inadequate options if the agent does not find the correct resource to be recommended.
Output	Inadequate automatic messages for critical cases.	If outside hackers change the messages, the critical cases may receive messages not related to their urgency, such as thank you messages or congratulations.
	Different roles may interact with the plugin.	If an inadequate number of persons interact with the plugin, they may change the plugin's functionalities.
Internal	The predictive models may not have the best performance in labeling the messages.	If the negative messages are not identified, students that need special attention may not receive the tutor's attention.
	The textual categorization takes into account only three attributes: sentiment, urgency, and confusion to define the student sentiment.	If the student's sentiment identification is wrong, he may receive the wrong intervention.
	Inadequate pre-processing techniques.	If the pre-processing techniques help the excluding negative sentiment, wrong interventions will be sent to the students.
	Inconsistent definition of ontology semantic rules.	Students may receive wrong interactions if the ontology does not identify negative sentiment.
	No identification of correct negative sentiment	If the negative sentiment is not identified, the student may receive wrong messages from the agent or the tutor.
	No adequate or sufficient definition of the type of intervention	If the type of intervention does not reflect the students' sentiment, they may receive wrong interactions.

3.5 Results

In this section we frame our discussion around the RQ described in the previous sections.

Trying to answer "How to identify security threats posed by vulnerabilities of Learning Management Systems plugins?", we can say that we need a methodology to support threat analysis and risk assessment of Virtual Learning Systems.

TABLE 2: Threat analysis and risk assessment

Asset	Attack path	Thread	Property	Attack feasibility	Impact rating	Risk determination	Risk treatment decision
Chat	Data entrance	Tampering	Integrity	Depends on the LMS security	High	Major	Avoided
Intervention	Interaction with the LMS	Tampering	Data Integrity	Depends on the LMS security	High	Severe	Avoided

As LMS is a sophisticated ecosystem the characterization and prevention of security vulnerabilities is important as many plugins depend on its dataset and repositories. The adoption of vulnerability item definition, threat analysis and risk assessment, and cybersecurity claims and goals, for the ecosystem and the plugins, may avoid damage to students, tutors, and teachers.

AI-based applications may have more vulnerabilities than other plugins, as they include the training process and the intelligent models. By identifying the possible threats and procedures vulnerabilities we can mitigate those issues.

The constraints of the case study may include the choice of one learning management system, Moodle, and PRED-INTER plugin, as the examples can be enriched if we detail more solutions and their vulnerabilities. We also focus only on consequences to students, teachers, and tutors.

4. FINAL REMARKS

We investigated the occurrence of security vulnerabilities caused by plugin dependability from Learning Management Systems. Actually, we are worried about threats to users' security. The case study focuses on a smart education solution as our example is an AI-based application.

We highlighted some security vulnerabilities found in educational software, mainly the Learning Management Systems and their plugins and we hope the study will motivate future research on web-based vulnerability detection and prevention in LMSs and educational software in general.

PRED-INTER is a safety-critical system, if we consider the whole LMS ecosystem. To assure its performance it has to be integrated into more interconnected platforms, databases, and applications.

In a broader context, from the results presented in the previous sections, developers must consider that the absence of catastrophic consequences (safety), the absence of improper system modifications made by nom authorized entity or person (integrity), and the ability of a system to protect itself against improper state alterations (integrity) as cited before, are attributes that will guarantee the safety and security of the plugin running.

Several venues for future exploration are possible. We are developing some artificial intelligent-based applications for LMS, and exploring their vulnerabilities to try to shade cyber attacks.

REFERENCES

ABDALLA, A. ; STROELE, V. ; Campos, F. ; DAVID, JOSÉ MARIA N. ; Braga, Regina . Plataforma de Ecossistema de Software para Sistemas de Recomendação. In: Simpósio Brasileiro de Sistemas de Informação (SBSI), 2018, Caxias do Sul. Porto Alegre: SBC, 2018. p. 1-8.

ALMEIDA, André; GOMES, Luciana de Queiroz Leal. Avaliação de Softwares Educacionais através de Indicadores de Qualidade. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 32. , 2021, Online. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação, 2021 . p. 249-258. DOI: https://doi.org/10.5753/sbie.2021.218685.

BÓBÓ, MÍRIA L. D. R. ; Campos, Fernanda ; STROELE, VICTOR ; DAVID, JOSÉ MARIA N. ; BRAGA, REGINA, TORRENT, TIAGO TIMPONI. Using Sentiment Analysis to Identify Student Emotional State to Avoid Dropout in E-Learning. International Journal Of Distance Education Technology,, v. 20, p. 1-24, 2022.

BOBO, M. ; Campos, F. ; STRÖELE, VICTOR ; DAVID, JOSÉ MARIA N. ; Braga, Regina . Identificação do Perfil Emocional do Aluno Através de Análise de Sentimento: Combatendo a Evasão Escolar. In: VIII Congresso Brasileiro de Informática na Educação (CBIE 2019), 2019, Brasilia. Anais do XXX Simpósio Brasileiro de Informática na Educação (SBIE 2019). Porto Alegre: SBC, 2019. v. 1. p. 1431-1440.

BRESSAN, L.; OLIVEIRA, A. L. ; CAMPOS, F. ; MONTECCHI, L. ; CAPILLA, R. ; PARKER, D. ; ASLANSEFAT, K. ; PAPADOPOULOS, Y. . Modeling the Variability of System Safety Analysis using State-Machine Diagrams. In: 8th International Symposium on Model-Based Safety Assessment, 2022, Munich. 2022. v. 1.

BRESSAN, L. ; OLIVEIRA, A. L. ; Campos, F. ; Capilla, R. . A variability modeling and transformation approach for safety-critical systems. In: 15th International Working Conference on Variability Modelling of Software-Intensive Systems (VaMoS?21), 2021, 2021, Krems, Áustri. 2021.

CAPUANO, N.; CABALLÉ, S. Multi-attribute categorization of mooc forum posts and applications to conversational agents. In: SPRINGER. International Conference on P2P, Parallel, Grid, Cloud and Internet Computing. [S.l.], 2019. p. 505–514.

COE, F. 2021. What is software security and why is it important? Accessed 04/23/2023, available from https://www.contentful.com/blog/software-security-to-deliver-digital-experiences-fast/

DEVLIN, J. et al. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805, 2018.

GOMES, J. et al. A hereditary attentive template-based approach for complex knowledge base question answering systems. Expert Systems with Applications, p. 117725, 2022. ISSN 0957-4174. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0957417422010089>. Acesso em: 09/06/2022.

LAUTENBACH, ALJOSCHA; ALMGREN, MAGNUS & OLOVSSON, TOMAS.Proposing HEAVENS 2.0 – an automotive risk assessment model. CSCS ’21, November 30, 2021, Ingolstadt, Germany

LUCA ALLODI, LUCA; MARCO CREMONINI, MARCO;MASSACCI, FÁBIO; SHIM, WOOHYUN. Measuring the accuracy of software vulnerability assessments: experiments with students and professionals. Empirical Software Engineering (2020) Springer 25:1063–1094. https://doi.org/10.1007/s10664-019-09797-4

MACHER, G.; SCHMITTNER, C., et al. ISO/SAE DIS 21434 Automotive Cybersecurity Standard - In a Nutshell. In: CASIMIRO, António et al. (Eds.). Computer Safety, Reliability, and Security. SAFECOMP 2020 Workshops. Cham: SpringerInternational Publishing, 2020. P. 123–135. ISBN 978-3-030-55583-2.

MARTINS, G. ; VEIGA, W. ; Campos, F. ; STRÖELE, VICTOR ; DAVID, JOSÉ MARIA N. ; Braga, Regina . Construção de Jogos Educacionais através de Modelo de Features. In: Simpósio Brasileiro de Sistemas de Informação (SBSI),, 2018, Caxias do Sul. Porto Alegre: SBC, 2018. p. 1-8.

MESA, Oslien et alli. Understanding vulnerabilities in plugin-based web systems: an exploratory study of wordpress. SPLC '18: Proceedings of the 22nd International Systems and Software Product Line Conference - Volume 1. September 2018Pages 149–159.

MORENO-MARCOS, P. M. et al. Prediction in moocs: A review and future research directions. IEEE Transactions on Learning Technologies, IEEE, v. 12, n. 3, p.384–401, 2018.

NERY, T. ; COELHO, G. ; Campos, F. ; Braga, Regina ; STRÖELE, VICTOR ; David, J. M. N. Uso de Proveniência de Objetos de Aprendizagem para Identificação do Estilo Preferencial de Aprendizagem. In: VIII Congresso Brasileiro de Informática na Educação (CBIE 2019),, 2019, Brasilia. Anais do XXX Simpósio Brasileiro de Informática na Educação (SBIE 2019). Porto Alegre: SBC, 2019. v. 1. p. 109-118.

NEVES, F. ; Campos, F. ; STROELE, V. ; DANTAS, MARIO ; Braga, Regina ; David, J. M. N. Assisted education: using predictive model to avoid school dropout in e-learning systems. In: Santi Caballé, Stavros Demetriadis and more. (Org.). Intelligent Systems and Learning Data Analytics in Online Education. 1ed.: Elsevier, 2021, v. 1, p. 1-.

OLIVEIRA, André Luiz de; BRAGA, Rosana; MASIERO, Paulo; PARKER, David, et al. Variability management in safety-critical systems design and dependability analysis.Journal of Software: Evolution and Process, v. 31, n. 8, e2202, Aug. 2019. ISSN 20477473. Available from: <http://doi.wiley.com/10.1002/smr.2202>.

RASCHKA, S.; PATTERSON, J.; NOLET, C. Machine learning in python: Main developments and technology trends in data science, machine learning, and artificial intelligence.Information, Multidisciplinary Digital Publishing Institute, v. 11, n. 4, p. 193, 2020.

ROSSI, DIEGO ; STRÖELE, VICTOR ; SOUZA, JAIRO ; Campos, Fernanda . Automatic classification of subjective attributes from student messages in virtual learning environments. In: Simpósio Brasileiro de Informática na Educação, 2022,

Brasil. Anais do XXXIII Simpósio Brasileiro de Informática na Educação (SBIE 2022), 2022. p. 871.

ROSSI, D. ; STROELE, VICTOR ; BRAGA, REGINA ; CABALLE, S. ; CAPUANO, N. ; Campos, F. ; DANTAS, MARIO ; LOMASCO, L. ; TOTI, D. . CAERS: A Conversational Agent for Intervention in MOOCs? Learning Processes.. In: Innovations in Learning and Technology for the Workplace and Higher Education, 2021. TLIC 2021., 2021. Lecture Notes in Networks and Systems, 2021. v. 349.

ROSSI, D. ; STROELE, V. ; Campos, F. ; BRAGA, REGINA ; DAVID, JOSE M. . Identifying pedagogical intervention in MOOCs learning processes: a conversational agent proposal. In: X Congresso Brasileiro de Informática na Educação (CBIE 2021), 2021. Anais do XXXII Simpósio Brasileiro de Informática na Educação (SBIE 2021), 2021. p. 849-860.

SCHNEIDER, Daniel et al. WAP: Digital Dependability Identities. In: PROCEEDINGS of the 2015 IEEE 26th International Symposium on Software Reliability Engineering (ISSRE). USA: IEEE Computer Society, 2015. (ISSRE '15), p. 324–329. ISBN 9781509004065. DOI: 10.1109/ISSRE.2015.7381825. Available from:<https://doi.org/10.1109/ISSRE.2015.7381825>.

BASILI, V. AND ROMBACH, D. 1988. The TAME project: Towards improvement-oriented software environments. IEEE Transactions on software engineering 14, 6 (1988), 758–773.

SOMMERVILLE, I. (2015) Software Engineering. 10th Edition, Pearson, London.