Towards Personalized Learning Paths to Empower Competence Development in Model Driven Engineering through the ENCORE platform

Antonio Bucchiarone
MoDiS Research Unit
Fondazione Bruno Kessler
Trento, Italy
bucchiarone@fbk.eu

Andrea Vázquez-Ingelmo
GRIAL Research Group
University of Salamanca
Salamanca, Spain
andreavazquez@usal.es

Gianluca Schiavo
i3 Research Unit
Fondazione Bruno Kessler
Trento, Italy
gschiavo@fbk.eu

Simone Barandoni
Department of Computer Science
University of Pisa
Pisa, Italy
simone.barandoni@phd.unipi.it

Alicia García-Holgado
GRIAL Research Group
University of Salamanca
Salamanca, Spain
aliciagh@usal.es

Francisco José García-Peñalvo
GRIAL Research Group
University of Salamanca
Salamanca, Spain
fgarcia@usal.es

Sébastien Mosser
McMaster University
McMaster Centre for Software Certification
Hamilton, Canada
mossers@mcmaster.ca

Alfonso Pierantonio
DISIM
Università degli Studi dell'Aquila
L'Aquila, Italy
alfonso.pierantonio@univaq.it

Steffen Zschaler, William Barnett
Department of Informatics
King's College London
London, UK
szschaler@acm.org, will.barnett@kcl.ac.uk

***Abstract*—Developing engaging learning experiences is costly and complicated. Open Educational Resources (OERs) offer the possibility of reuse, creating opportunities for more efficient and effective development of learning, including the potential for truly personalised learning through adaptive learning management platforms. To make this a reality, we need to address two challenges: (1) for OERs to become effective there need to be tools that allow them to be reused efficiently from high-level designs of learning experiences, and (2) for such high-level tools to work, we need to establish a robust infrastructure that treats OERs as components, akin to software components, complete with well-defined interfaces. The latter is particularly challenging for learning resources on MDE, because of the often complex tool environments required. In this paper, we propose a platform for personalized learning design based on OERs and provide initial insights into component formats within the context of teaching about model-driven engineering.**

***Index Terms*—Open Educational Resources (OERs), Personalized learning management platform, Model-driven engineering.**

I. Introduction

Open Educational Resources (OERs) [1] offer the potential of providing unparalleled options for the creation of individualized, efficient, and engaging learning experiences. These adaptable and reusable materials have the capacity to enhance the educational environment, especially when combined with adaptive learning management platforms. However, realizing this promise requires overcoming two key difficulties.

Firstly, for OERs to be effective, there is a need for tools that facilitate their efficient reuse from high-level designs of learning experiences. This implies a shift from traditional, monolithic learning tools towards a more modular, component-based approach. Second, such high-level tools require an architecture that allows OERs to be reused as components, similar to software components, with well-defined interfaces.

This is particularly challenging in the context of teaching about model-driven engineering (MDE), given the complexity of the required tool environments [2], [3]. Despite the promising potential of OERs, their application in the field of model-driven engineering (MDE) is indeed currently limited. One of the primary reasons for this is the scarcity of high-quality, relevant OERs in this domain. The scattered nature of existing resources and the lack of a centralized repository further exacerbate this issue, making it difficult for educators and learners to find and utilize these resources effectively.

In light of these challenges, there is a pressing need for a systematic approach that not only increases the availability of high-quality OERs in the field of MDE but also ensures their effective and efficient reuse. This paper aims to address this gap by proposing a conceptual model to develop a platform for personalized learning based on OERs, with a focus on enhancing the quantity and quality, as well as the reuse, of OERs in the MDE field.

Previous works have tackled the design and creation of educational tools and frameworks in different learning scenarios through conceptual models and meta-modeling. For instance, [4] described a model-driven approach designed to support learning in public administrations. The proposed method attempts to improve knowledge exchange and cooperation among public employees by making important information more accessible, in addition to encouraging a better awareness of processes and working environments. By utilizing various model types and promoting collaboration, civil servants can acquire and apply knowledge effectively, leading to improved performance and decision-making.

A similar idea is behind the work presented in [5], in which a meta-model for developing learning ecosystems using the Model-Driven Architecture (MDA) approach is proposed. The meta-model aims to address the challenges associated with the definition and evolution of learning ecosystems by providing a high-level abstraction and an architectural pattern, and it is focused on the information flows and software tools needed to comply with the stakeholders' requirements and the managerial objectives.

In this context, some solutions aim at providing meta-models to design web-based educational platforms. In [6], the authors propose a meta-model to facilitate the conceptual design of web-based educational applications by capturing their underlying content and navigational structure. This meta-model includes elements such as links, static pages and learning resources. However, the learning resources are only characterized by their objectives and optional meta-data.

On the other hand, other works focused specifically on modeling collaborative learning and generating tools for their support. In [7] a new systematic model-driven methodological approach is proposed for the design of flows of learning activities and the semi-automatic generation of collaborative learning tools. Their collaborative learning meta-model is the input for a graphical editor which allows educators to generate learning tools for different scenarios and areas.

The work described in [8] is also focused on modeling collaborative learning practices. In this case, authors propose an Educational Modelling Language (EML) meta-model to enable the modelling of the information managed and transmitted between tasks, as well as the roles involved and their assignment to different tasks, and shows how the suggested meta-model may be utilized to simulate coordination concerns in educational processes such as learners, tutors, materials, tasks, assignments, and document flows.

Finally, other authors have developed model-driven and meta-modeling solutions for specific educational applications, such as educational authoring tools and serious games. For example, a meta model-based textual language for authoring educational courses is detailed in [9]. For specifying the structure, content, sequencing, and assessment of educational courses, the authors provide four domain-specific languages. The proposed software also contains a code creation module that generates executable courses in a variety of formats, including HTML, SCORM, and IMS-QTI.

In the context of serious games, the work presented in [10] proposes a preliminary meta-model for educational games in higher education. By integrating knowledge requirements, transferable skills, and course results to game production, the meta-model attempts to improve the development of high-quality and engaging educational games. It modularizes domain-specific bodies of information, learning taxonomies, and skill-based challenges, and situates learning opportunities within a narrative in which students advance by outperforming non-player opponents.

In general, the aforementioned works primarily focus on providing a holistic perspective of particular educational scenarios and workflows. In this context, the ENCORE conceptual model explicitly targets the essential characteristics of Open Educational Resources (OERs) and learning paths, as fundamental components in educational processes. By harnessing this conceptual model, our aim is to devise tools that facilitate personalized learning paths and establish a community-oriented database encompassing well-defined OERs.

II. Conceptual Model

The ENCORE conceptual model (as depicted in Figure 1) is built around the concept of competence, acknowledging the significance of competences in both personal and professional growth. Each *Competence* is defined by their characteristics and represents a distinct set of abilities, knowledge, and behaviors that individuals can acquire and apply in various contexts. To better understand and assess competences, each *Competence Item* is structured and categorized into different *levels* of proficiency or expertise. These levels of proficiency help to measure a person's mastery of a particular competence, indicating the extent to which they have developed and applied it successfully. For instance, a competence item can be referred to as a competence at a specific proficiency level, such as beginner, intermediate, advanced, expert or master. These proficiency levels serve as a guideline to evaluate and measure the depth of understanding and practical application of each competence.

To establish a more comprehensive framework for competences, the ENCORE conceptual model explores potential links with existing *Frameworks*, such as O*NET[1], ESCO[2], EntreComp[3]. By connecting competences to these established frameworks, individuals can better align their skill development with recognized industry standards and qualifications, enabling them to make more informed decisions about their personal and professional growth.

A competence can be described as a set of *Concepts* that individuals must study, learn, and understand to develop specific skills and abilities. Competences are not isolated skills; instead, they encompass a range of related knowledge, principles, and practices that work together to achieve a particular *Learning Goal*.

[1] O*NET (Occupational Information Network): https://www.onetonline.org/

[2] ESCO (European Skills, Competences, Qualifications and Occupations): https://esco.ec.europa.eu/

[3] EntreComp (European Entrepreneurship Competence Framework): https://ec.europa.eu/social/main.jsp?catId=1317&langId=en

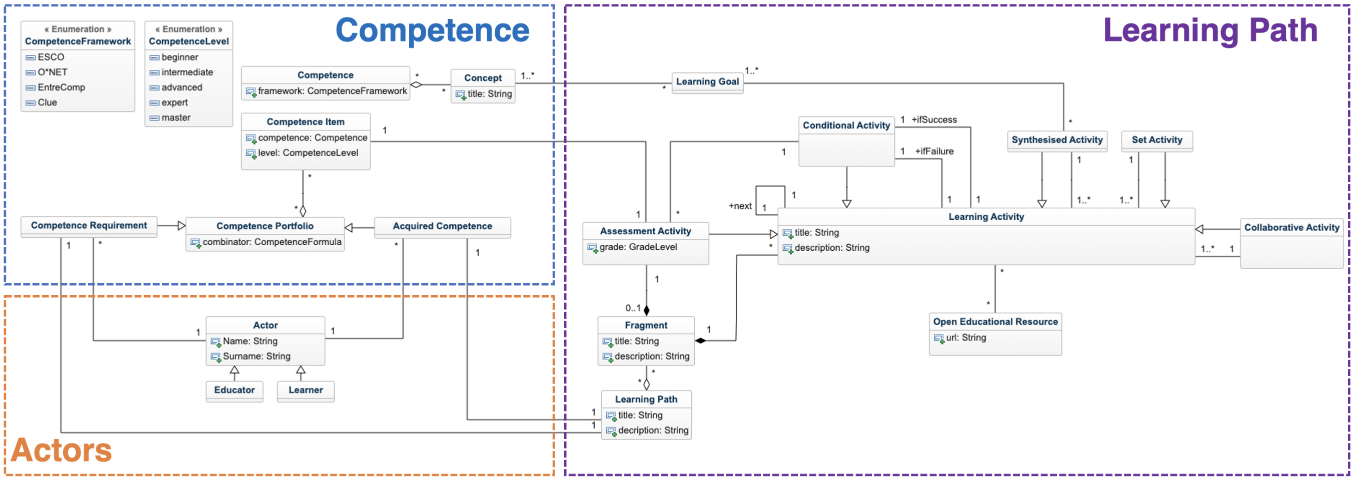

Fig. 1. ENCORE Conceptual Model.

The *Competence Portfolio* serves as an aggregation of various competence items, representing a diverse range of skills, knowledge, and abilities that individuals possess or aspire to acquire. These competences are organized and structured within the portfolio to enable the identification of potential learning paths and progressions.

The *Combinator*, an essential component of the competence portfolio, plays a crucial role in guiding learners through their skill development journey. It is a logical formula that can incorporate both *Competence Requirements* and *Acquired Competences*. The requirements indicate which competences should be acquired before progressing to others. This ensures a logical and coherent advancement in skill development, allowing learners to build a strong foundation before tackling more advanced topics.

On the other hand, the acquired competences represent the ultimate objectives or outcomes that learners aim to achieve after completing specific learning paths.

Each *Educator* has the autonomy to define *Learning Paths* for their learners. These learning paths are designed to be decomposed into a set of *Learning Fragments*, which are centered around the same topic or subject matter. These learning fragments allow for a more granular and focused approach to education, enabling learners to concentrate on specific aspects of a competence while gradually building a comprehensive understanding.

Learners can follow these paths, acquiring competences in a well-organized manner, and ultimately reach their desired end competences, which signify their mastery and proficiency in a particular domain. This approach fosters a systematic and efficient approach to skill development, empowering learners to achieve their learning goals and excel in their personal and professional endeavors.

Each learning fragment consists of a set of interconnected *Learning Activities* and serve as a guided process that helps learners achieve specific learning objectives. The activities within each fragment are designed to be executed in a particular order, facilitated by the *"next"* relation.

As learners progress through each learning fragment, they follow a structured sequence of activities that build upon one another, providing a coherent and well-organized learning experience. Each activity serves a unique purpose, contributing to the overall attainment of the learning objectives defined for that fragment.

The "next" relation ensures a logical flow of learning, indicating the sequential order in which activities should be completed. This sequential execution allows learners to steadily advance through the fragment, with each activity laying the groundwork for the subsequent ones.

By employing this systematic approach to learning, learners can effectively grasp complex concepts and develop their competences in a step-by-step manner.

A learning activity can take various forms, catering to different learning styles and objectives. These activities include:

- *Conditional Activity*: it offers learners different next steps based on their performance or outcome in the preceding activity. If learners meet the predefined *success* criteria or demonstrate a sufficient level of proficiency in the activity, they proceed along the success path. If learners do not meet the success criteria or struggle to achieve the desired outcomes, they are directed along the *failure path*. This path may offer remedial activities, additional support, or alternative learning materials tailored to address their specific learning needs.
- *Synthesised Activity*: it is not fully defined by the educator but is synthesised from a specific *learning goal* (i.e., a sets of concepts to be mastered). By harnessing specific recommendation systems [11] or generative AI techniques [12], [13], the system can seamlessly propose learning activities that align with the designated learning objectives, utilizing an existing set of learning fragments.
- *Set Activity*: within this activity, learners have the freedom to select which activities they wish to engage with, based on their individual interests, learning preferences,

and prior knowledge. By granting learners the autonomy to select their preferred sub-activities, the set activity promotes personalized learning experiences. Learners can focus on areas they find most challenging, delve deeper into topics they are passionate about, and apply their prior knowledge to build connections between different sub-activities.

- *Collaborative Activity*: it involves learners working together in a group setting to achieve shared learning goals. This type of activity promotes teamwork, communication, and cooperation, encouraging learners to exchange ideas, share perspectives, and collectively solve problems.
- *Assessment Activity*: it plays a crucial role in evaluating learners' achievement of specific competences within the overall learning path. The Assessment Activity is designed to measure learners' proficiency and understanding of the targeted competence. It may consist of quizzes, exams, practical exercises, or any other evaluative methods that measure learners' knowledge and skills related to the learning objectives. Moreover, it can be seamlessly integrated into Conditional Activities as well to influence their progression within the learning path.
  The Assessment Activity includes an attribute called "grade," representing the Grade Level defined for the specific competence being assessed. This grade corresponds to predefined criteria that align with the proficiency levels expected of learners at different stages of their learning journey. The grade serves as a quantitative measure of learners' performance, providing valuable feedback and insights into their competence development.

Open Educational Resources (OER) offer a valuable and versatile resource that can be utilized across all the learning activities mentioned previously. Whether in Set Activities, Collaborative Activities, Synthesised Activities, or even Assessment Activities, OER provides a wide array of freely accessible educational materials that can enrich the learning experience. By exploiting OERs across these learning activities, educators can enhance the quality and accessibility of education while promoting lifelong learning for a broader and more diverse audience.

III. Method

This section describes the application of the ENCORE conceptual model through the presentation of a lifecycle example, along with a practical scenario in MDE education.

A. LifeCycle

The ENCORE lifecycle consists of four distinct steps.

The first step, **Discover**, involves educators delving into relevant concepts and competences that will serve as the foundation for their learning activities. The identification of these competences takes place within specific domains or reference frameworks and is subsequently stored in a dedicated database known as the ENCORE skill database [14]. Moreover, this process necessitates the linkage of each competence to the relevant educational materials, including Open Educational Resources (OERs). These educational resources are stored in a dedicated repository called the ENCORE repository [14].

Moving on to the second step, **Collect**, educators review the extracted educational materials within the OERs and carefully select the most relevant ones. They have the flexibility to export these materials to other Learning Management Systems, such as Moodle[4] or Canvas[5], or use them to enhance and refine a learning path based on the concepts covered in the OERs.

In the subsequent step, **Plan**, educators have the ability to convert the primary concepts from the collected OERs into a concept map using semantic similarity algorithms. They can then personalize the generated map by adding new concepts, making edits to existing ones, and arranging them as needed. The resulting concept map can be transformed into a learning path, where relevant learning activities are associated with each node or concept. Educators have the freedom to determine the order of the learning activities and include various types of educational material. These learning activities follow the classification described in the conceptual model (Conditional, Synthesised, Collaborative, Set and Assessment Activities). Furthermore, the resulting learning path can be shared with students in the subsequent phase. Additionally, the learning paths can be utilized to validate the assignment of open badges[6], depending on the successful coverage of concepts throughout the path.

Lastly, in the **Execute** phase, students gain access to the learning paths crafted by educators. They can utilize notebook interfaces to explore the learning activities presented within each node of the path, allowing them to personalize their learning experience by accessing different resources. These notebook interfaces enable students to engage with the materials in a flexible and personalized manner.

B. Running Scenario - Model-driven Arduino

Teaching model-driven engineering faces a significant challenge due to the required level of abstraction for understanding its concepts. Learners often encounter confusion as they perceive a gap between the systems they model and the systems they code. This issue, known as the "model-code gap" [15], is characterized by differences in: i) vocabulary, for instance the vocabulary used in models (modules, components, protocols, associations, etc.) versus the vocabulary used in code (packages, classes, variables, functions, etc.); ii) levels of abstraction, especially during the analysis phase of the software development process), and iii) the specific nature of source code. This disparity poses a challenge for students as they transition between concrete source code and abstract models, often leading to a lack of comprehension regarding the relationships and utility of these artifacts.

A variety of works in the literature have attempted to address this issue using various methodologies, such as combining UML and OCL [16], adapting materials to the audience [17], using interactive exercises to visually understand the

[4]https://moodle.org/
[5]https://www.instructure.com/canvas
[6]https://openbadges.org/

implications of UML diagrams [18], or teaching modeling languages before programming languages [19]. On the other hand, certain active techniques, such as project-based learning (PBL), attempt to facilitate understanding of this topic by encouraging student participation in the development of projects that are relevant to real-world contexts. These techniques aim to boost students' enthusiasm for such a complicated topic, as well as their sense of the value of software engineering in real-world circumstances [20].

Following this idea, we propose a learning scenario for teaching MDE concepts with a real-world example in the domain of the Internet of Things (IoT), where related concepts can be introduced through a practical learning path.

C. Discovery and Collection of OERs

Before building the learning path for the previously described scenario, it is necessary to collect Open Educational Resources related to MDE applied to IoT. First, we analyzed existing Repositories of OERs (ROERs) focused on software engineering and MDE. In this context, the MDENet network [21] develops and provides learning resources as well as running regular online training sessions. These resources have been curated and made available through their community platform [22].

The OERs from MDENet were collected following a scraping approach, and organized into the ENCORE database storing the meta-data attributes from the Dublin Core Metadata Element Set (DCMES) [23], [24]:

- **Title**. An OER should have a title describing its content.
- **Description**. The description is crucial in the context of the ENCORE project. OERs must be clearly described to extract the GDE skills addressed in its content.
- **Subject**. The subject of the OER depicts the field tackled by the resource, and it is also crucial to identify the skills addressed through the content.
- **Creator**. The author or authors of the OER.
- **Contributor**. Entity or entities that contributed to the OER content.
- **Publisher**. Entity or entities in charge of making the resource available.
- **Publication date**. The date in which the OER was published.
- **Type**. Category of the resource (image, dataset, text, etc.).
- **Format**. Technical format of the resource (application/pdf, image/gif, etc.).
- **Source**. Reference to other resources from which the OER was derived.
- **Language**. The language of the OER.
- **Coverage**. The applicability of the resource.
- **Rights**. Information related to the OER's rights.
- **Relation**. This attribute is represented to the "related to" relationship in the domain model, and depicts related resources to a certain OER.

Among the MDENet resources, there is a tutorial of MDE in IoT using Arduino, an open-source electronics platform based on easy-to-use hardware and software, which is aligned and fits with the learning scenario previously described.

This tutorial covers different aspects of MDE and domain specific languages (DSL) following a bottom-up approach from code to models [25] and from models to DSLs, and it is divided into different educational resources tackling concepts such as programming and modeling a finite state machine or designing a DSL. The OERs collected from this tutorial can be combined with other activities to create personalized learning paths using the ENCORE technological ecosystem.

D. OER to Map of Concepts

After collecting the necessary OERs, the ENCORE lifecycle includes the Plan phase, where the educator, based on these OER, must first select the concepts to be introduced in the learning path and then proceed to specify the learning activities to assign to the learners. To achieve this the ENCORE platform provides a method to generate concepts graphs starting from the collected OERs. Concepts graphs are the representation of the main notions mentioned in a text, organized as a network whose nodes are the concepts and the edges between them express a semantic relation. The methodology is divided in two main phases: the first is focused on extracting the concepts from OERs' descriptions, the second addresses the generation of the concept graphs.

1) Concept Extraction: To assess the identification of concepts in a given text, we developed a lexicon-based information extraction approach: concepts are extracted by searching the terms of a predefined lexicon in a text; if a certain term is matched, the corresponding concepts is considered found. First, we created a lexicon of concepts. Since the whole work is done to support educational processes, we aimed at gathering terms which can be considered notions that someone might need to learn or teach. For this reason, we relied on the European Skills, Competences, Qualifications, and Occupations (ESCO) database[7], and on Wikipedia. We begun by taking the standard list of 3,059 knowledge from ESCO. Then, to enlarge this set of concepts, we automatically searched all these knowledge on Wikipedia. We saved all the hyperlinks found in each Wikipedia page visited in the collection of concepts. We repeated this operation on all the new entities found through the hyperlinks: we automatically visited their Wikipedia pages and stored all the hyperlinks found there. By merging the ESCO knowledge and the entities retrieved in the two iterations performed on Wikipedia, we obtained the final lexicon of concepts, composed of 162,256 terms. Once the lexicon was ready, we were able to search its terms in texts. We applied this methodology to the OERs collected in the Discovery and Collect phases obtaining a list of concepts which are mentioned in their descriptions.

2) Graph Generation: The purpose of the second step of the methodology is generating a graph based on the concepts extracted from a given text. The nodes of the graph correspond to the concepts. The nodes should be connected by edges

[7]https://esco.ec.europa.eu/en

which express a semantic relation between concepts. The preliminary operation needed for the generation of the graph is the identification of the semantic relations. For this task we relied on Wikipedia: we assumed that if a certain concept A appears in the Wikipedia page of another concept B as an hyperlink (redirecting to the Wikipedia page of concept A), it means that there exists a relation between them, in particular meaning that A is contained in B. The reason for this assumption is the fact that in Wikipedia pages it is common to mention concepts which are kindred to the main topic and to insert hyperlinks to redirect the user to their pages, if existing. The relations identified on Wikipedia have a direction: if A is found in the page of B, but not vice versa, there exists a relation going from A to B, but not necessarily from B to A. We developed an algorithm (exposed as an ENCORE API) which automatically iterates over all the concepts extracted from a text: for each of them, it opens their Wikipedia pages and searches all the other concepts found in the text among the hyperlinks found there. This operation creates, for a given text, a table containing the combinations of all the concepts extracted and a flag to denote whether the semantic relation exists or not between each pair. This table is then utilised to create a network. To restrain the dimension of the graphs, we set a limit, in order to show the 50 nodes with the highest degree centrality (i.e., the number of relations) only. Furthermore, the size of each node is proportional to the number of occurrences of the corresponding concept in the documents. Applying this process to the learning scenario considered meant taking the concepts extracted from the OERs and identifying the relations between them.

E. Learning Path Design and Execution

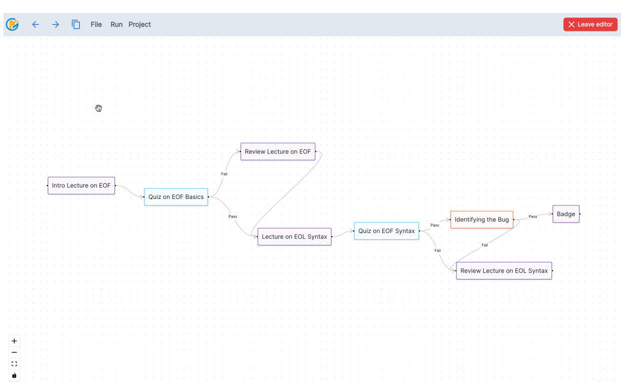

Fig. 2. Educator design tool for learning paths (Epsilon Learning Path).

The educator, guided by the overall concepts graph extracted in the previous phase makes the decision to initiate the learning process by focusing on specific concepts related to `code generation`, `model-to-model transformation`, and `model validation`. These concepts will serve as essential building blocks to explain the intricate relationship and significance between real-world code and abstract models. By introducing these concepts, the educator aims to provide a clear and comprehensive understanding of how these aspects interrelate and are instrumental in practical applications as IoT.

From the selected concepts, the ENCORE platform is able to propose a filtered subset of OERs. These recommended resources can be effectively harnessed by the educator to craft a dedicated learning path tailored to the learning objectives.

In the specific running scenario, the educator decides to use the resources proposed and related to the Epsilon framework [26]. In particular the educator wants students to first learn about the Epsilon Object Language (EOL) since it is the core expression language of Epsilon. Once the students have completed the EOL learning objective, they move to learn the basics of the second language, the Epsilon Generation Language (EGL) to understand how it can be used to transform models into various types of textual artefact, including code (e.g. Java), reports (e.g. in HTML/LaTeX), formal specifications, or even entire applications comprising code in multiple languages (e.g. HTML, Javascript and CSS).

The educator has identified three types of specific learning activities required to define the scenario: lessons, quizzes and modelling exercises (as depicted in Figure 2). In the event that a student fails a quiz or a modeling exercise, the learning path provides them with a `Review Lecture` and an additional `Quiz` to reinforce their understanding before they proceed. Once students have completed the first section on the EOL, they move to the second part, where they can learn the remaining language (i.e., EGL) following the same pattern as the first section. Gamification has the role of motivating the students while executing the learning activities. This is done thanks to a set of game elements (i.e., badges) used to link the successful execution of learning activities to the advancement in the game narrative (i.e., BADGE nodes in Figure 2).

The ENCORE Execution Engine allows students to access and follow learning paths through standard Visual Studio Code notebooks (Figure 3). Paths defined in the ENCORE editor can be downloaded and distributed to learners for execution using VS Code notebooks.

An illustrative video is available to demonstrate the execution of the ENCORE life-cycle presented in this paper. It provides a visual walkthrough of the entire process, offering a clear and comprehensive understanding of how educators can create effective learning paths and how learners can engage with the materials and activities included in the path.

IV. Future Directions

To better frame the contribution of this work, we outline future promising directions aimed at strengthening model-driven engineering teaching and education. These directions involve harnessing the full potential of open educational resources in MDE and fostering collaborative learning practices.

A. OERs in MDE

In order for MDE learning activities to be packageable as OERs, we need to solve a key challenge: any MDE activity requires specific advanced tools to be accessible to the learner. Setting up and configuring these tools can be a

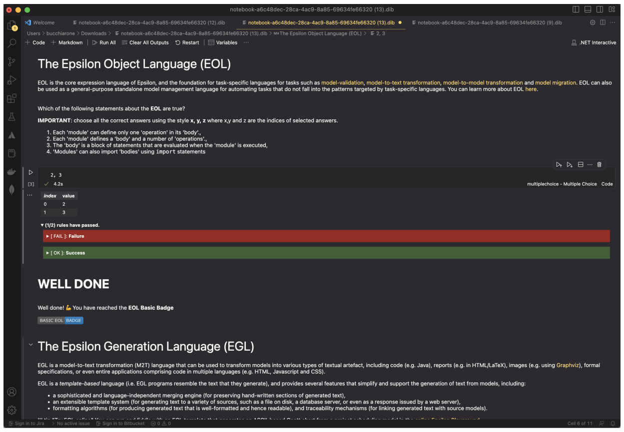

Fig. 3. EOL part of the learning path.

substantial challenge and anecdotal evidence from across the MDE community suggests that this can be a challenge that is difficult to overcome when teaching MDE [3]. In the world of OERs, these issues are even more important: each OER should be packaged as an independent unit and this requires the ability to fully specify its dependencies on specific tools and their configuration without relying on the learner having these tools installed and correctly configured already.

For MDE OERs to easily integrate with the ENCORE platform, we, therefore, require

1) A standardised format for describing and packaging MDE learning resources, and
2) An infrastructure that can interpret this packaging format, access the MDE tools required, and enable learners to engage with the educational resource.

In MDENet[8], we are developing a playground-based solution for the above challenges [27]. In summary, it expands on ideas from web-based playgrounds, such as the Epsilon Playground [28], but provides a mechanism for integrating other MDE tools easily. Learning activities are specified by providing: (1) A GitHub repository with the relevant source files and project structures; and (2) A declarative specification, identifying the relevant tools and how they will be used as part of the learning activity.

This is only the first step towards a market place of reusable MDE OERs. The infrastructure does not yet support automated assessment and cannot yet fully integrate with the ENCORE learning paths as described here. We are organising a community workshop co-located with MODELS 2023[9] to further explore requirements and architecture for such a platform.

In upcoming work, we'll leverage existing learning materials in Model-Driven Engineering (MDE), including repositories like ReMoDD [29]. Our aim is to adapt these for compatibility with the ENCORE learning platform. Additionally, we'll outline crucial concepts and their framework, fostering coherent learning paths. The Educators Symposium's MDE knowledge initiative [30] will guide our approach. We'll also highlight the value of cross-disciplinary learning. Analyzing analogous disciplines with established knowledge documents offers insights for optimizing content structure and enhancing learning experiences on platforms.

B. Tools for Collaborative Modeling Activities

In a distributed setting, collaborative modeling [31], [32] enables numerous people to work at once on the same modeling product. It comprises using methods, plans, and resources that allow various stakeholders to jointly manage, coordinate, and control various system components. In order to reduce the possibility of conflicts and divergence, access-control and coordination rules, such as locking mechanisms, model versioning systems, conflict management, and support for presence awareness, enable participants to see who else is currently working on which particular portion of the model. JJodel [33] is a cloud-based, reflective and collaborative modeling platforms that aims at minimizing the accidental complexity due to the accumulated technical debt exposed by some of the current modeling platforms. For instance, it is based on a cloud-based infrastructure that nullify costly installation and upgrading procedure, while providing the users with the possibility of collaboratively work on the same artifact. In collaborative modeling environments, the Client-Server Reactive Real-Time Architecture (CSRRTA) is a well-liked method that enables numerous users to collaborate on a shared model simultaneously. A central server is used by CSRRTA as the model's authoritative source of truth, enabling real-time collaboration and prompt updates on any modifications made simultaneously by participants. The server, which manages concurrent modifications, stores the model data, and broadcasts updates to all associated clients, is at the center of this design. All users will have a consistent and current view of the model thanks to this.

Integrating JJodel with ENCORE poses a challenge, especially for collaborative activities involving user groups. Managing evolving personalized learning paths, including collaborative external activities using tools like JJodel, demands careful consideration.

The challenge is synchronizing JJodel's collaborative activities with ENCORE's personalized learning paths. This demands seamless data exchange, ensuring JJodel's collaboration enhances learners' journeys. Managing groups, permissions, and material access in JJodel is crucial for a secure, smooth collaborative experience.

Integrating JJodel and ENCORE empowers educators and learners for collaborative modeling. This fosters teamwork, knowledge sharing, and collective problem-solving. Synergizing these platforms offers valuable group exploration of complex concepts, ensuring personalized paths evolve dynamically with JJodel's external collaborative activities.

V. Conclusion

This paper outlines challenges in creating learning experiences and the role of Open Educational Resources (OERs)

[8]www.mde-network.org
[9]https://modellingtoolsforteaching.github.io/

in addressing them. We introduce the ENCORE conceptual model, highlighting OERs and personalized learning paths as key educational components. Our goal is to build tools for personalized learning and a community-driven OER database. Looking ahead, we see potential in leveraging OERs for MDE education and fostering collaborative learning to enrich the learning journey.

To unlock the platform's potential, we must engage the model engineering education community. Integrating tools and refining our approach for openness and personalization are vital next steps. By collaboratively tackling these challenges, we pave the way for enhanced, efficient, and engaging learning experiences.

Acknowledgment

This work has been partially supported by the Erasmus+ Programme of the European Union ERASMUS-EDU-2021-PI-ALL-INNO, Project ENCORE - ENriching Circular use of OeR for Education (Reference number 101055893). Zschaler and Barnett's contribution was partly funded by the UK Engineering and Physical Sciences Research Council (EPSRC) through the MDENet grant (EP/T030747/1).

References

[1] "Open Educational Resources (OER)," https://en.wikipedia.org/wiki/Open_educational_resources, accessed: 2023-07-19.

[2] F. Ciccozzi, M. Famelis, G. Kappel, L. Lambers, S. Mosser, R. F. Paige, A. Pierantonio, A. Rensink, R. Salay, G. Taentzer, A. Vallecillo, and M. Wimmer, "How do we teach modelling and model-driven engineering? a survey," in *Proc. of the 21st ACM/IEEE Int. Conf. on Model Driven Engineering Languages and Systems: Companion Proceedings*, ser. MODELS '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 122–129.

[3] A. Bucchiarone, J. Cabot, R. F. Paige, and A. Pierantonio, "Grand challenges in model-driven engineering: an analysis of the state of the research," *Softw. Syst. Model.*, vol. 19, no. 1, pp. 5–13, 2020.

[4] G. De Angelis, A. Pierantonio, A. Polini, B. Re, B. Thönssen, and R. Woitsch, *Modeling for Learning in Public Administrations—The Learn PAd Approach*. Cham: Springer International Publishing, 2016, pp. 575–594.

[5] A. García-Holgado and F. J. García-Peñalvo, "A metamodel proposal for developing learning ecosystems," in *Learning and Collaboration Technologies. Novel Learning Ecosystems*, P. Zaphiris and A. Ioannou, Eds. Cham: Springer International Publishing, 2017, pp. 100–109.

[6] S. Retalis, A. Papasalouros, and M. Skordalakis, "Towards a generic conceptual design meta- model for web-based educational applications," 2002.

[7] Y. Arroyo, A. I. Molina, M. A. Redondo, and J. Gallardo, "Learn-ciam: A model-driven approach for the development of collaborative learning tools," *Applied Sciences*, vol. 11, no. 6, 2021.

[8] M. Caeiro-Rodríguez, M. L. Nistal, and L. Anido-Rifón, "An eml meta-model proposal for the modelling of collaborative educational practices," in *VII Simposio Internacional de Informática Educativa (SIIE'05). Leiría, Portugal. ISBN*. Citeseer, pp. 972–95 205.

[9] D. Pérez-Berenguer and J. García-Molina, "Indieauthor: A metamodel-based textual language for authoring educational courses," *IEEE Access*, vol. 7, pp. 51 396–51 416, 2019.

[10] C. S. Longstreet and K. M. Cooper, "Developing a meta-model for serious games in higher education," in *2012 IEEE 12th International Conference on Advanced Learning Technologies*, 2012, pp. 684–685.

[11] F. L. da Silva, B. K. Slodkowski, K. K. A. da Silva, and S. C. Cazella, "A systematic literature review on educational recommender systems for teaching and learning: Research trends, limitations and opportunities," *Education and Information Technologies*, vol. 28, no. 3, p. 3289–3328, sep 2022.

[12] "AI is the key to scaling personalized, one-on-one instruction," https://www.eschoolnews.com/digital-learning/2022/07/26/ai-intelligent-bots/, accessed: 2023-07-19.

[13] R. H. Mogavi, C. Deng, J. J. Kim, P. Zhou, Y. D. Kwon, A. H. S. Metwally, A. Tlili, S. Bassanelli, A. Bucchiarone, S. Gujar, L. E. Nacke, and P. Hui, "Exploring user perspectives on chatgpt: Applications, perceptions, and implications for ai-integrated education," *CoRR*, vol. abs/2305.13114, 2023.

[14] "ENCORE database," https://encore-db.grial.eu/, accessed: 2023-07-19.

[15] G. Fairbanks, *Just enough software architecture: a risk-driven approach*. Marshall & Brainerd, 2010.

[16] L. Burgueño, A. Vallecillo, and M. Gogolla, "Teaching uml and ocl models and their validation to software engineering students: an experience report," *Computer Science Education*, vol. 28, no. 1, pp. 23–41, 2018.

[17] S. Moisan and J.-P. Rigault, "Teaching object-oriented modeling and uml to various audiences," in *International Conference on Model Driven Engineering Languages and Systems*. Springer, 2009, pp. 40–54.

[18] S. Frezza and W. Andersen, "Interactive exercises to support effective learning of uml structural modeling," in *Proceedings. Frontiers in Education. 36th Annual Conference*. IEEE, 2006, pp. 7–12.

[19] C. Starrett, "Teaching uml modeling before programming at the high school level," in *Seventh IEEE International Conference on Advanced Learning Technologies (ICALT 2007)*. IEEE, 2007, pp. 713–714.

[20] M. Daun, A. Salmon, T. Weyer, K. Pohl, and B. Tenbergen, "Project-based learning with examples from industry in university courses: an experience report from an undergraduate requirements engineering course," in *2016 IEEE 29th International Conference on Software Engineering Education and Training (CSEET)*. IEEE, 2016, pp. 184–193.

[21] "MDENet Community Web Site," www.mde-network.org, accessed: 2023-07-19.

[22] "MDENet Community Platform," https://community.mde-network.org/, accessed: 2023-07-19.

[23] J. Kunze and T. Baker, "The dublin core metadata element set," Tech. Rep., 2007.

[24] S. Weibel, J. Kunze, C. Lagoze, and M. Wolf, "Dublin core metadata for resource discovery," Tech. Rep., 1998.

[25] L. Gonnord and S. Mosser, "Practicing domain-specific languages: From code to models," in *Proceedings of the 21st ACM/IEEE International Conference on Model Driven Engineering Languages and Systems: Companion Proceedings*, ser. MODELS '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 106–113.

[26] "Eclipse Epsilon," https://www.eclipse.org/epsilon/, accessed: 2023-07-19.

[27] W. Barnett, S. Zschaler, A. Boronat, A. Garcia-Dominguez, and D. Kolovos, "An online education platform for teaching MDE," in *Proc. Educators Symposium at MODELS 2023*, 2023.

[28] D. Kolovos and A. Garcia-Dominguez, "The Epsilon playground," in *Proc 25th Int'l Conference on Model Driven Engineering Languages and Systems: Companion Proceedings*, ser. MODELS'22. Association for Computing Machinery, 2022, pp. 131–137. [Online]. Available: https://doi.org/10.1145/3550356.3556507

[29] R. B. France, J. M. Bieman, and B. H. C. Cheng, "Repository for model driven development (remodd)," in *Models in Software Engineering, Workshops and Symposia at MoDELS 2006, Genoa, Italy, October 1-6, 2006*, ser. Lecture Notes in Computer Science, T. Kühne, Ed., vol. 4364. Springer, 2006, pp. 311–317.

[30] L. Burgueño, F. Ciccozzi, M. Famelis, G. Kappel, L. Lambers, S. Mosser, R. F. Paige, A. Pierantonio, A. Rensink, R. Salay, G. Taentzer, A. Vallecillo, and M. Wimmer, "Contents for a model-based software engineering body of knowledge," *Softw. Syst. Model.*, vol. 18, no. 6, pp. 3193–3205, 2019.

[31] M. Franzago, D. Di Ruscio, I. Malavolta, and H. Muccini, "Collaborative model-driven software engineering: a classification framework and a research map," *IEEE Transactions on Software Engineering*, vol. 44, no. 12, pp. 1146–1175, 2017.

[32] D. Di Ruscio, A. Di Salle, D. Di Vincenzo, J. Di Rocco, A. Pierantonio, and G. Tinella, "JJodel – A reflective cloud-based modeling framework," 2023, submitted for publication.

[33] D. Di Vincenzo, J. Di Rocco, D. Di Ruscio, and A. Pierantonio, "Enhancing syntax expressiveness in domain-specific modelling," in *2021 ACM/IEEE International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C)*. IEEE, 2021, pp. 586–594.