DigiFit4All – Conceptualization of a Platform to Generate Personalized Open Online Courses (POOCs)

Stefan Pasterk, Lukas Pagitz, Albin Weiß, and Andreas Bollin

Department of Informatics Didactics, University of Klagenfurt, Austria
{stefan.pasterk,lukas.pagitz,
albin.weiss,andreas.bollin}@aau.at

Abstract. The need for evaluated digital resources arose over the last years as blended learning applications became even more popular and distance learning essential. Besides well-known *massive open online courses (MOOCs)*, approaches called *personalized open online courses (POOCs),* which present individual learning resources to the users, are developed. Personalization is, in most cases, based on different information about the users. *DigiFit4All* is a project to develop a platform for POOCs, including open teaching and learning resources for lower and higher education. In the background, a competency model is used, which enables the definition of learning outcomes for courses, the determination of learning paths, and the assignment of learning resources. With the help of pre-tests, competencies that are already known can be identified. The learning resources are integrated into a *learning management system (LMS)* which imports them from open repositories. This contribution gives detailed information about the concepts of the *DigiFit4All* project and compares it to other approaches.

Keywords: POOC, e-learning, digital learning resources, competency models

1 Introduction

In our society, the relevance of digital teaching and learning is constantly increasing. Various forms of blended learning, e-learning or even distance learning are widespread and partly necessary due to the current pandemic. A lot of material is available online in the form of massive open online courses (MOOCs). These platforms differ in several aspects concerning technological background and usability. The presentation of the learning resources is, in most cases, the same for all participants, and heterogeneity is not considered. Further, MOOCs often follow an asynchronous form and present their content in the form of videos, quizzes or reading tasks. Approaches that offer individually prepared courses and incorporate heterogeneity are called *personalized open online courses (POOCs).* They often base on learner profiles and an analysis of the results from prior assessments [1].

The project *DigiFit4All* started in May 2020 and has the aim to develop a platform for POOC creation. Based on competency models, which are stored as graphs, teachers can create their own courses and identify learning paths to reach the goals of the course. Students can attend individual courses by participating in pre-tests to determine competency profiles. In addition to the platform, resources for computer science and digital education topics are developed during the project. A cooperation between the University of Klagenfurt[1], the Danube University Krems[2], the Johannes Kepler University Linz[3], and the Vienna University of Technology[4] ensures that the different target groups of students in lower and higher education, teachers, and administrative staff can be covered.

This contribution summarizes findings from the research literature (section 2) that influenced design decisions during the conceptualization of the *DigiFit4All* project, and different approaches are compared (section 3). In section 4, the components of the *DigiFit4All* system are explained in more detail, and the technological background is described. Section 5 presents an overview of the user roles in the project and sample workflows for teachers and students.

2 Related Work

To get a first overview of existing systems that can generate *personal open online courses*, several approaches that have been published already are reviewed, and the most relevant are briefly described below.

In 2007, Leung et al. [2] identified students' diversity as a significant issue in terms of e-learning. To answer how course materials can be presented student-centered, they suggested an approach following the elaboration theory of instruction combined with the so-called educational ontology.

El Mawas et al. [3] discussed the personalization of massive open online courses in the context of lifelong learning. Their architecture consists of a *content management system (CMS)* and a *learning management system (LMS)* and supports different roles such as pedagogical engineer, teacher, and learner. Subjects, topics, and concepts are stored in *domain models*, while the individual results are saved in *learner models*.

Due to the low completion rates of MOOCs, Brinton et al. [4] developed the so-called MIICs. MIIC stands for *mobile integrated and individualized course* and is an *adaptive educational system* that updates students' user models based on their interaction with the course. MIICs offer multiple learning paths, whereby one path consists of several segments. One segment represents the 'smallest unit of knowledge' offered in different versions (depending on the path selected for the learner).

Another aspect in the context of MOOCs is big data. Compared to traditional classroom learning settings, online courses allow collection and analysis of learners' behavior as all actions performed by users can be logged. Xi et al. [5] described how the collected data could be interpreted using a behavior analysis model. They described a model consisting of six components, whereby the *predictive model* is the centerpiece of it. Students can access the course content, and based on their interaction with it, the

[1] https://www.aau.at
[2] https://www.donau-uni.ac.at
[3] https://www.jku.at
[4] https://www.tuwien.at

adaption engine updates their personal model. Teachers can intervene through the so-called intervention engine, which allows manual adjustments in the otherwise automatic system.

Analyzing the related work showed that many ideas in personalizing MOOCs had been made already. However, unfortunately, most of the introduced prototypes are no longer available, or the work on this subject was discontinued. However, some of the concepts that were described can or will be re-used in the *DigiFit4All* project in the presented or a similar way.

3 Components for POOCs

3.1 Used Components in Existing Projects

Analyzing related work showed that the models described in the papers consist of several components, which will be identified and explained in this section.

A *pre-test* is used to determine the prior knowledge of a student before they start with the course. Based on their individual result during this evaluation, the course is adjusted to their personal needs. In the approach of Leung et al. [2], a pre-test has to be performed by the students, which sets the ground for the individual study path, while El Mawas et al. [3] ask the learner to fill a so-called positioning questionnaire as a basis for the learner model created by the platform. Additionally, statements by the user are considered for the initiation of the model. The approaches suggested by Brinton et a. [4] and Xi et al. [5] do not contain any pre-test, and the learning paths are calculated later on.

A *post-test* is a test that has to be performed by a learner, and it is used to check the knowledge once a given activity, exercise or segment was finished. An automated mechanism that adjusts the learner model based on the users' behavior is not considered a post-test. The model of Leung et al. is the only one that let their learners submit post-tests, while the other models use different approaches of measuring the performance.

Learning paths allow the individualization of how someone learns. A person's path is adjusted to their personal needs, either depending on post-tests or calculations in the background. In the approach of El Mawas et al., the learning path is based on the positioning questionnaire and static, while the model of Leung et al. allows changes in the path depending on the pre- and post-tests. The platform by Xi et al. allows automatic as well as manual adjustments of the learning paths, and the approach of Brinton et al. considers the users' behavior in each segment to select the next one.

User behavior can be considered to create dynamic personalized courses, depending on how a learner interacts with the platform. As just described, Brinton et al. select the segment and version of the following learning object based on the results yielded in the current one. As the approach of El Mawas et al. is static, the user's behavior is not analyzed during the course. The same applies to the model of Leung et al., as it depends on the tests performed by the user. Xi et al.'s platform uses behavioral data to recommend activities to the learner.

A *repository* is a platform where learning material that can be linked to learning objects is stored. Repositories can either be open platforms where everyone is able to provide course material that can even be used outside a course or closed platforms that do not share their contents with the public. Three of the researched papers do not describe where and how the material is stored in detail; only Brinton et al. tell that all files are stored in EPUB containers, and videos are streamed via HTTP.

MOOCs are held in *learning management systems (LMS)* that offer a variety of features for course management. The analyzed approaches used different platforms for the integration of their personalized courses: Leung et al. used Moodle, El Mawas et al. used edX, and both Brinton et al. and Xi et al. did not specify the platform or used a custom-created website.

Exams are defined as actual exams that are relevant for the grade of a learner in the subject of the course. None of the four analyzed approaches supports writing an exam within the personalized course.

Learning objects can be used to modularize content for learning scenarios. They represent the teaching and learning materials and are defined following the *IEEE Standard for Learning Object Metadata (LOM)* [6]. Compared to segments in the approach of Brinton [4], learning objects collect 'smallest units of knowledge' to a didactically meaningful package. This concept appears only in the work of Leung [2].

A *microservice architecture* is a method to develop software based on small services with well-defined interfaces to easily add new services. In none of the analyzed approaches a comparable architecture is described.

3.2 Comparison of the approaches

The comparison of the approaches is based on the descriptions of the methods in the respective papers. In case any information was not found for a specific component, it is considered as not existent. As presented in table 1, all components include learning paths for course personalization. The pre-test approach is only used by two out of four models, and only Leung et al. included post-tests as well. None of the ideas presented used an open repository which allows uploading learning material. The components of the *DigiFit4All* approach together with the technical background will be described in the following section.

4 Components of the DigiFit4All Project

4.1 Overview of the Project

During the development process of the *DigiFit4All* project, literature was reviewed and approaches from related work were analyzed. Existing expertise, previous work and research results of the partner institutions had the strongest influence on the project's basic idea.

Table 1. Component comparison of personalized online course systems

Component	Approach				
	Leung [2]	El Mawas [3]	Brinton [4]	Xi [5]	DF4A
Pre-test	Yes	Yes	No	No	Yes
Post-test	Yes	No	No	No	Yes
Learning paths	Yes	Yes	Yes	Yes	Yes
User behavior	No	No	Yes	Yes	No
Repository	No	No	Yes	No	Yes
LMS Integration	Yes	Yes	No	No	Yes
Exams	No	No	No	No	Yes
Learning objects	Yes	No	No	-	Yes
Microservice architecture	-	No	No	No	Yes

Based on this prior knowledge, the following components were selected to be part of this project.

- Competency models: The background calculations for the personalization are based on competency models and learning paths within the models.
- Learning objects: The teaching and learning materials to reach competencies are developed in the form of *learning objects*.
- Assessment: Course participants are assessed before a course starts, to get information about their knowledge and which competencies they already have acquired. After finishing a course, the learners are assessed again to find out which competencies they reached during the course.
- Learning management systems (LMS): The *learning objects* are presented in form of courses in chosen LMS.

These components work together to create personalized open online courses in the *DigiFit4All* project and are described in more detail in the following sections.

4.2 Competency Models and Learning Objects

As already mentioned, the personalization of the courses in the *DigiFit4All* project is based on competency models. For this purpose, national and international curricula, educational standards, and competency models for different target groups are collected and analyzed. As a central element, the learning outcomes of the models, here called *competencies*, are part of this process. In a first step, the competencies have to be standardized because some of them contain more than one outcome. This leads to *subcomptencies* which are *part of* competencies in their original form. The models are mapped to a graph-based representation form and stored in a graph database as Pasterk and Bollin describe it [7, 8]. Competencies represent the nodes, and dependencies between them are displayed as directed edges. Pasterk defines the two dependency types *expands* and *requires*, which show that a competency *C1* is necessary to reach another competency *C2* within the same topic (*C2 expands C1*) or from another topic

(*C2 requires C1*) [9]. Nodes and edges are sometimes given by the curricula, educational standards, or competency models, but in many cases, the dependencies have to be added by experts. As the models are in general developed for one target group, i.e., primary school children or students of an undergraduate program, Pasterk presents a method to combine them to one overall model, here called the *main index*. This approach uses so-called *intersector nodes*, which collect a set of similar competencies from different models in one single node. *Intersector nodes* inherit all dependencies from their included competencies, thereby connecting competency models to each other [9]. In the *DigiFit4All* project, the approach from Pasterk and Bollin is applied with some adaptations concerning the *intersector nodes*. Similar competencies are not collected in sets but are connected over an *equal to* relation. If a competency *A* is similar to another already existing competency *B* their respective nodes are set to be *equal to* another. In the main index only the node of the first competency B is shown and used for calculations. Again, the existing competency inherits all dependencies from competencies which are later added and set to be *equal to* it. For the purpose of collecting, analyzing and combining the models, the *Graph-based Environment for Competency and Knowledge-Item* Organization (GECKO)[5] platform presented by Pasterk and Bollin in 2017 [7, 8] and described by Pasterk 2020 in detail [9] is adapted and extended by some functionalities.

In the graph-based representation, *central competencies* of the models and the *main index* can be identified and *learning paths* can be calculated. To find *central competencies*, the centrality values of the nodes are measures. This method is used in network analysis and adds information about important competencies [9]. *Learning paths* include competencies, which are necessary to reach a selected competency, by following the direction of the edges back to a selected starting node. Especially in the *main index* more than one learning path to a given competency can exist. In the *DigiFit4All* project, the *learning paths* support teachers, and lecturers to find necessary prerequisites for targeted competencies and to include them in their courses. Additional components, which are new in the *DigiFit4All* project, are *learning objects*. The mentioned LOM [6] standard includes different forms of materials, also non-digital ones, which are excluded from the *DigiFit4All* project. It is required that the materials are tagged with metadata to record information about them. Additionally, the *learning objects* are in subject to further project-specific requirements like a time limit and the highest possible degree of independence to other learning objects. These two requirements are important for the reuse of learning objects in different courses and contexts. *Learning objects* are developed to reach given competencies and are directly linked to corresponding competencies in the GECKO system.

4.3 Self-assessment for Personalization

At the beginning and the end of a course, assessment questions are addressed to learners and serve to evaluate individual skills and competencies acquired before and after a course. The results of these pre-tests are used to determine the competencies

---

[5] https://gecko.aau.at

that the learner would still have to reach to pass the course positively. These are then highlighted in the personalized online course, while topics that have already been mastered remain accessible but are kept in the background. Items for the post-tests are used to check whether a competency has been achieved after finishing the course. They can be part of the exam and, with that, the grading. Both types of questioning, assessment questions and test items, are carried out via the *KAUA (Košice and Alpen-Adria University Assessment)*[6] platform of the Department of Informatics Didactics at the University of Klagenfurt. For the purposes of the *DigiFit4All* project, the platform is modified and extended. The idea behind the *KAUA* platform is to use hash values to identify users instead of storing specific user data [10].

4.4 Support for Learning Management Systems

To allow teachers to create courses for their students, integrations for *learning management systems (LMS)* are being created. Once the corresponding plugin for the LMS is installed, it will allow teachers to import courses for their classes. Students are then able to perform the pre-test and use their personalized course. As a first LMS, Moodle – which was already used by Leung et al. [2] in their implementation of personalized courses – will be supported as it is open-source, therefore available for free and open for further modifications.

4.5 Technical Background

The *DigiFit4All* system is designed as a microservice architecture. Most of the services are Spring Boot applications and therefore implemented in Java. During the project, two Single Page Applications (SPA) are also developed. Among other things, the application also provides an interface to an external repository and LMS. The repository in turn provides an interface to the LMS. Figure 1 below shows a section of this architecture. The API *Gateway Services* serves as the central access point to the backend. All stateless communication runs through this gateway, so the different microservices cannot be addressed directly. In figure 1, the most important services of *GECKO* and *KAUA* are visible. The *Auth Service* manages all users and authentication. The *Graph Service* manages the Neo4J graph database, which is used to map the various competency models. The *Liza Service* provides various recommendations when creating a competency model. This application is implemented in Python and communicates with the rest of the Spring-based application via the *Liza Service Sidecar*. The *Library* is not a standalone service but is added by each core service as a dependency. The *Survey Service* handles the creation of questions and questionnaires. The created questions can be added to surveys, competencies or learning objects. The survey data and the results of the tests for the personalization of the different users are stored anonymously. Each time a user logs in, a hash value is generated from the user credentials in the *UID Service*. This hash value serves as an identifier for the user.

---

[6] https://kaua.aau.at

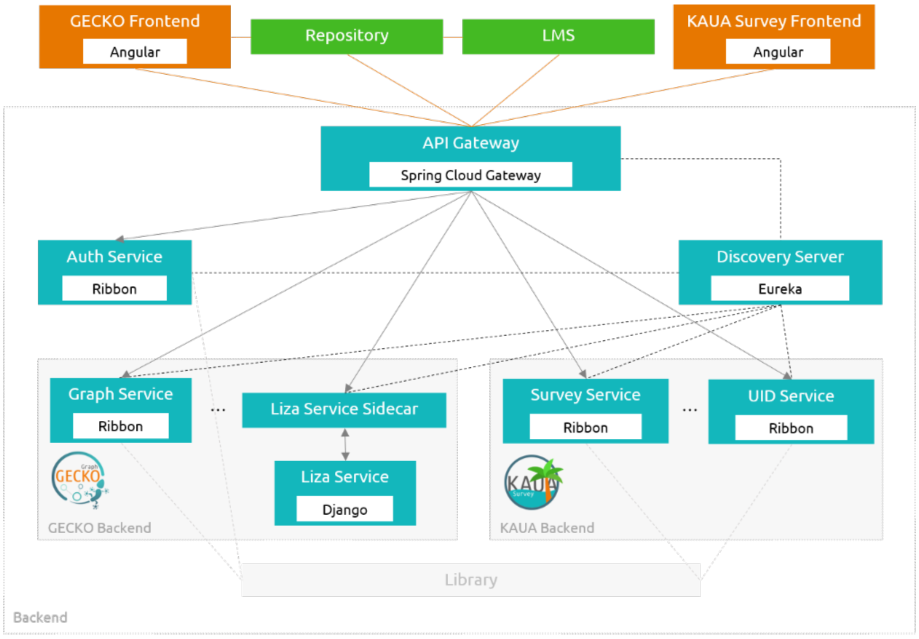

Fig. 1. Detail of the *DigiFit4All* system architecture

5 Roles and Workflows in the DigiFit4All Project

5.1 User Roles

Within the *DigiFit4All* project, different user roles exist for different functionalities in *GECKO* as well as in *KAUA*. In *GECKO*, users can register for free but at this stage only have the possibility to view the *main index* as a list of competencies or in its graph representation. Every user can send a request to get the role of a *GECKO-author* or of a *GECKO-lecturer*. *GECKO-authors* are responsible for the competency models collected in *GECKO* and are able to upload or build their own models for their selected target group. The individual models have to be submitted and are evaluated by an expert team. In contrast to that, *GECKO-lecturers* use the given competencies in the *main index* to represent their goals for own courses. This allows teachers or lecturers to follow the suggestions of the system or to choose their own ways. Two of the three roles existing in *KAUA* work in a similar matter. *KAUA-authors* develop questions and question modules of different forms and for different purposes. The *KAUA-lecturers* combine given and evaluated modules to prepare their own surveys. The last role which does not require a typical registration is the *KAUA-student* role. An enrolled student can access their personal course through the LMS using their student data. After submitting the pre-test, students can start with the personalized online course. In the following section, the workflows for lecturers and learners within the project are described in more detail.

5.2 Sample Workflows

Workflow for teachers. Based on the knowledge about the components of the project, the typical workflow for a teacher or lecturer starts with the assignment of competencies to a course in *GECKO*. They need to login to the platform with a corresponding account and define a new course or work on an existing one. Starting to work on a new model, the first step is to select the target competencies from the *main index*. As an example, a teacher wants to introduce simple SQL statements in their databases course. For this category several competencies can be found but the following two are selected by the teacher:

1. Learners formulate SQL queries using a projection.
2. Learners formulate SQL queries with a selection.

These competencies are based on the *ACM/IEEE Computer Science Curriculum 2013*[7] but go more into detail. In the background, the system calculates necessary prerequisites for the target competency and suggests the results to the teacher. They receive a list of additional competencies including competencies from the topics *relation schema* and *datatypes*. However, for the teacher only competencies for SQL statements are of interest. That is why for the two selected target competencies, the following list of competencies is added to the course:

3. Learners can reproduce what is meant by a projection in SQL queries.
4. Learners understand and can explain what is meant by a projection in SQL queries.
5. Learners can reproduce what is meant by a selection in SQL queries.
6. Learners understand and can explain what is meant by a selection in SQL queries.

Competencies 3 and 4 are prerequisites for (1), and 5 and 6 for (2), as they deal with the same topics on different cognitive levels. The teacher decides whether as foreknowledge suggested competencies should be added to the course or not (step 2). In the given example the teacher also adds the prerequisites to the course. After the competencies are selected, the teacher gets a list of teaching and learning materials that can be used to reach the corresponding competencies (step 3). Again, the teacher selects those materials that are appropriate for the course (step 4). This step includes the option to create a final exam with questions related to selected competencies and marked to be exam questions. For the competencies in the database-example, the teacher chooses interactive videos to be appropriate for the course but chooses not to have a final exam through the system. These steps are illustrated in figure 2 and show the actions taken in the *GECKO* platform. Additionally, a teacher can create own surveys and tests in *KAUA*. Again, existing modules for different purposes can be chosen from the existing collection, or own questions can be defined. Once the teacher enabled the connection to *GECKO*, the course created can be selected by its title. The learning objects corresponding to the selected competencies are imported from the repository into the LMS section.

[7] https://www.acm.org/binaries/content/assets/education/cs2013_web_final.pdf

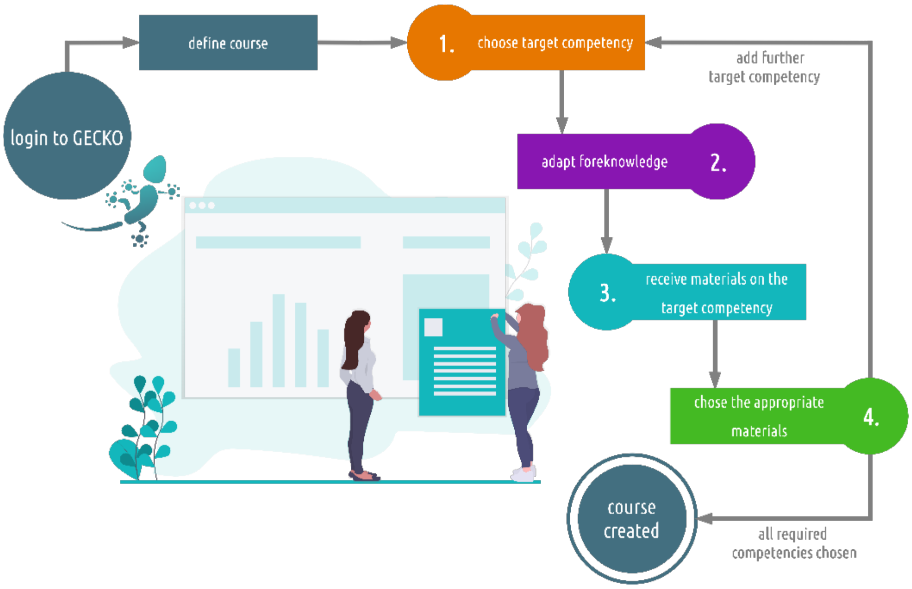

Fig. 2. The workflow for teachers and lecturers in the *DigiFit4All* project

Workflow for learners. As mentioned above, the students do not register to the system on their own. In higher education, a student enrolls for a course in the affiliated institution's usual system. Through the student account, they get access to the used LMS and with it to the learning resources. For students in schools, the administrator or the teacher must generate a list of student accounts in the LMS system of the school. With that account, the students can enter the online course. From this point on, the workflow for all kind of students is the same. The first step is the pre-test. Depending on the LMS, the students can participate directly in the LMS or have to follow a given link to *KAUA*. Based on the results of the pre-test, a competency profile of each student is created and stored. Looking at the example for a database course with competencies in SQL statements, a section of the pre-test would include self-assessment questions in the following form and for the corresponding competencies (see the referencing numbers in the brackets):

- Have you heard about projections in the context of databases and SQL statements? (Competency 3)
- Can you explain a projection in the context of databases and SQL statements? (Competency 4)
- Have you created an own SQL query including a projection? (Competency 1)
- Have you heard about selections in the context of databases and SQL statements? (Competency 5)
- Can you explain a selection in the context of databases and SQL statements? (Competency 6)
- Have you created an own SQL query including a selection? (Competency 2)

As a next step, the student has access to the learning material of the online course. With the help of the competency profile, the course is personalized for each student. This means, the sections for already reached competencies are put into the background and the other sections are highlighted. Through this approach, the student identifies on the first glance which sections are important for them. However, they still have access to all learning resources, as they are relevant for the exam.

So, for the given example it can be assumed that a student has already read something about SQL statements and knows the difference between projection and selection but has never created an own query. This student only replies with 'No' to the questions for competencies 1 and 2. With this profile, the course for this student includes four collapsible sections which contain the interactive videos for competencies 3 to 6. Two already opened sections present the two videos for competencies 1 and 2, which are not yet reached by the student. After a student has worked through all the materials, they participate in a post-test. This test depends on the selected competencies for the course and not on the profiles of the students. Therefore, even questions about topics, which the student already knew about before participating in the course, can be included. Results from the pre-test have no influence on the post-test.

In case of the given example about SQL statements, post-test questions can include the following questions or tasks:

- What is a projection in the context of databases?
- What is necessary to create a simple SQL query?
- What is the result of a given query over a given table?
- Create a query to get given results from a given table.

It has to be mentioned that the post-test is not the same as an exam, which has to be passed to receive a grade for the course. The post-test informally shows the progress of participating students but does not influence the grades.

6 Conclusion

This paper presents an analysis of existing approaches for *personalized open online courses (POOC)* and discusses differences and similarities. Research in related work shows that several prototypes were created; however, most of them are no longer available as they only served as proof of concepts or were discontinued for other reasons. Several components used in the mentioned approaches could be identified and compared to each other as well as to the *DigiFit4All* project. The results show that all selected approaches use *learning paths* in the background. Three of five approaches include *pre-tests* and some kind of *LMS integration*. Considering other points like *post-tests*, *exams* or *user behavior,* the approaches differ more. A detailed description of the structure of the *DigiFit4All* project includes the use of *competency models* in the background to create courses and calculate *learning paths*, the development of independent and reusable *learning objects*, the necessity of assessment before and after the course, *learning management systems* support as well as the technical background. By explaining the roles and two sample workflows, it is shown what users (both learners and lecturers) can do on the *DigiFit4All* platform. The project started one year ago, and a first prototype of the *GECKO* platform is already online, including some first competency models. In a next step, a repository will be created and linking learning objects to competencies will be supported, as well as the import into learning management systems.

References

1. Sanchez-Gordon, S., Luj´an-Mora, S.: Moocs gone wild. In: INTED2014 Proceedings. pp. 1449–1458. 8th International Technology, Education and Development Conference, IATED (2014)
2. Leung, E.W.C., Li, Q.: An experimental study of a personalized learning environment through open-source software tools. IEEE Transactions on Education 50(4),331–337 (2007)
3. El Mawas, N., Gilliot, J.M., Garlatti, S., Euler, R., Pascual, S.: As one size doesn't fit all, personalized massive open online courses are required. In: McLaren, B.M., Reilly, R., Zvacek, S., Uhomoibhi, J. (eds.) Computer Supported Education. pp.470–488. Springer International Publishing, Cham (2019)
4. Brinton, C.G., Rill, R., Ha, S., Chiang, M., Smith, R., Ju, W.: Individualization for education at scale: Miic design and preliminary evaluation. IEEE Transactions on Learning Technologies 8(1), 136–148 (2015)
5. Xi, J., Chen, Y., Wang, G.: Design of a personalized massive open online course platform. International Journal of Emerging Technologies in Learning (iJET) 13(04), 58–70 (2018)
6. IEEE: Standard for learning object metadata (ieee 1484.12.1-2020) (2020), https://standards.ieee.org/standard/1484121-2020.html
7. Pasterk, S., Bollin, A.: A graph-based approach to analyze and compare computer science curricula for primary and lower secondary education. In: Proceedings of the 2017 ACM Conference on Innovation and Technology in Computer Science Education. pp. 365–365. ITiCSE '17, ACM, New York, NY, USA (2017)
8. Pasterk, S., Bollin, A.: Graph-based analysis of computer science curricula for primary education. In: 2017 IEEE Frontiers in Education Conference (FIE). pp.1–9 (2017)
9. Pasterk, S.: Competency-Based Informatics Education in Primary and Lower Secondary Schools. Ph.D. thesis, University of Klagenfurt - Department of Informatics Didactics (2020)
10. Bollin, A., Kesselbacher, M., Mößlacher, C.: Ready for computing science? a closer look at personality, interests and self-concept of girls and boys at secondary level. In: Kori, K., Laanpere, M. (eds.) Informatics in Schools. Engaging Learners in Computational Thinking. pp. 107–118. Springer International Publishing, Cham (2020)