

IO5 -INSTRUCTIONS FOR WS

INSTRUCTIONS FOR AN INTERACTIVE CONTENT DEVELOPMENT WORKSHOP

Table of contents

I. INTRODUCTORY THOUGHTS.....	2
II. SUGGESTED STEPS FOR DIGITAL CONTENT DEVELOPMENT.....	2
2.1 Introduction: getting to know the application	3
2.2 Developing a pedagogical concept: who-what-for-whom-when-how?.....	4
2.2.1 Content development requires thinking about: who-what-for-whom-when-how?.....	4
2.2.3 Definition of the structure: difficulty levels and their criteria.....	5
2.3. Preparation of analogue content: draft terms of reference.....	5
2.4 Script writing phase.....	5
2.5 Creative editing: digitisation	6
2.6 Planning the learning activity.....	6
2.6.1. Replayability	6
2.6.2 Ascending/descending.....	6
2.6.3. Playing order of tasks.....	6
2.6.4 How to play the path: linear-synthetic, test-practice	7
2.6.5. Structure of the superunit	7
2.7 Testing.....	7

I. INTRODUCTORY THOUGHTS

During the project, the teachers involved in the collaborative work, needs assessment and testing have shown an intense interest in the possibility of creating independent content. In this document, we would like to help those interested in the methodology of digital content development. As one of the main strengths of the LTHP application is the ease of content development, this professional aspect also plays a significant role in dissemination and multiplication activities, and task writing is often the central topic of workshops promoting the educational tool. The description aims to provide professional guidance for the implementation of an interactive workshop.

The Instructions for an interactive content development workshop discusses the steps of content development in a practice-oriented way, providing all relevant information and the documentation to be prepared at each stage. During the implementation of the workshop, the session leader simply needs to follow the steps in the document to share effective content development experiences with the participants.

II. SUGGESTED STEPS FOR DIGITAL CONTENT DEVELOPMENT

During the content development workshop, the suggested steps in the process of writing a self-assigned task should be discussed with the participants. For each step, it is necessary to put the task into practice and to perform the respective actions in the LTHP system. The proposed 7 stages of content development are based on the 6-step digital curriculum development model, which can be easily adapted to the LTHP system for the creation of stand-alone content.

The steps of the e-learning curriculum structure are not mandatory, but can provide guidance and help for beginners. The implementation of the individual steps can be freely adapted to local needs, but it is always worth considering the specific criteria. The documents for each step should not always be prepared, they may be omitted altogether, and it may be worthwhile to improve the documentation already prepared with the new aspects. As the LTHP is working with an innovative methodology that predefines many elements of the learning/teaching process, some steps may require less elaboration in a stand-alone format.

The proposed process of digital curriculum development is a multi-stage process consisting of 7 activity phases, during which the original concept and plans may need to be revisited several

times in the light of the completed tasks, and tasks may need to be rewritten in order to maintain unity. However, once you get to grips with the logic of how the application works and how to creatively create content, it can become an enjoyable and useful part of your teaching routine.

During the workshop, it is necessary to work on content development along the suggested steps:

1. introductory phase: getting acquainted with the application, creating test tasks
2. developing a pedagogical concept of who-what-for-whom-when-how
3. preparation of analogue content: outline of tasks
4. script writing: assigning tasks to game engines, preparing all illustrations, pictures, images, texts to be used, final text/instruction
5. planning the learning activity: planning the activity means planning in advance how the students will interact with the educational content, in our case, determining the necessary system settings.
6. creative editing: digitisation
7. testing

2.1 Introduction: getting to know the application

Before anyone starts to create their own content, it is worth getting to know the application and the possibilities it offers. There are several ways to do this, you can view the finished content after registering, or you can view the demo material created for this purpose without registering. Creating test exercises: before designing the learning material, it is advisable to familiarise yourself with the different game engines, creating a test exercise with each one, so that you are clear about the formal and content criteria to which the content should be adapted. This step is of paramount importance for subsequent development work, otherwise there is a risk that the pre-written tasks will not fit the logic of the platform.

2.2 Developing a pedagogical concept: who-what-for-whom-when-how?

The conceptualisation process essentially addresses the following questions: pedagogical objectives, themes, structure, and how the content is to be taught. It gives the framework, technical, content and visual direction of the curriculum, defines the technical, formal and quantitative criteria, the levels of difficulty and the criteria for grading them, the skills to be developed and the objectives for knowledge development, the types of questions to be asked, the structure of the tasks, etc.

2.2.1 Content development requires thinking about: who-what-for-whom-when-how?

Learning and teaching are inseparable from the context in which they take place. Different objectives, didactic principles, content and methods are justified and successful in different contexts. Before you want to produce content or use existing content, you need to identify the following factors that will determine the development process:

- individual variables of the students: age, prior knowledge, motivation, local needs, socio-cultural background, etc.
- forms of education: classroom (e.g. school, classroom), home (homework), online, blended/mixed
- pedagogical purpose: learning new material, revision, practice, monitoring, assessment, catching up, preparation for admission
- knowledge and competences to be developed: what exactly do I want to improve?
and what competences are being developed
- method of curriculum development: individual or group. It is also possible to produce content cooperatively, based on a common concept, by breaking down a topic into sub-topics and dividing them within a group
- curricular characteristics: level jump - linear

In linear content development, one set of tasks is created, whereas in the case of level-jumping, a multi-level set of tasks is created, which is made up of superunits (A-S-M). The consequences of this in terms of content structure: in the case of linear content, there is the possibility to welcome students before the tasks, e.g. to prepare them for learning, to tune in pdf. However, in the case of synoptic content, only superunits can be used, otherwise the system will not move on to the next task.

- students' experience with digital learning applications
- attitude: attitude to learning

2.2.3 Definition of the structure: difficulty levels and their criteria

Once the basic questions have been answered, it is necessary to decide whether to create linear (task set) or level-jump (task system) content. If it is linear, there is nothing to do, but even in this case it is worth defining the order and rhythm of the tasks, e.g. from simpler tasks to more complex ones.

If you want to create a level-jumping curriculum, you need to structure the planned tasks according to the methodology, according to several levels of difficulty in order to differentiate them. The levels of difficulty can even be aligned to the grading. Another suggested leveling could be: minimum, easy, medium, difficult, competition.

For each level, several tasks of the same task type should be designed, so that if the student drops down a level, he/she does not get the same task again that he/she has already encountered once. A minimum of 3 tasks per level is recommended, but this can be considerably more if the topic requires it. However, it is important to note that at the top level, if there is no further progression, the system will run through all the tasks, so it is worth rationalising the number of tasks here.

In the sense of multi-round task writing, it is useful to develop each of the Basic Tasks first, and then to develop the Support Tasks and Explanations for the Basic Tasks.

2.3. Preparation of analogue content: draft terms of reference

Experience has shown that this section can be omitted at the discretion of the writer. In the manuscript writing step, methodologically thought-out (but not yet final) educational content is created on the basis of the objectives and themes set out in the concept, which forms the background and basis of the curriculum. The outline is the system of information/content to be processed, not the concrete tasks themselves.

2.4 Script writing phase

Scriptwriting is in fact a further development of scriptwriting: the outline drawn up in the previous phase is supplemented by the selection of appropriate game engines for the tasks, in terms of content, form and quantity. At this stage, it is worthwhile to collect all the professional material to be used and prepare it for digitisation, such as images, videos, texts, etc.

2.5 Creative editing: digitisation

During the digitisation process, the tasks are recorded in the Teaching Interface, the pre-written tasks are used to create the game engines.

2.6 Planning the learning activity

Planning the learning activity: activity planning is the pre-planning of the students' activities related to the educational content, in our case the definition of the necessary system settings.

This phase involves thinking through the learning activity, specifically planning what students will do in the system. In our case, the learning activity is defined by the LTHP application methodology. In fact, when designing the learner activity, it is necessary to make system settings that determine the progression through the curriculum. These are:

2.6.1. Replayability

If a pathway is replayable, the student can open and work on it several times, otherwise he/she can only move through it once.

2.6.2 Ascending/descending

Depending on his/her results, the student moves up or down in the curriculum between levels of difficulty. The settings that determine the movement between levels can be modified to distinguish between difficult, medium and easy ascents and descents:

Difficult drop: falling down to the previous level after the third wrong solution

Moderately difficult fall: the student falls to the previous level after the second wrong answer

Easy fall: falls to the next level after the first wrong answer

Difficult climb: student moves up to a higher level after the third correct solution

Moderate difficulty: the student moves up to the next level after the second correct answer

Easy promotion: the student moves up a level after the first good solution

2.6.3. Playing order of tasks

Within the difficulty level: if you tick the random play option, students will receive the tasks in random order. If not ticked, they will follow the order in which they are arranged on the interface. For linear courseware, the layout can be used to define the learning curve, e.g. from simple to complex.

2.6.4 How to play the path: linear-synthetic, test-practice

A common misunderstanding arises from the confusion between curriculum and pathway setting, so it is essential to separate the two. Students will not be working directly with content in the database, this will first need to be shared with the current group of students in the form of a pathway. The created linear or synoptic learning content will be seen by the students according to the playback mode of the path, so it is useless to have synoptic content if the playback mode is not selected. For linear learning content, the linear playback mode should be selected, while for the jump to level path, the jump to level path should be selected, and it is always advisable to check this before working with students.

Test - practice

The playback mode of the Basic Tasks and Help Tasks can be either test or practice.

Depending on the pedagogical purpose, the test setting can be used to assess the students' performance, in which case the system will allow the student to move on to the next task, both for good and bad answers. On the other hand, when playing in practice mode, the student cannot go further until he/she gives the correct answer to the question asked.

2.6.5. Structure of the superunit

The number of repetitions depends on the number of times the student is given back the Basic Task, the Help Task and the Explanation in case of a failed solution. Different solutions are possible, the principle is that the Basic Task should be repeated at least once.

Handling of accidental wrong answers: currently, the interface does not allow the possibility to modify an answer once marked before the time given to solve the problem has elapsed, even if you have changed your mind. To overcome this problem, a solution could be to give the student the basic task twice in succession, in case he/she has inadvertently marked an incorrect answer:

A-A-S-A-M

For more complex tasks, it is reasonable to repeat the basic task more than once (A-A-S-A-M-A), while for easier tasks, it is less often. It can be motivating if the student is given the original problem again after the explanation, because knowing that he/she can try again, he/she is more likely to read the explanation.

2.7 Testing

The assignment-writing process concludes with testing, during which hidden errors can be revealed, the content of the course material can be finalised on the basis of what has been learned, and minor errors can be corrected to make the material smoother to use next time.