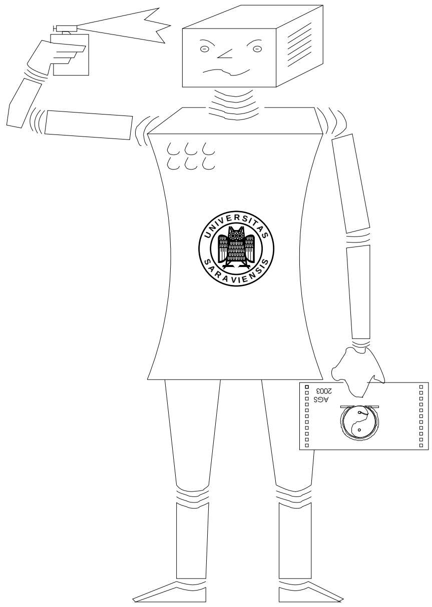
SEKI-REPORT ISSN 1437-4447

UNIVERSITÄT DES SAARLANDES
FACHBEREICH INFORMATIK
D–66123 SAARBRÜCKEN
GERMANY
WWW: http://www.ags.uni-sb.de/



Pedagogical Rules in ActiveMath and their Pedagogical Foundations

Carsten Ullrich
FR Informatik, Saarland Univ.
cullrich@ags.uni-sb.de

SEKI Report SR–2003–03

This SEKI Report was internally reviewed by:

Erica Melis
FR Informatik, Universität des Saarlandes, D–66123 Saarbrücken, Germany
E-mail: melis@activemath.org
WWW: http://www.activemath.org/~melis/

Editor of SEKI series:

Claus-Peter Wirth
FR Informatik, Universität des Saarlandes, D–66123 Saarbrücken, Germany
E-mail: cp@ags.uni-sb.de
WWW: http://www.ags.uni-sb.de/~cp/welcome.html

Pedagogical Rules in `ActiveMath` and their Pedagogical Foundations

Carsten Ullrich
FR Informatik, Saarland Univ.
`cullrich@ags.uni-sb.de`

July 22, 2003

Abstract

One added value an intelligent learning environment offers over traditional learning media is adaptivity to the individual needs of the user. `ActiveMath` offers adaptivity by dynamically assembling learning materials to a course individually tailored to the learner's needs. Its course generation uses pedagogical rules that declaratively represent instructional knowledge. These rules are based on instructional design theories, learning outcomes, and individual differences. This technical report serves to explain the pedagogical principles behind the currently available rules and to describe the instantiation of these principles in the rules of `ActiveMath`.

1 Motivation

One advantage a dynamic E-Learning system offers over static traditional learning media is adaptivity. It can select from the learning materials those that are most suited for the individual user. However, adaptivity in itself is only a technical feature. Its full potential for learning will only be reached if implementation is paired with pedagogical expertise.

`ActiveMath` offers several kinds of adaptivity; namely with respect to content, presentation, and feedback. In this paper, I will focus on adaptivity with respect to selection and composition of content. This report serves to describe the pedagogical principles on which `ActiveMath`'s adaptivity is based on, and to provide a detailed explanation of the instantiation of these principles in the rules of `ActiveMath`.

I will start by describing the pedagogical theories that served as a basis for my work (Section 2) and then present the concrete instantiations of these theories in the rules used in `ActiveMath` currently (Section 3).

2 Pedagogical Background

Each of the following sections describes a different aspect of relevant pedagogical information. I will start by presenting Merrill's Instructional Transaction Theory, a very general instructional design theory that describes what good teaching materials should look like. Successfully applying an instructional design theory requires to make explicit the intended learning outcome of a course. Possible learning outcomes are discussed in Section 2.2. Equally important for effective teaching are individual characteristics of the learner. They are covered in Section 2.3.

One word on explaining my choices of pedagogical theories. Myriads of pedagogical theories exist. Obviously, I can not claim completeness, but I tried to focus on those theories that subsume and integrate most of the work done in the specific area.

2.1 Merrill's Instructional Transaction Theory

Instructional design theories describe how to design teaching materials that are effective (how well is learned), efficient (effectivity divided by time taken), and appealing with respect to the learning goal (see [11]). They do not explain how a learner learns (the topic of learning theories), but what the teaching materials should be organized and should look like. In the following, I will describe an instructional (meta) theory that subsumes several instructional design theories.

In [9], Merrill analyzes several approaches to instructional design ([14], [1], [7], [4], [10], [5], [15], and [13], most of them compiled in [11]) and lays out the common underlying principles. He identifies five stages that need to be present in an instructional interaction so that successful learning can arise:

Problem "Learning is facilitated when learners are engaged in solving real-world problems." Students are more engaged if in the beginning of the course or section, they can see what task they will be able to solve after completing the course. Preferably, the task is a concrete, authentic, real world problem, instead of just an enumeration of objectives. The problem should neither be to easy nor to complex. If no adequate problem is found, a difficult problem can be presented, with the necessary support. Comparing successive problems helps the student the tune their mental model.

Activation "Learning is facilitated when relevant previous experience is activated." The activities in this stage should not consist of mere reading of previously seen items, but of an active re-activation of previously learned information. The learner should recall, describe, or apply the relevant knowledge.

Demonstration "Learning is facilitated when the instruction demonstrates what is to be learned rather than merely telling information about what is to be learned." Showing the students what to learn (examples) is better for learning than telling them. The demonstrated knowledge should be in accordance to the intended outcome of learning.

Application "Learning is facilitated when learners are required to use their new knowledge or skill to solve problems." The given exercises should be selected with respect to the learning goals. Feedback, error diagnosis and correction, as well as (fading) support are necessary.

Integration "Learning is facilitated when learners are encouraged to integrate (transfer) the new knowledge or skill into their everyday life." Essential for learning is motivation. Learners are most motivated if they can demonstrate their gained skill for themselves or in public. They should have the possibility to reflect, discuss, and defend their new knowledge. The learned materials should be made part of the student's life.

Optimally, in a course, all five stages take place. However, Merrill's Instructional Transaction Theory is a very general theory, and not specifically targeted at E-Learning purposes. It describes an optimal instructional interaction, whereas in existing E-Learning systems, the extent to which the different stages of the theory can be realized depend on technical constraints. For instance, not all systems provide the technical means necessary for collaborative learning. Still, even if applied only partly, the theory can serve as a guideline for the creation and composition of learning materials.

2.2 Learning Outcomes

Designing a successful course requires to be precise about the intended goal of the learning. A course that teaches and assesses learning by heart of basic definitions in algebra looks different than a course that enables the learner to determine whether a given structure is a group. These different results of learning are called *learning outcomes*. Learning outcomes specify what the learner has learned regarding her cognitive abilities with respect to the content.

A common classification used to describe learning outcomes is Bloom's taxonomy of educational objectives [2]. The following list (adapted from [3]) provides an overview on a subset of the different types of learning.

Knowledge Recall of data. Examples: Recite a policy. Quote prices from memory to a customer. Knows the safety rules. Key Words: defines, describes, identifies, knows, labels, lists, matches, names, outlines, recalls, recognizes, reproduces, selects, states.

Comprehension Understand the meaning, translation, interpolation, and interpretation of instructions and problems. State a problem in one's own words. Examples: Rewrites the principles of test writing. Explain in one's own words the steps for performing a complex task. Translates an equation into a computer spreadsheet. Key words: comprehends, converts, defends, distinguishes, estimates, explains, extends, generalizes, gives examples, infers, interprets, paraphrases, predicts, rewrites, summarizes, translates.

Application Use a concept in a new situation or unprompted use of an abstraction. Applies what was learned in the classroom into novel situations in the workplace. Examples: Use a manual to calculate an employee's vacation time. Apply laws of statistics to evaluate the reliability of a written test. Key Words: applies, changes, computes, constructs, demonstrates, discovers, manipulates, modifies, operates, predicts, prepares, produces, relates, shows, solves, uses.

2.3 Individual Differences

As each individual is different, the same content or learning experience will lead to different learning achievement for each learner. The generation of learning materials for a specific student should reflect her unique personality. I will subsume the causes that directly depend on the user and that influence the effect of learning materials under the term of *individual differences*.

I distinguish two kinds of individual differences: those that form a basic, inborn traits of the learner (her cognitive style) and those that describe external circumstances (such as her field of study).

2.3.1 Cognitive Styles

The *cognitive style* of a learner is reflected in the way she manages a learning task. It is considered one of the most stable user characteristics that influences a person's general achievement in learning situations. A multitude of models exists that describe models of cognitive style. In [12], Riding and Rayner provide a unifying model of research and literature produced in the field of cognitive styles since the forties. Their model distinguishes between a wholist-analytic and a verbaliser-imager dimension:

Wholist-Analytic: "... whether an individual tends to organise information into wholes or parts."

Verbaliser-Imager: "... whether an individual is inclined to represent information during thinking verbally or in mental pictures."

The cognitive style of a learner has effect on both, selection of content and performance. If given free choice, a learner typically chooses those learning materials that best reflects her cognitive style, even if she is not aware of it. Additionally, on subsequent testing, a learner shows best performance, when the learning materials were in accordance with her cognitive style.

However, one should not over-generalize these findings. More often than not, the subjects of the studies were children between the age of 11 and 16, tested on easy task such as reading and recall. Whether the cognitive style still shows a significant effect, when applied to real-world problem solving, remains an open question. Additionally, in general, those learning materials that consisted of structured, not too large units preferably containing an illustrating picture, led to the best performance for all types of learners.

2.3.2 Additional Factors

Individuals differ in less fundamental aspects than cognitive style. Other circumstances, such as the field of study or a learner or her actual state of knowledge effect the learning. As Merrill states, presenting learning materials that have a relation to their daily life, helps students to learn by increasing the motivation and to retain the learned knowledge by allowing a better integration.

Applied to an university setting, the presented learning materials should be related to the learner's subject of study and correspond to his educational level. Furthermore, the content and its difficulty should correspond to the learner's mastery of the content. It should neither be too difficult, nor to easy. The former has a de-motivating, the latter a boring effect.

3 Pedagogical Knowledge in `ActiveMath`

As already mentioned, in this report I will focus on adaptivity with respect to content selection and composition. In `ActiveMath`, pedagogical knowledge about adaptivity is represented in rules. These rules act on the content that was retrieved from the database in a first step. They select from and compose the learning materials to form a course. In this section, I will explain how the pedagogical theories I described are captured in the rules. The rules determine the adaption to the learner along the dimensions *scenario, content goals, user preferences,* and *user mastery of concepts.*

3.1 Scenarios

By choosing a scenario, the learner can explicitly state her intended learning outcome. `ActiveMath` offers the following scenarios:

The *overview* generates a course that provides the student with an overview on the chosen concepts.

Four scenarios encompass *guided tours* with different emphasis based on Bloom's taxonomy of the cognitive domain: *knowledge* generates a course that enables the student to recall/describe/name the chosen concepts; *comprehension* generates a course that enables the student to explain/identify/grasp the chosen concepts); and *application* that generates a course that enables the student to apply/use the chosen concepts. The fourth scenario, in principle the union of the above scenarios, teaches the student about the chosen concepts without focusing on a cognitive domain. These scenarios use the `ActiveMath` extension *competence-level* of the `OMDoc` [6] metadata. Using this metadata, an author can encode whether the learning outcome of an element mainly targets knowledge, comprehension, or application.

Two additional scenarios support the student in active problem solving by selecting *only exercises*, or exercises and the corresponding concepts (scenario *exam preparation*).

All the above scenarios select unknown and well-know learning material (albeit the latter is covered less extensively), and prefer unknown examples and exercises over already seen ones. This behavior changes in two additional scenarios: The *rehearsal* scenario chooses items that the learner has already seen; the *terse* scenario, similar to a complete guided tour, removes all well mastered content.

A last scenario, the *Polya-style* proof presentation, targets the presentation of proofs in a supportive manner. For more details, see [8].

The scenarios respect Merrill's Instructional Transaction Theory by providing instances of each stage, if possible. They first arise the learner's interest by presenting an introduction and/or motivation to the concept, then present the learning material itself (definitions and assertions), and eventually support integration by presenting exercises and examples, and elaborations that help to put it into context.

3.2 Content Goals

Equally important as adaptation of the generated courses to the intended learning outcomes is the adaption to the intended goals of the user with respect to the content. For instance, Anton needs to rehearse basic knowledge about semi groups, whereas Eva is interested in cyclic groups. Instead of presenting the same course to both learners, an intelligent E-Learning system offers the capability of presenting only the content the user wants and needs to learn.

`ActiveMath` realizes this behavior by explicitly allowing the learner to chose the concepts he wants to learn about. `ActiveMath` then automatically selects the necessary prerequisites. By this mean, no unnecessary content is provided, thereby stimulating student motivation as well as contributing to the efficency of a course.

3.3 User Preferences

In its current state, the student model of `ActiveMath` has no representation of the cognitive style of a learner. As I argued in Section 2.3.1, the cognitive style of a learning material has no dramatic effect on learning performance, given the learning materials meet certain standards, (being structured and not too long). Fortunately, `OMDoc` directly imposes these criteria as most `OMDoc` elements represent content with about a single paragraph length. Therefore, I argue that cognitive styles can be neglected within `ActiveMath` without loosing the benefit of adaptivity.

However, `ActiveMath`'s user model represents several other properties of the user that are important with respect to individual differences. *Field* describes the domain a learner studies, *educational level* the level of education a user has reached (for instance primary education or university first cycle), and a third property covers the *external systems* a learner is capable to use.

These values are reflected in `ActiveMath`'s extensions of the `OMDoc` metadata *field, learning context*, and *system*. The scenarios provide support of these individual differences by preferring examples and exercises that correspond to the user's field, learning context, and known systems.

3.4 Student Mastery Level

For each concept, `ActiveMath`'s student model stores mastery level values that represent how well a concept is known by the student. A mastery value for a concept is a triple *(knowledge, comprehension, application)* with each position corresponding to the Bloom objective of the same name.

The mastery values directly influence the amount and difficulty of the presented exercises and examples. The rules distinguish between un-mastered, medium mastered, and well mastered concepts. For un-mastered concepts, the course generation selects a large amount of easy, some fair, and a small amount of difficult exercises and examples. The selection is shifted accordingly, if a concept is medium or well know. For the concrete number of exercises, please see Appendix A.

4 Conclusion

In this report, I layed out the pedagogical basis of ActiveMath's current course generation. The pedagogical rules reflect requirements that arise from an instructional design theory; by providing learning materials for each of the different stages necessary for successful learning as far as possible within ActiveMath. Furthermore, the rules take into account relevant individual differences, the prior knowledge of the learner, her intended learning outcome, as well as her goal concepts.

Future work will target the evaluation of the effects of ActiveMath's adaptivity, as well as enhancing adaptivity by providing a redesign of the course data structure that contains additional information about interactions with the content and that allows for refinement during the instructional interaction.

References

[1] T. Andre. Selected microinstructional methods to facilitate knowledge construction: implications for instructional design. In R. D. Tennyson, F. Schott, N. Seel, and S. Dijkstra, editors, *Instructional Design: International Perspective: Theory, Research, and Models*, volume I, pages 243–267. Lawrence Erlbaum Associates, Mahwah, NJ, 1997.

[2] B.S. Bloom, editor. *Taxonomy of educational objectives: The classification of educational goals: Handbook I, cognitive domain.* Longmans, Green, New York, Toronto, 1956.

[3] D. Clark. ISD manual - analysis, design, develop, implement, evaluate (ADDIE) - a complete guide to training. Available online: `http://www.nwlink.com/~donclark/hrd/sat.html`, 1999.

[4] H. Gardner. Multiple approaches to understanding. In C. M. Reigeluth, editor, *Instructional Design Theories and Models: A New Paradigm of Instructional Theory*, volume II, pages 69–89. Lawrence Erlbaum Associates, 1999.

[5] D. Jonassen. Designing constructivist learning environments. In C. M. Reigeluth, editor, *Instructional Design Theories and Models: A New Paradigm of Instructional Theory*, volume II, pages 215–239. Lawrence Erlbaum Associates, 1999.

[6] M. Kohlhase. OMDoc: Towards an OPENMATH representation of mathematical documents. Seki Report SR-00-02, Fachbereich Informatik, Universität des Saarlandes, 2000. See also `http://www.mathweb.org/omdoc`.

[7] B. McCarthy. *About Learning.* Excell Inc., Barrington, IL, 1996.

[8] E. Melis and C. Ullrich. Polya-scenarios in ActiveMath. In *AI in Education, AIED-2003.* IOS Press, 2003. accepted.

[9] M. D. Merrill. First principles of instruction. *Educational Technology Research & Development*, 2001. submitted.

[10] L. M. Nelson. Collaborative problem solving. In C. M. Reigeluth, editor, *Instructional Design Theories and Models: A New Paradigm of Instructional Theory*, volume II, pages 241–267. Lawrence Erlbaum Associates, 1999.

[11] C. M. Reigeluth. *Instructional Design Theories and Models: A New Paradigm of Instructional Theory*, volume II. Lawrence Erlbaum Associates, Mahwah, NJ, 1999.

[12] R. Riding and S. Rayner. *Cognitive Styles and Learning Strategies.* David Fulton Publishers, London, 1998.

[13] R. C. Schank, T. R. Berman, and K. A. Macperson. Learning by doing. In C. M. Reigeluth, editor, *Instructional Design Theories and Models: A New Paradigm of Instructional Theory*, volume II, pages 161–181. Lawrence Erlbaum Associates, 1999.

[14] D. L. Schwartz, X. Lin, S. Brophy, and J. D. Bransford. Toward the development of flexibly adaptive instructional desings. In C. M. Reigeluth, editor, *Instructional Design Theories and Models: A New Paradigm of Instructional Theory*, volume II, pages 183–213. Lawrence Erlbaum Associates, 1999.

[15] J. J. G. van Merriënboer. *Training Complex Cognitive Skills.* Educational Technology Publications, Inc., Englewood Cliffs, 1997.

A The Pedagogical Rules of `ActiveMath`

The appendix contains the concrete pedagogical rules. The effect of a rule is straightforward:

```
(defrule name_of_the_rule
  if_part
  =>
  then_part)
```

If `if_part` evaluates to true, then `then_part` is executed. Expressions starting with a question-mark denotate variables, `(userValuesKnowledge ?uvk&:(<= ?uvk 0.3))` is a short-form for a boolean test that evaluates to true if `?uvk` is smaller or equal to 0.3.

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;; Scenario detailled Guided Tour
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

(defrule PageRequestForDetailedGuidedTourForLowKnowledge
  (or (scenario DetailedGuidedTour) (scenario Essential))
  (Concept (MBaseObject ?C)
           (userValuesKnowledge ?uvk&:(<= ?uvk 0.3)))
  =>
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 2)
                    (item_type motivation)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 4)
                    (item_type introduction)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 6)
```

```
                    (item_type already_selected)
                    (SelectedMBaseItem (create$ (call ?C getMBaseID)))))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 10)
                    (item_type general)))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 12)
                    (item_type proof)))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 14)
                    (item_type genericconcept)))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 18)
                    (item_type example)
                    (difficulty 0.3) (competenceLevel "knowledge")))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 20)
                    (item_type example)
                    (difficulty 0.3) (competenceLevel "comprehension")))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 22)
                    (item_type example)
                    (difficulty 0.5) (competenceLevel "knowledge")))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 24)
                    (item_type example)
                    (difficulty 0.5) (competenceLevel "knowledge")))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 26)
                    (item_type example)
                    (difficulty 0.7) (competenceLevel "knowledge")))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 28)
                    (item_type example)
                    (difficulty 0.3) (competenceLevel "comprehension")))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 30)
                    (item_type example)
                    (difficulty 0.3) (competenceLevel "comprehension")))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 32)
                    (item_type example)
                    (difficulty 0.5) (competenceLevel "application")))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 34)
                    (item_type example)
                    (difficulty 0.5) (competenceLevel "comprehension")))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 36)
                    (item_type example)
                    (difficulty 0.7) (competenceLevel "comprehension")))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 38)
                    (item_type example)
                    (difficulty 0.7) (competenceLevel "comprehension")))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 40)
                    (item_type example)
                    (difficulty 0.3) (competenceLevel "application")))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 41)
                    (item_type example)
                    (difficulty 0.3) (competenceLevel "application")))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 42)
                    (item_type example)
                    (difficulty 0.5) (competenceLevel "application")))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 44)
                    (item_type example)
                    (difficulty 0.5) (competenceLevel "application")))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 46)
                    (item_type example)
                    (difficulty 0.7) (competenceLevel "application")))
```

```
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 47)
                  (item_type example)
                  (difficulty 0.7) (competenceLevel "application")))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 48)
                  (item_type exercise)
                  (difficulty 0.3) (competenceLevel "knowledge")))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 50)
                  (item_type exercise)
                  (difficulty 0.3) (competenceLevel "comprehension")))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 52)
                  (item_type exercise)
                  (difficulty 0.5) (competenceLevel "knowledge")))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 54)
                  (item_type exercise)
                  (difficulty 0.5) (competenceLevel "knowledge")))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 56)
                  (item_type exercise)
                  (difficulty 0.7) (competenceLevel "knowledge")))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 58)
                  (item_type exercise)
                  (difficulty 0.3) (competenceLevel "comprehension")))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 60)
                  (item_type exercise)
                  (difficulty 0.3) (competenceLevel "comprehension")))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 62)
                  (item_type exercise)
                  (difficulty 0.5) (competenceLevel "application")))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 64)
                  (item_type exercise)
                  (difficulty 0.5) (competenceLevel "comprehension")))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 66)
                  (item_type exercise)
                  (difficulty 0.7) (competenceLevel "comprehension")))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 68)
                  (item_type exercise)
                  (difficulty 0.3) (competenceLevel "application")))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 70)
                  (item_type exercise)
                  (difficulty 0.3) (competenceLevel "application")))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 72)
                  (item_type exercise)
                  (difficulty 0.5) (competenceLevel "application")))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 74)
                  (item_type exercise)
                  (difficulty 0.5) (competenceLevel "application")))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 76)
                  (item_type exercise)
                  (difficulty 0.7) (competenceLevel "application")))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 77)
                  (item_type exercise)
                  (difficulty 0.7) (competenceLevel "application")))
(assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 78)
                  (item_type elaboration)))
)
```

```
(defrule PageRequestForDetailedGuidedTourForMediumKnowledge
  (or (scenario DetailedGuidedTour) (scenario Essential))
  (Concept (MBaseObject ?C)
           (userValuesKnowledge ?uvk&:(> ?uvk 0.3)&:(< ?uvk 0.8)))
  =>
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 2)
                    (item_type motivation)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 4)
                    (item_type introduction)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 6)
                    (item_type already_selected)
                    (SelectedMBaseItem (create$ (call ?C getMBaseID)))))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 10)
                    (item_type general)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 12)
                    (item_type proof)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 14)
                    (item_type genericconcept)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 22)
                    (item_type example)
                    (difficulty 0.5) (competenceLevel "knowledge")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 24)
                    (item_type example)
                    (difficulty 0.5) (competenceLevel "knowledge")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 26)
                    (item_type example)
                    (difficulty 0.7) (competenceLevel "knowledge")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 30)
                    (item_type example)
                    (difficulty 0.3) (competenceLevel "comprehension")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 32)
                    (item_type example)
                    (difficulty 0.5) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 34)
                    (item_type example)
                    (difficulty 0.5) (competenceLevel "comprehension")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 36)
                    (item_type example)
                    (difficulty 0.7) (competenceLevel "comprehension")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 40)
                    (item_type example)
                    (difficulty 0.3) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 42)
                    (item_type example)
                    (difficulty 0.5) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 44)
                    (item_type example)
                    (difficulty 0.5) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 46)
                    (item_type example)
                    (difficulty 0.7) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 47)
                    (item_type example)
                    (difficulty 0.7) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 48)
```

```
                    (item_type exercise)
                    (difficulty 0.3) (competenceLevel "knowledge")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 54)
                    (item_type exercise)
                    (difficulty 0.5) (competenceLevel "knowledge")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 56)
                    (item_type exercise)
                    (difficulty 0.7) (competenceLevel "knowledge")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 58)
                    (item_type exercise)
                    (difficulty 0.3) (competenceLevel "comprehension")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 62)
                    (item_type exercise)
                    (difficulty 0.5) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 64)
                    (item_type exercise)
                    (difficulty 0.5) (competenceLevel "comprehension")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 66)
                    (item_type exercise)
                    (difficulty 0.7) (competenceLevel "comprehension")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 70)
                    (item_type exercise)
                    (difficulty 0.3) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 72)
                    (item_type exercise)
                    (difficulty 0.5) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 74)
                    (item_type exercise)
                    (difficulty 0.5) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 76)
                    (item_type exercise)
                    (difficulty 0.7) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 77)
                    (item_type exercise)
                    (difficulty 0.7) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 86)
                    (item_type elaboration)))
  )

(defrule PageRequestForDetailedGuidedTourForHighKnowledge
  (or (scenario DetailedGuidedTour) (scenario Essential))
  (Concept (MBaseObject ?C)
           (userValuesKnowledge ?uvk&:(>= ?uvk 0.8)))
  =>
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 2)
                    (item_type motivation)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 4)
                    (item_type introduction)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 6)
                    (item_type already_selected)
                    (SelectedMBaseItem (create$ (call ?C getMBaseID)))))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 10)
                    (item_type general)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 12)
                    (item_type proof)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 14)
```

```
                     (item_type genericconcept)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 26)
                     (item_type example)
                     (difficulty 0.7) (competenceLevel "knowledge")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 32)
                     (item_type example)
                     (difficulty 0.5) (competenceLevel "comprehension")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 36)
                     (item_type example)
                     (difficulty 0.7) (competenceLevel "comprehension")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 42)
                     (item_type example)
                     (difficulty 0.5) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 44)
                     (item_type example)
                     (difficulty 0.5) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 46)
                     (item_type example)
                     (difficulty 0.7) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 47)
                     (item_type example)
                     (difficulty 0.7) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 56)
                     (item_type exercise)
                     (difficulty 0.7) (competenceLevel "knowledge")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 64)
                     (item_type exercise)
                     (difficulty 0.5) (competenceLevel "comprehension")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 66)
                     (item_type exercise)
                     (difficulty 0.7) (competenceLevel "comprehension")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 72)
                     (item_type exercise)
                     (difficulty 0.5) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 74)
                     (item_type exercise)
                     (difficulty 0.5) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 76)
                     (item_type exercise)
                     (difficulty 0.7) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 77)
                     (item_type exercise)
                     (difficulty 0.7) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 86)
                     (item_type elaboration)))
  )


;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;; Scenario detailled Guided Tour Knowledge
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

(defrule PageRequestForDetailedGuidedTourKForLowKnowledge
  (scenario DetailedGuidedTourK)
  (Concept (MBaseObject ?C)
           (userValuesKnowledge ?uvk&:(<= ?uvk 0.3)))
```

```
  =>
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 2)
                    (item_type motivation)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 4)
                    (item_type introduction)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 6)
                    (item_type already_selected)
                    (SelectedMBaseItem (create$ (call ?C getMBaseID)))))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 18)
                    (item_type example)
                    (difficulty 0.3) (competenceLevel "knowledge")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 20)
                    (item_type example)
                    (difficulty 0.3) (competenceLevel "knowledge")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 22)
                    (item_type example)
                    (difficulty 0.5) (competenceLevel "knowledge")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 24)
                    (item_type example)
                    (difficulty 0.5) (competenceLevel "knowledge")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 26)
                    (item_type example)
                    (difficulty 0.7) (competenceLevel "knowledge")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 28)
                    (item_type exercise)
                    (difficulty 0.3) (competenceLevel "knowledge")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 30)
                    (item_type exercise)
                    (difficulty 0.3) (competenceLevel "knowledge")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 32)
                    (item_type exercise)
                    (difficulty 0.5) (competenceLevel "knowledge")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 34)
                    (item_type exercise)
                    (difficulty 0.5) (competenceLevel "knowledge")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 36)
                    (item_type exercise)
                    (difficulty 0.7) (competenceLevel "knowledge")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 38)
                    (item_type elaboration)))
  )

(defrule PageRequestForDetailedGuidedTourKForMediumKnowledge
  (scenario DetailedGuidedTourK)
  (Concept (MBaseObject ?C)
           (userValuesKnowledge ?uvk&:(> ?uvk 0.3)&:(< ?uvk 0.8)))
  =>
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 2)
                    (item_type motivation)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 4)
                    (item_type introduction)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 6)
                    (item_type already_selected)
                    (SelectedMBaseItem (create$ (call ?C getMBaseID)))))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 18)
                    (item_type example)
```

```
                    (difficulty 0.3) (competenceLevel "knowledge")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 19)
                    (item_type example)
                    (difficulty 0.5) (competenceLevel "knowledge")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 20)
                    (item_type example)
                    (difficulty 0.5) (competenceLevel "knowledge")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 22)
                    (item_type example)
                    (difficulty 0.7) (competenceLevel "knowledge")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 24)
                    (item_type exercise)
                    (difficulty 0.3) (competenceLevel "knowledge")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 26)
                    (item_type exercise)
                    (difficulty 0.5) (competenceLevel "knowledge")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 27)
                    (item_type exercise)
                    (difficulty 0.5) (competenceLevel "knowledge")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 28)
                    (item_type exercise)
                    (difficulty 0.7) (competenceLevel "knowledge")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 30)
                    (item_type elaboration)))
  )

(defrule PageRequestForDetailedGuidedTourKForHighKnowledge
  (scenario DetailedGuidedTourK)
  (Concept (MBaseObject ?C)
           (userValuesKnowledge ?uvk&:(>= ?uvk 0.8)))
  =>
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 2)
                    (item_type motivation)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 4)
                    (item_type introduction)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 6)
                    (item_type already_selected)
                    (SelectedMBaseItem (create$ (call ?C getMBaseID)))))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 18)
                    (item_type example)
                    (difficulty 0.7) (competenceLevel "knowledge")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 20)
                    (item_type example)
                    (difficulty 0.7) (competenceLevel "knowledge")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 22)
                    (item_type exercise)
                    (difficulty 0.7) (competenceLevel "knowledge")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 24)
                    (item_type exercise)
                    (difficulty 0.7) (competenceLevel "knowledge")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 26)
                    (item_type elaboration)))
  )
```

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;; Scenario detailled Guided Tour Comprehension
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

(defrule PageRequestForDetailedGuidedTourCForLowComprehension
  (scenario DetailedGuidedTourC)
  (Concept (MBaseObject ?C)
           (userValuesComprehension ?uvk&:(<= ?uvk 0.3)))
  =>
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 2)
                    (item_type motivation)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 4)
                    (item_type introduction)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 6)
                    (item_type already_selected)
                    (SelectedMBaseItem (create$ (call ?C getMBaseID)))))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 18)
                    (item_type example)
                    (difficulty 0.3) (competenceLevel "comprehension")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 20)
                    (item_type example)
                    (difficulty 0.3) (competenceLevel "comprehension")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 22)
                    (item_type example)
                    (difficulty 0.5) (competenceLevel "comprehension")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 24)
                    (item_type example)
                    (difficulty 0.5) (competenceLevel "comprehension")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 26)
                    (item_type example)
                    (difficulty 0.7) (competenceLevel "comprehension")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 28)
                    (item_type exercise)
                    (difficulty 0.3) (competenceLevel "comprehension")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 30)
                    (item_type exercise)
                    (difficulty 0.3) (competenceLevel "comprehension")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 32)
                    (item_type exercise)
                    (difficulty 0.5) (competenceLevel "comprehension")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 34)
                    (item_type exercise)
                    (difficulty 0.5) (competenceLevel "comprehension")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 36)
                    (item_type exercise)
                    (difficulty 0.7) (competenceLevel "comprehension")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 38)
                    (item_type elaboration)))
  )

(defrule PageRequestForDetailedGuidedTourCForMediumComprehension
  (scenario DetailedGuidedTourC)
  (Concept (MBaseObject ?C)
           (userValuesComprehension ?uvk&:(> ?uvk 0.3)&:(< ?uvk 0.8)))
  =>
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 2)
```

```
                      (item_type motivation)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 4)
                      (item_type introduction)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 6)
                      (item_type already_selected)
                      (SelectedMBaseItem (create$ (call ?C getMBaseID)))))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 18)
                      (item_type example)
                      (difficulty 0.3) (competenceLevel "comprehension")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 19)
                      (item_type example)
                      (difficulty 0.5) (competenceLevel "comprehension")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 20)
                      (item_type example)
                      (difficulty 0.5) (competenceLevel "comprehension")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 22)
                      (item_type example)
                      (difficulty 0.7) (competenceLevel "comprehension")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 24)
                      (item_type exercise)
                      (difficulty 0.3) (competenceLevel "comprehension")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 26)
                      (item_type exercise)
                      (difficulty 0.5) (competenceLevel "comprehension")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 27)
                      (item_type exercise)
                      (difficulty 0.5) (competenceLevel "comprehension")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 28)
                      (item_type exercise)
                      (difficulty 0.7) (competenceLevel "comprehension")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 30)
                      (item_type elaboration)))
  )

(defrule PageRequestForDetailedGuidedTourCForHighComprehension
  (scenario DetailedGuidedTourC)
  (Concept (MBaseObject ?C)
           (userValuesComprehension ?uvk&:(>= ?uvk 0.8)))
  =>
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 2)
                      (item_type motivation)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 4)
                      (item_type introduction)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 6)
                      (item_type already_selected)
                      (SelectedMBaseItem (create$ (call ?C getMBaseID)))))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 18)
                      (item_type example)
                      (difficulty 0.7) (competenceLevel "comprehension")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 20)
                      (item_type example)
                      (difficulty 0.7) (competenceLevel "comprehension")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 22)
                      (item_type exercise)
                      (difficulty 0.7) (competenceLevel "comprehension")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 24)
```

```
                    (item_type exercise)
                    (difficulty 0.7) (competenceLevel "comprehension")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 26)
                    (item_type elaboration)))
  )

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;; Scenario detailled Guided Tour Application
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

(defrule PageRequestForDetailedGuidedTourAForLowApplication
  (scenario DetailedGuidedTourA)
  (Concept (MBaseObject ?C)
           (userValuesApplication ?uvk&:(<= ?uvk 0.3)))
  =>
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 2)
                    (item_type motivation)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 4)
                    (item_type introduction)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 6)
                    (item_type already_selected)
                    (SelectedMBaseItem (create$ (call ?C getMBaseID)))))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 10)
                    (item_type general)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 12)
                    (item_type proof)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 14)
                    (item_type genericconcept)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 16)
                    (item_type method)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 18)
                    (item_type example)
                    (difficulty 0.3) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 20)
                    (item_type example)
                    (difficulty 0.3) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 22)
                    (item_type example)
                    (difficulty 0.5) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 24)
                    (item_type example)
                    (difficulty 0.5) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 26)
                    (item_type example)
                    (difficulty 0.7) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 28)
                    (item_type exercise)
                    (difficulty 0.3) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 30)
                    (item_type exercise)
                    (difficulty 0.3) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 32)
                    (item_type exercise)
                    (difficulty 0.5) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 34)
                    (item_type exercise)
```

```
                     (difficulty 0.5) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 36)
                     (item_type exercise)
                     (difficulty 0.7) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 38)
                     (item_type elaboration)))
  )

(defrule PageRequestForDetailedGuidedTourAForMediumApplication
  (scenario DetailedGuidedTourA)
  (Concept (MBaseObject ?C)
           (userValuesApplication ?uvk&:(> ?uvk 0.3)&:(< ?uvk 0.8)))
  =>
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 2)
                     (item_type motivation)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 4)
                     (item_type introduction)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 6)
                     (item_type already_selected)
                     (SelectedMBaseItem (create$ (call ?C getMBaseID)))))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 10)
                     (item_type general)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 12)
                     (item_type proof)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 14)
                     (item_type genericconcept)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 16)
                     (item_type method)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 18)
                     (item_type example)
                     (difficulty 0.3) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 19)
                     (item_type example)
                     (difficulty 0.5) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 20)
                     (item_type example)
                     (difficulty 0.5) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 22)
                     (item_type example)
                     (difficulty 0.7) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 24)
                     (item_type exercise)
                     (difficulty 0.3) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 26)
                     (item_type exercise)
                     (difficulty 0.5) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 27)
                     (item_type exercise)
                     (difficulty 0.5) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 28)
                     (item_type exercise)
                     (difficulty 0.7) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 30)
                     (item_type elaboration)))
  )
```

```
(defrule PageRequestForDetailedGuidedTourAForHighApplication
  (scenario DetailedGuidedTourA)
  (Concept (MBaseObject ?C)
           (userValuesApplication ?uvk&:(>= ?uvk 0.8)))
  =>
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 2)
                    (item_type motivation)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 4)
                    (item_type introduction)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 6)
                    (item_type already_selected)
                    (SelectedMBaseItem (create$ (call ?C getMBaseID)))))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 10)
                    (item_type general)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 12)
                    (item_type proof)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 14)
                    (item_type genericconcept)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 16)
                    (item_type method)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 18)
                    (item_type example)
                    (difficulty 0.7) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 20)
                    (item_type example)
                    (difficulty 0.7) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 22)
                    (item_type exercise)
                    (difficulty 0.7) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 24)
                    (item_type exercise)
                    (difficulty 0.7) (competenceLevel "application")))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 26)
                    (item_type elaboration)))
  )


;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;; Scenario Exam
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

(defrule PageRequestForExamForLowKnowledge
  (scenario Exam)
  (Concept (MBaseObject ?C)
           (userValuesKnowledge ?uvk&:(<= ?uvk 0.3)))
  =>
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 12)
                    (item_type exercise) (difficulty 0.3)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 14)
                    (item_type exercise) (difficulty 0.3)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 16)
                    (item_type exercise) (difficulty 0.3)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 17)
                    (item_type exercise) (difficulty 0.3)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 18)
                    (item_type exercise) (difficulty 0.5)))
```

```
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 20)
                    (item_type exercise) (difficulty 0.5)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 21)
                    (item_type exercise) (difficulty 0.5)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 22)
                    (item_type exercise) (difficulty 0.5)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 24)
                    (item_type exercise) (difficulty 0.7)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 26)
                    (item_type exercise) (difficulty 0.7)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 28)
                    (item_type exercise) (difficulty 0.7)))
  )

(defrule PageRequestForExamForMediumKnowledge
  (scenario Exam)
  (Concept (MBaseObject ?C)
           (userValuesKnowledge ?uvk&:(> ?uvk 0.3)&:(< ?uvk 0.8)))
  =>
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 16)
                    (item_type exercise) (difficulty 0.3)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 17)
                    (item_type exercise) (difficulty 0.3)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 18)
                    (item_type exercise) (difficulty 0.5)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 20)
                    (item_type exercise) (difficulty 0.5)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 21)
                    (item_type exercise) (difficulty 0.5)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 22)
                    (item_type exercise) (difficulty 0.5)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 23)
                    (item_type exercise) (difficulty 0.5)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 24)
                    (item_type exercise) (difficulty 0.7)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 26)
                    (item_type exercise) (difficulty 0.7)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 28)
                    (item_type exercise) (difficulty 0.7)))
  )

(defrule PageRequestForExamForHighKnowledge
  (scenario Exam)
  (Concept (MBaseObject ?C)
           (userValuesKnowledge ?uvk&:(>= ?uvk 0.8)))
  =>
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 16)
                    (item_type exercise) (difficulty 0.3)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 17)
                    (item_type exercise) (difficulty 0.3)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 18)
                    (item_type exercise) (difficulty 0.5)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 20)
                    (item_type exercise) (difficulty 0.5)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 23)
                    (item_type exercise) (difficulty 0.5)))
```

```
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 24)
                    (item_type exercise) (difficulty 0.7)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 26)
                    (item_type exercise) (difficulty 0.7)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 28)
                    (item_type exercise) (difficulty 0.7)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 30)
                    (item_type exercise) (difficulty 0.7)))
  )

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;; Scenario ExamPrep
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

(defrule PageRequestForExamPrepForLowKnowledge
  (scenario ExamPrep)
  (Concept (MBaseObject ?C)
           (userValuesKnowledge ?uvk&:(<= ?uvk 0.3)))
  =>
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 2)
                    (item_type already_selected)
                    (SelectedMBaseItem (create$ (call ?C getMBaseID)))))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 6)
                    (item_type proof)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 8)
                    (item_type genericconcept)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 10)
                    (item_type method)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 12)
                    (item_type exercise) (difficulty 0.3)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 14)
                    (item_type exercise) (difficulty 0.3)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 16)
                    (item_type exercise) (difficulty 0.3)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 17)
                    (item_type exercise) (difficulty 0.3)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 18)
                    (item_type exercise) (difficulty 0.5)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 20)
                    (item_type exercise) (difficulty 0.5)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 21)
                    (item_type exercise) (difficulty 0.5)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 22)
                    (item_type exercise) (difficulty 0.5)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 24)
                    (item_type exercise) (difficulty 0.7)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 26)
                    (item_type exercise) (difficulty 0.7)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 28)
                    (item_type exercise) (difficulty 0.7)))
  )

(defrule PageRequestForExamPrepForMediumKnowledge
  (scenario ExamPrep)
  (Concept (MBaseObject ?C)
           (userValuesKnowledge ?uvk&:(> ?uvk 0.3)&:(< ?uvk 0.8)))
```

```
  =>
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 2)
                    (item_type already_selected)
                    (SelectedMBaseItem (create$ (call ?C getMBaseID)))))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 10)
                    (item_type general)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 12)
                    (item_type proof)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 14)
                    (item_type genericconcept)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 16)
                    (item_type method)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 16)
                    (item_type exercise) (difficulty 0.3)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 17)
                    (item_type exercise) (difficulty 0.3)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 18)
                    (item_type exercise) (difficulty 0.5)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 20)
                    (item_type exercise) (difficulty 0.5)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 21)
                    (item_type exercise) (difficulty 0.5)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 22)
                    (item_type exercise) (difficulty 0.5)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 23)
                    (item_type exercise) (difficulty 0.5)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 24)
                    (item_type exercise) (difficulty 0.7)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 26)
                    (item_type exercise) (difficulty 0.7)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 28)
                    (item_type exercise) (difficulty 0.7)))
  )

(defrule PageRequestForExamPrepForHighKnowledge
  (scenario ExamPrep)
  (Concept (MBaseObject ?C)
           (userValuesKnowledge ?uvk&:(>= ?uvk 0.8)))
  =>
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 6)
                    (item_type already_selected)
                    (SelectedMBaseItem (create$ (call ?C getMBaseID)))))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 12)
                    (item_type proof)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 14)
                    (item_type genericconcept)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 16)
                    (item_type method)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 16)
                    (item_type exercise) (difficulty 0.3)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 17)
                    (item_type exercise) (difficulty 0.3)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 18)
                    (item_type exercise) (difficulty 0.5)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 20)
                    (item_type exercise) (difficulty 0.5)))
```

```
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 23)
                    (item_type exercise) (difficulty 0.5)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 24)
                    (item_type exercise) (difficulty 0.7)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 26)
                    (item_type exercise) (difficulty 0.7)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 28)
                    (item_type exercise) (difficulty 0.7)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 30)
                    (item_type exercise) (difficulty 0.7)))
  )

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;; Scenario Overview
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;


(defrule PageRequestForOverviewForLowKnowledge
  (scenario Overview)
  (Concept (MBaseObject ?C)
           (userValuesKnowledge ?uvk&:(<= ?uvk 0.3)))
  =>
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 2)
                    (item_type motivation)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 6)
                    (item_type already_selected)
                    (SelectedMBaseItem (create$ (call ?C getMBaseID)))))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 14)
                    (item_type genericconcept)))
  )

(defrule PageRequestForOverviewMediumKnowledge
  (scenario Overview)
  (Concept (MBaseObject ?C)
           (userValuesKnowledge ?uvk&:(> ?uvk 0.3)&:(< ?uvk 0.8)))
  =>
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 6)
                    (item_type already_selected)
                    (SelectedMBaseItem (create$ (call ?C getMBaseID)))))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 14)
                    (item_type genericconcept)))
  )

(defrule PageRequestForOverviewHighKnowledge
  (scenario Overview)
  (Concept (MBaseObject ?C)
           (userValuesKnowledge ?uvk&:(>= ?uvk 0.8)))
  =>
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 6)
                    (item_type already_selected)
                    (SelectedMBaseItem (create$ (call ?C getMBaseID)))))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 14)
                    (item_type genericconcept)))
  )
```

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;; Scenario Rehearsal


;; in a rehearsal we choose examples and exercises we have already seen

(defrule WantAlreadySeenItems
  (scenario Rehearsal)
  ?fact <- (Prefer (alreadySeen nil))
  =>
  (modify ?fact (alreadySeen TRUE)))

(defrule PageRequestForRehearsalForLowKnowledge
  (scenario Rehearsal)
  (Concept (MBaseObject ?C)
           (userValuesKnowledge ?uvk&:(<= ?uvk 0.3)))
  =>
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 6)
                    (item_type already_selected)
                    (SelectedMBaseItem (create$ (call ?C getMBaseID)))))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 12)
                    (item_type proof)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 14)
                    (item_type genericconcept)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 16)
                    (item_type method)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 18)
                    (item_type example)
                    (difficulty 0.3) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 10)
                    (item_type example)
                    (difficulty 0.3) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 12)
                    (item_type example)
                    (difficulty 0.3) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 14)
                    (item_type example)
                    (difficulty 0.5) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 16)
                    (item_type example)
                    (difficulty 0.5) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 18)
                    (item_type example)
                    (difficulty 0.5) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 20)
                    (item_type example)
                    (difficulty 0.7) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 22)
                    (item_type example)
                    (difficulty 0.7) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 24)
                    (item_type exercise)
                    (difficulty 0.3) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 25)
                    (item_type exercise)
```

```
                     (difficulty 0.3) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 26)
                     (item_type exercise)
                     (difficulty 0.3) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 28)
                     (item_type exercise)
                     (difficulty 0.5) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 30)
                     (item_type exercise)
                     (difficulty 0.5) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 32)
                     (item_type exercise)
                     (difficulty 0.5) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 34)
                     (item_type exercise)
                     (difficulty 0.7) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 36)
                     (item_type exercise)
                     (difficulty 0.7) (alreadySeen TRUE)))
  )

(defrule PageRequestForRehearsalMediumKnowledge
  (scenario Rehearsal)
  (Concept (MBaseObject ?C)
           (userValuesKnowledge ?uvk&:(> ?uvk 0.3)&:(< ?uvk 0.8)))
  =>
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 6)
                     (item_type already_selected)
                     (SelectedMBaseItem (create$ (call ?C getMBaseID)))))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 12)
                     (item_type proof)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 14)
                     (item_type genericconcept)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 15)
                     (item_type method)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 17)
                     (item_type example)
                     (difficulty 0.3) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 18)
                     (item_type example)
                     (difficulty 0.5) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 20)
                     (item_type example)
                     (difficulty 0.5) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 22)
                     (item_type example)
                     (difficulty 0.5) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 24)
                     (item_type example)
                     (difficulty 0.7) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 26)
                     (item_type example)
                     (difficulty 0.7) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 28)
                     (item_type example)
                     (difficulty 0.7) (alreadySeen TRUE)))
```

```
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 37)
                    (item_type exercise)
                    (difficulty 0.3) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 38)
                    (item_type exercise)
                    (difficulty 0.5) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 40)
                    (item_type exercise)
                    (difficulty 0.5) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 42)
                    (item_type exercise)
                    (difficulty 0.5) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 44)
                    (item_type exercise)
                    (difficulty 0.7) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 46)
                    (item_type exercise)
                    (difficulty 0.7) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 48)
                    (item_type exercise)
                    (difficulty 0.7) (alreadySeen TRUE)))
  )

(defrule PageRequestForRehearsalHighKnowledge
  (scenario Rehearsal)
  (Concept (MBaseObject ?C)
           (userValuesKnowledge ?uvk&:(>= ?uvk 0.8)))
  =>
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 6)
                    (item_type already_selected)
                    (SelectedMBaseItem (create$ (call ?C getMBaseID)))))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 12)
                    (item_type proof)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 14)
                    (item_type genericconcept)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 16)
                    (item_type method)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 20)
                    (item_type example)
                    (difficulty 0.3) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 22)
                    (item_type example)
                    (difficulty 0.5) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 24)
                    (item_type example)
                    (difficulty 0.7) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 26)
                    (item_type example)
                    (difficulty 0.7) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 28)
                    (item_type example)
                    (difficulty 0.7) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 30)
                    (item_type example)
                    (difficulty 0.7) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 31)
```

```
                    (item_type exercise)
                    (difficulty 0.3) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 32)
                    (item_type exercise)
                    (difficulty 0.5) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 34)
                    (item_type exercise)
                    (difficulty 0.7) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 36)
                    (item_type exercise)
                    (difficulty 0.7) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 38)
                    (item_type exercise)
                    (difficulty 0.7) (alreadySeen TRUE)))
  (assert (PageItem (forWhat ?C) (forConceptPage ?C) (PageNumber 1) (PositionOnPage 40)
                    (item_type exercise)
                    (difficulty 0.7) (alreadySeen TRUE)))
)
```