

Mastery Grids

Personalized Adaptive Web Systems Lab, School of Information Sciences, University of Pittsburgh
Learning+Technology research group, Aalto University, 2015

Mastery Grids is a visually-rich, interactive, adaptive social E-learning portal that provides access to multiple kinds of smart learning content. Its most critical feature is open social learner modeling integrated with adaptive navigation support. Mastery Grids is supported by adaptive social learning framework Aggregate, which supports several kinds of open learner models, social comparison, and recommendation. Our past research shows that Mastery Grids effectively increases learners' performance, motivation, engagement and retention.



Figure 1: Mastery Grids' knowledge and progress visualization of Me, Me vs Group, Group, other learners embedding open learner model of oneself and social comparisons of a student with the group or other learners.

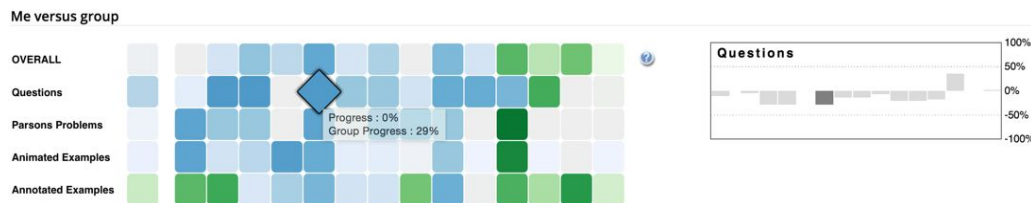


Figure 2: Mastery Grids' social comparison by learning content (resource) type.

❑ Open Learner Model

Mastery Grids combines open learner model with social comparison empowering open social learner model. It enables a learner to be aware of his/her own strength and weakness, and stimulates a learner to work harder. Also, it encourages learners to catch up with others and to follow the good learning paths of the peers.

❑ Knowledge and Progress Visualization

Mastery Grids presents and compares student learning progress and knowledge by colored grids from four perspectives as shown in Figure 1: "Me" (my progress grid), "Me vs group" (comparison grid), "Group" (group grid), and "Students in the class" (performance of all of the learners in the class).

❑ Social Comparison

Social comparison is implemented by "Me vs Group" and "Me vs other learners". Such comparisons can be further conducted in fine-grained level by learning content (resource) type as shown in Figure 2.

Multi-Type Smart Learning Contents

Mastery Grids provides multiple types of interactive, smart learning contents from three domains (Java, SQL, and Python), which has been used for undergraduate and graduate level programming and database classes since 2013. Courses are organized by topics with timelines aligning with learning contents (Figure 1). Learning contents contain semantics problems, parson problems, annotated examples, and animated examples.

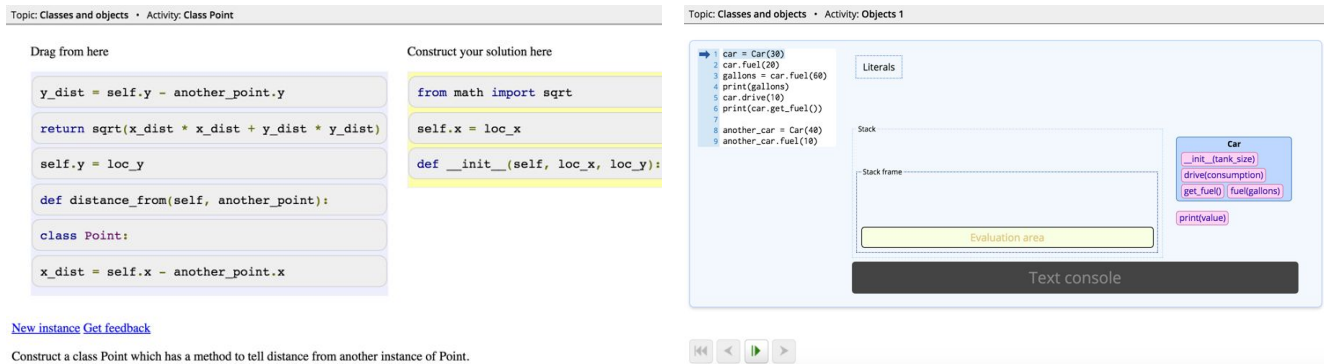


Figure 3: The left side shows a parson problem for Python. In such problems, students are asked to drag different fragments of a code to construct a complete code in order to achieve a task. The right side shows an animated examples for Python. In such examples, the line-by-line execution of a piece of code is visualized by animations.

Adaptive Navigation Support

Mastery Grids provides adaptive navigation support across different types of learning contents and different topics (knowledge modules). It tracks user activities and updates learner knowledge in a centralized user modeling server, based on which recommendation is provided. Recommendation is applied with reactive and/or proactive approach(es). The reactive approach activates in reaction to a student's activity (Figure 1 with red stars), while the pro-active approach proactively suggests learning materials (Figure 1 and Figure 4).

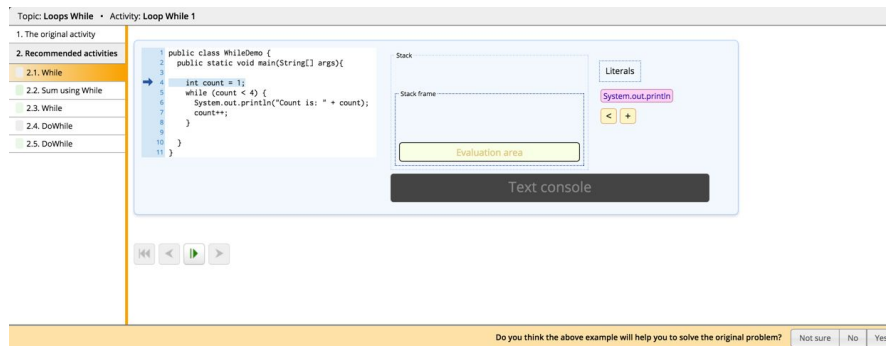


Figure 4: Recommendation of examples after a learner's failure in a Java problem.

Authoring Tools

We have developed four major types of authoring tools for the project: 1) Content Authoring for creating different kinds of smart learning contents, 2) Course Authoring for creating adaptive courses that use the content, 3) Group Authoring for managing users and groups, as well as 4) Authoring Portal to access different authoring tools.

Resources and Documentations

Our project website with detailed information can be accessed here [bit.ly/OSLM_MG](http://adapt2.sis.pitt.edu/wiki/Adaptive_Navigation_Support_and_Open_Social_Learner_Modeling_for_PAL) which is a shortcut for http://adapt2.sis.pitt.edu/wiki/Adaptive_Navigation_Support_and_Open_Social_Learner_Modeling_for_PAL. All related code can be downloaded from Github: <https://github.com/PAWSLabUniversityOfPittsburgh>, <https://github.com/acos-server/>.