



# {APFEL}

## Adaptive Programming-Feedback for E-Learning

Dominic Lohr

### System Design



ALEA is an **adaptive learning assistant** that can provide customized content for learners based on **semantically annotated course materials** [1].

Learning objects are annotated following the **Y-model framework** [2], making relevant details available to the system (e.g., prerequisite knowledge, relevant concepts, learning objectives, answer classes).

ALEA incorporates a **learner model** capable of offering insights into the concepts learners grasped within a specific cognitive dimension.

[alea.education](http://alea.education)

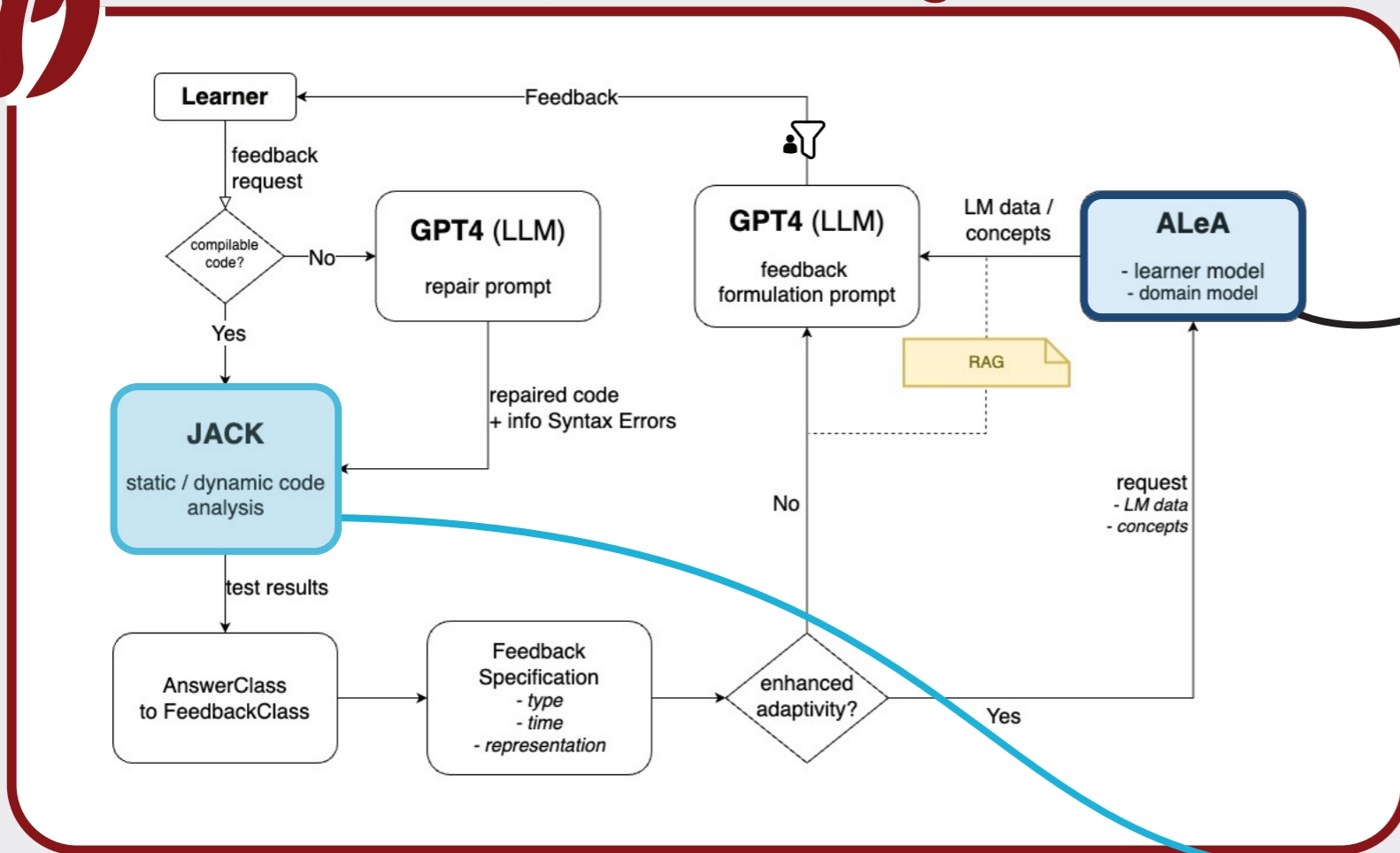


JACK is an **e-assessment system** that includes functionality for grading programming exercises [3].

Thanks to its modular architecture, we can make use of its capabilities for **static and dynamic analysis**.

Based on the results of these analyses, we map the learner's program code to **answer classes** following the Y-model framework [2].

[jack-community.org](http://jack-community.org)



#### RUN

RUN offers the basic functionality of an IDE. The program code can be executed to display **console output** or **compiler messages** [KM-CE]\*.

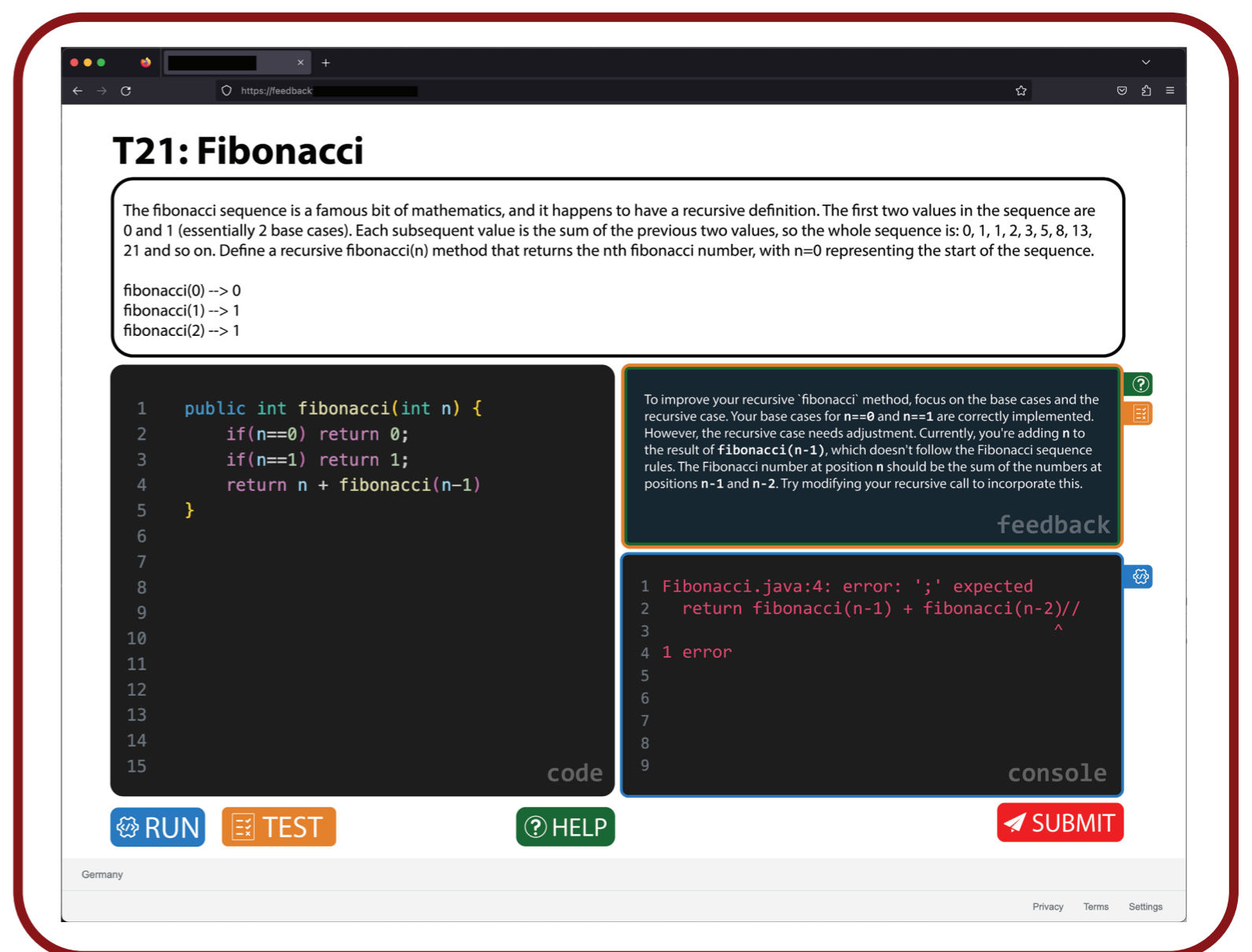
#### TEST

TEST enables detailed feedback on the current **state of the answer**, including results of the static and dynamic analysis [KM-TF]\*, as well as descriptions of the answer classes assigned to the current answer [KTC, KM-SE]\*.

Using LLM-based code repair, this is also possible if the code contains syntax errors.

#### HELP

HELP enables APFEL's **adaptive functionality**. It generates LLM output by leveraging a comprehensive context, including analysis results, answer classes, learner model insights, and additional pertinent metadata about the task and the course. This information is dynamically integrated into prompts (via RAG) to generate varied types of feedback [KC, KH]\*. [4]



### Functionality

[1] Kruse et al. 2023. Learning with ALEA: Tailored experiences through annotated course material. *INFORMATIK23*.

[2] Lohr et al. 2023. The Y-Model - Formalization of Computer Science Tasks in the Context of Adaptive Learning Systems. *GeCon23*.

[3] Striewe. 2016. An Architecture for Modular Grading and Feedback Generation for Complex Exercises. *Science of Computer Programming*.

[4] Lohr, Keuning, Kiesler. 2024. You're (Not) My Type - Can LLMs Generate Feedback of Specific Types for Introductory Programming Tasks? *JCAL24 (accepted)*

[ ] \* feedback types by Keuning et al. 2018