Specification: Human reading assessment system

The system is a web platform where can students log in to take personalised reading tests, and teachers can log in to check results from tests.

This specification is for the UI, web server and database components, and integration with pre-developed machine learning modules for document recommendation and text evaluation (training, model selection etc. are not included in this specification).

On logging in, students are given a stream of reading tests followed by questions. The student

1. is presented with the headings/introductions of a choice of two or more texts, given by a recommender module
2. picks one of the texts – or chooses to stop
3. is presented with the full text, and can start reading
4. is given a questionnaire to assess understanding
    - this can include various multiple-choice and optionally free-form questions
5. is given a questionnaire on how well they liked the text

See https://www.uis.no/nb/lesesenteret/human for a mock-up which gives an idea of how the above process could look.

Answers per student for the two types of questions (understanding of the text versus preference for the text) are stored in tables in a database, and are used as parameters to the document recommender module which suggests new texts for the next reading assignment.

Answers are checked against a table of correct answers, except the free-form questions which are scored by a machine learning module. Scores per user per text are stored in a table.

Each text in the corpus has a set of understanding-questions, which are rated for difficulty, so each text can be assigned a *difficulty score* from the difficulty of its questions. Each student can also be assigned a *reading level* by calculating the IRT theta score from the set of previous (correct vs incorrect) answers to the understanding questions (weighted by question difficulty). The document recommender calculates theta, and should only suggest texts which are not too difficult or too easy compared to the student's level (e.g. if the student's IRT theta is 0.5, we may want to include only texts which are rated for difficulty 0.5 ± 0.1).

For each text the student has read, the system stores their answers to the preference questionnaire. On asking for new document recommendations, these answers (along with the difficulty range) are given to the similarity search module, which looks up a set of similar documents from the corpus.

Teachers can log in to the system to see each student's progress, including their development over time, including graphs such as

- correct answers per text
- correct answers per difficulty

Possible architecture sketch: