Course analysis for dbtech23

Version 2024-05-27_01

Author

Johan Jansson (jjan@kth.se), Course coordinator

Description of the course evaluation process

Earlier course rounds have indicated a very poor number of responses to course surveys, and no interactive discussion with the students, such that no meaningful information could be extracted. This year a course survey was carried out by student program representatives - with generally very positive feedback. An anonymous survey form was also available which generated one very positive review, where the student even wanted to use my pedagogical approach and the motivation-based design of the course as material for a project course. The course coordinator Johan Jansson was also available for all students and held a continuous discussion with the students about course goals, course design, etc., gathering feedback throughout the course. Several students emailed feedback during the course, also generally positive. Finally an engaged and productive feedback meeting was held with the student representatives. Overall this approach generated a broad set of feedback, and since the feedback was collected continuously, the course could also be adapted in real-time in response to the needs of the students.

Description of meetings with students

The course coordinator Johan Jansson was available for all students in lecture halls at least once per week, and also for booking online meetings. A course assistants was similarly available in the final part of the course. There was significant interaction with the students in these sessions, with in-depth discussions with typically 20-30 students per week on topics that they raised or that arose in their projects. An engaged and productive feedback meeting was held with the student representatives after the examination deadlines.

Course design

The course design is a new format based on projects in groups where the students chose to work on something in their own interest, and they submit the source code ("runnable" and "editable" in a reproducible setting such as Jupyter Notebooks is recommended) together with making an oral presentation as examination. A lab part can either be integrated in the project or carried our standalone in a similar way, but without the oral presentation. The purpose of the lab part is to make sure all students come into contact with a set of standard concepts, which may not be included in the project of their own choosing. The students were required to cover the course goals in their project and lab reports, in a constructive alignment setting which also made sure to activate the students' motivation, and make use of peer learning The course design was carried out in the direction of the KTH Future Framework for Education, this is something Johan Jansson has implemented in courses since 2006 at KTH. The course went very well, with very enthusiastic and active students showing high competence. The students were given the opportunity to use and develop with state-of-the-art distributed database Spark.

Students' workload

The students carried out extensive project development, consistent with the course goals.

Students' results on the course

The course went very well, with very enthusiastic and active students showing high competence.

Students' answers to open questions

See "Summary of students' opinions" below.

Summary of students' opinions

The students highlighted that the course had great design with students creating their own projects which gave freeom, great that the course coordinator Johan was available so much. Some students expressed that it was a bit unclear what the definition of a "database" is, and asked if it's possible to give more structured information about this. I think this is a relevant viewpoint. The concept of what a "database" is has changed significantly during the last 50 years. In the 1970s and 1980s programming languages and abstraction were primitive, and there was a need to have separate abstractions and systems to handle data for computer programs. With the advent of third- and fourth-generation programming languages such as Python from the 2000s, the abstractions have reached such a high level that the concept of "database" can be argued is integrated in the language, see e.g. the "Pickle" and "dictionary" abstractions in Python. In this round of the course it was a deliberate choice to encourage the students to form their own viewpoint of the concept of "database".

Overall impression

The course may need a reform of course goals. Modern programming languages have "database" concept build-in, e.g. Python "Pickle", this is enough to learn to grasp the concept and enough for the absolute majority of applications, also "closer to the data structures" used in programming. Also should be modernized - SQL vs. NoSQL, data with computation, etc.

Analysis

Female students were generally positive. The design of choosing a project which the students themselves were motivated to carry out has a high chance of creating motivation, which is likely to overcome traditional imbalances between e.g. gender, background, disability, etc.

Prioritized course development

See “Overall impression” and “Summary of students’ opinions” above.

Other information you want to share