

Code Expert EduHubDays 2022

Dr. David Sichau
Software Engineer
15. February 2022



Overview

1. What is Code Expert
2. Benefits
3. Gamification Elements
4. Hands On
5. Discussion

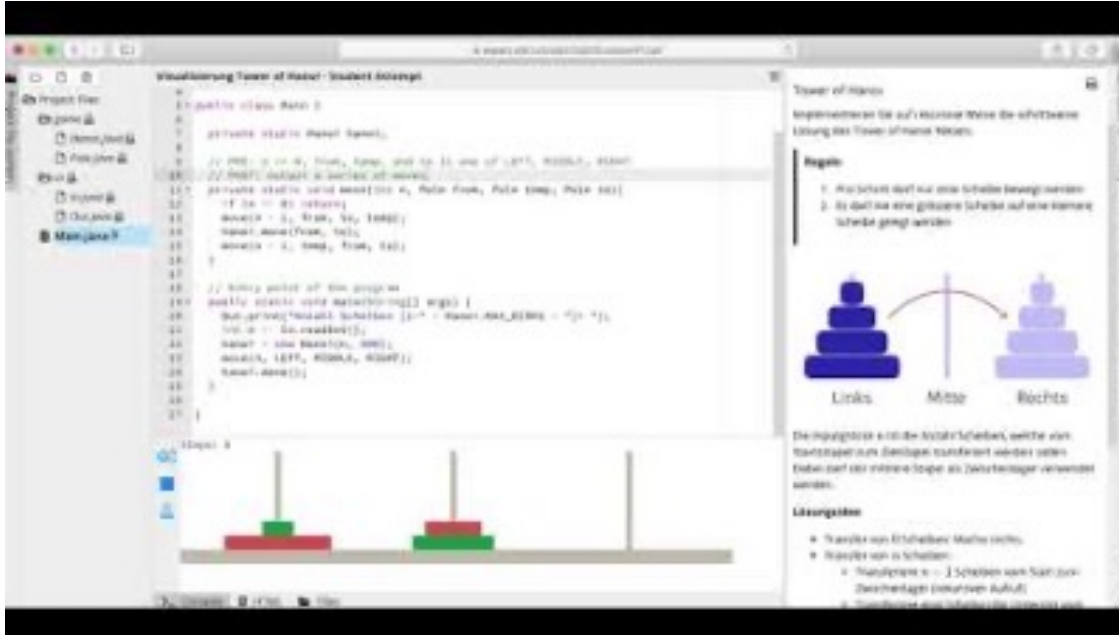
Code Expert

The screenshot displays the Code Expert interface with a filter bar at the top. The filter bar includes 'Filters Show more', 'Sort by date', and status options: 'To do', 'Passed', 'Failed', and 'Feedback'. Below the filter bar, there are four exercise sections, each with a 'Perfect score' indicator and a 'Closed' date. Each exercise section contains a list of tasks with their respective scores and completion status.

Exercise	Task	Score	Status
Exercise 1: The First C++ Program Closed 4 months ago	Task 1: Hello World	10	Passed
	Task 2: Multiply with 29	50	Passed
Exercise 2: Expressions & Integers Closed 4 months ago	Task 1: Expressions	200	Passed
	Task 2: Representation of Integers	100	Passed
	Task 3: Equivalent Resistance	300	Passed
Exercise 3: Boolean expressions & Basic loops Closed 4 months ago	Task 1: Expression Evaluation	100	Passed
	Task 2: From Natural Language to C++	100	Passed
	Task 3: From decimal to binary representation	200	Passed
	Task 4a: Fibonacci primes	200	Passed
	Task 4b: Fibonacci overflow check	200	Passed
	Task 1.5: two-complement integer representation (Optional)	0	To do
Exercise 4: Loops and Floats Closed 3 months ago			

- Code Expert is an online IDE and learning management tool for computer science education
- Developed since 2018 at D-INFK at ETH Zurich
- So far 18'600 users
- 73 courses use Code Expert
- Over 16 million Jobs executed
- Used in high stakes exams
- Developed by a team of 5 developers

Code Expert is used in:



- **Introductory programming classes** for life science and engineering students (2000 students/semester)
- **Advanced programming classes** for Computer Science students (1000 students/semester)
- **23 courses** in current semester
- **1028 programming tasks** (leading to 2.5 Mio code submissions)

Exercise delivery dashboard

The dashboard displays the following exercises:

Exercise	XP
Modul 6: Theorie	10
Modul 6: E.Tutorial Teil 1	10
1) Teil I: Vektoren (Luftdaten)	30
2) Teil II: Matrizen (Boss-Puzzle)	30
Modul 6: E.Tutorial Teil 2	10
1) Case Study I: Wärmeausbreitung in einem Metallstab	30
2) Case Study II: Wärmeausbreitung in einer Platte	30

Exercise	XP
Modul 1: Präsentation	500
Modul 2: Präsentation	500
Modul 3: Präsentation	500
Modul 4: Präsentation	500
Modul 5A und 5B: Präsentation	500
Modul 6: Präsentation	0 of 500

- Students are guided through the course
- **Gamification:** collect XP and unlock Bonus Exercises
- Students always know where they stand and what is next
- Easy to filter and sort the tasks.

Online IDE

The screenshot shows an online IDE interface for a task titled "2) Teil II: Matrizen (Boss-Puzzle) - Student Attempt".

Code Editor: The main area contains Python code for a 3x3 matrix game. The code defines a function `ausgabe(spielbrett)` that prints the matrix and a loop that updates a cell in the matrix.

```
1 import numpy as np
2
3 def ausgabe(spielbrett):
4     print(2)
5     print(" S 1 2 3")
6     print("Z  -----")
7     for i in range(0, 3):
8         print(i + 1, "|", end=" ")
9         for j in range(0, 3):
10            print(spielbrett[i][j], end=" ")
11        print()
12    print()
13
14
15 brett = np.array([[3, 4, 8], [7, 0, 5], [2, 1, 6]], int)
16
17 leer_zeile = 1
18 leer_spalte = 1
19
20 print(1)
21
22 for i in range(10):
23     ausgabe(brett)
24     zeile = int(input("Zeilennummer: ")) - 1
25     spalte = int(input("Spaltennummer: ")) - 1
26     brett[leer_zeile][leer_spalte] = brett[zeile][spalte]
27     brett[zeile][spalte] = 0
28     leer_zeile = zeile
29     leer_spalte = spalte
```

Progress Table: A table showing the number of tests passed, failed, and errors.

COUNT	PASS	FAIL	ERRORS
3	3	0	0

Task Description (Right Panel): The task is titled "Teil II: Matrizen (Boss-Puzzle)". It instructs the user to note commands from a tutorial in the Python file `main.py` and to click the "Run" button to execute the program. It also mentions that test cases will be automatically executed and that the user can view the test results in the HTML tab.

Buttons: There are several buttons: "Create new Submission" (blue), "Test" (blue play button), and "HTML" (blue flask icon).

Online IDE

The screenshot displays an online IDE interface. On the left is a file explorer showing a project structure with files like `main.py`, `test_runner.py`, and `template.html`. The central editor shows Python code for a matrix puzzle solution. Below the code is a test runner window with a table of results:

COUNT	PASS	FAIL	ERRORS
3	2	1	0

The test runner also shows a progress bar and details for each test case, including one that failed: "Teste, ob die Funktion `ausgabe` das Spielfeld ausgibt." On the right, a tutorial panel titled "Teil II: Matrizen (Boss-Puzzle)" provides instructions and a "Publish Solution & Template" button.

- **Gamification:** Instant Feedback generation
- Support many different programming languages
- No setup required

Review

The screenshot shows the Code Expert submission review interface. The top bar indicates the submission is for '2) Teil II: Matrizen (Boss-Puzzle) - Submission Feedback' and is marked as 'A Job is Running'. The interface is divided into several sections:

- Project Files:** A tree view on the left shows the project structure, including files like `cx_data`, `cx_description`, `scripts`, `test_runner`, and `main.py`.
- Code Editor:** The main area displays the Python code for `main.py`. The code defines a function `ausgabe(spielbrett)` and uses NumPy to create a 3x3 grid. It also includes a loop for user input and a function call to `ausgabe(brett)`.
- Console:** The bottom of the code editor shows the output of the program, which is a 3x3 grid of numbers: `S 1 2 3`, `Z -----`, `1 | 3 4 8`, `2 | 7 0 5`, `3 | 2 1 6`.
- Feedback Panel:** On the right, a panel for 'Markus Dahinden' shows the submission feedback. The feedback text reads: 'Sauber programmiert. Die Ausgabe des Spielfelds ist korrekt und gut gelöst. Auch der Einsatz einer Funktion für die Ausgabe macht hier Sinn, da Sie sonst den Code für die Ausgabe in Zeile 24 und 32 duplizieren müssten.' Below the feedback, it shows 'Autograde: 1 / Score: 1 of 1' and a progress bar.

- **Manual, high-quality feedback** by teaching assistants (either written or orally)

Analytics



- **Lecturers** gain an understanding about the learning process of their students with the provided **analytics**
- The analytics allow lecturers to detect problems in their course

Code Expert Benefits for Users

Students

- No setup required
- Immediate Feedback
- Single platform
- High motivation due to gamification elements

Teaching Assistants

- Receive submissions
- Provide Feedback
- Easy Communication
- Analytics

Lecturers

- Single Plattform
- Flexible setup of courses
- Design a motivating learning path
- Very flexible to set up projects and autograding
- Analytics to spot problems

Code Expert Advantages & Limits

Advantages

Runs on own Infrastruktur

Scalable architecture

Language independent

Different didactic models can be implemented and configured

Continuous development

Limits

Focus on computer science education

Not a general-purpose learning management tool

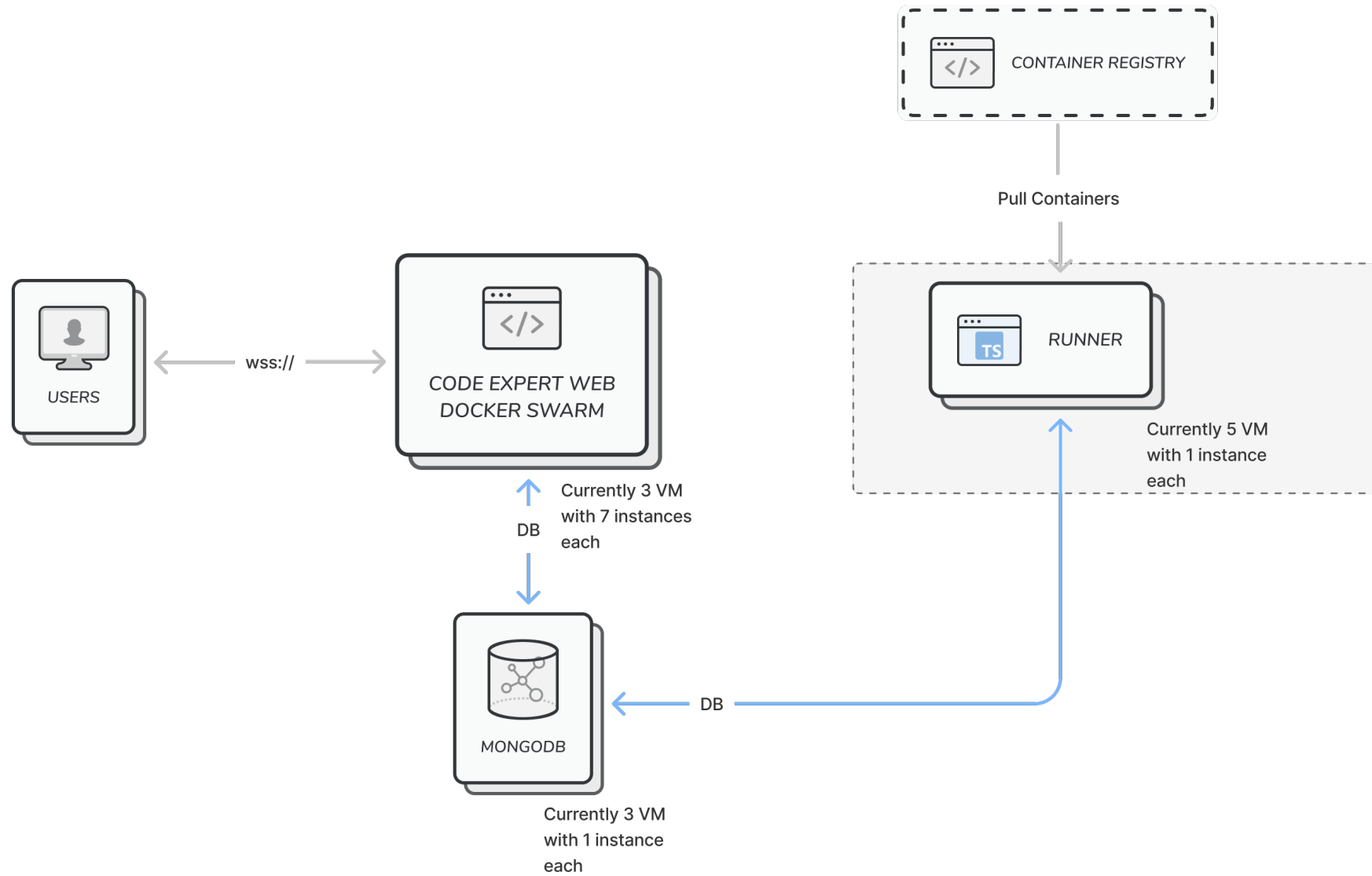
Currently limited question types

Runs on a shared infrastructure no HPC code possible

Not a full-fledged IDE

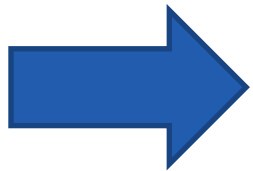
Complex Setup

Tech Slide



Gamification Elements

- We have different gamification elements on different levels, which helps lecturers to increase engagement and motivation
- Students can **compete** against the cohort and can view their position compared to the cohort
- One can **gain experience points** by solving tasks
- **Bonus exercises** that can be only **unlocked** if a certain amount of experience points have been collected



Feedback from our users suggest that the **combination of gamification** elements with **content** helps, that the students stay **motivated** through the whole semester and stay **active**.

Hands-On

Filters Show more ▾ Sort by date ↓↑

To do Passed Failed Feedback

To do

First Steps Due by Wed 16.2.22 23:57

Theory	10 XP	⋮
E.Tutorial	10 XP	⋮
Python as calculator		⋮
Temperatur conversion	20 XP	⋮
Strings and swapping	20 XP	⋮
Circle Math	20 XP	⋮

Bonus exercise

Individual Task Locked (due by Wed 16.2.22 23:58)

0% **Earn XP to unlock bonus material**
To unlock this bonus exercise, you need to earn at least 40 XP in the following exercises:

First Steps	0 of 80 XP
-------------	------------

Bonus exercise

Presentation Locked (due by Wed 16.2.22 23:59)

0% **Earn XP to unlock bonus material**
To unlock this bonus exercise, you need to earn at least 60 XP in the following exercises:

First Steps	0 of 80 XP
Individual Task	Not open yet

- You can not try out Code Expert
- Signup to eduHub course:
<https://expert.ethz.ch/enroll/SS22/eduHub>
- Follow the course ask if you have any questions
- Submit your code and present your individual tasks

Feedback & Discussion

<https://www.figma.com/file/kkkwSOZDy9B81QFPufSPKh/EduHubDays>

Interested

- If you are interested in using Code Expert, please let us know.
- We plan to run some pilot cases with courses outside of the ETH to collect some experience
- Contact us via expert@inf.ethz.ch



Thank you for your
attention

Code Expert Team