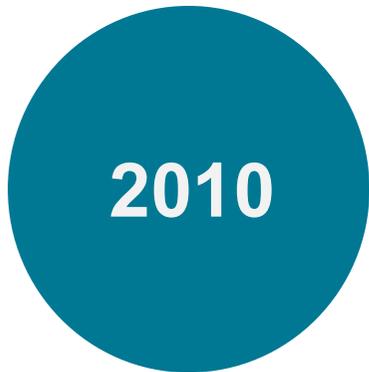


E-Assessment an der ETH Zürich – Ein Praxisbericht

Bild: Alessandro Della Bella

Infrastruktur

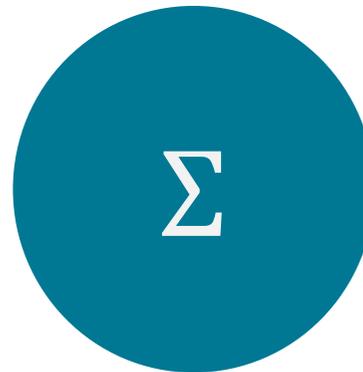
Eckdaten



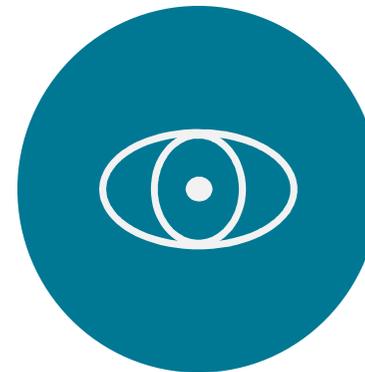
Seit 2010



separate
Instanz



summativ
selektiv



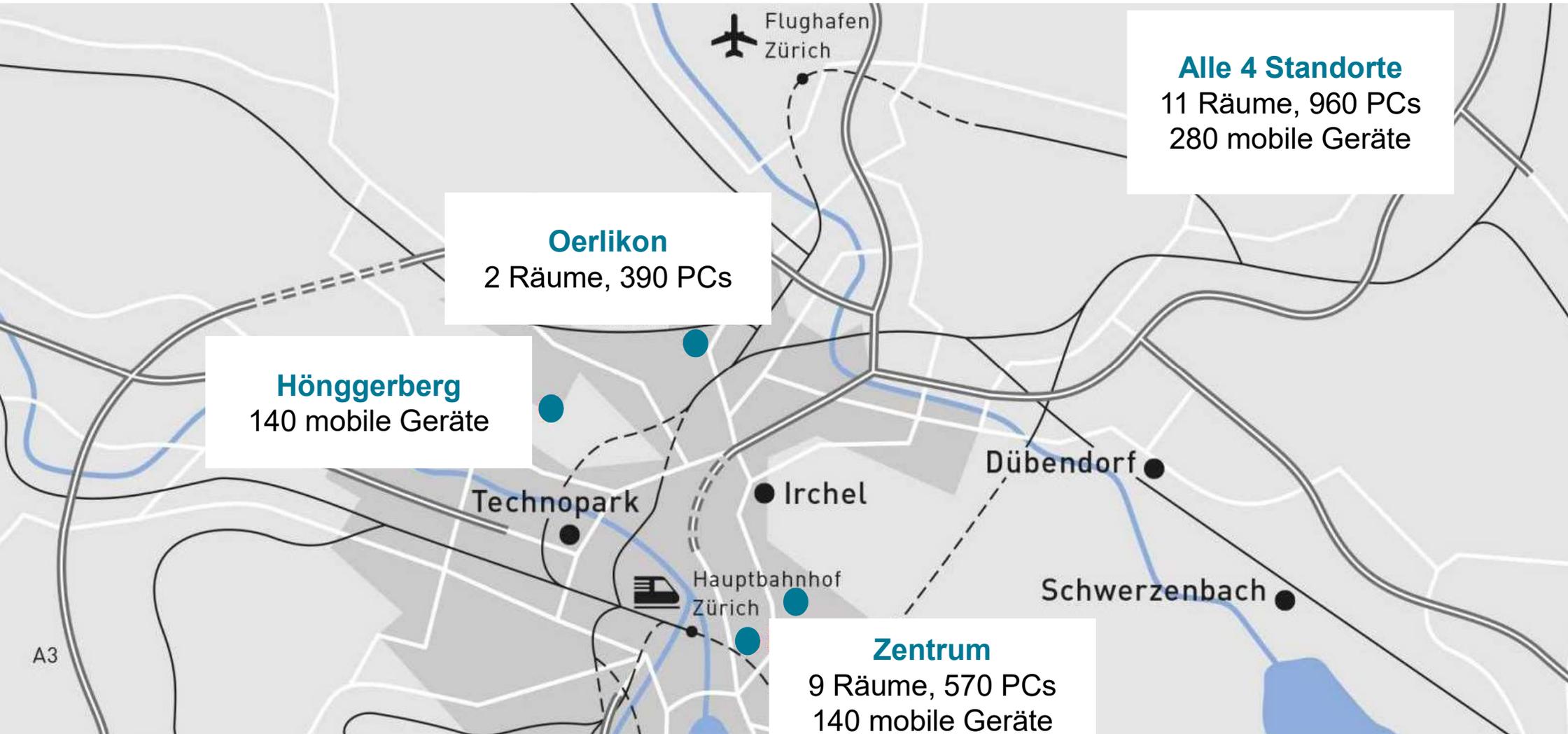
beaufsichtigt



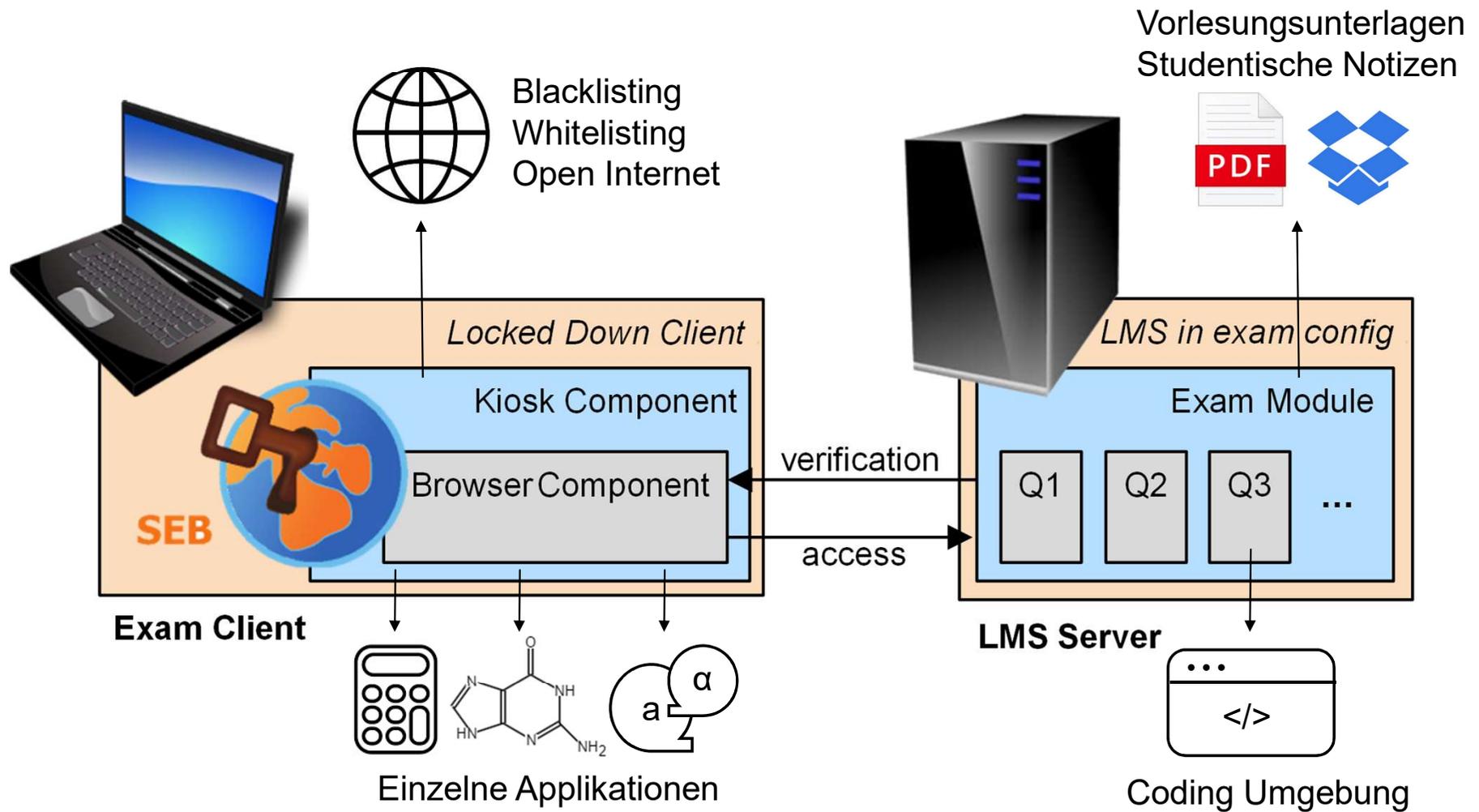
abgesichert

2022: 300 Prüfungen à 25'000 Einzelleistungen

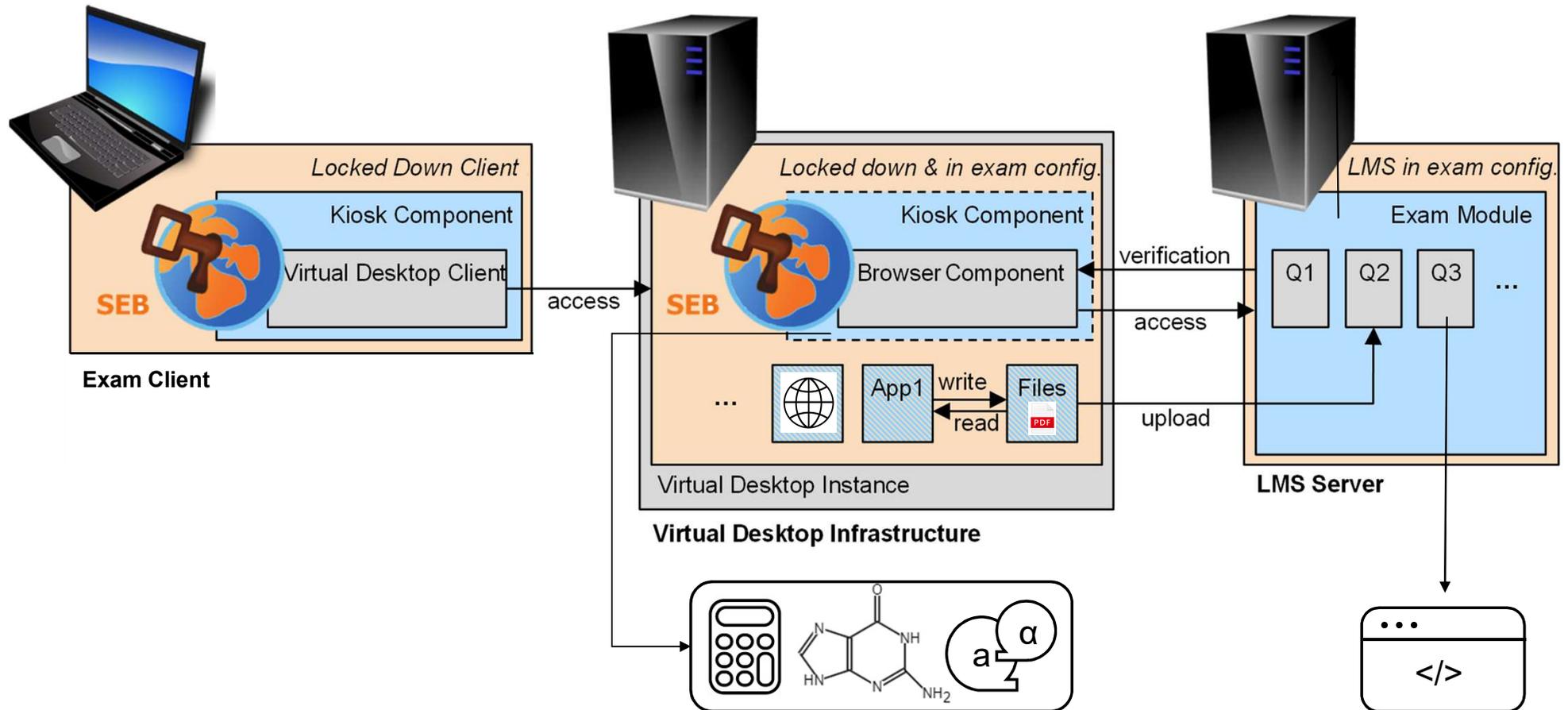
Standorte & Räume



Moodle und Safe Exam Browser: Absicherung via SEB-config-files

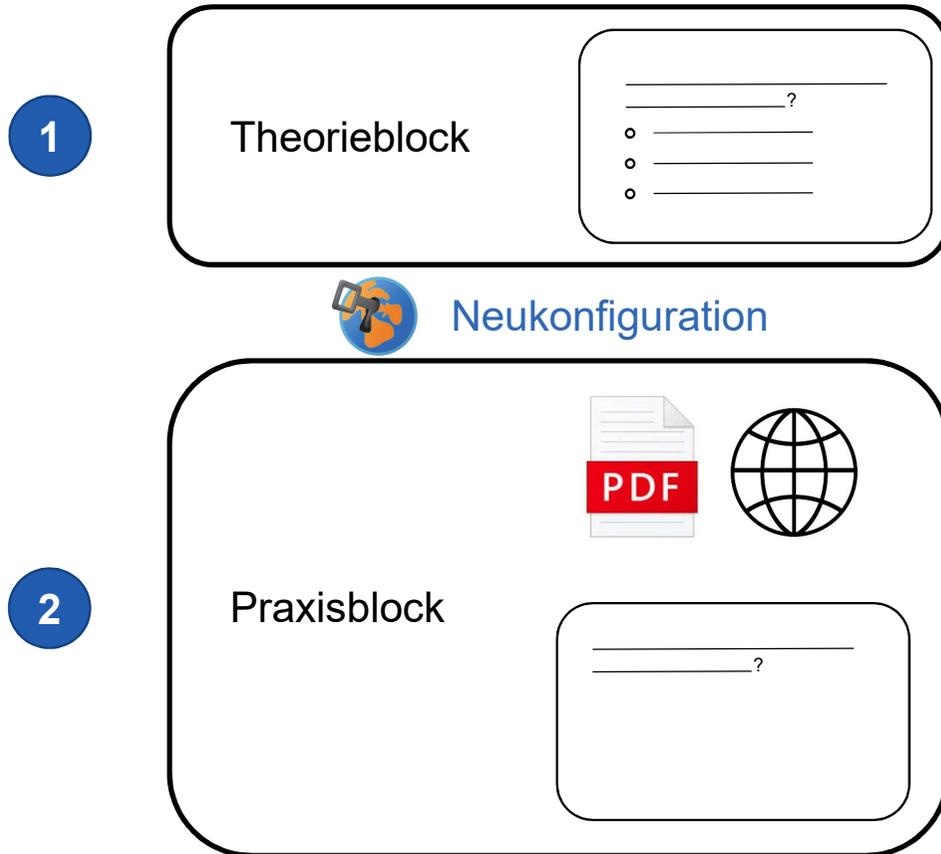


Softwareeinbindung via Virtuelle Desktops: zusätzliche Sicherheitsstufe



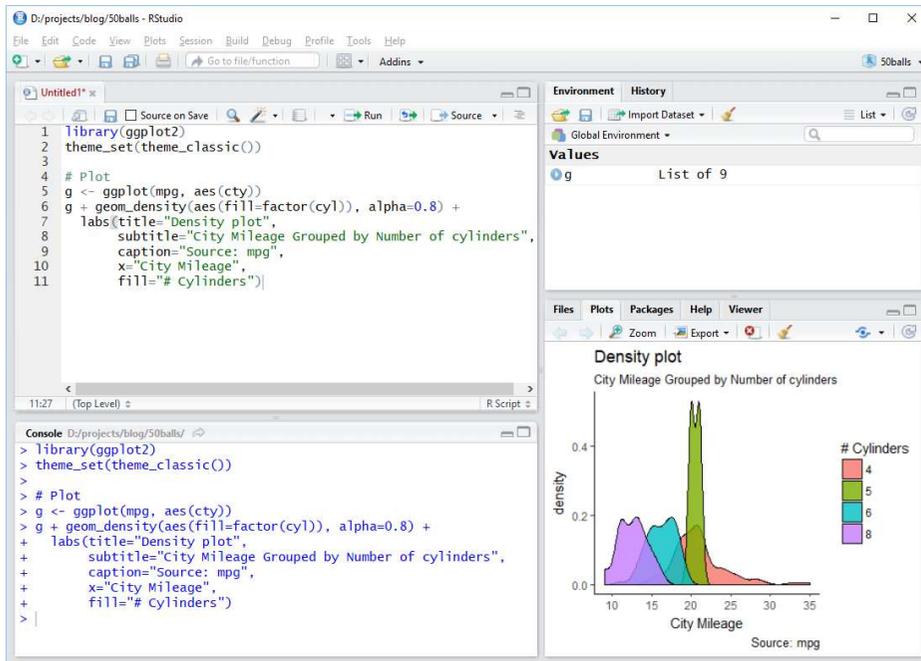
Praxiseinblick: On Campus

2-Phasen Prüfung: Impfeempfehlungen



- 120 Minuten
- 35 Studierende (MSc)
- Tool: Moodle
- Aufgabe:
 - 1 Theoriefragen
 - 2 Impfeempfehlung basierend auf Case Studies
- Aufgabenformat:
 - 1 geschlossen
 - 2 offen

Software & LMS: Datenauswertung mit RStudio



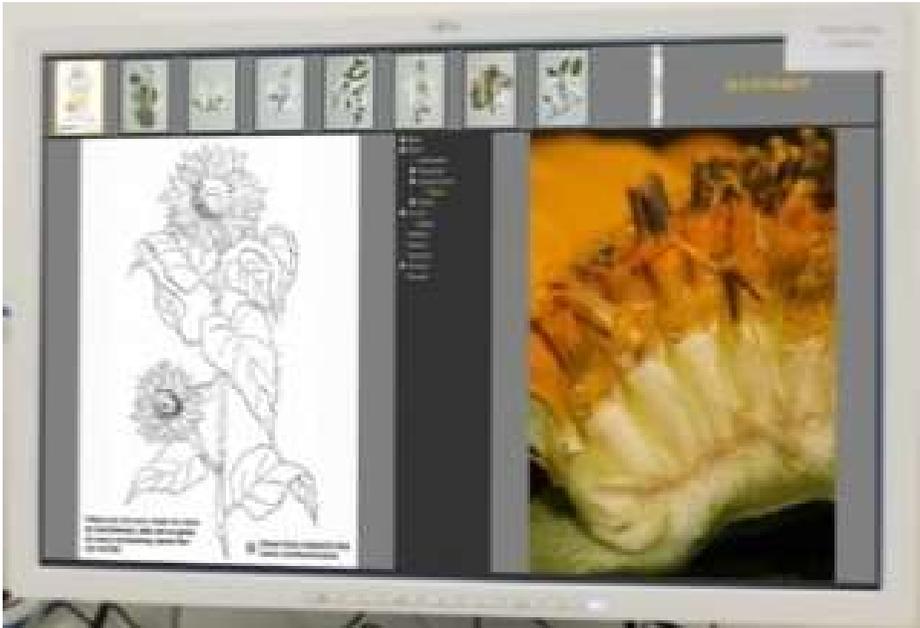
- 180 Minuten
- 350 Studierende (BSc)
- Tool: RStudio
- Auftrag: Datenauswertung und -analyse mit RStudio
- Aufgabenformat: geschlossen

Hardware Anbindung: Diagnose mittels Mikroskop



- 65 Minuten
- 20 Studierende (MSc)
- Tool: Laptops & Mikroskope mit Kamera, Bildbearbeitungssoftware
- Auftrag: Diagnose von Pilzkrankheiten in Pflanzen anhand von Mikroskopaufnahmen
- Aufgabenformat: Dateiabgabe

Simulation: Virtuelles Herbarium



- 90 Minuten
 - 60min Papier
 - 30min Computer
- 100 Studierende (BSc)
- Tool: Fotokatalog
- Aufgabe Computer-Teil: Bestimmung von 20 Pflanzen anhand artspezifischer Merkmale
- Aufgabenformate: offen und geschlossen

Vollwertige Programmierumgebung: Coding unter praxisnahen Bedingungen

ETH zürich

Moodle You are logged in as Hermann Lehner

Question 13
Not yet answered
Marked out of 10.00
Flag question
Edit question

```
Binary Trees - Student Attempt
6  if (root == nullptr) return 0;
7  // leaf node
8  if (root->left == nullptr && root->right == nullptr){
9  // **** TODO ****
10 }
11 // calculate children
12 int left = validate(root->left);
13 int right = validate(root->right);
14
15 // **** TODO ****
16
17 // default
18 return -1;
19 }
```

variable 'left' [-Wunused-variable]
int left = validate(root->left);
/var/lib/cxrun/projectfiles/validate.h:13:6: warning: unused variable 'right' [-Wunused-variable]
int right = validate(root->right);
Compilation successful
Console

gilt. Für T2 gilt die Eigenschaft nicht, da die erwartete Gleichung für den Knoten mit Schlüssel 3 nicht gilt: $3 \neq 2 + 0$.

T1:

```
graph TD
  3((3)) --- 2((2))
  3 --- 1((1))
```

T2:

```
graph TD
  5((5)) --- 3((3))
  5 --- 2((2))
  3 --- 2((2))
```

Aufgabe: Vervollständigen Sie die Implementierung der Funktion `validate`, so dass die Funktion genau dann -1 zurückgibt, wenn der übergebene Baum die Bedingung nicht erfüllt. Andernfalls kann eine beliebige Zahl (ungleich -1) zurückgegeben werden. Die Lösung muss rekursiv sein.

Beachten Sie, dass bereits einige Testfälle erfüllt werden. Punkte

- 2 x 300 Minuten
- 160 Studierende (MSc)
- Tool: CodeExpert mit CodeJudge, Debugger und Live-Kommentarfunktion
- Aufgabe: Lauffähige Algorithmen basierend auf passenden Modellen für 6 praxisnahe, in Prosa beschriebene Probleme
- Aufgabenformat: Dateiabgabe

Online-Datenbanken und Open Internet: Heiz-, Kühl-, Lüftungssysteme

energieschweiz

Suche Menü

1 Standort / Technologie

Heerenschürlistrasse 8 8051 Zürich

Solarwärme
 Photovoltaik

Manuelle Auslegung

2 Bewohner im Haus / System

Bewohner im Haus

System

3 Orientierung / Neigung

Orientierung der Module
Süd

Dachneigung

4 Grösse der Anlage

Leistung auf Dach

Leistung auf Fassade

Batterie

Start Monatsdaten PDF-Report

Weitere Einstellungen Renditerechner

Referenz-Wetterdaten

Ergebnisse Simulation

Gesamtstromproduktion	25'634 kWh/Jahr
Solarstrom selber verbraucht	18'028 kWh/Jahr
Eigenverbrauchsanteil	70.3 %
Solarstrom ans Netz abgegeben	7'606 kWh/Jahr
Kosten schlüsselfertige Anlage	49'040 CHF
Kleine Einmalvergütung KLEIV	12'100 CHF
Amortisationsdauer der Anlage	5 Jahre

Jahresverlauf

- 120 Minuten
- 300 Studierende (BSc)
- Tool: Online-Datenbanken
- Aufgabe: Berechnungen und Evaluation von Heiz-, Kühl- und Lüftungsparametern an unterschiedlichen Standorten
- Aufgabenformat: offen und geschlossen

Jupyter Notebooks: Analyse von Big Data

```
6. Dropping the duplicate rows

This is often a handy thing to do because a huge data set as in this case contains more than 10000 rows. I remove all the duplicate value from the data-set. For example prior to removing I had 11989 of duplicate data.

In [7]: df.shape
Out[7]: (11914, 10)

In [8]: duplicate_rows_df = df[df.duplicated()]
print("number of duplicate rows: ", duplicate_rows_df.shape)
number of duplicate rows: (989, 10)

Now let us remove the duplicate data because it's ok to remove them.

In [9]: df.count() # Used to count the number of rows
Out[9]: Make          11914
Model          11914
Year           11914
HP             11845
Cylinders      11884
Transmission   11914
Drive Mode     11914
MPG-H          11914
MPG-C          11914
Price          11914
dtype: int64

So seen above there are 11914 rows and we are removing 989 rows of duplicate data.
```

- 180 Minuten
- 300 Studierende (BSc, MSc)
- Tool: Jupyter Notebooks
- Aufgabe: Theorie, Analyse, Datenbankabfrage
- Aufgabenformat: geschlossen; offen (Code für Datenbankabfrage)