

ReCode Data Contract

## What specific data Backend needs from Frontend

## **User Authentication & Roles**

**Data Needed from Frontend:**

* **Login page:**
  + email (string)
  + password (string)
* **Registration page:**
  + email (string)
  + password (string)
  + full\_name (string)
  + role (enum: student/lecturer)
  + module\_code (string)

## **Slide Upload & Topic Management (Lecturer)**

Lecturers upload slides -> backend parses slides and suggests topics->lecturers can edit topics.

**Data Needed from Frontend:**

* Slide file upload (pptx/pdf)
* File name
* Edited topic list:
  + topic\_name (string)
  + importance\_weight (float)
  + status (active/archived)

## **Challenge Creation & Question Selection (Lecturer)**

Lecturer assembles challenges -> links questions to topics.

**Data Needed from Frontend:**

* Selected topic ID
* Module\_id
* question\_ids (array of objects):
  + question\_id (string)
  + difficulty (enum: bronze/silver/gold
* Challenge metadata:
  + title (string)
  + week\_number (int)
  + open\_date (datetime)
  + close\_date (datetime)

## **Challenge Attempt & Code Submission (Student)**

Students submit code via editor (CodeMirror) -> backend marks with Judge0.

**Data Needed from Frontend:**

* user\_id
* challenge\_id
* question\_id
* submitted\_code (string)
* language (string)
* start\_timestamp (datetimel)
* end\_timestamp (datetime)
* hints\_used (int)
* resubmission\_count (int)

## **Hints & Resubmission Enforcement**

Backend tracks usage and applies penalties from actions done through frontend for ELO point calculations

**Data Needed from Frontend:**

* Hint requests:
  + hint\_id (string)
  + user\_id (string)
  + challenge\_id (string)
* Resubmission attempts:
  + user\_id (string)
  + challenge\_id (string)
  + question\_id (string)

## **Notifications & Reminders**

Notify students about new challenges and deadlines.

**Data Needed from Frontend:**

* Trigger when lecturer selects questions (event trigger)
* Or a set day of week of lecturers choosing when notification goes out

## **Paste/Plagiarism & Editor Telemetry**

Tracks editor events to detect similarity or pasting activity

**Data Needed from Frontend:**

* event\_type (paste or focus or blur)
* Challenge\_id
* question\_id
* user\_id

## **File Uploads & Storage**

Backend stores slide files, exports

**Data Needed from Frontend:**

* file upload (pptx/pdf/txt/csv/xlsx)

## **Challenge Eligibility Engine (Ruby/Emerald Unlocks)**

Backend computes unlocks based on participation and challenge completions through frontend.

**Data Needed from Frontend:**

* user\_id (string)
* challenge\_completion\_ids (array of string)
* current\_participation\_percent (float)