Website Requirements:

Python >= 3.9

Poetry to manage packages

Docker and docker compose to manage containerisation

Postgres database server

Django – python backend

HTMX + css frontend

Website Pages:

1. A homepage with:
   1. Heading: “convince.me”
   2. Explanation Text: “You've approached me at my bridge! But nothing I've seen has made me want to let you past. Write a short message that would convince me to let you through...”
   3. An open text field for users to response to a given question
   4. A submit/respond button that links up to the backend to:
      1. Run validation tests on the user\_input
      2. Checks to see if the user\_input exists in the database
      3. If not, constructs a ‘teacher’ API request to OpenAI (wrapping it up into a prompt with a pre-set system message and wrapper around the user input) to get feedback and a writing score.
      4. Constructs a second, ‘character’ API request to OpenAI (wrapping it up into a prompt with a pre-set system message, the scores from the teacher AI response and a wrapper around the user input) to get an ‘in character’ response to the user input.
   5. Background image
   6. Character image/lottiefiles moving image.
2. Response Page
3. Review Others Page with:
   1. Heading: “Help me get smarter”
   2. Explanation Text: “Vote on the response that you like the best.”
   3. Background image (ensure it doesn’t change size as the content length changes)
   4. 2 columns; each column containing:
      1. A ‘character’ response from the database
      2. A thumbs up button to vote for that response. When clicked;
         1. runs an ELO scoring formula to calculate the change in scores
         2. Gives feedback to the user with the change in rating scores floating above each button/response; ie: +20(green) -20(red)
         3. logs winning response id, winning response elo, losing response id, losing response elo and change in elo score in a log table in the database
         4. updates the ELO score and number of votes in the main database table for the two responses (keeping all other fields the same).
         5. Selects 2 new responses to be reviewed and displays them ‘refreshing’ the responses displayed.