Hello and welcome, Dr. Le Thi Ngoc Hanh, to our project demo. I'm Quan, and I'm going to demo our CV builder web app project.

This is a tool with which the user can upload a resume written in a custom domain specific language, which the system will automatically convert to a nicely formatted preview and give the option to export as a PDF or PNG.

Now, let's go through the setup and the workflow. First of all, I generate the parser and the visitor files from the grammar with "python run.py gen". Ok, they are generated successfully.

We also have this line here (python run.py parse), which is used for debugging and double-checking the results before running the application. Because the application is working perfectly, we don't need to run this.

Then, I will run the web application using "python app.py". This starts the Flask server, and I can open a browser and go to 127.0.0.1:5000.

Okay, this is our user interface. I will just focus on our main function right there, where user can upload a CV file presented in a DSL format. Let me choose an example right there. Now the frontend sends the file to the backend via a POST request to the parser. And after I click the "Generate Beautiful Resume" button... Okay, it shows us a preview right here.

And I will talk about the workflow. The backend reads the file and use the integrated CV lexer and CV parser to build a parse tree, then it uses the custom CV to dict visitor to convert the parse tree to convert the parse tree into a Python dictionary. The backend sends the JSON back to the front end, and the front end renders a live preview of the CV based on the JSON. Finally, the user can choose to download in PDF or PNG. I will save it in this location. Okay, I got it. And this is our result. What if I don't want to save as PNG? I will choose "Download PDF" and save in this location. I already got this file here, so no need to save it anymore.

And so in short, the workflow is like this: the backend will parse with the ANTLR to JSON and convert it, the frontend renders and finally exports it for downloading. That is how the whole system works from start to finish.

That is the end of our project demo. Thank you for watching. Should you have any questions, I'm here to answer. Thank you.