Opening (0:00 – 0:30)

Narration:

“Hello, and welcome to the presentation of my project: JSON to CSV Converter. This project was developed as part the challenge 2. The goal was to automate the integration of JSON files into CSV files, to improve efficiency in generating scientific production reports.”

Problem & Objective (0:30 – 1:15)

Narration:

“JSON is a common format for data exchange, but it is not user-friendly for analysis in tools like Excel. Manual conversion is time-consuming and prone to errors.

the objective was to build a Java application that reads JSON files, validates their structure, transforms them, and generates accurate CSV outputs—ready to use in Excel.”

Solution Design (2:00 – 3:00)

Narration:

“The solution was built in Java 17 with a modular design.

It includes three main components:

JSON Reader – to read and validate files.

Data Mapper – to transform nested structures into flat rows.

CSV Writer – to generate well-formatted CSV files.

The application runs as a desktop program. Parameters like input file, output file, and delimiter can be passed through the command line or a configuration file.”

Transformation Algorithm (3:00 – 3:45)

Narration:

“The transformation algorithm works as follows:

Each JSON object becomes a row in the CSV.

Nested objects are flattened using dot notation, like address.city.

Arrays are joined with the pipe symbol |.

Missing values become empty cells.

Headers are inferred automatically from the JSON keys.”

Demonstration (3:45 – 5:00)

Narration:

“Now let’s see the program in action. I will run the converter from the command line.”

Type command:

java -jar converter-1.0.0.jar sample\_data/input.json sample\_data/output.csv ";"

“Here we pass the input file, the output file, and the delimiter. The program successfully creates a CSV file.”

Results & Benefits (5:00 – 5:45)

Narration:

“With this solution, JSON data can be automatically converted into structured CSV files.

The benefits are:

Faster processing time.

Reduced risk of errors.

Direct compatibility with Excel.

Reusability for future analysis.”

Verification Checklist (5:45 – 6:15)

Narration:

“To verify quality, the program includes:

JSON validation and error messages.

Configurable parameters.

Automated unit tests for JSON reading, CSV writing, and error handling.

Clear JavaDoc and documentation.”

Conclusion & Closing (6:15 – 6:45)

Narration:

“In conclusion, the JSON to CSV Converter project met all backlog requirements.

It is modular, reliable, and efficient. This tool will help reduce integration time and improve the accuracy of scientific production reports.

Thank you, see you next time