* Full name: Iván Kaleb Ramírez Torres
* NAO ID: 3357
* Date: September 17th, 2025
* Name of the pathway in which you are enrolled: Bécalos Techno Ready
* Title of the Challenge: **Challenge 2 – Back End Java for Information Processing**

# Tracking Tables

## Table 1 – Requirements list

|  |  |
| --- | --- |
| Sprint | Requirements |
| Sprint 1:  **Start planning the project for**  **Challenge 2** | 1. Constructing a Backlog for Challenge 2. 2. Build a Roadmap integrating all the requirements, dates, participants and Sprints for Challenge 2. 3. Create a presentation that documents:  * The basic concepts of the SCRUM methodology * The characteristics of a JSON and a CSV file * Include descriptive comments in the code to explain the basic functionality of each developed component |
| Sprint 2:  **Develop the desktop program using Java**  **For Challenge 2** | 1. Implement in Java using a popular JSON manipulation library. 2. Create a Java class with a function to **read JSON files** (open, parse, handle exceptions, test in isolation). 3. Use a library to **write CSV files** in Java. 4. Create a Java class with a function to **write data into CSV** (configure delimiters, handle exceptions). 5. Add **JavaDoc documentation** for all classes and functions (purpose, parameters, return values). 6. Create a **GitHub repository** with: 7. Organized code and documentation 8. README.md (project description + setup/run instructions) 9. Proper access permissions for reviewers |
| Sprint 3:  **Develop a desktop Java program** | 1. Develop a desktop Java program to **convert JSON into CSV**. 2. Implement functionality:  * JSON file reading (reuse Sprint 2 function). * Data transformation (map JSON into CSV structure). * CSV file writing (reuse Sprint 2 function). * Transformation algorithm (document mapping/validation logic). * Parameter configuration (filenames, delimiters via CLI or config file).  1. Add **JavaDoc documentation** (purpose, parameters, return values, usage examples). 2. Use a **verification checklist** to confirm required functionality and quality. 3. Ensure all **documentation is complete and clear**. 4. Update **GitHub repository** with Sprint deliverables (organized files, updated docs). |
| Final Project:  **Document Analysis & Results for**  **Challenge 2** | Make a video presentation explaning Analysis & Result of the Challenge 2 |

## Table 2: Prioritize list – Challenge 2

|  |  |  |  |
| --- | --- | --- | --- |
| Requirements | Stages (Steps) | Time Estimation | Deliverables |
| Constructing a backlog for Challenge 2 | 1. Identify tasks 2. Prioritize tasks 3. Document backlog | 3h | Backlog document |
| Build a roadmap integrating requirements, dates, participants, and sprints | 1. Define timeline 2. Assign participants 3. Organize sprints | 4h | Roadmap file (timeline) |
| Create a presentation: SCRUM, JSON, CSV, with code comments | 1. Collect content 2. Design slides 3. Export presentation | 5h | Presentation (PDF/Word/Slides) |
| Implement in Java using a JSON library | 1. Choose library 2. Add dependency | 2h | Java project setup with JSON dependency |
| Create a function to read JSON files (open, parse, handle exceptions, test in isolation) | 1. Open JSON file 2. Parse into object/list 3. Handle exceptions | 6h | Java class JsonReader.java with tests |
| Use a library to write CSV files | 1. Add CSV library 2. Verify configuration | 2h | Configured CSV writing dependency |
| Create a function to write data into CSV (configure delimiters, handle exceptions) | 1. Define CSV structure 2. Write rows 3. Handle exceptions | 6h | Java class CsvWriter.java with tests |
| Add JavaDoc documentation for all classes and functions | 1. Write comments 2. Generate docs | 3h | JavaDoc comments in code |
| Create GitHub repository with structured code & documentation | 1. Create repo 2. Push files | 1h | GitHub repo link |
| Add README.md (description + setup/run instructions) | 1. Create file 2. Add description 3. Add setup steps | 2h | README.md file |
| Provide access permissions for reviewers | 1. Open repo settings 2. Add reviewers | 0.5h | Repo access granted |
| Develop desktop Java program to convert JSON into CSV | 1. Integrate reader/writer 2. Run program 3. Debug issues | 8h | Main program file |
| Data transformation: map JSON into CSV structure | 1. Define mapping rules 2. Implement transformation 3. Validate output | 5h | Transformation algorithm implemented |
| Parameter configuration (filenames, delimiters via CLI/config file) | 1. Define parameters 2. Implement parser 3. Test options | 3h | Config options added |
| Document mapping/validation logic in algorithm | 1. Write explanation 2. Add validation details | 2h | Documentation updates |
| Add JavaDoc (purpose, parameters, usage examples) | 1. Write comments 2. Add usage examples 3. Generate docs | 3h | Updated JavaDoc |
| Use verification checklist to confirm functionality & quality | 1. Create checklist 2. Validate items | 2h | Completed checklist |
| Ensure all documentation is clear and complete | 1. Review files 2. Correct issues 3. Finalize docs | 2h | Updated documentation set |
| Update GitHub repo with Sprint 3 deliverables | 1. Commit changes 2. Push to repo | 1h | Updated repo |
| Make a video presentation explaining analysis & results | 1. Prepare script 2. Record video 3. Edit final file | 6h | Video presentation file |

As the User Stories was an exercise already made in Challenge 1, All this backlog was made according to Challenge 2 requirements for All 3 Sprints and Final Project.