ASSIGNMENT NAME: HYP/Technology/BE
DESCRIPTION: Evaluation of the backend of the application
COURSE: Hypermedia applications
STUDENT/GROUP:
ADDITIONAL CONSTRAINTS: To pass the back-end part, each criteria grade should at least be "partially" or more. Besides, the web application should be up and running on a remote cloud platform such as Heroku at the time of the exam; no projects with a non-functioning application will be accepted. The back-end documentation should be available as a single page through a specific url of the web application you are building (e.g., `yourwebapp.com/docs/be`). The page must 1) contain a link to the browsable openAPI specification, 2) a link to the private repository containing the source code (provided that you've given access to `vittorio.zaccaria@polimi.it`) and 3) should fulfill each criteria in the dimensions below.

DOCUMENTATION (30 %)	fully (100%)	fairly (70%)	partially (60%)	slightly (30%)	not at all (0%)
Provides a link to the REST OpenApi specification	☐	☐	☐	☐	☐
Provides a link to the on-line repository where application code is stored	☐	☐	☐	☐	☐
Describes how each criteria in the "Specification", "Implementation" and "Testing" dimensions below have been addressed	☐	☐	☐	☐	☐
Provides an overview of the architecture of the application through a diagram	☐	☐	☐	☐	☐
Describes the data model used for managing the application through a conventional (e.g. ER diagram) formalism	☐	☐	☐	☐	☐
Describes the tasks assigned to each member of the group	☐	☐	☐	☐	☐
Describes and points to relevant real APIs that have inspired the specification	☐	☐	☐	☐	☐
Each of the members of the team describes what was the most important thing they've learned while developing this part of the project, what questions remained unanswered, how they will use what they've learned their everyday life	☐	☐	☐	☐	☐

SPECIFICATION (30 %)	fully (100%)	fairly (70%)	partially (60%)	slightly (30%)	not at all (0%)
Adheres to common practices to partition the web application (static assets vs. application data, front-end data vs application data etc.)	☐	☐	☐	☐	☐
Is documented through an OpenAPI specification	☐	☐	☐	☐	☐
Defines a concise yet comprehensive (concerning the front-end requirements) set of REST endpoints	☐	☐	☐	☐	☐
Defines a concise, simple model for data	☐	☐	☐	☐	☐

IMPLEMENTATION (20 %)	fully (100%)	fairly (70%)	partially (60%)	slightly (30%)	not at all (0%)
Adheres to the provided specification	☐	☐	☐	☐	☐
Provides session management enabling access control	☐	☐	☐	☐	☐
Integrate either a relational or a no-SQL database for managing the data model	☐	☐	☐	☐	☐
Differentiate code and data to enable application re-targeting and/or internationalization	☐	☐	☐	☐	☐

	fully (100%)	fairly (70%)	partially (60%)	slightly (30%)	not at all (0%)
Provides a clear and synthetic overview of static assets used when deploying the web app	☐	☐	☐	☐	☐

TESTING (20 %)	fully (100%)	fairly (70%)	partially (60%)	slightly (30%)	not at all (0%)
Provides command line scripts (or an appropriate online page) for remotely testing the REST API	☐	☐	☐	☐	☐