

Internet Multimedia Engineering

EXAM

Cloud and Network Programming

Alex Carrega – *alessandro.carrega@unige.it*

16th May 2025

Problem

You have to implement an **archive** of the **football teams** and **players**. This archive has to be managed through a user Representational State Transfer (REST) Application Programming Interface (API). In more details, it is necessary to define:

- an **entity** to represent the **teams**;
- an **entity** to represent the **players**.

In particular, in each **team** must be present the following data:

- **name** (indicate the name of the team);
- **year of foundation**;
- **city**.

Instead, in each **player** must be present the following data:

- **name**;
- **birth date**;
- **role**

You have to implement the **history** of the **player**. In more details, you have to specify the **period of date** (*start*, *end*) *in a team*.

Important

- *Only the user with admin role can create, updated and delete ALL the teams and players.*
- *Basic user can create, update, delete only created teams and players.*

For each question the following bonus are applied:

0.50 points if the code is correctly indented;

1.00 points if the *variable names* are *related to the context* (it is more *simple the understanding* of the code):

Examples

- If I have to refer to the book entity model, the correct name can be *book*, *books*, etc. and *not* *entity_1*, *entity_a*, or *entityA*;
- If I have to refer to the book - customer relationship model, the correct name can be *BookCustomer*, *book_customer*, etc. and *not* *relationship_1*, *relationship_a*, or *relationshipA*;

0.50 points if the *variable* and *function names* are written in *english* (de-facto standard in the *programming*);

0.75 points if the *code* is well *commented* in *english* (improves the *understanding* of the code).

9 Points 1. Creation, update, deletion of a team.

9 Points 2. Creation, update, deletion of a player.

9 Points 3. Creation, update, deletion of the history of the player.

9 Points 4. Add a *standard authorization mechanism* to all the functions defined in the REST APIs.

9 Points 5. Additional *entity* for the *player role* with *relative relationship* to the *players* and the needed operations using the REST API.

Extra

Tips

- *poetry* (<https://python-poetry.org>) is valid tool to *manage python environment* and *avoid possible problems with package dependencies*.
- *Guide*: <https://realpython.com/dependency-management-python-poetry>.

Instructions for Delivery

- *Zip your files (only directory and python scripts with extension .py).*

- **Rename the zip file with the following syntax:** your-id-number-surname_name.zip.
- **Option A** (highly recommended)
 - Go the URL of this teaching in **Aulaweb**: .
 - Enter to the assignment **2025-XX-XX [XXh XX 2025]** and upload the file.
- **Option B**
 - **Send an email to `alessandro.carrega@unige.it` with subject “Exam 2025-XX-XX your-id-number”, your zip file as attachment, and no text.**
- **Option C**
 - **Send your zip file as attachment in the chat with `alessandro.carrega@unige.it` of Microsoft Teams channel of Cloud and Internet Programming - 104779 class.**