Player DB Microservice

Your assignment is to create a microservice which serves the contents of the People.csv through a REST API. You are free to choose whatever programming language you are comfortable with, SDKs, web frameworks, databases, and online resources to complete this exercise.

Requirements:

Provide a REST API to your service
Package the application for distribution. Some examples of this:
- Docker image (preferred)
- Tomcat WAR
- Static binary
- others...

Add some unit tests to cover the core logic

REST API

The service should expose two REST endpoints:

GET /api/players - returns the list of all players
GET /api/players/{playerID} - returns a single player by it's ID
Given a player line of the CSV file:

Example Data:

```
playerID,birthYear,birthMonth,birthDay,birthCountry,birthState,birthCity,deathYear,dea
thMonth,deathDay,deathCountry,deathState,deathCity,nameFirst,nameLast,nameGiven,weight
,height,bats,throws,debut,finalGame,retroID,bbrefID
aardsda01,1981,12,27,USA,CO,Denver,,,,,,,David,Aardsma,David
Allan,215,75,R,R,2004-04-06,2015-08-23,aardd001,aardsda01
```

The json representation of a player should be the following, where each field name is equal to the CSV column name:

```
{
   "playerID": "aardsda01",
   "birthYear": 1981,
   "birthMonth": 12,
   "birthDay": 27,
   "deathYear": null,
   "deathMonth": null,
   "deathDay": null,
   "nameFirst": "David",
   "nameLast": "Aardsma",
   "height": 75,
   ...
   ...
}
```

Rate Limiting

The service should expose an additional REST API endpoint:

POST /api/sleep - sleeps for the specified duration in seconds, then returns.
Example Request Payload:

```
{
```

```
   "duration": 10
}
```
Response Payload:

```
{}
```
The API should only run a maximum of 5 concurrent requests at a time. If a sixth request is made while there are already 5 in-flight requests yet to be completed, the API should block until one of the in-flight requests complete, before executing the sleep.

Additional Requirements:
Pagination:
The API should support pagination of response

The service should expose a REST endpoint:

GET /api/players/paginate - returns the list of all players paginating 25 players per responses