

Obligatory assignment 2:

wRESTling bots

DATA2410, Spring 2021

- **This is a group assignment.**
- The assignment does not count towards your grade but must be completed and passed for you to be able to enter as an exam candidate.
- Allowed programming languages are python3, java, or javascript. Other languages may be acceptable on request, depending on resources.

Contents:

Introduction:	1
Task 1: RESTful API server	1
Users:	1
Chat-Rooms:	2
Room users:	2
Messages:	2
Task 2: Clients and bots.	2

Introduction:

We've all made some pretty bad bots in the past. In this assignment you'll make a brand new modern version of the chat room that instead of using raw sockets implement a RESTful API. You'll still have bots; you are allowed to use the bots you received in portfolio1 and to take some inspiration from what you may have invented yourself before, but you are not allowed to copy or reuse any code.

Task 1: RESTful API server

Implement a RESTful API for a chat server. The server API will contain at least these resources and operations:

Users:

Route: /api/users

Operations: Get all, get one, add one (e.g. register), delete one (e.g. deregister)

Chat-Rooms:

Route: /api/rooms

Operations: Get all, get one, add one.

Room users:

Route: /api/room/<room-id>/users

Operations: Get all, add one(e.g. join)

Restrictions: Only registered users can join

Messages:

Route: /api/room/<room-id>/messages

Operations: Get all

Restrictions: Only users in the room can get messages.

Route: /api/room/<room-id>/<user-id>/messages

Operations: Get all, add one

Restrictions:

- Only users in the room can get or add messages. Only registered user-id's should be permitted as <user-id>
- All requests except the one adding a new user, will require userID to be passed in with the request as JSON.

Task 2: Clients and bots.

1. Implement a "client program" or library, that can connect to server described above and perform all of the supported operations.
2. Implement at least 4 "chat bots". You can take inspiration from bots you have made before, but not copy any code from any previous work except the code given to you as part of the course (e.g. you can reuse the bad bots from portfolio 1 as a minimal implementation). The bots should all share the "client code" from step 1, but add their own behaviours, which at least must include:
 - Register as a user once
 - Join an existing room and create a new one
 - Post several messages in all the rooms it's a part of and also fetch all messages in those rooms. Great bots respond to other messages. Lesser bots just post things. Both are welcome.
 - Implement a basic user interface for a human to observe and optionally interact with the chat rooms. The user interface can be a simple terminal program or a web page that lets the user select which of the actions accessible in the API it wants to take (for example, fetch all messages from room 42) and then asks for input to that action where applicable (for example, if the action is to post a message to room 42, allow the user to type in the message they want to send)