**Aufgabe 1.2.1 : The blackboard registration**

socat - UDP-LISTEN:24000,crlf

accepting UDP connection from AF=2 172.19.0.3:34589

{"blackboard\_port":5000}

1. Discover the blackboard service

curl 172.19.0.3:5000

"definitions": {},

"info": {

"title": "Cool product name",

"version": "0.0.0"

},

"paths": {

"/blackboard/deliverables/{id}": {

"get": {

"description": "",

"responses": {},

"summary": "Details about a single deliverable",

"tags": [

"deliverable"

]

}

},

"/blackboard/deliveries": {

"get": {

"description": "",

"responses": {},

"summary": "Lists the deliveries",

"tags": [

"delivery"

]

}

},

"/blackboard/deliveries/{id}": {

"get": {

"description": "",

"responses": {},

"summary": "Details about a single delivery",

"tags": [

"delivery"

]

}

},

"/blackboard/quests": {

"get": {

"description": "",

"responses": {},

"summary": "Lists the quests available",

"tags": [

"quests"

]

}

},

"/blackboard/quests/{id}": {

"get": {

"description": "",

"responses": {},

"summary": "Shows details about the quest",

"tags": [

"quest"

]

}

},

"/blackboard/quests/{id}/tasks": {

"get": {

"description": "",

"responses": {},

"summary": "Lists the tasks to be fulfilled to solve the quest",

"tags": [

"task"

]

}

},

"/blackboard/tasks/{id}": {

"get": {

"description": "",

"responses": {},

"summary": "Details about a single task",

"tags": [

"task"

]

}

},

"/map": {

"get": {

"description": "",

"responses": {},

"summary": "Your friendly map, telling you where locations are found",

"tags": [

"location"

]

}

},

"/map/{name}": {

"get": {

"description": "<br/>Locations do update itself. Do not try to update th em. You don't know the key ;)",

"responses": {},

"summary": "Information about a location on the map",

"tags": [

"location"

]

},

"put": {

"description": "<br/>Locations do update itself. Do not try to update th em. You don't know the key ;)",

"responses": {},

"summary": "Information about a location on the map",

"tags": [

"location"

]

}

},

"/users": {

"get": {

"description": "Lists the available users of the system.<br/>You may reg ister yourself with a post of {\"name\":\"<name>\", \"password\":\"<password>\"} ",

"responses": {},

"summary": "The list of users",

"tags": [

"users"

]

},

"post": {

"description": "Lists the available users of the system.<br/>You may reg ister yourself with a post of {\"name\":\"<name>\", \"password\":\"<password>\"} ",

"responses": {},

"summary": "The list of users",

"tags": [

"users"

]

}

},

"/users/{name}": {

"get": {

"description": "",

"responses": {},

"summary": "Shows details about a single user",

"tags": [

"user"

]

}

},

"/users/{name}/encryption\_key": {

"get": {

"description": "This will provide information about the encryption key u sed when<br/>comminicating in passwordless mode (hopefully soon to come)<br/>Use r has to be authenticated by basic auth to use this resource.",

"responses": {},

"summary": "Gives the users encryption key.",

"tags": [

"user"

]

}

}

},

"swagger": "2.0"

}

2. Register a /users ( {“name”:”<user>”, “password”:”<pass>”} )

curl -d '{"name":"jannikb", "password":"jannikb"}' -H "Content-Type: application/json" -X POST 172.19.0.3:5000/users

{

"encryption": {

"HMACK": "sha256",

"algorithm": "AES",

"encryption": "Fernet",

"key": "w+BE0MiiO47k5tD5SKUHeTl7Xv3HaiV0x1Xkzc/ziDM=\n",

"key\_encoding": "base64",

"keylength": "128",

"mode": "CBC",

"padding": "PKCS7"

},

"message": "Created User",

"object": [

{

"\_links": {

"encryption\_key": "/users/jannikb/encryption\_key",

"self": "/users/jannikb"

},

"deliverables\_done": [],

"delivered": [],

"ip": null,

"location": null,

"name": "jannikb"

}

],

"status": "success"

}

3. Login via get /login and receive your authentication token

curl --user jannikb:jannikb 172.19.0.3:5000/login

{

"message": "Welcome. To use your token set header 'Authorization:Token <tokenvalue>'",

"token": "Z0FBQUFBQlo3MnlnaWlDVlJKZG5RbUQ1Um9odk1LTTZ0WktnTG1mZEViWUpoQjBhbkJaaGd0YVEtZG1XeE1DcnJzWmZMeE9ralhqRnh5LURfMU9qaWtlXzBITm15Z1ROeUNLZU1iUTI3QXdZOVE0YkVvaklMb2Fud2l1bXBXTlp5U0xsNkE0MklfTWg1aFAxenJFNkprY0VzUEhaZDVmZ3dBPT0=",

"valid\_till": 1508891936.457143

}

4. Check your login via /whoami

curl 172.19.0.3:5000/whoami -H "Accept: application/json" -H 'Authorization: Token Z0FBQUFBQlotWHM5b1VGZ1FlQndlbURFcXU2SGEzRE1jRzJVSTFtV1ZMLTluSFVJcjE5RXR6bVBNOTVONDE2b2pjQ2ZSZ2JRRk9EcUV2Q2dfWnctOE1yZms3VENFM21wcEwwWFA0Z3pSUW5OZ1RRbDI1TFRqUS1GWnRjVDQ1Q2RvYllqOVRpdURXTXBzNVhvaEpvcWtXV1NwWG5KV1V4aGdBPT0='

{

"message": "You are authenticated",

"user": {

"\_links": {

"encryption\_key": "/users/jannikb/encryption\_key",

"self": "/users/jannikb"

},

"deliverables\_done": [],

"delivered": [],

"ip": null,

"location": null,

"name": "jannikb"

}

}

**Aufgabe 1.3 : Questing**

curl 172.19.0.3:5000/blackboard/quests

{

"objects": [

{

"\_links": {

"deliveries": "/blackboard/quests/1/deliveries",

"self": "/blackboard/quests/1",

"tasks": "/blackboard/quests/1/tasks"

},

"deliveries": [

1,

2,

5,

8,

9,

13,

14,

15

],

"description": "Welcome young adventurer! I'm so glad you are here. We are in grave danger and we are so relieved you answered the call. You should first go an see the king to be welcomed to the empire",

"id": 1,

"name": "A kingly welcome",

"reward": 1,

"tasks": [

"/blackboard/tasks/1"

]

}

],

"status": "success"

}

curl 172.19.0.3:5000/blackboard/tasks/1 -H 'Authorization: Token <tokenvalue>'

**Aufgabe 1.4 : Check the map**

1.Find the host for solving the quest, by querying the /map/<location>

curl 172.19.0.3:5000/map/Throneroom

2.Answer this question: Is the map a yellow pages or white pages server and what is the

difference between?

**Aufgabe 1.5 : Go get your hands dirty**

1.Visit the location of the quest.

curl 172.19.0.4:5000/visits -H "Accept: application/json" -H 'Authorization: Token Z0FBQUFBQlotWHM5b1VGZ1FlQndlbURFcXU2SGEzRE1jRzJVSTFtV1ZMLTluSFVJcjE5RXR6bVBNOTVONDE2b2pjQ2ZSZ2JRRk9EcUV2Q2dfWnctOE1yZms3VENFM21wcEwwWFA0Z3pSUW5OZ1RRbDI1TFRqUS1GWnRjVDQ1Q2RvYllqOVRpdURXTXBzNVhvaEpvcWtXV1NwWG5KV1V4aGdBPT0='

2. solve the quest by posting to the resource and follow any further given instructions

curl -X POST 172.19.0.4:5000/visits -H "Accept: application/json" -H 'Authorization: Token Z0FBQUFBQlotWHM5b1VGZ1FlQndlbURFcXU2SGEzRE1jRzJVSTFtV1ZMLTluSFVJcjE5RXR6bVBNOTVONDE2b2pjQ2ZSZ2JRRk9EcUV2Q2dfWnctOE1yZms3VENFM21wcEwwWFA0Z3pSUW5OZ1RRbDI1TFRqUS1GWnRjVDQ1Q2RvYllqOVRpdURXTXBzNVhvaEpvcWtXV1NwWG5KV1V4aGdBPT0='

{

"message": "\nWelcome to the Throneroom! Well they have told you to come visit t he King?\nWell, do you have an audience? NO? Then go away and tell your supervisor that the King does not have time\nfor every new greenhorn in town. But consider y our quest as fullfilled - at least you visited the Throneroom.\nGive this token ba ck as a symbol of fullfillment ( quest deliveries, POST {\"tokens\":{\"<task\_uri>\ ":token}} )\n",

"token": "Z0FBQUFBQlotWF91YXNmTlh1OFYtMnN2Y2xzMjExLTJEQ0lZYk9RRUFHREl2eE16LUQtYU tIZnFFSmdLOHdJS3V3Q01wZHkyeXBLS1NpTmJLc3RUc1VqRk1QamdvUFRKcnBBWDZGM3NwWC1oTkFaSkdS VjNnTjdvQXBUd2EwcmtBZ20tMmZFOVZUdHcwZjFGRzEtN05mYjlxejZWaHBQcFlxakV5WXN0c3U5Zjc3bl ZodllFLVB6LXdHRFdXQ1BXNThMUjF3SXpTTml5VnRGWFhUQzlTd1R5OXdaWTlLc1VyMzA3dW01clNNYlU4 azIxVmF3UWFPamZXNUYtMU4zVS0tdFpWRnB6eXhiazI1bFEtbFYwNU1hOUFOWnQxSy1PLXRoeHRkV0NmTX o2OXExeW1VcjcxUHFReTExNi1zQjk3RlF4LVhiaWo3RzhIQ0xyYk5saTdQcmdwaFdGdVcyZkVzb25CaGJv eDU2U2hYT1ZHNlBzdjZGR1IxekJkZ01OTnd2cVJlV3FEVTJfXzdSejJqOHVYWTB3NGNxRk56WGJaOU5waG JwWWVDaG9wMFdHSUV3NHhJajhvaXRlOEc0QjBwQkVEcmc5VjhxQg=="

curl -i -X POST 172.19.0.3:5000/blackboard/quests/1/deliveries -H "Content-Type: application/json" -H "Authorization: Token Z0FBQUFBQlotWHM5b1VGZ1FlQndlbURFcXU2SGEzRE1jRzJVSTFtV1ZMLTluSFVJcjE5RXR6bVBNOTVONDE2b2pjQ2ZSZ2JRRk9EcUV2Q2dfWnctOE1yZms3VENFM21wcEwwWFA0Z3pSUW5OZ1RRbDI1TFRqUS1GWnRjVDQ1Q2RvYllqOVRpdURXTXBzNVhvaEpvcWtXV1NwWG5KV1V4aGdBPT0=" -d '{"tokens":{"/blackboard/tasks/1":""}}'

}

HTTP/1.0 201 CREATED

Content-Type: application/json

Location: http://172.19.0.3:5000/blackboard/deliveries/35

Content-Length: 253

Server: Werkzeug/0.12.2 Python/3.5.4

Date: Wed, 01 Nov 2017 09:38:38 GMT

{

"message": "Created Delivery",

"object": [

{

"deliverables": [

25

],

"id": 35,

"quest": 1,

"timestamp": "2017-11-01T10:38:38.860083+00:00",

"user": "jannikb"

}

],

"status": "success"

}

**Aufgabe 1.6: Specify a tavern**

Sooner or later it is important to design a brand new service. For that API-Definitions or APIDocumentations

come in handy.

Your task is to specify a new „tavern“ service where heroes just like you are welcome and

may offer help to others as well as seek help themselves. If you are a warrior and need the

help of a priest, the tavern should be the place to find the help you need. To be clear: Don’t

write code, but write specification/documentation like swagger or raml.

1.) Heroes must be able to offer help, indicating which hero class (like mage, warrior,

priest….) they belong to and how they are reachable

2.) The offers shall be deletable and update-able

3.) Other heroes should be able to view a list of all offers

4. ) Heroes should have a nice way to get help of a specific class (they don‘t like to talk to a

bunch of priests if they need a plumber)

5.) Argue if this is rather a white pages or a yellow pages service and which does fit the job

better