

COMP0034 Coursework_1

General introduction of file structures:

In the rest API, I define 3 tables in the database including player table, trainer table and the data table. The player table and the trainer table are the tables containing information of users. The data table contains the results of output of machine learning models and the input features reading from the excel. Detailed structure of the database is shown in the fig.1.

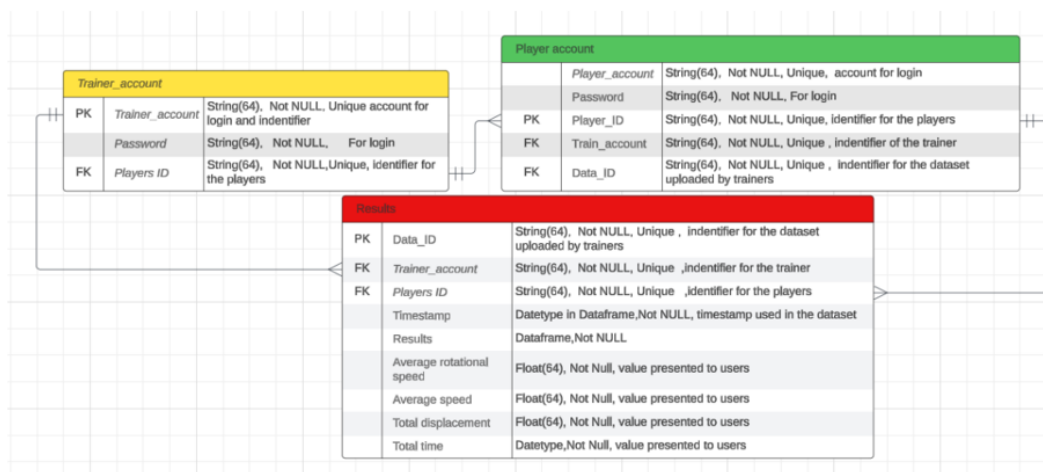


fig.1 The ERD diagrams of the database

The structure of API module is shown in fig.2:

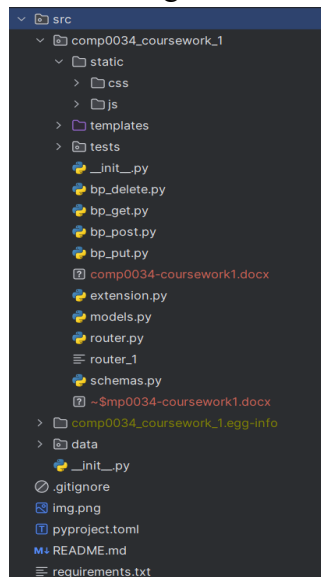


fig.2 The structure of the API

Application part:

As shown in fig.2, the API contains several parts. `__init__.py` contains the setting of the `create_app()` functions to create the flask instance. I used blueprints to make the structures of the module clearer and easy to debug. `bp_delete`, `bp_get`, `bp_put` and `bp_post` contain functions with certain http requests. I deploy the error handlers in the blueprints. Detailed explanation is within code. Extension file is used to import db and ma instances, which avoids location errors when other files importing db and ma. In `models.py`, I defined the models of the databases from designs of ERD in fig.1. The router file is used for directly call API using code. `schemas.py` avails packages Marshmallow is used to serialize and deserialize JSON files and database objects in `schemas.py`. The data file contains the csv file of IMU data.

Test part:

Overall, I run 20 tests for the routes in API. Not only that I tested all the functions in blueprints, but also, I tested some edge situations and whether some errors raised normally with the functions in blueprints. I also tested with different values and see whether the test codes run normally.

As the fig.1, test codes locate in the test directory. In `conftest.py`, I defined some json variables for the player, trainer, and data row, which makes the structure clearer and easier to debug. Other files are used for the testing of certain functions. Detailed explanations are inside the test files.

The fig.3 demonstrates that all the tests have passed, and fig.4 demonstrates the coverage reports:

```
[...] PS E:\Programming_Assignments\comp0034-cw11-King-in-black> coverage run --pytest
===== test session starts =====
platform win32 -- Python 3.7.9, pytest-7.4.4, pluggy-1.2.0
rootdir: E:\Programming_Assignments\comp0034-cw11-King-in-black
configfile: pyproject.toml
testpaths: src\comp0034_coursework_1\tests
collected 20 items

src\comp0034_coursework_1\tests\test_delete_functions.py ..... [ 25%]
src\comp0034_coursework_1\tests\test_get_functions.py ..... [ 55%]
src\comp0034_coursework_1\tests\test_post_functions.py ..... [ 85%]
src\comp0034_coursework_1\tests\test_put_functions.py ... [100%]

===== 20 passed in 59.08s =====
```

fig.3 the test screen shot.

```
(.venv) PS E:\Programming_Assignments\comp0034-cw1i-King-in-black> coverage report
```

Name	Stmts	Miss	Branch	BrPart	Cover
src\comp0034_coursework_1__init__.py	53	3	10	4	89%
src\comp0034_coursework_1\bp_delete.py	38	4	6	0	91%
src\comp0034_coursework_1\bp_get.py	43	13	6	0	65%
src\comp0034_coursework_1\bp_post.py	64	26	16	1	54%
src\comp0034_coursework_1\bp_put.py	35	4	8	3	84%
src\comp0034_coursework_1\extension.py	4	0	0	0	100%
src\comp0034_coursework_1\models.py	36	0	0	0	100%
src\comp0034_coursework_1\router.py	11	11	2	0	0%
src\comp0034_coursework_1\tests\test_delete_functions.py	56	0	0	0	100%
src\comp0034_coursework_1\tests\test_get_functions.py	50	0	0	0	100%
src\comp0034_coursework_1\tests\test_post_functions.py	33	0	0	0	100%
src\comp0034_coursework_1\tests\test_put_functions.py	35	0	0	0	100%
TOTAL	535	61	48	8	85%

fig.4 the coverage reports

From fig.4, we can see that not all the contents of blueprints are tested. That should be the reason of error handler and exception I set. I tested some of the errors but not all of them.

Tools and Techniques:

Instead of importing logging packages, I directly allow pytest to report info level of information from the system configuration to help me to debug.

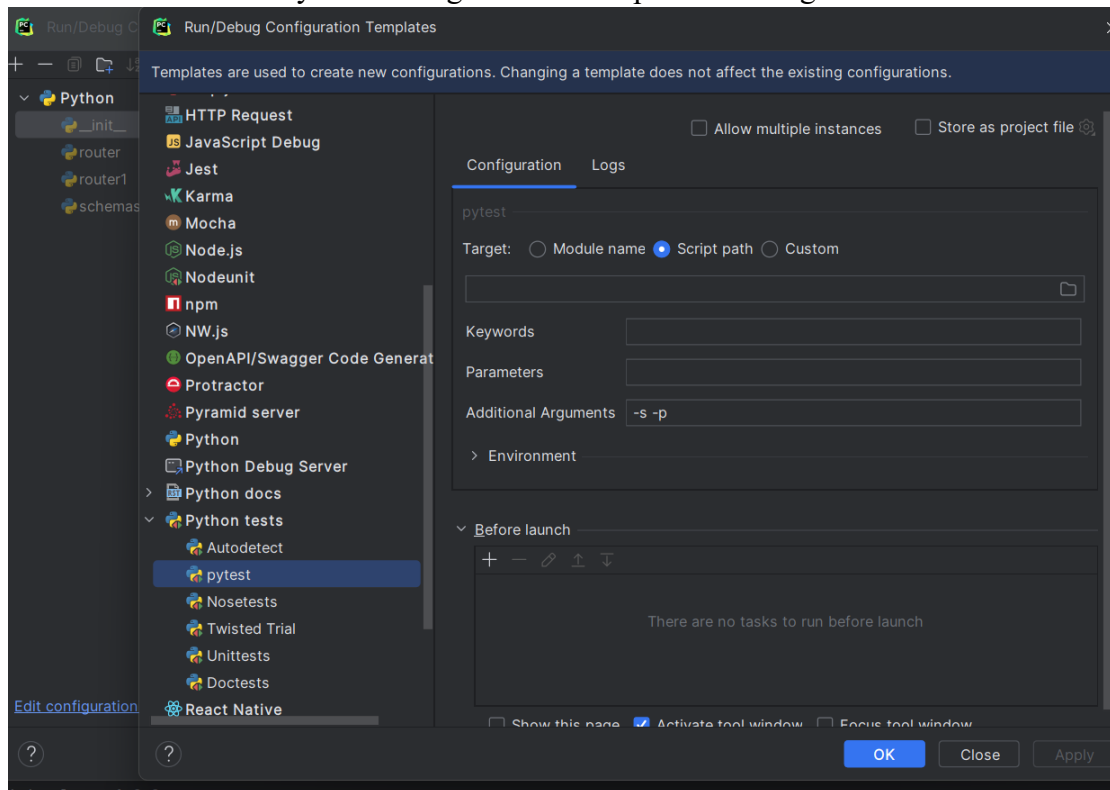


fig.5 the configuration of pytest

Also, I used DB Browser to help me to check the content of database, which helped me to debug the API.

DB Browser for SQLite - E:\Programming_Assignments\comp0034-cw1i-King-in-black\instance\IMU_data.sqlite

文件(F) 编辑(E) 查看(V) 工具(T) 帮助(H)

新建数据库(N) 打开数据库(O) 写入更改(W) 倒退更改(R) 打开工程(P) 保存工程(V) 附加数据库(A) 关闭数据库(C)

数据库结构 浏览数据 编辑备注 执行 SQL

表(T): data

	Data_ID	Dataset_ID	Player_ID	Trainer_ID	timestamp	accX	accY	accZ	gyroX	gyr
1	1	0	NULL	NULL	2062. 9	-0. 496517	3. 785628	8. 954828	-0. 142849	
2	2	0	NULL	NULL	2063. 0	-0. 470263923076923	3. 85022415384615	9. 20642030769231	0. 0319186923076922	0. 0452671
3	3	0	NULL	NULL	2063. 1	-0. 344954625899281	3. 8288434676259	9. 00808051798561	0. 053389417266187	0. 0497821
4	4	0	NULL	NULL	2063. 2	-0. 433005701482004	3. 87208025829217	8. 8217618743825	0. 0211040134086097	0. 0348893
5	5	0	NULL	NULL	2063. 3	-0. 460661514069188	4. 03840173868501	8. 90557713339415	-0. 0804636052908567	0. 0799161
6	6	0	NULL	NULL	2063. 4	-0. 724542123184802	4. 04735024506665	8. 60316728323748	-0. 0017138730324406	-0. 0036801
7	7	0	NULL	NULL	2063. 5	-0. 982852329397539	4. 50310074615458	8. 8116537809683	-0. 204484607869441	0. 100658
8	8	0	NULL	NULL	2063. 6	-1. 08680541918148	4. 36455903412465	8. 48738215882936	-0. 156697647058333	0. 047408
9	9	0	NULL	NULL	2063. 7	-0. 766120413711482	4. 57155308450129	8. 42899529842024	-0. 121720305658489	0. 0252422
10	10	0	NULL	NULL	2063. 8	-0. 888811248590556	4. 63417119510407	8. 32251956079749	0. 0220369571747145	0. 010408
11	11	0	NULL	NULL	2063. 9	-0. 927085042378044	4. 88274429887102	8. 22152578933181	0. 0159378763480756	-0. 0587721
12	12	0	NULL	NULL	2064. 0	-0. 997663950221939	5. 20942166327105	9. 10021330377012	0. 117559116910116	-0. 279924
13	13	0	NULL	NULL	2064. 1	-0. 544175212765829	5. 09974658149557	9. 96065812831373	-0. 379096644109935	-0. 0748402
14	14	0	NULL	NULL	2064. 2	-0. 0129947384235513	3. 49888459787994	10. 6500747714687	0. 406471144340483	0. 0633623
15	15	0	NULL	NULL	2064. 3	-0. 0180440215995171	3. 04649797287273	8. 89862660630925	0. 205319040415832	-0. 050082
16	16	0	NULL	NULL	2064. 4	-0. 300375007695197	3. 25258149274894	7. 83712727732337	0. 174134711990511	-0. 14856
17	17	0	NULL	NULL	2064. 5	-0. 386226502419428	3. 93670724870816	7. 31571458252366	-0. 0215024866554927	-0. 154032
18	18	0	NULL	NULL	2064. 6	-0. 290016050688555	4. 53906897484582	7. 7617877492992	0. 097951454049631	-0. 084739
19	19	0	NULL	NULL	2064. 7	-0. 335652515211871	5. 40892959255485	9. 01971173262518	0. 197635336226475	0. 171884
20	20	0	NULL	NULL	2064. 8	-1. 06898685458913	6. 08660617779008	10. 2281163198297	-0. 143908199143966	-0. 098200
21	21	0	NULL	NULL	2064. 9	-1. 19552175637806	4. 39901105331937	9. 66237259594299	-0. 012565659741816	-0. 0124634
22	22	0	NULL	NULL	2065. 0	-1. 01291572691285	3. 65246731599347	8. 51429937877929	-0. 0572741979226851	-0. 081603
23	23	0	NULL	NULL	2065. 1	-0. 583014418073449	3. 54565059479794	7. 64282291363297	0. 0139138406232614	-0. 021268
24	24	0	NULL	NULL	2065. 2	-0. 302948325421956	4. 27474287843959	7. 34145827408981	-0. 0224811478130318	-0. 0014804
25	25	0	NULL	NULL	2065. 3	-0. 466737597626601	5. 16239966353195	7. 90436518222699	-0. 135300744343919	-0. 2562
26	26	0	NULL	NULL	2065. 4	-0. 758654679287988	5. 5835646990596	9. 57141155466814	-0. 0650881233031739	-0. 378933

Fig.6 the evidence of using DB browser

The menu is in Chinese, but the content is shown in Fig.6.

Also, I use git for version control often; here are part of records:

Console x Update Info: 2024/2/12 11:03 x

Q-	* Cc	Branch	User	Date	Paths
●		Merge remote-tracking branch 'origin/main'	origin & main	Pikaquu_tesla	1 hour ago
●		Change a little in md		Pikaquu_tesla	1 hour ago
●		Rename main.yml to coverage.yml		Z*	1 hour ago
●		Add token		Z*	Today 10:53
●		Create pytest_coverage.yml		Z*	Today 10:48
●		Finish the construction of modules		Pikaquu_tesla	Today 8:31
●		Add some explanations		Pikaquu_tesla	Today 7:52
●		Use black package to reformat		Pikaquu_tesla	Today 7:44
●		Merge remote-tracking branch 'origin/main'		Pikaquu_tesla	Yesterday 21:40
●		Add explanation and revise the markdown		Pikaquu_tesla	Yesterday 21:40
●		Create pylint.yml		Z*	Yesterday 17:40
●		Merge remote-tracking branch 'origin/main'		Pikaquu_tesla	Yesterday 17:38
●		Test CI		Pikaquu_tesla	Yesterday 17:38
●		Create python-app.yml		Z*	Yesterday 17:38
●		Upload content of requirements.txt		Pikaquu_tesla	Yesterday 17:35
●		Finish the test part of the code		Pikaquu_tesla	Yesterday 17:16
●		Finish test of blueprints of delete, start to construct the put functions' test functions.		Pikaquu_tesla	Yesterday 16:17
●		Construct the delete of blueprints		Pikaquu_tesla	Yesterday 12:42
●		finish test code of get		Pikaquu_tesla	Yesterday 12:24
●		Finish the test code of post; and almost finish test code of get		Pikaquu_tesla	Yesterday 11:51
●		bug fixed		Pikaquu_tesla	Yesterday 8:40
●		Not sure why still bugging		Pikaquu_tesla	Yesterday 7:48
●		Successfully debugs the upload csv.		Pikaquu_tesla	2024/2/10 16:28
●		Finish the http response for errors in the all post functions.		Pikaquu_tesla	2024/2/10 12:53
●		Write the http response for errors in the post functions		Pikaquu_tesla	2024/2/10 12:47

Fig.7 the evidence of using git

Also, I add explanation of file in Readme.md and dependencies in requirement.txt.

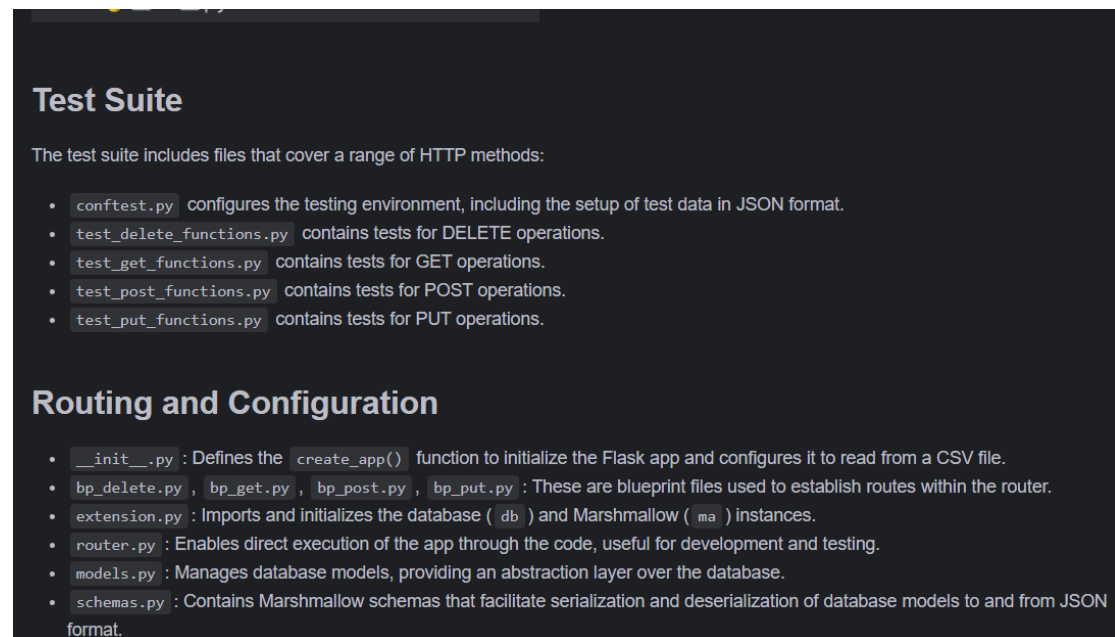


Fig.8 A part of Readme.md

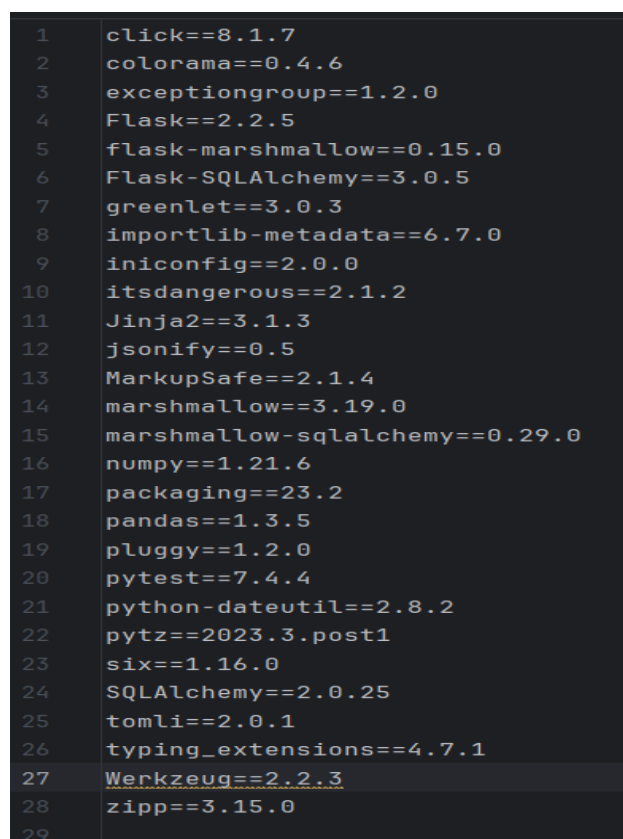


Fig.9 the dependencies in requirements.

Also, I add .gitignore to ignore the virtual environment I used for the project.











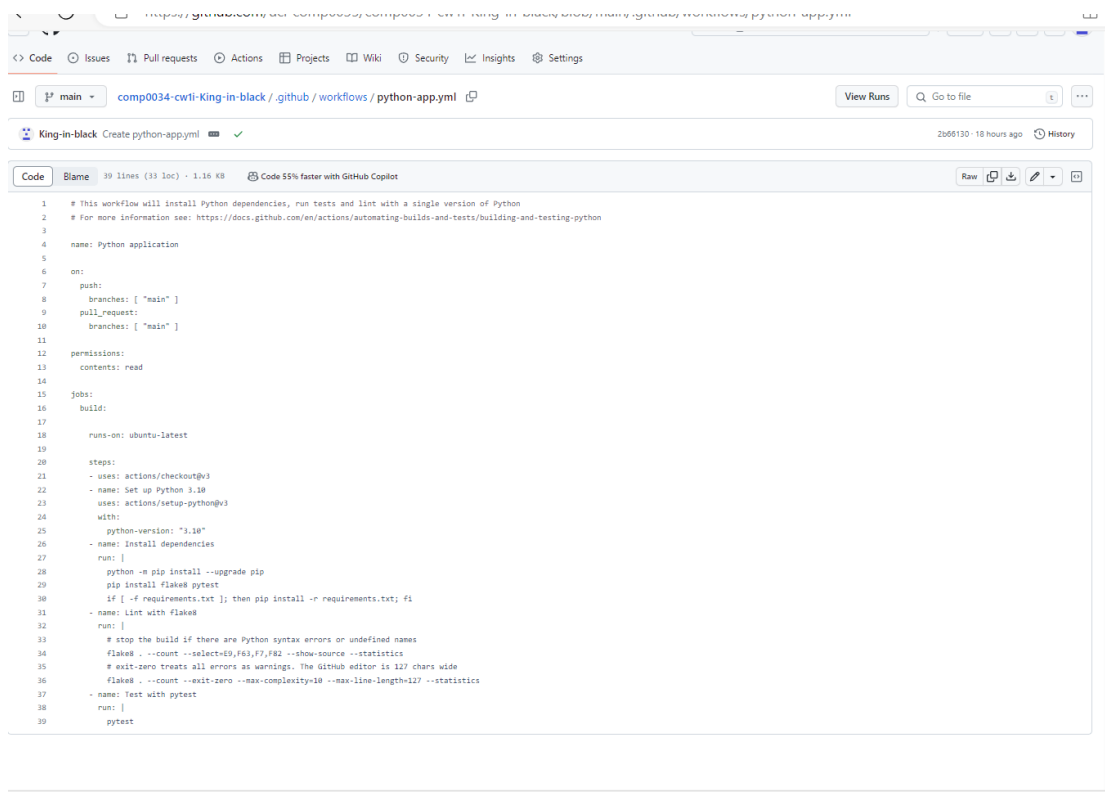
	King-in-black Merge remote-... 	8149e21 · 1 hour ago	 74 Commits
	.github/workflows	Rename main.yml to cover...	1 hour ago
	src	Finish the construction of ...	3 hours ago
	.gitignore	Initial commit	2 weeks ago
	README.md	Change a little in md	1 hour ago
	img.png	Add explanation and revis...	14 hours ago
	pyproject.toml	Update the format of the ...	last week
	requirements.txt	Upload content of require...	18 hours ago

Fig.10 the exists of ignore file

I used CI to use pytest and deploy the workflow in the GitHub. The .github zip in src file demonstrates the CI test code in yml format.



```

1 # This workflow will install Python dependencies, run tests and lint with a single version of Python
2 # For more information see: https://docs.github.com/en/actions/automating-builds-and-tests/building-and-testing-python
3
4 name: Python application
5
6 on:
7   push:
8     branches: [ "main" ]
9   pull_request:
10    branches: [ "main" ]
11
12 permissions:
13   contents: read
14
15 jobs:
16   build:
17
18     runs-on: ubuntu-latest
19
20     steps:
21     - uses: actions/checkout@v3
22     - name: Set up Python 3.10
23       uses: actions/setup-python@v3
24       with:
25         python-version: "3.10"
26     - name: Install dependencies
27       run: |
28         python -m pip install --upgrade pip
29         pip install flake8 pytest
30         if [ -f requirements.txt ]; then pip install -r requirements.txt; fi
31     - name: Lint with flake8
32       run: |
33         # stop the build if there are Python syntax errors or undefined names
34         flake8 . --count --select=E9,F63,F7,F82 --show-source --statistics
35         # exit-zero treats all errors as warnings. The GitHub editor is 127 chars wide
36         flake8 . --count --exit-zero --max-complexity=10 --max-line-length=127 --statistics
37     - name: Test with pytest
38       run: |
39         pytest

```

Fig.11 the exists of yml in workflow.

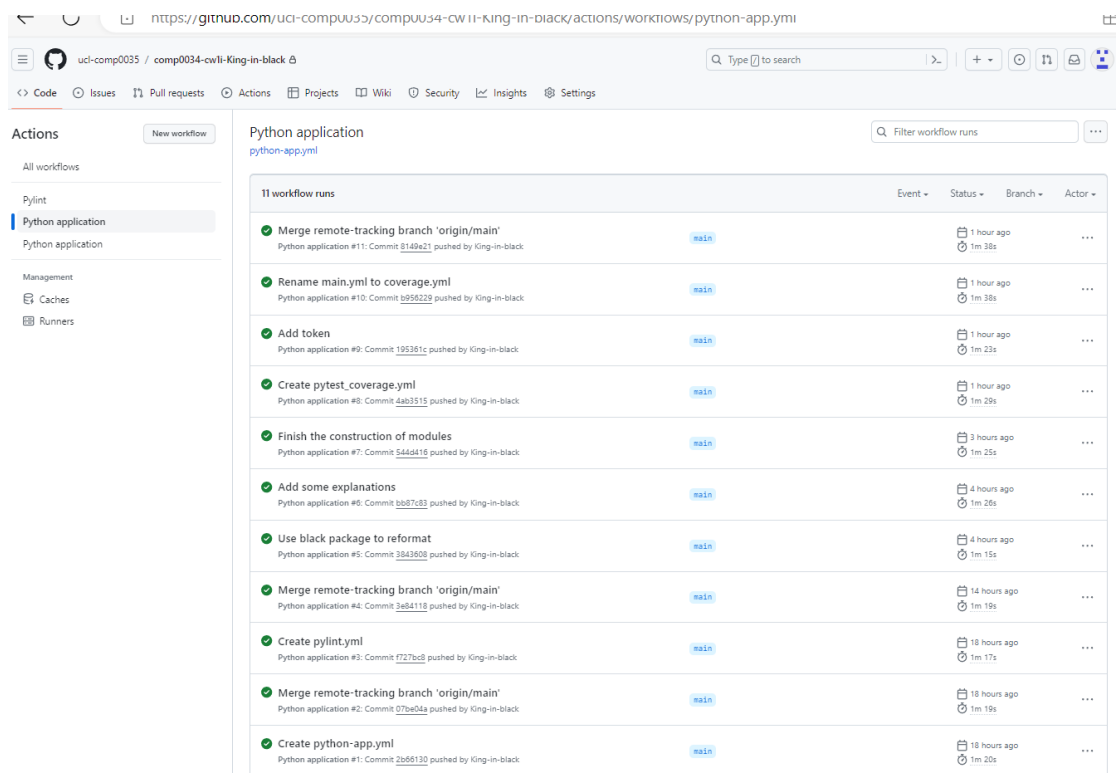


Fig.12 the screen shot of the actions for the projects

I also used black package to lint the files to make the files format in a good style:

```

===== 20 passed
(.venv) PS E:\Programming_Assignments\comp0034-cw11-King-in-black> black src
All done! 🎉📁👉
17 files left unchanged.
(.venv) PS E:\Programming_Assignments\comp0034-cw11-King-in-black>

```

Fig.13 the evidence of using of black packages.

I define pyproject.toml

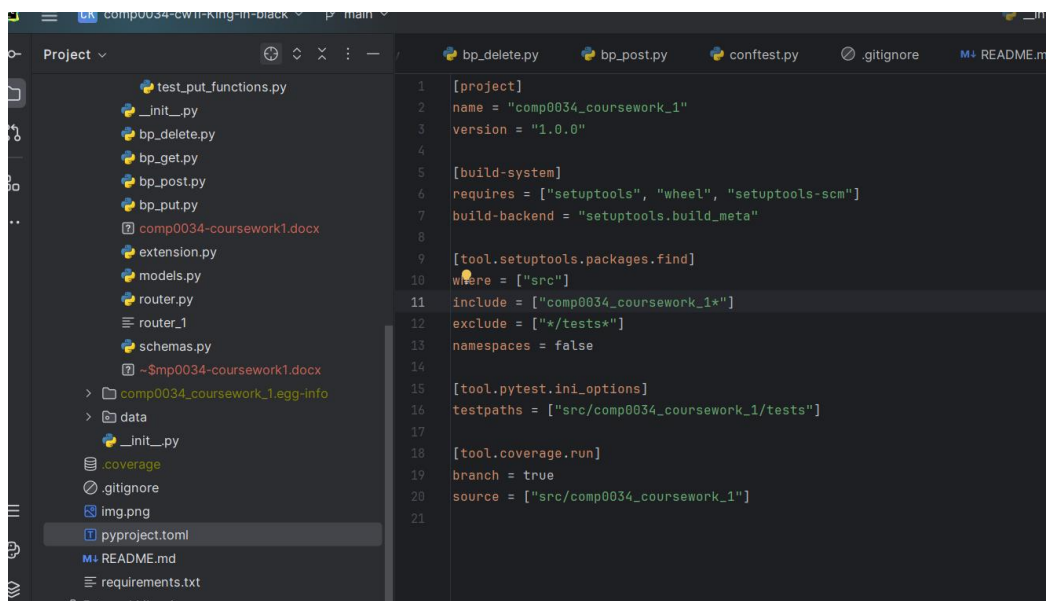


Fig.14 the evidence of defining pyproject.toml.

Improvements I would like to do but I do not have time:

I would like to change the password type of players and trainers to hash type of password. But the workload for that is large, which needs at least 8 hours to do that. All my test functions and blueprints about it need to change. I hardly have time to do that.

References:

Acknowledgement of AI used:

Chatgpt4.0 and Chatgpt 3.5(published by OPEN AI) is used in help me for coding and proving solutions for my projects. Some of codes are generated and I correct them to fit my project. Some of the codes like yml for CI is directly generated without any changes from the Chatgpt. They also suggested new methods to use in debugging. They also advised new ideas about some concepts taught in comp0034. Also, I used it to polish some written works in the Readme markdown.

Cite of the dataset:

AlSahly,Abdullah. (2022). accelerometer_gyro_mobile_phone_dataset. UCI Machine Learning Repository. <https://doi.org/10.3390/s22176513>.

GitHub URL:

<https://github.com/ucl-comp0035/comp0034-cw1i-King-in-black.git>