

Course: IT202-008-S2025

Assignment: Milestone 2 API Testing

Student: Dylan W. (dw347)

Status: Submitted | Worksheet Progress: 100%

Potential Grade: 10.00/10.00 (100.00%)

Received Grade: 0.00/10.00 (0.00%)

Grading Link: <https://learn.ethereallab.app/assignment/v3/IT202-008-S2025/milestone-2-api-testing/grading/dw347>

# Instructions

1. Ensure you read all instructions and objectives before starting.
2. Ensure you added your API choice to the Google doc shared on Canvas (selections must be unique, no two students can use the same exact API)
3. Create a new branch from Milestone2 called M7-Homework
  1. `git checkout Milestone2` (ensure proper starting branch)
  2. If you don't have Milestone2 yet, create it from Milestone1
  3. `git pull origin Milestone2` (ensure history is up to date)
  4. `git checkout -b M7-Homework` (create and switch to branch)
4. Follow the lesson from
  - Add `testApi.php` to your project folder
  - Immediately record to history
    - ☐ `git add public_html`
    - ☐ `git commit -m "adding M7 HW baseline files"`
    - ☐ `git push origin M7-Homework`
    - ☐ Create a Pull Request from M7-Homework to Milestone2 and keep it open
5. Fill out the below worksheet
  - Ensure you add the UCID/Date and details comments as you work
  - Get the example working with your chosen API
6. Once finished, click "Submit and Export"
7. Locally add the generated PDF to a folder of your choosing inside your repository folder and move it to Github
  1. `git add .`
  2. `git commit -m "adding PDF"`
  3. `git push origin M7-Homework`
  4. On Github merge the pull request from M7-Homework to Milestone2
  5. On Github create and merge the pull request from Milestone2 to dev
  6. On Github create a pull request from dev to prod and immediately merge. (This will trigger the prod deploy to make the heroku prod links work)
8. Upload the same PDF to Canvas
9. Sync Local
  1. `git checkout dev`
  2. `git pull origin dev`
  3. `git checkout Milestone2`
  4. `git pull origin Milestone2`

100%

## Section #1: ( 2 pts.) Analyze Api Choice

100%

### Task #1 ( 0.40 pts.) - API Link

#### Url Prompt

**Weight:** 20%**Objective:** API Link

URL #1

<https://rapidapi.com/therundown/api/therundown>

URL

<https://rapidapi.com/therundown/api/1>

Saved: 4/12/2025 12:24:45 PM

100%

### Task #2 ( 0.80 pts.) - Quota and Limiations

#### Text Prompt

**Weight:** 40%**Objective:** Quota and Limiations

Your Response:

1000 requests per hour



Saved: 4/12/2025 9:39:59 PM

100%

### Task #3 ( 0.80 pts.) - Goal Data

#### Text Prompt

**Weight:** 40%**Objective:** Goal Data**Details:**

- Clearly note what type of data you'll be using
- Clearly note the specific properties you intend to use (it's likely not all of them)

Your Response:

I will be using the player data from a certain team. I will be taking the display name(first & last), position, jersey number, debut year and team but team is taken from a seperate endpoint.



Saved: 4/12/2025 5:59:59 PM

100%

## Section #2: ( 6 pts.) Verify Api Choice

100%

### Task #1 ( 6 pts.) - Edit the `testApi.php` file (and related `.env` conte

#### Combo Task:

**Weight:** 100%

**Objective:** Edit the `testApi.php` file (and related `.env` content) to test your chosen API

**Details:**

- The code will need to be adjusted since your API differs from the one I chose
- You'll need to apply the API `.env` items to Heroku's config vars both `dev` and `prod`
- Be mindful of your API quota (may need to initially use a cached response until you get it working)

#### ≡ Image Prompt

**Weight:** 40%

**Details:**

Two screenshots are expected

1. Snippet of relevant code showing solution (with ucid/date comment) 2. Don't show any private details like the API key
2. Full output of executing the `testApi.php` (visit the proper file on Heroku dev after a manual deploy)

code



## heroku output



 Saved: 4/12/2025 9:18:47 PM

## ≡/ Url Prompt

**Weight: 20%**

### Details:

- Direct link to the file in the homework related branch from Github (should end in `.php`)
- Direct link to the file on Heroku Prod (Just grab the base prod url and manually enter the path to the file)

URL #1

[https://github.com/dyylan2018/dw347-IT201808M7-Homework/public\\_html/project/testApi.php](https://github.com/dyylan2018/dw347-IT201808M7-Homework/public_html/project/testApi.php)



UHL

<https://github.com/dyylan2018/dw>:



URL #2

<https://dw347-it202-008-prod-39957dda50b0.herokuapp.com/project/testApi.php>



UHL

<https://dw347-it202-008-prod-3995>



 Saved: 4/12/2025 9:18:47 PM

### ≡/ Text Prompt

**Weight: 40%**

**Details:**

Briefly explain what you needed to do in order to get this test working

Your Response:

I had to find an api that provided enough requests for one month. once found, I narrowed down the data I would use so the api wouldnt have to make several calls for different pieces of data. then i input the api host name, the endpoint, and the key into the proper files and updated the form. once done, i tested the api page to see if it worked.

 Saved: 4/12/2025 9:18:47 PM

## 100% Section #3: ( 2 pts.) Misc

### 100% Task #1 ( 0.67 pts.) - Github Details

**Combo Task:**

**Weight: 33.33%**

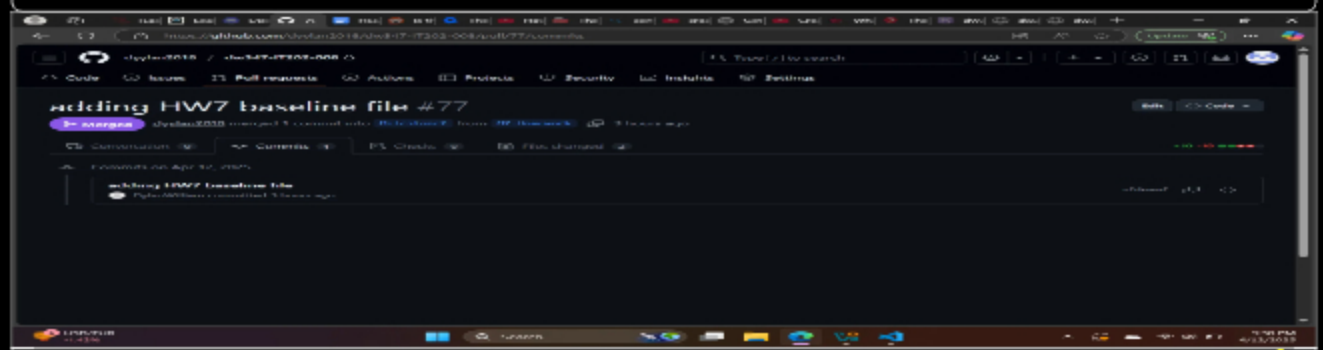
**Objective: Github Details**

 **Image Prompt**

**Weight: 60%**

**Details:**

From the Commits tab of the Pull Request screenshot the commit history Following minimum should be present



commits



Saved: 4/12/2025 9:32:43 PM

## Url Prompt

Weight: 40%

Details:

Include the link to the Pull Request (should end in `/pull/#`)

URL #1

<https://github.com/dyylan2018/dw347-IT202-008/pull/7>



URL

<https://github.com/dyylan2018/dw347-IT202-008/pull/7>



Saved: 4/12/2025 9:32:43 PM

100%

## Task #2 ( 0.67 pts.) - WakaTime - Activity

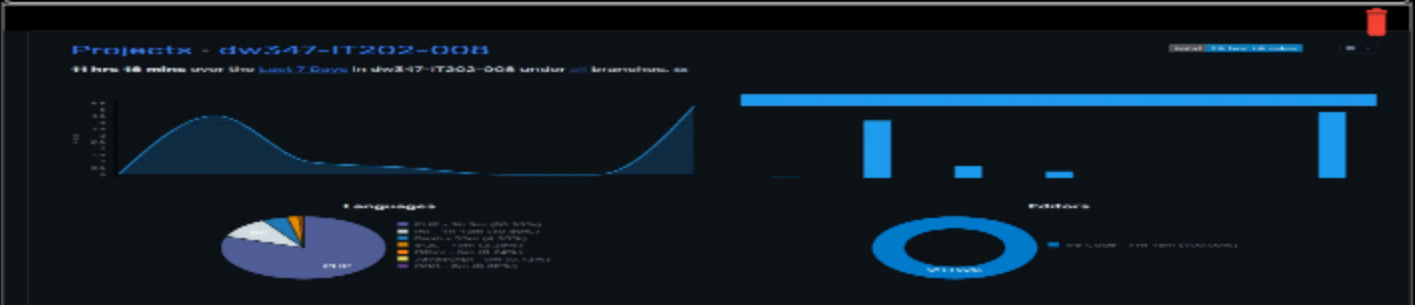
### Image Prompt

Weight: 33.33%

Objective: *WakaTime - Activity*

Details:

- Visit the WakaTime.com Dashboard
- Click `Projects` and find your repository
- Capture the overall time at the top that includes the repository name
- Capture the individual time at the bottom that includes the file time
- Note: The duration isn't relevant for the grade and the visual graphs aren't necessary



overall



Briefly answer the question (at least a few decent sentences)

Your Response:

inputting the api details into the code



Saved: 4/12/2025 9:39:16 PM



### Task #3 ( 0.33 pts.) - What was the hardest part of the assign

#### ≡ Text Prompt

**Weight:** 33.33%

**Objective:** *What was the hardest part of the assignment?*

**Details:**

Briefly answer the question (at least a few decent sentences)

Your Response:

solving all the errors that came up. fixing the files location and more.



Saved: 4/12/2025 9:39:41 PM