

Wprowadzenie

Ten dokument ma na celu opisać punkty końcowe (endpoints) komunikacji klient-serwer (API) dla projektu PZZ UŚ 2025. W tym dokumencie będą wskazane punkty końcowe, jakie dane i w jakiej formie one będą otrzymywali, jakie metody będą przejmowali i w jakiej formie będą zwracali odpowiedzi.

Ponieważ Flask podtrzymuje RESTful API ([link 1](#)¹, [link 2](#)²), będziemy korzystali z tych podstawowych funkcjonalności.

Punkty końcowe

<strona>/api/pselector

- **IN:** GET request
- **OUT:** JSON, zawierający nazwy kierunków / grup kierunków (np. Podyplomowe) w postaci:
{ "Nazwa kierunku": ID }
- **Przykład:**
{
 "Fizyka Medyczna II stopnia": 28687,
 "Informatyka I stopnia (niestacjonarna)": 6921,
 "Informatyka II stopnia": 6912,
 ...
}
- **Frontend:** Zwykły GET request z oczekiwaniem odpowiedzi
- **Backend:** Powinien w odpowiedni sposób sparsować kursy.json i zwrócić do frontendu odpowiedni JSON powyższej formy.
- **On error:** Nie przewidujemy errorów, jednak jeżeli kursy.json nie będzie dostępny, powiedzmy, że będzie zwracany pusty JSON-objekt: {}

<strona>/api/pselector?id_kier=<id>

- **IN:** GET request, gdzie id = id właściwego kierunku
- **OUT:** JSON, zawierający wszystkie lata, grupy i podgrupy kierunku lub błąd
- **Przykład (dla informatyki stosowanej):**
-> <strona>/api/pselector?id_kier=28698
<-

```
{  
  "IS-1 rok": {  
    "ID": 28699,  
    "Grupy": {  
      "IS-Gr.1(gr.01/1 rok, gr.02/1 rok)": {  
        "ID": 31429,  
        "Podgrupy": {  
          "IS/ 1 rok/ gr. 01": {  
            "ID": 28723,
```

¹ <https://flask-restful.readthedocs.io/en/latest/reqparse.html>

² <https://auth0.com/blog/developing-restful-apis-with-python-and-flask/>

```
"Podgrupy": {}
},
"IS/ 1 rok/ gr. 02": {
  "ID": 28724,
  "Podgrupy": {}
}
},
"IS-Gr.2(gr.03/1 rok, gr.04/1 rok)": {
  "ID": 31430,
  "Podgrupy": {
    "IS/ 1 rok/ gr. 03": {
      "ID": 28736,
      "Podgrupy": {}
    },
    "IS/ 1 rok/ gr. 04": {
      "ID": 28742,
      "Podgrupy": {}
    }
  },
  "IS-Gr.3(gr.05/1 rok, gr.06/1 rok)": {
    "ID": 31433,
    "Podgrupy": {
      "IS/ 1 rok/ gr. 05": {
        "ID": 28738,
        "Podgrupy": {}
      },
      "IS/ 1 rok/ gr. 06": {
        "ID": 28739,
        "Podgrupy": {}
      }
    },
    "IS-Gr.4(gr.07/1 rok, gr.08/1 rok)": {
      "ID": 53150,
      "Podgrupy": {
        "IS/ 1 rok/ gr. 07": {
          "ID": 53151,
          "Podgrupy": {}
        },
        "IS/ 1 rok/ gr. 08": {
          "ID": 53152,
          "Podgrupy": {}
        }
      },
      "IS-Gr.5(gr.09/1 rok, gr.10/1 rok)": {
        "ID": 85011,
        "Podgrupy": {
          "IS/ 1 rok/ gr. 09": {
            "ID": 85012,
            "Podgrupy": {}
          },

```

```

    "IS/ 1 rok/ gr. 10": {
      "ID": 85013,
      "Podgrupy": {}
    }
  },
  "IS-2 rok": {
    "ID": 28700,
    "Grupy": {
      "IS-Gr.1 (gr.01/2 rok, gr.02/2 rok)": {
        "ID": 31460,
        "Podgrupy": {
          "IS/2 rok/gr.01": {
            "ID": 28793,
            "Podgrupy": {}
          },
          "IS/2 rok/gr.02": {
            "ID": 28794,
            "Podgrupy": {}
          }
        }
      },
      "IS-Gr.2 (gr.03/2 rok, gr.04/2 rok)": {
        "ID": 31461,
        "Podgrupy": {
          "IS/2 rok/gr.03": {
            "ID": 28795,
            "Podgrupy": {}
          },
          "IS/2 rok/gr.04": {
            "ID": 28796,
            "Podgrupy": {}
          }
        }
      },
      "IS-Gr.3 (gr.05/2 rok, gr.06/2 rok)": {
        "ID": 31462,
        "Podgrupy": {
          "IS/2 rok/gr.05": {
            "ID": 28797,
            "Podgrupy": {}
          },
          "IS/2 rok/gr.06": {
            "ID": 55896,
            "Podgrupy": {}
          }
        }
      }
    }
  },
  "IS-3 rok": {
    "ID": 28701,

```

```

    "Grupy": {
      "IS/3 rok/gr.1": {
        "ID": 31353,
        "Podgrupy": {}
      },
      "IS/3 rok/gr.2": {
        "ID": 29379,
        "Podgrupy": {}
      },
      "IS/3 rok/gr.3": {
        "ID": 29377,
        "Podgrupy": {}
      },
      "IS/3 rok/gr.4": {
        "ID": 29374,
        "Podgrupy": {}
      }
    },
    "IS-4 rok": {
      "ID": 85354,
      "Grupy": {
        "IS/4 rok/gr.1": {
          "ID": 85355,
          "Podgrupy": {}
        },
        "IS/4 rok/gr.2": {
          "ID": 85356,
          "Podgrupy": {}
        },
        "IS/4 rok/gr.3": {
          "ID": 85357,
          "Podgrupy": {}
        },
        "IS/4 rok/gr.4": {
          "ID": 85358,
          "Podgrupy": {}
        }
      }
    }
  }
}

```

- **Frontend:** Ma przechowywać JSON kierunków po otrzymaniu poprzedniego zapytania. Ma wysłać odpowiedni ID zgodnie z wyborem użytkownika. Ma sparsować odpowiedź API z latami, grupami i podgrupami, aby dodać nazwy do odpowiednich elementów interfejsu — nie wysyłając zapytania za każdym razem do serwera zmniejszymy obciążenie naszego serwera. Także trzeba dodać specjalny przypadek dla błędu.
- **Backend:** Ma w odpowiedni sposób sparsować kursy.json, aby wysłać obiekt, zaczynając od "Lata" szukanego kierunku.
- **On error:** Ktoś może wprowadzić niewłaściwy id lub jakiś inny napis. Niezbędne jest zabezpieczenie (walidacja) zapytania, a już później zabezpieczone od *Key Error* sprawdzanie słownika. Jeżeli kierunek z takim ID nie zostanie znaleziony, trzeba

wysłać odpowiedź postaci:

```
{
  "error": "Nie znaleziono kierunku"
}
```

<strona>/api/pselector?<cokolwiek poza id_kier>

- **IN:** GET request
- **OUT:** HTTP Response 404 Not Found.

<strona>/api/getplan?for=<id>

- **IN:** GET request, gdzie id = id właściwej grupy/podgrupy
- **OUT:** JSON, zawierający listę, opisującą harmonogram zajęć, lub błąd, jeżeli planu dla tej grupy nie wykryto
- **Przykład (dla IS 3 rok 4 grupa):**
-> <strona>/api/getplan?for=29374

```
{
  "PN": {},
  "WT": {
    "Pracownia programowania zespołowego cz.1, lab. live online": {
      "Start": 0945,
      "End": 1115,
      "Prowadzący": "dr Katarzyna Schmidt",
      "ID_Prowadzący": 28623,
      "Sala": "",
      "Występowanie": "Semestr letni"
    }
  },
  "SR": {},
  ...
}
```

- **Uwaga 1:** Imię prowadzącego backend dostaje z HTML planu dla prowadzącego (chorę ale cóż. To jest zapytanie do uczelnianego planu z type=10 i id prowadzącego (jest na planie tygodnia dla grupy):
"http://plan.ii.us.edu.pl/plan.php?type=10&id=28623&winW=691&winH=816&loadBG=000000"
- **Uwaga 2:** Format godzin startu i końca zajęć jeszcze nie został ustawiony, ponieważ nie prowadziliśmy konwersacji z pozycji w przeglądarce na godziny.
- **Frontend:** Trzeba będzie odpowiednio przechowywać info o wybranym przedmiocie i podgrupie, aby wysłać zapytanie z prawidłowym ID. Trzeba będzie sparsować JSON, aby uzupełnić klocki z zajęciami. Także trzeba obsłużyć error.
- **Backend:** Trzeba wysłać zapytanie na
plan.ii.us.edu.pl/plan.php?type=0&id=<id>&winW=1694&winH=940&loadBG=000000
(winW i winH mogą ulec zmianie), przetworzyć odpowiedź, konwertując umieszczenie klocków z zajęciami w absolutnych pozycjach na godziny i dni (czyli trzeba będzie jakoś to obliczyć). Potem trzeba pobrać ID prowadzących i wysłać odpowiednie zapytania do planu prowadzących (type=10), aby dostać się do ich imienia. Zapakowany JSON trzeba zwrócić do frontendu.

- **On error:** Ktoś może wprowadzić niewłaściwy id lub jakiś inny napis. Niezbędne jest zabezpieczenie (walidacja) zapytania, a już później sprawdzić, czy plan rzeczywiście istnieje. Jeżeli plan dla ID nie zostanie znaleziony, trzeba wysłać odpowiedź postaci:
{
 "error": "Nie znaleziono planu. Może on jeszcze nie został umieszczony"
}

<strona>/api/getplan?<cokolwiek poza for>

- **IN:** GET request
- **OUT:** HTTP Response 404 Not Found.

<strona>/api/*

- **IN:** PUSH, PUT, DELETE, CONNECT etc requests
- **OUT:** HTTP Response 405 Method Not Allowed

Strony od poziomu Frontendu:

/ — main, strona z wyborem kierunku/roku/grupy etc + plan

/about — strona o projekcie

/privacy — strona z kilkoma słowami odnośnie polityki prywatności

[Navbar]

[Content]