
APE-learn Platform



Class Diagram-v1.0

Η Ομάδας μας και η Συμβολή της στο παρόν Τεχνικό Κείμενο

Μέλος	ΑΜ	Ρόλος
Πραμαντιώτης Χρήστος	1067457	Συντάκτης (Editor)
Πεντεσκούφης Παναγιώτης	1067470	Συνεργάτης (Contributor)
Τσιρίκος Αναστάσιος	1070926	Ομότιμος Κριτής (Peer Reviewer)
Αποστολόπουλος Γεώργιος	1067456	Ομότιμος Κριτής (Peer Reviewer)

Πληροφορίες σχετικά με το Class Diagram

- Με κίτρινο χρώμα αναπαρίστανται οι Κλάσεις που δηλώνουν backEnd operations με τη Βάση.
- Με γαλάζιο χρώμα υποδηλώνεται μία νέα προστιθέμενη Κλάση.
- Με άσπρο χρώμα αναπαρίστανται Κλάσεις GUI.
- Σε κάθε Κλάση το σύμβολο '-' δηλώνει κάποια μεταβλητή, ενώ το σύμβολο '+' δηλώνει κάποια μέθοδο που χρησιμοποιεί η Κλάση.

Για την εξαγωγή Διαγράμματος UML (για αναγνώριση μεταβλητών και συναρτήσεων και κατόπιν απευθείας πέρασμα στο Draw.io) χρησιμοποιήθηκε το παρακάτω Python script:

```
import os
```

```
import re
```

```
def parse_java_files(directory):
```

```
    classes_details = {}
```

```
    for root, dirs, files in os.walk(directory):
```

```
        for file in files:
```

```
            if file.endswith(".java"):
```

```
                filepath = os.path.join(root, file)
```

```
                with open(filepath, 'r', encoding='utf-8') as f:
```

```
                    content = f.read()
```

```
            # Regex to capture class name
```

```
            class_names = re.findall(r'\bclass\s+(\w+)', content)
```

```

if class_names:

    class_name = class_names[0]

    classes_details[class_name] = {"fields": [], "methods": []}

    # Regex to capture fields

    fields =
    re.findall(r'(?:(?:public|protected|private|static|\s)\s+[\w<>\[\]]+\s+(\w+);', content)

    classes_details[class_name]["fields"] = fields

    # Regex to capture methods

    methods =
    re.findall(r'(?:(?:public|protected|private|static|\s)\s+[\w<>\[\]]+\s+(\w+)\s*\(\s*.*?\s*\s*{', content)

    classes_details[class_name]["methods"] = methods

    return classes_details


def generate_plantuml(classes_details):

    plantuml_code = "@startuml\n"

    for class_name, details in classes_details.items():

        plantuml_code += f"class {class_name} {\n"

        for field in details["fields"]:

            plantuml_code += f"    - {field}\n"

        for method in details["methods"]:

            plantuml_code += f"    + {method}()\n"

```

```
plantuml_code += "}\n"
plantuml_code += "@enduml"
return plantuml_code

directory = r'C:\path\to\my\Project\src'
classes_details = parse_java_files(directory)
plantuml_code = generate_plantuml(classes_details)
print(plantuml_code)
```

Εδώ γίνονται εκτεταμένες χρήσεις από κανονικές εκφράσεις (regex) με σκοπό την αναγνώριση Κλάσεων, μεταβλητών και μεθόδων.

Στην επόμενη σελίδα βρίσκεται το Class Diagram. Σε περίπτωση που δεν είναι κάτι ευκρινές μπορείτε να επισκεφθείτε το profile του APE-learn στο GitHub:

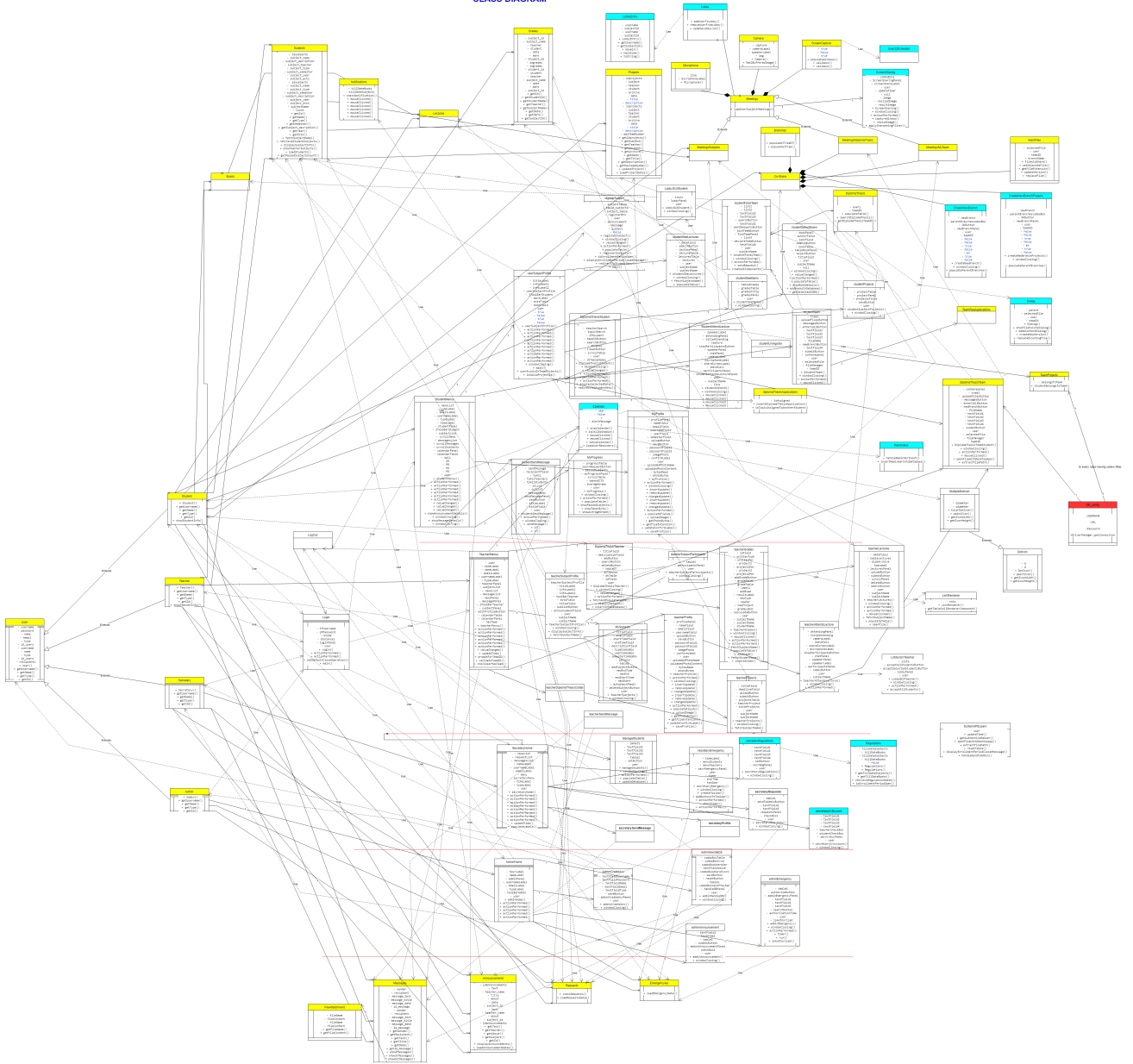
Στο Branch 'Third Deliverable':

- ✓ <https://github.com/prChristos/APE-learn/tree/Fourth-Deliverable>

ή στο Branch 'Diagrams':

- ✓ <https://github.com/prChristos/APE-learn/tree/Diagrams>

CLASS DIAGRAM



Εργαλεία που χρησιμοποιήθηκαν στο παρόν Τεχνικό Κείμενο

- Για τη συγγραφή, χρησιμοποιήθηκε το Microsoft Word, ενώ η μετατροπή του διαγράμματος Pert σε Pdf, πραγματοποιήθηκε μέσω του Latex, έτσι ώστε να επιτευχθεί καλύτερη ανάλυση.

- Για τη σχεδίαση του Class Diagram, χρησιμοποιήθηκε το Draw.io.