

## ΕΠΙΚΟΙΝΩΝΙΑ ΑΝΘΡΩΠΟΥ ΜΗΧΑΝΗΣ ΕΡΓΑΣΙΑ 3

Υλοποίηση τμήματος του ιστοχώρου του Υπουργείου  
Εργασίας και Κοινωνικών Υποθέσεων  
(<https://www.yrakp.gr/>)

ΜΕΛΗ:

ΑΝΤΥΠΙΑΣ ΙΟΡΔΑΝΗΣ(1115201700004)

ΓΙΑΝΝΙΟΣ ΝΙΚΟΛΑΟΣ(1115201700020)

## **ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ**

1. ΕΙΣΑΓΩΓΗ
2. ΣΧΕΔΙΑΣΤΙΚΕΣ ΕΠΙΛΟΓΕΣ
3. DATABASE
4. ΕΝΔΕΙΚΤΙΚΟΙ ΧΡΗΣΤΕΣ ΕΦΑΡΜΟΓΗΣ
5. DOCKER
6. ΤΡΕΞΙΜΟ LOCALLY
7. BACKEND
8. FRONTEND

## **ΕΙΣΑΓΩΓΗ**

Η εργασία υλοποιήθηκε με React στο frontend, Django στο backend και με βάση δεδομένων PostgreSQL, ενώ για το deployment χρησιμοποιήθηκε το docker-compose. Η εργασία ακολουθεί σε μεγάλο βαθμό τις προδιαγραφές της εκφώνησης. Δεν έχει υλοποιηθεί το BONUS ερώτημα.

## **ΣΧΕΔΙΑΣΤΙΚΕΣ ΕΠΙΛΟΓΕΣ**

Ο εργαζόμενος μπορεί να υλοποιήσει όλες τις φόρμες τις οποίες όμως πρέπει να εγκρίνει ο εργοδότης. Ο εργοδότης μπορεί να υλοποιήσει όλες τις φόρμες πέρα από την άδεια ειδικού σκοπού, ενώ μπορεί οποιαδήποτε στιγμή να εγκρίνει ή να απορρίψει φόρμες. Οι χρήστες έχουν τη δυνατότητα να αλλάζουν πέρα από τα άλλα στοιχεία τους την εταιρεία τους αλλά όχι username και ρόλο.

## **DATABASE**

Για να γίνει import το αρχείο της ΒΔ που περιλαμβάνουμε αρκούν:  
Δημιουργία της βάσης είτε από το terminal(psql) είτε με κάποιο database tool(πχ dbeaver):

### ΤΡΕΞΙΜΟ LOCALLY

Για να τρέξει η local db αντί της cloud db, σε περίπτωση που κάτι προκύψει με την online db που χρησιμοποιούμε αρκεί να μπει σε comment η πρώτη γραμμή του αρχείου backend/server/.env(DJANGO\_ENV=production -> // DJANGO\_ENV=production) και μετά:

```
CREATE ROLE qfsjgwmf WITH LOGIN PASSWORD 'fkSINUsGtKhCc-ZGkY8_oWXZJelaDdgA';
```

```
CREATE DATABASE qfsjgwmf;
```

Και στο terminal η εκτέλεση της εντολής:  
`pg_restore -d qfsjgwmf sdi1700004.sql`

### DOCKER:

Για να χρησιμοποιηθεί local db στο docker στο αρχείο backend/server/.env πέρα από την προηγούμενη τροποποίηση πρέπει επίσης να γίνει DOCKER=1 και μετά:

```
docker-compose build --no-cache  
docker-compose up --force-recreate --no-deps
```

```
pg_restore --verbose --host=localhost --port=5433 --username=qfsjgwmf --format=c --
dbname=qfsjgwmf sdi1700004.sql (με Password: fkSINUsgtKhCc-
ZGkY8_oWXZJelaDdgA)
```

## ΕΝΔΕΙΚΤΙΚΟΙ ΧΡΗΣΤΕΣ ΕΦΑΡΜΟΓΗΣ

### EMPLOYEE

```
{
  "username": "ni87",
  "password": "hardpassword"
},
```

### EMPLOYER

```
{
  "username": "ni878",
  "password": "easypassword"
}
```

## DOCKER

Για να τρέξει η εφαρμογή με το docker αρκούν οι εντολές:

```
docker-compose build --no-cache
docker-compose up --force-recreate --no-deps
```

Η εφαρμογή ακούει στο localhost:80. Σε περίπτωση που το port χρησιμοποιείται ήδη μπορεί να φανεί χρήσιμη η εντολή `sudo /etc/init.d/apache2 stop`.

## ΤΡΕΞΙΜΟ LOCALLY

### Backend:

In backend/server:

```
linux: sudo python3 -m venv env / Windows: py -m venv env
linux: source env/bin/activate / Windows: .\env\Scripts\activate
pip install -r ../requirements.txt
python manage.py runserver
```

### Frontend:

In frontend/react:

```
npm install
npm start
```

## BACKEND

Για το authenticate των χρηστών χρησιμοποιείται JWT token.

### API:

http://localhost:8000:

(register) POST: /authentication/register

example: {

```
"password": "hardpassword",
"email": "n7k@gmail.com",
"first_name": "n7is",
"last_name": "sdhj",
"amka": "1781",
"afm": "1871",
"username": "ni87",
"phone": "1127118",
"role": "employer",(employer/employee)
"has_child_under_12": "True",(boolean)
"company": "UoA"
```

}

(login) POST: /authentication/api/token/:

{

```
"username": "nikos",
"password": "hardpassword"
```

}

(logout) POST: /authentication/logout/blacklist/

GET: /authentication/profile -> gets logged in user's profile

POST: /forms/adeiaeidikoyskopoy:

{

```
"from_time": "2021-01-17",
"to_time": "2021-01-19",
"children": [{"age": 10, "school": "12o"}, {"age": 12, "school": "13o"}]
```

}

GET: /forms/adeiaeidikoyskopoy/<int:form\_id>

POST: /forms/adeiaeidikoyskopoy/<int:form\_id>

{

```
"status": -1 (-1: NOT_APPROVED, 0: PENDING, 1: APPROVED)
```

}

POST: /forms/adeiaergasias:

IF USER IS EMPLOYER:

{

```
"from_time": "2021-01-17",
"to_time": "2021-01-19",
"employee_username": "admin2"
}
IF USER IS EMPLOYEE:
{
  "from_time": "2021-01-17",
  "to_time": "2021-01-19"
}
GET: /forms/adeiaergasias/<int:form_id>
POST: /forms/adeiaergasias/<int:form_id>
{
  "status": -1 (-1: NOT_APPROVED, 0: PENDING, 1: APPROVED)
}
```

```
POST: /forms/anastoliergasias:
IF USER IS EMPLOYER:
{
  "from_time": "2021-01-17",
  "to_time": "2021-01-19",
  "employee_username": "admin2"
}
IF USER IS EMPLOYEE:
{
  "from_time": "2021-01-17",
  "to_time": "2021-01-19"
}
```

```
GET: /forms/anastoliergasias/<int:form_id>
POST: /forms/anastoliergasias/<int:form_id>
{
  "status": -1 (-1: NOT_APPROVED, 0: PENDING, 1: APPROVED)
}
```

```
POST: /forms/tilergasia:
IF USER IS EMPLOYER:
{
  "from_time": "2021-01-17",
  "to_time": "2021-01-19",
  "employee_username": "admin2"
}
IF USER IS EMPLOYEE:
{
  "from_time": "2021-01-17",
  "to_time": "2021-01-19"
}
```

```
GET: /forms/tilergasia/<int:form_id>
POST: /forms/tilergasia/<int:form_id>
```

```
{
  "status": -1 (-1: NOT_APPROVED, 0: PENDING, 1: APPROVED)
}
```

GET: /forms/myforms -> gets logged in user's forms

POST: /authentication/update\_profile

```
{
  "email": "ok@gmail.com",
  "first_name": "n7is889",
  "last_name": "sdhj880",
  "amka": "1781880",
  "afm": "1871880",
  "phone": "112711888",
}
```

POST: /authentication/update\_password

```
{
  "old_password": "hardpassword",
  "new_password": "easypassword",
  "new_rpassword": "easypassword"
}
```

POST: /appointments/create\_appointment:

IF NOT LOGGED IN

```
{
  "email": "nikos@gmail.com",
  "phone": "1",
  "date": "2021-01-31",
  "time": "18:00"
}
```

IF LOGGED IN

```
{
  "date": "2021-01-31",
  "time": "18:00"
}
```

GET: /appointments/myappointments

GET: /appointments/appointment\_by\_date

```
{
  "date": "2020-01-11"
}
```

## **FRONTEND:**

Όλες οι σελίδες περιλαμβάνουν ένα header που τους δίνει πρόσβαση στο menu και στις υπόλοιπες σελίδες του ιστότοπου, αλλά και στο προφίλ τους και ένα footer με τους τρόπους επικοινωνίας με το υπουργείο.

Για το frontend επιλέξαμε ένα μινιμαλιστικό theme, ώστε να μην κουράζεται ο χρήστης.

Σε κάθε σελίδα υπάρχει το path της σελίδας ώστε ο χρήστης να έχει τη δυνατότητα να πλοηγείται πιο εύκολα.

Αρκετές σελίδες που υπήρχαν στην αρχική σελίδα του υπουργείου αλλά δε ζητούνται στην άσκηση έχουν μπει ως links για να «γεμίσει» ο ιστόχωρος αλλά βγάζουν την ενημέρωση ότι είναι υπο-κατασκευή.

Στην αρχική σελίδα περιλαμβάνονται links για τις πιο δημοφιλείς σελίδες σε αυτό το διάστημα, καθώς και links προς τις σελίδες που αφορούν την κάθε ομάδα χρηστών.