



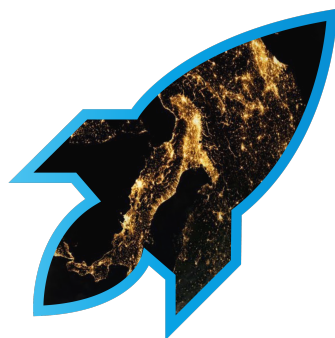
ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ  
Εθνικόν και Καποδιστριακόν  
Πανεπιστήμιον Αθηνών  
— ΙΔΡΥΘΕΝ ΤΟ 1837 —

ΤΕΧΝΟΛΟΓΙΕΣ ΕΦΑΡΜΟΓΩΝ ΔΙΑΔΙΚΤΥΟΥ

ΕΑΡΙΝΟ ΕΞΑΜΗΝΟ 2020

ΥΠΟΧΡΕΩΤΙΚΗ ΕΡΓΑΣΙΑ

# ΕΦΑΡΜΟΓΗ ΔΙΑΔΙΚΤΥΟΥ ΕΝΟΙΚΙΑΣΗΣ ΔΩΜΑΤΙΩΝ / ΚΑΤΟΙΚΙΩΝ



Αριθμός Μητρώου(ΑΜ):

**1115201700207**

**1115201700203**

Ονοματεπώνυμο:

Κωνσταντίνος ΓΕΩΡΓΙΟΥ

Λεωνίδας ΕΦΡΑΙΜ

ACADEMIC YEAR 2019-2020

---

# ΠΕΡΙΕΧΟΜΕΝΑ

<b>1</b>	<b>ΕΙΣΑΓΩΓΗ</b>	<b>3</b>
<b>2</b>	<b>ΕΓΚΑΤΑΣΤΑΣΗ</b>	<b>3</b>
2.1	BACKEND . . . . .	3
2.1.1	NodeJs . . . . .	3
2.1.2	Express . . . . .	4
2.1.3	PostgresQL . . . . .	4
2.2	FRONTEND . . . . .	6
2.2.1	React . . . . .	6
2.2.2	MVC . . . . .	6
<b>3</b>	<b>BACKEND</b>	<b>7</b>
3.1	ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ . . . . .	7
3.2	REST API BACKEND . . . . .	7
3.3	AUTHENTICATION . . . . .	8
<b>4</b>	<b>FRONTEND</b>	<b>8</b>
4.1	BOOTSTRAP . . . . .	8
4.2	REST API FRONTEND AXIOS . . . . .	8

# 1

---

## ΕΙΣΑΓΩΓΗ

# 2

---

## ΕΓΚΑΤΑΣΤΑΣΗ

### 2.1 BACKEND

Αρχικά μελετήσαμε διάφορα μοντέλα και τεχνολογίες όσο αφορά την δομή του server που θα κατασκευάζαμε. Τελικά καταλήξαμε στην παρακάτω δόμη όσο αφορά τις τεχνολογίες που θα χρησιμοποιηθούν στο νωτιαίο άκρο της εφαρμογής (Backend).

NodeJs

Express

PostgresQL

#### 2.1.1 NODEJS

Η απόφαση μας να χρησιμοποιήσουμε NodeJs ληφθηκε με βάση της δυνατότητες και την χρηστικότητα που μας παρέχει. Κατα κυριο λογο την επιλέξαμε γιατί μας δίνει την δυνατότητα να χρησιμοποιήσουμε μόνο μια γλώσσα προγραμματισμου (Javascript)

τόσο στο back-end όσο και στο front-end.Επίσης η Javascript είναι εύκολη στην κατανόηση και στην χρήση της.

### 2.1.2 EXPRESS

write your text here.Express

### 2.1.3 POSTGRESQL

Έχουμε επιλέξει την PostgreSQL για βάση επειδή είναι μια SQL like βάση δεδομένων στην οποία μπορούμε να γράψουμε πιο κατανοητά στον χρήστη ερωτήματα. Επίσης η PostgreSQL υποστηρίζει JSON αρχεία που μας διευκολύνει με την React και την NodeJs.

Η εγκατάσταση της PostgreSQL γίνεται με την παρακάτω εντολή

**sudo apt-get install postgresql**

Στην συνέχεια συνδεθηκαμε στην postgres

**psql -U <user name>**

και δημιουργσαμε την βάση με την εντολή

**CREATE DATABASE Breezbnb**

Αφου δημιουργήθηκε η βάση φορτώσαμε τους πίνακες και τις ρυθμίσεις της βάσης

**cat database/config.sql | psql -d <database name> -U <user name>**

**cat models/\* | psql -d <database name> -U <user name>**

**node database/generate-seed.js > database/seed.sql cat database/seed.sql  
| psql -d <database> -U <user>**

## Φόρτωση Βάσης στον Server

Για να φορτώσουμε την βάση στον server δημιουργήσαμε ένα αρχείο με κατάληξη

.env το οποίο αντιστοιχούσε στις κατάλληλες μεταβλητές με τα στοιχεία σύνδεσης της βάσης. Τα στοιχεία αυτά φορτώνοντε στις συνάρτησεις που είναι υπεθυνες για την επικοινωνία με την βάση

```
.env
1 # Port that server is listening
2 PORT = 5000
3 # Database credentials
4 DATABASE_URI = 'postgresql://postgres:apoel@localhost:5432/brezzebnb?currentSchema=public'
5 DATABASE_USER = "postgres"
6 DATABASE_USER_PW= "password"
7 DATABASE_HOST = "localhost"
8 DATABASE_PORT = 5432
9 DATABASE = "brezzebnb"
10
```

## Διαχείριση Βάσης Δεδομένων - OmniDB

Την διαχείριση της βάσης δεδομένων την κάναμε με την εφαρμογή OmniDB.

Group	Technology	Connection String	Server	Port	Database	User	Title	SSH Tunnel	SSH Server	SSH Port	SSH User	SSH Password	SSH Key	Actions
postgres			ec2-54-228-209-117.eu...	5432	d2mhib04g7bu15	odkqxcwhgacy	postgres			22				
postgres			ec2-54-228-209-117.eu...	5432	d2mhib04g7bu15	odkqxcwhgacy	breeze			22				

Καθορίζοντας τα στοιχεία σύνδεσης της βάσης μας μπορούμε να δούμε απο την διεπαφή τα δεδομένα των πινάκων μας καθώς και να τρέξουμε queries

The screenshot shows the OmniDB web interface. On the left, there's a tree view of the database structure. The main area displays a SQL query and its results.

**Query:**

```
SELECT t.user_id
       , t.user_name
       , t.first_name
       , t.last_name
       , t.email
       , t.password
       , t.phone
       , t.user_role
       , t.picture
FROM public.users t
```

**Results:**

	user_id	user_name	first_name	last_name	email	password	phone	user_role	picture
1	2c1c1d7b-7130-4ea3-93e3-7...	Jayden5	Jayden	Ward	jayden@ward.com	password	484129588	host	picture
2	38575966-2497-49be-96b4-6...	Imogen7	Imogen	Neal	imogen@neal.com	password	932970106	admin	picture
3	488b74fb-2399-4f0c-883a-fc...	Caleb4	Caleb	Clarke	caleb@clarke.com	password	941252561	ghost	picture
4	594b77b8-5332-494a-8b40-5...	Eleanor4	Eleanor	Goodwin	eleanor@goodwin.com	password	872587182	guest	picture
5	756ac978-a827-4c61-a458-e...	Hazel5	Hazel	Chaney	hazel@chaney.com	password	879596216	host	picture
6	7937232a-ba67-42bb-862c-4...	Finn8	Finn	Fritz	finn@fritz.com	password	363599057	host	picture
7	a22cc5f1-1162-4125-bd5f-d0...	Mia7	Mia	Cline	mia@cline.com	password	223960330	ghost	picture
8	bfcf2754-55b5-452c-b683-cb...	Violet3	Violet	Trujillo	violet@trujillo.com	password	799307740	host	picture
9	dc018063-decd-4a52-93f5-0...	Isla9	Isla	Hooper	isla@hooper.com	password	194803206	host	picture
10	fbbf63de-9732-4936-9319-3...	Maevae3	Maevae	Chaney	maevae@chaney.com	password	596639463	host	picture

## 2.2 FRONTEND

Όσο αφορά το μετωπιαίο ακρό (frontend) αποφασίσαμε να χρησιμοποιήσουμε React η οποία είναι JavaScript βιβλιοθήκη που μας βοηθά να αναπτύξουμε μια UserInterface εφαρμογή. Επίσης χρησιμοποιήσαμε το μοντέλο MVC για όσο αφορά την διαχείριση της εφαρμογής.

### 2.2.1 REACT

Η React έχει αναπτυχθεί από το Facebook και το community η οποία μας είναι σε Javascript

### 2.2.2 MVC

write your text here.MVC

---

## BACKEND

### 3.1 ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ

Στην βάση δεδομένων δημιουργήσαμε τους παρακτω πίνακες:

listings (Πίνακας με τα καταληματα)

Rentalsreserved (Πίνακας με τις κρατήσεις)

messages(Πίνακας με τα μηνύματα)

reviews(Πίνακας με τις κριτικές)

users(Πίνακας με του χρήστες)

Επίσης χρησιμοποισαμε ένα configuration για της ρυθμίσεις στην βάση Εκεί ρυθμίσαμε το format της ημερομηνίας έτσι ώστε να είναι το Ευρωπαϊκό, καθορίσαμε enums για τα user roles και τα property types και φορτώσαμε το uuid ossf το οποίο μας κάνει generate μοναδικά hash έτσι ώστε να τα χρησιμοποιούμε για primary key σε διάφορους πίνακες

### 3.2 REST API BACKEND

Για την επικοινωνία με την βάση δεδομένων χρησιμοποιήσαμε REST API. Στην ουσία είναι συναρτήσεις οι οποίες κάνουν διάφορα queries στην βάση για να πάρουν κάποια δεδομένα και να τα επιστρέψουν στον χρήστη. Για να κληθεί η συνάρτηση πρέπει να κάρουμε ένα request στον αντιστοιχο σύνδεσμο και να του δώσουμε τις σωστές

παραμέτρους στην συνάρτηση

Για παράδειγμα αν θέλουμε να δούμε τα στοιχεία κάποιου διαμερίσματος το οποίο είναι διαθέσιμο προς ενοικίαση θα πρέπει να κάνουμε ένα `get request` στο σύνδεσμο `localhost:5000/listings/ae867f1c-1b93-495e-9a32-d26dea7ac749`

οπου το `ae867f1c-1b93-495e-9a32-d26dea7ac749` είναι το `id` του καταλήματος.

Το `request` αυτό θα καλέσει την κατάλληλη συνάρτηση η οποία θα εκτελέσει ένα `SELECT query` στην βάση για να πάρει τα στοιχεία του καταλήματος και να τα επιστρέψει σε μορφή `JSON`

### 3.3 AUTHENTICATION

write you text here for authentication

# 4

---

## FRONTEND

### 4.1 BOOTSTRAP

write you text here for BOOTSTRAP

### 4.2 REST API FRONTEND AXIOS

write you text here for REST API FRONTEND AXIOS