



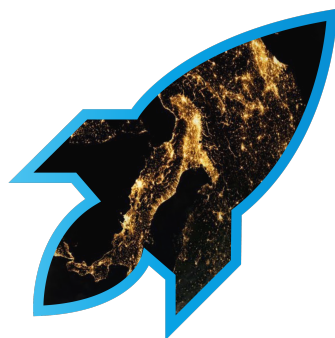
ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ
Εθνικόν και Καποδιστριακόν
Πανεπιστήμιον Αθηνών
— ΙΔΡΥΘΕΝ ΤΟ 1837 —

ΤΕΧΝΟΛΟΓΙΕΣ ΕΦΑΡΜΟΓΩΝ ΔΙΑΔΙΚΤΥΟΥ

ΕΑΡΙΝΟ ΕΞΑΜΗΝΟ 2020

ΥΠΟΧΡΕΩΤΙΚΗ ΕΡΓΑΣΙΑ

ΕΦΑΡΜΟΓΗ ΔΙΑΔΙΚΤΥΟΥ ΕΝΟΙΚΙΑΣΗΣ ΔΩΜΑΤΙΩΝ / ΚΑΤΟΙΚΙΩΝ



Αριθμός Μητρώου(ΑΜ):

1115201700207

1115201700203

Ονοματεπώνυμο:

Κωνσταντίνος ΓΕΩΡΓΙΟΥ

Λεωνίδας ΕΦΡΑΙΜ

ΑΚΑΔΗΜΑΪΚΗ ΧΡΟΝΙΑ 2019-2020

ΠΕΡΙΕΧΟΜΕΝΑ

1	ΕΙΣΑΓΩΓΗ	3
2	ΕΓΚΑΤΑΣΤΑΣΗ	4
2.1	BACKEND	4
2.1.1	NodeJs	4
2.1.2	Express	5
2.1.3	PostgresQL	5
2.2	FRONTEND	7
2.2.1	React	7
2.2.2	MVC	7
2.3	ΔΗΜΙΟΥΡΓΙΑ ΔΙΑΧΕΙΡΗΣΤΗ ΣΕΛΙΔΑΣ(ADMIN)	7
2.4	ΠΑΓΚΟΣΜΙΟΣ ΙΣΤΟΣ (HEROKU APP)	7
3	BACKEND	9
3.1	ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ	9
3.2	REST API BACKEND	9
3.3	AUTHENTICATION	10
4	FRONTEND	10
4.1	BOOTSTRAP	10
4.2	REST API FRONTEND AXIOS	10

ΕΙΣΑΓΩΓΗ

Στο πλαίσιο του μαθήματος κληθήκαμε να υλοποιήσουμε μια διαδικτυακή εφαρμογή που είχε σκοπό την ενοικίαση δωματίων και κατοικιών. Επειδή μας δόθηκε μεγάλος βαθμός ελευθερίας όσον αφορά την υλοποίηση και τη σχεδίαση της εφαρμογής, κάναμε μια μελέτη στο διαδίκτυο με βάση τις ενότητες και τα αντικείμενα που διδαχθήκαμε στο μάθημα, με σκοπό να βρούμε τις πιο κατάλληλες τεχνολογίες που θα χρησιμοποιούσαμε για την υλοποίηση μας.

ΕΓΚΑΤΑΣΤΑΣΗ

2.1 BACKEND

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

NodeJs

Express

PostgresQL

2.1.1 NODEJS

Η απόφασή μας να χρησιμοποιήσουμε NodeJs λήφθηκε με βάση τις δυνατότητες και τη χρηστικότητα που μας παρέχει. Κατά κύριο λόγο την επιλέξαμε γιατί μας δίνει τη δυνατότητα να χρησιμοποιήσουμε μόνο μια γλώσσα προγραμματισμού (Javascript) τόσο στο back-end όσο και στο front-end. Επίσης η Javascript είναι εύκολη στην κατανόηση και στη χρήση της. Η εγκατάσταση της NodeJs γίνεται με την εντολή **sudo apt install nodejs**

Επίσης χρειάζεται και η εγκατάσταση της Node Package Manager (NPM) η οποία είναι μια βιβλιοθήκη ανοιχτού κώδικα, που έχει Node.js πακέτα και γίνεται εγκατάσταση με την εντολή **sudo apt install npm**

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 ερωτήματα στον χρήστη. 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

Στη συνέχεια συνδεθήκαμε στην postgres

psql -U <user name>

και δημιουργήσαμε τη βάση με την εντολή

CREATE DATABASE Breezbnb

Αφού 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 τη βάση στον server δημιουργήσαμε ένα αρχείο με κατάληξη .env το οποίο αντιστοιχούσε στις κατάλληλες μεταβλητές με τα στοιχεία σύνδεσης της βάσης. Τα στοιχεία αυτά φορτώνονται στις συναρτήσεις που είναι υπεύθυνες για την επικοινωνία με τη 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

Τη διαχείριση της βάσης δεδομένων την κάνουμε 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
postgresql	postgresql		ec2-54-228-209-117.eu...	5432	d2mhib04g78u15	odqxycwhgtgacy	postgres	<input type="checkbox"/>		22				
postgresql	postgresql		ec2-54-228-209-117.eu...	5432	d2mhib04g78u15	odqxycwhgtgacy	brezze	<input type="checkbox"/>		22				

Καθορίζοντας τα στοιχεία σύνδεσης της βάσης μας, μπορούμε να δούμε από τη 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 is divided into a console for SQL queries and a results pane showing the output of the query.

SQL Query:

```

1 SELECT t.user_id
2       , t.user_name
3       , t.first_name
4       , t.last_name
5       , t.email
6       , t.password
7       , t.phone
8       , t.user_role
9       , t.picture
10 FROM public.users t

```

Query Results:

user_id	user_name	first_name	last_name	email	password	phone	user_role	picture
2c1c1d76-7130-4ea3-93e3-7...	Jayden5	Jayden	Ward	jayden@ward.com	password	484129588	host	picture
38575966-2497-49be-96b4-6...	Imogen7	Imogen	Neal	imogen@neal.com	password	932970106	admin	picture
488b74fb-2399-4f0c-883a-fc...	caleb4	Caleb	Clarke	caleb@clarke.com	password	941252561	ghost	picture
594b77b8-5332-494a-8b40-5...	eleanor4	Eleanor	Goodwin	eleanor@goodwin.com	password	872587182	guest	picture
756ac978-a827-4c61-a458-e...	hazel5	Hazel	Chaney	hazel@chaney.com	password	879596216	host	picture
793732a-ba67-42bb-8b2c-4...	finn8	Finn	Fritz	finn@fritz.com	password	363599057	host	picture
a22c5f1-1162-4125-bd5-d0...	mia7	Mia	Cline	mia@cline.com	password	223960330	ghost	picture
bfc2754-55b5-452c-b683-cb...	violet3	Violet	Trujillo	violet@trujillo.com	password	799307740	host	picture
dc018063-dedd-4a52-93f5-0...	isla9	Isla	Hooper	isla@hooper.com	password	194803206	host	picture
fbbf33de-9732-4936-9319-3...	maeve3	Maeve	Chaney	maeve@chaney.com	password	596639483	host	picture

Properties pane shows:

Property	Value
Database	d2mhib04g78u15
Schema	public
Table	users
OID	24139610
Owner	odqxycwhgtgacy
Size	8192 bytes

2.2 FRONTEND

Όσον αφορά το μετωπιαίο άκρο (frontend) αποφασίσαμε να χρησιμοποιήσουμε React, η οποία είναι JavaScript βιβλιοθήκη που μας βοηθά να αναπτύξουμε μια UserInterface εφαρμογή. Επίσης χρησιμοποιήσαμε το μοντέλο MVC όσον αφορά τη διαχείριση της εφαρμογής.

2.2.1 REACT

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

2.2.2 MVC

write your text here.MVC

2.3 ΔΗΜΙΟΥΡΓΙΑ ΔΙΑΧΕΙΡΗΣΤΗ ΣΕΛΙΔΑΣ (ADMIN)

Για την δημιουργία του διαχειρηστή της σελίδας φτιάξαμε το αρχείο database/admin.js το οποίο ορίζει τα στοιχεία του admin της σελίδας και στην συνέχεια τα καταχωρεί στην βάση με την εντολή

node database/admin.js

Τα στοιχεία σύνδεσης του διαχειρηστή είναι

USERNAME:**admin**

PASSWORD:**admin**

2.4 ΠΑΓΚΟΣΜΙΟΣ ΙΣΤΟΣ (HEROKU APP)

Παράλληλα με την τοπική εφαρμογή, ανεβάσαμε στον παγκόσμιο ιστό την εφαρμογή μας, με τη βοήθεια του Heroku app. Το Heroku είναι μια cloud πλατφόρμα η οποία μας παρέχει δωρεάν hosting για τον server μας. Έτσι λοιπόν ανεβάσαμε τους 2 server μας (Backend - Frontend) καθώς και τη βάση μας στους παρακάτω συνδέσμους

BACK-END

breezebnb.herokuapp.com

WEB SERVER

breezebnb.herokuapp.com NA MPI TO FRONT END

BACKEND

3.1 ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ

Στη βάση δεδομένων δημιουργήσαμε τους παρακάτω πίνακες:

listings (Πίνακας με τα καταλύματα)

Rentalsreserved (Πίνακας με τις κρατήσεις)

messages(Πίνακας με τα μηνύματα)

reviews(Πίνακας με τις κριτικές)

users(Πίνακας με τους χρήστες)

Επίσης χρησιμοποιήσαμε ένα configuration για τις ρυθμίσεις στη 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 χρησιμοποιήσουμε για primary key σε διάφορους πίνακες

3.2 REST API BACKEND

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

παραμέτρους στη 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`

όπου το `ae867f1c-1b93-495e-9a32-d26dea7ac749` είναι το `id` του καταλύματος.

Το `request` αυτό θα καλέσει την κατάλληλη συνάρτηση η οποία θα εκτελέσει ένα `SELECT query` στη βάση για να πάρει τα στοιχεία του καταλύματος 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