

# ΥΣ14 - Τεχνολογίες Εφαρμογών Διαδικτύου

Χειμερινό εξάμηνο 2019-20



Ιωάννης Κατσαρός: 1115201500234  
Χριστίνα Κατσαρλίνου: 1115201500068

[Αρχιτεκτονική Συστήματος](#)

[Στόχος της εργασίας](#)

[Λεπτομέρειες Αρχιτεκτονικής Συστήματος](#)

[Σχετικά με το Backend](#)

[Λειτουργία Backend](#)

[Βάση Δεδομένων](#)

[Ρόλοι των Users](#)

[Διαχείριση σφαλμάτων](#)

[Γέμισμα της Βάσης](#)

[Σχετικά με το Frontend](#)

[Αποτελέσματα](#)

[API Documentation](#)

[Base URL](#)

[API Requests & Responses](#)

[Frontend Screenshots](#)

## Αρχιτεκτονική Συστήματος

Η υλοποίηση του **Backend** έγινε βάσει **Java** και **Spring Boot**, ενώ η υλοποίηση του **Frontend** έγινε βάσει **Angular**. Ως βάση δεδομένων χρησιμοποιήθηκε η **MySQL** ενώ η επικοινωνία μεταξύ του front και του back γίνεται μέσω REST API. Το authentication και authorization των users και των ρόλων τους γίνονται χρησιμοποιώντας JSON Web Tokens.

## Στόχος της εργασίας

Στόχος της εργασίας είναι η υλοποίηση ολοκληρωμένου συστήματος καταχώρισης δωματίων / κατοικιών από μέλη (οικοδεσπότες) της πλατφόρμας, αναζήτησης διαθέσιμων χώρων για κράτηση από απλούς guests ή εγγεγραμμένους χρήστες και η διαχείριση των μελών του συστήματος από τους ίδιους τους διαχειριστές της πλατφόρμας. Οι χρήστες έχουν πρόσβαση στην εφαρμογή μέσω σύγχρονου φυλλομετρητή παγκόσμιου ιστού (web browser).

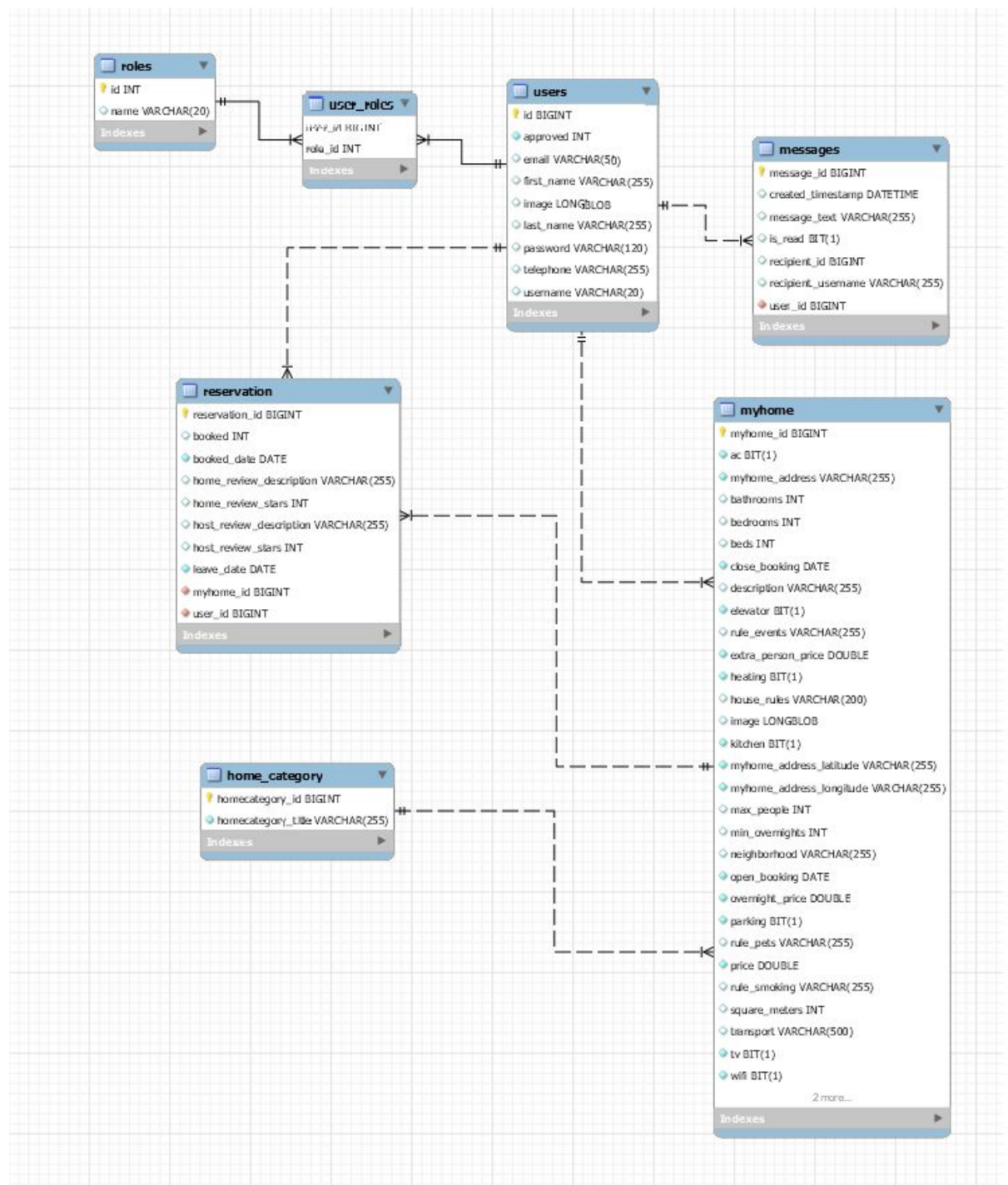
# Λεπτομέρειες Αρχιτεκτονικής Συστήματος

## Σχετικά με το Backend

### Λειτουργία Backend

Το Backend κομμάτι έχει υλοποιηθεί σε Java και Spring Boot και χρησιμοποιείται σχεσιακή βάση δεδομένων MySQL, ενώ παράλληλα υποστηρίζει ένα RESTful Application Programming Interface (API) για τη διαχείριση των δεδομένων του HomeRent. Για authorization και authentication έχουν δημιουργηθεί κατηγορίες χρήστη (Ρόλοι) και χρησιμοποιείται το Spring Security το οποίο παράγει μοναδικά Json Web Tokens που στέλνονται στο frontend με κάθε σωστή σύνδεση χρήστη και το οποίο χρησιμοποιείται προκειμένου να μπορεί έπειτα το frontend να κάνει τα requests προς το backend. Στην περίπτωση που ένας χρήστης προσπαθήσει να στείλει κάποιο request προς κάποιο endpoint στο οποίο δεν πρέπει να έχει πρόσβαση λόγω του Ρόλου του, η απάντηση του backend είναι forbidden. Το application είναι δομημένο με MVC και έχουν υλοποιηθεί τα κατάλληλα microServices για την επικοινωνία των layers του MVC και για την ευκολότερη επεκτασιμότητα του προγράμματος, όπως και έχει γίνει ο κατάλληλος διαχωρισμός των controllers και του μοντέλου που χρησιμοποιείται. Το front στα requests του δεν μιλάει απευθείας με την βάση δεδομένων, αλλά για λόγους ασφαλείας γίνεται ο μετασχηματισμός του request που στάλθηκε από το front ή που θα σταλεί στο front (ανάλογα του τι είδους request έγινε) σε Data transfer object και μέσω αυτού μετά γίνεται η επικοινωνία με την database καλώντας το κατάλληλο service, μετασχηματίζοντας την απάντηση του service σε Data transfer object και έπειτα επιστρέφοντας το σαν απάντηση στο front. Ακόμα ανάλογα με το format που θα δώσει ο χρήστης (admin) σε export endpoints (δηλαδή αν δοθεί ?format=xml) τότε η απάντηση που επιστρέφεται από το backend είναι σε μορφή xml, αλλιώς αν δοθεί ?format=json η απάντηση που επιστρέφεται είναι σε json μορφή την οποία παρουσιάζει το frontend. Για το Build του project έχει χρησιμοποιηθεί το Apache Maven.

## Βάση Δεδομένων



## Ρόλοι των Users

Δυνατότητα σύνδεσης χρηστών με βάση διαφορετικούς ρόλους που θα ορίζονται από τον λογαριασμό που θα συνδεθούν.

Admin: Διαχειριστής πλατφόρμας που ορίζεται ως ROLE\_ADMIN

Host: Οικοδεσπότης που ορίζεται ως ROLE\_MODERATOR

Tenant: Ενοικιαστής που ορίζεται ως ROLE\_USER

Guest: Απλός user που δεν είναι εγγεγραμμένος

## Διαχείριση σφαλμάτων

Οι κλήσεις στο API επιστρέφουν τα κατάλληλα HTTP Status Codes σε περίπτωση σφάλματος, όπως, ενδεικτικά φαίνονται στον ακόλουθο πίνακα.

401	Not authorized	Σε περίπτωση που ο χρήστης δεν έχει τη δικαιοδοσία να εκτελέσει μια ενέργεια.
404	Not found	Σε περίπτωση που ο ζητούμενος πόρος δεν υπάρχει.
400	Bad request	Σε περίπτωση που τα δεδομένα εισόδου δεν είναι έγκυρα (π.χ. κενό υποχρεωτικό πεδίο).

## Γέμισμα της Βάσης

Για το γέμισμα της βάσης δεδομένων δημιουργήθηκε ένα php script το οποίο παράγει έναν διαχειριστή, έναν user (για test της πλατφόρμας), 100 διαφορετικούς hosts και έναν random αριθμό από σπίτια για κάθε host. Επίσης τα σπίτια έχουν random τιμές στα πεδία τους οπότε είναι εντελώς διαφορετικά μεταξύ τους. Το php script έχει δοθεί και βρίσκεται στα files της πλατφόρμας. Υλοποιήθηκε σε php καθώς ανέβηκε ενδεικτικά σε έναν server για να έχουμε και οι 2 απομακρυσμένα απευθείας σύνδεση σε random sql τιμές με κάθε ανανέωση του backend.

## Σχετικά με το Frontend

Το Frontend κομμάτι έχει υλοποιηθεί σε TypeScript και Angular και δομείται με φακέλους που περιέχουν ξεχωριστά τις λειτουργίες (javascript) και το σχεδιαστικό κομμάτι (html/css) όπως και γίνεται καταμερισμός της δομής του project με βάση τα components και την λειτουργία κάθε component. Η πιστοποίηση των χρηστών γίνεται ανταλλάσσοντας ένα json web token και για κάθε λειτουργία του front γίνεται κλήση ενός endpoint από το API όπου επιστρέφει το αναμενόμενο result.

## Αποτελέσματα

Έχουν υλοποιηθεί όλες οι απαιτήσεις του project.

## API Documentation

### Base URL

Το REST API θα είναι διαθέσιμο στο ακόλουθο base URL για όλες τις εργασίες του συστήματος: <https://localhost:8080/api>

Τα επιμέρους Resources (REST endpoints) που θα διατίθενται μέσω του API είναι προσβάσιμα μέσω του παραπάνω base URL, ως εξής: {baseURL}/{path-to-resource}

## API Requests & Responses

---

API Documentation

---

SAMPLE BODY REQUEST:

/api/auth/signup

POST

```
{
  "username": "admin",
  "firstname": "admin",
  "lastname": "admin",
  "email": "admin@homerent.com",
  "role": ["admin"],
  "password": "123456",
  "telephone": "+306911111111"
```

```
}
```

NOTE: We have 3 roles: admin (the admin), mod (the host), user (the tenant)

SAMPLE RESPONSES:

```
{
  "message": "User registered successfully!"
}
```

```
{
  "message": "Error: Username is already taken!"
}
```

```
{
  "message": "Error: Email is already taken!"
}
```

```
{
  "message": "User registered successfully but must be approved by an Administrator!"
}
```

-----  
SAMPLE BODY REQUEST:

POST

/api/auth/signin

```
{
  "username": "admin",
  "password": "123456"
}
```

SAMPLE RESPONSE:

```
{
  "id": 1,
  "username": "admin",
  "firstName": "admin",
  "lastName": "admin",
  "email": "admin@homerent.com",
  "telephone": "+30691111111",
  "approved": "1",
  "roles": [
    "ROLE_ADMIN"
  ],
  "tokenType": "Bearer",
  "accessToken":
"eyJhbGciOiJIUzUxMiJ9.eyJzdWUiOiJhZG1pbilzImIhdCI6MTU5ODY1Mzg4MSwiZXhwIjoxNk4NzQwMjgxfQ.Inrcak4ODx0emZ5xJI7WGDSorc4hk33zGddSAskomognPwmLgdAZ9Rdby8q8K_jBnNTkda1oNdEx2pEAVfXSg"
```

```
}
```

-----  
ADMINS ONLY:

GET

/api/admin/users

SAMPLE RESPONSE:

```
[  
  {  
    "id": 1,  
    "username": "admin",  
    "firstname": "admin",  
    "lastname": "admin",  
    "email": "admin@homerent.com",  
    "telephone": "+306911111111",  
    "approved": 1,  
    "image": null  
  },  
  {  
    "id": 2,  
    "username": "nikos",  
    "firstname": "Nikos",  
    "lastname": "Papadopoulos",  
    "email": "nikos@homerent.com",  
    "telephone": "+306911111111",  
    "approved": 0,  
    "image": null  
  },  
  {  
    "id": 3,  
    "username": "giorgos",  
    "firstname": "Giorgos",  
    "lastname": "Papathanasiou",  
    "email": "giorgos@homerent.com",  
    "telephone": "+306911111111",  
    "approved": 1,  
    "image": null  
  },  
  {  
    "id": 4,  
    "username": "user",  
    "firstname": "Sir",  
    "lastname": "User",  
    "email": "user@homerent.com",  
    "telephone": "+306911111111",  
    "approved": 1,  
  }  
]
```



```
    "image": null
  }
]
```

---

ADMINS ONLY:

GET

/api/admin/users/{id}

SAMPLE REQUEST:

localhost:8080/api/admin/users/1

SAMPLE RESPONSE:

```
{
  "id" : 1,
  "username" : "admin",
  "firstname" : "admin",
  "lastname" : "admin",
  "email" : "admin@homerent.com",
  "telephone" : "+306911111111",
  "approved" : 1,
  "image" : null
}
```

---

ADMINS ONLY:

PUT

/api/admin/users

SAMPLE BODY REQUEST: (right now you can change only firstname and lastname)

```
{
  "id" : 1,
  "username" : "admin",
  "firstname" : "THIS IS MY NEW NAME",
  "lastname" : "admin",
  "email" : "admin@homerent.com",
  "telephone" : "+306911111111",
  "approved" : 1,
  "image" : null
}
```

SAMPLE RESPONSE: (the changed user)

```
{
  "id" : 1,
  "username" : "admin",
  "firstname" : "THIS IS MY NEW NAME",
  "lastname" : "admin",
  "email" : "admin@homerent.com",
```

```
"telephone" : "+306911111111",
"approved" : 1,
"image" : null
}
```

-----  
VISIBLE FOR ALL AUTHENTICATED:

GET

localhost:8080/api/secure/user

SAMPLE BODY REQUEST:

```
{
  "id" : 1,
  "username" : "admin",
  "firstname" : "admin",
  "lastname" : "admin",
  "email" : "admin@homerent.com",
  "telephone" : "+306911111111",
  "approved" : 1,
  "image" : null
}
```

-----  
VISIBLE FOR ALL AUTHENTICATED:

PUT

NOTE: if you add "password": "newpassword" will change the password too. If you don't add it in json, password will stay the same as before.

localhost:8080/api/secure/user/update

SAMPLE BODY REQUEST:

```
{
  "id" : 1,
  "username" : "admin",
  "firstname" : "THIS IS MY NEW NAME",
  "lastname" : "admin",
  "email" : "admin@homerent.com",
  "telephone" : "+306911111111",
  "approved" : 1,
  "image" : null
}
```

SAMPLE RESPONSE: (the changed user)

```
{
  "id" : 1,
  "username" : "admin",
  "firstname" : "THIS IS MY NEW NAME",
```

```
"lastname" : "admin",  
"email" : "admin@homerent.com",  
"telephone" : "+306911111111",  
"approved" : 1,  
"image" : null  
}
```

-----  
VISIBLE FOR ALL AUTHENTICATED:

DELETE  
localhost:8080/api/secure/user/delete

SAMPLE REQUEST:  
localhost:8080/api/secure/user/delete

SAMPLE RESPONSE:  
{"Status": "Successful Deletion"}

-----  
ADMINS ONLY:  
DELETE  
/api/admin/users/{id}

SAMPLE REQUEST:  
/api/admin/users/1

SAMPLE RESPONSE:  
{"Status": "Successful Deletion"}

-----  
ADMINS ONLY:  
POST  
/api/admin/users/{id}/approve

SAMPLE REQUEST:  
/api/admin/users/5/approve

SAMPLE RESPONSE:  
{"Status": "User Approved"}

-----  
ADMINS ONLY:  
GET  
localhost:8080/api/admin/users/unapproved

SAMPLE RESPONSE:

```
[  
  {  
    "id": 2,
```

```
    "username": "nikos",
    "firstname": "Nikos",
    "lastname": "Papadopoulos",
    "email": "nikos@homerent.com",
    "telephone": "+306911111111",
    "approved": 0,
    "image": null
  }
]
```

---

HOSTS ONLY:

GET

/api/host/{userId}/homes

SAMPLE REQUEST:

/api/host/1/homes

SAMPLE RESPONSE 1:

```
[ {
  "id" : 1,
  "ownerId" : 1,
  "ownerUsername" : "admin",
  "reservations" : [ ],
  "openBooking" : "2020-01-01",
  "closeBooking" : "2020-10-01",
  "price" : 50.0,
  "address" : "Aghsilaou",
  "latitude" : "0.543545",
  "longitude" : "0.553545",
  "description" : "It is a nice place to stay",
  "squareMeters" : 78,
  "overnightPrice" : 50.0,
  "extraPersonPrice" : 100.0,
  "maxPeople" : 5,
  "minOvernights" : 1,
  "beds" : 3,
  "bathrooms" : 4,
  "bedrooms" : 3,
  "transport" : "1",
  "neighborhood" : "0",
  "houseRules" : "There are not house rules",
  "ac" : false,
  "elevator" : false,
  "heating" : false,
  "kitchen" : false,
  "parking" : false,
```

```
"tv" : false,  
"wifi" : false,  
"smoking" : "Yes",  
"pets" : "Yes",  
"events" : "No"  
}]
```

#### SAMPLE RESPONSE 2:

```
{  
  "message": "Host Isn't approved yet by administrator"  
}
```

#### HOSTS ONLY:

POST

/api/home/new

#### SAMPLE REQUEST:

localhost:8080/api/host/home/new

In Body:

```
{  
  "ownerId" : 3,  
  "ownerUsername" : "giorgos",  
  "reservations" : [ ],  
  "openBooking" : "2020-01-01",  
  "closeBooking" : "2020-10-01",  
  "image" : null,  
  "price" : 50.0,  
  "address" : "Aghsilaou",  
  "latitude" : "0.543545",  
  "longitude" : "0.553545",  
  "homeCategory" : {  
    "homeCategoryTitle" : "Shared Room"  
  },  
  "description" : "It is a nice place to stay",  
  "squareMeters" : 78,  
  "overnightPrice" : 50.0,  
  "extraPersonPrice" : 100.0,  
  "maxPeople" : 5,  
  "minOvernights" : 1,  
  "beds" : 3,  
  "bathrooms" : 4,  
  "bedrooms" : 3,  
  "transport" : "1",  
  "neighborhood" : "0",  
  "houseRules" : "There are not house rules",  
}
```

```
"elevator" : true,  
"heating" : true,  
"kitchen" : true,  
"parking" : true,  
"tv" : true,  
"wifi" : true,  
"ac" : true,  
"smoking" : "Yes",  
"pets" : "Yes",  
"events" : "No"  
}
```

-----  
HOSTS ONLY:

PUT

/api/home/update

SAMPLE REQUEST:

localhost:8080/api/host/home/update

In Body:

```
{  
  "id" : 5,  
  "ownerId" : 3,  
  "ownerUsername" : "giorgos",  
  "reservations" : [ ],  
  "openBooking" : "2020-01-01",  
  "closeBooking" : "2020-10-01",  
  "image" : null,  
  "price" : 50.0,  
  "address" : "Aghsilaou",  
  "latitude" : "0.543545",  
  "longitude" : "0.553545",  
  "homeCategory" : {  
    "homeCategoryTitle" : "Shared Room"  
  },  
  "description" : "It is a nice place to stay",  
  "squareMeters" : 78,  
  "overnightPrice" : 50.0,  
  "extraPersonPrice" : 100.0,  
  "maxPeople" : 5,  
  "minOvernights" : 1,  
  "beds" : 3,  
  "bathrooms" : 4,  
  "bedrooms" : 3,  
  "transport" : "1",  
  "neighborhood" : "0",  
}
```

```
"houseRules" : "There are not house rules",
"elevator" : true,
"heating" : true,
"kitchen" : true,
"parking" : true,
"tv" : true,
"wifi" : true,
"ac" : true,
"smoking" : "Yes",
"pets" : "Yes",
"events" : "No"
}
```

---

HOSTS ONLY:

POST

/api/host/home/{id}/delete

SAMPLE REQUEST:

localhost:8080/api/host/home/2/delete

SAMPLE RESPONSE 1:

```
{
  "Status" : "Successful Deletion"
}
```

SAMPLE RESPONSE 2:

```
{
  "Status" : "User not found"
}
```

---

HOSTS ONLY:

POST (add photo to home)

localhost:8080/api/host/home/{id}/image

SAMPLE REQUEST:

localhost:8080/api/public/home/2/image

Body Request form-data

with KEY imagefile and VALUE the actual photo

SAMPLE RESPONSE:

redirect:/home/2/show

---

PUBLIC FOR ALL:

GET (all homes without filters)

/api/public/homes/all

SAMPLE RESPONSE:

```
[ {
  "id" : 1,
  "ownerId" : 3,
  "ownerUsername" : "giorgos",
  "reservations" : [ {
    "reservationId" : 1,
    "bookedHomelId" : 1,
    "bookedDate" : "2020-09-02",
    "leaveDate" : "2020-09-29",
    "isBooked" : 0,
    "userIdBooked" : 4,
    "userNameBooked" : "user",
    "hostReviewStars" : 0,
    "hostReviewDescription" : null,
    "homeReviewStars" : 0,
    "homeReviewDescription" : null
  } ],
  "openBooking" : "2020-01-01",
  "closeBooking" : "2020-10-01",
  "image" : null,
  "price" : 50.0,
  "address" : "Aghsilaou",
  "latitude" : "0.543545",
  "longitude" : "0.553545",
  "homeCategory" : {
    "id" : 2,
    "homeCategoryTitle" : "Shared Room"
  },
  "description" : "It is a nice place to stay",
  "squareMeters" : 78,
  "overnightPrice" : 50.0,
  "extraPersonPrice" : 100.0,
  "maxPeople" : 5,
  "minOvernights" : 1,
  "beds" : 3,
  "bathrooms" : 4,
  "bedrooms" : 3,
  "transport" : "1",
  "neighborhood" : "0",
  "houseRules" : "There are not house rules",
  "elevator" : true,
  "heating" : true,
  "kitchen" : true,
```



```
"parking" : true,
"tv" : true,
"wifi" : true,
"ac" : true,
"smoking" : "Yes",
"pets" : "Yes",
"events" : "No"
}]
```

---

#### PUBLIC FOR ALL:

GET (with url parameters as filters)

/api/public/homes?people={integer}&latitude={double}&longitude={double}&arrivalDate={yyy  
y-MM-dd}&departureDate={yyyy-MM-dd}

NOTE: (How filtering works)

1. We search for homes with people <= max people number
2. We search for homes in the range of 30km of latitude and longitude
3. We search for homes that are open for booking between the arrival and departure dates
4. We search for homes that have not any reservations between the arrival and departure dates

#### SAMPLE REQUEST:

localhost:8080/api/public/homes?people=4&latitude=0.543545&longitude=0.553541&arrival  
Date=2020-09-30&departureDate=2020-09-31

#### SAMPLE RESPONSE 1:

```
{
  "homes" : [ {
    "id" : 1,
    "ownerId" : 3,
    "ownerUsername" : "giorgos",
    "reservations" : [ {
      "reservationId" : 1,
      "bookedHomeld" : 1,
      "bookedDate" : "2020-09-02",
      "leaveDate" : "2020-09-29",
      "isBooked" : 0,
      "userIdBooked" : 4,
      "userNameBooked" : "user",
      "hostReviewStars" : 0,
      "hostReviewDescription" : null,
      "homeReviewStars" : 0,
      "homeReviewDescription" : null
    } ],
    "openBooking" : "2020-01-01",
    "closeBooking" : "2020-10-01",
```

```

    "image" : [],
    "price" : 50.0,
    "address" : "Aghsilaou",
    "latitude" : "0.543545",
    "longitude" : "0.553545",
    "homeCategory" : {
      "id" : 2,
      "homeCategoryTitle" : "Shared Room"
    },
    "description" : "It is a nice place to stay",
    "squareMeters" : 78,
    "overnightPrice" : 50.0,
    "extraPersonPrice" : 100.0,
    "maxPeople" : 5,
    "minOvernights" : 1,
    "beds" : 3,
    "bathrooms" : 4,
    "bedrooms" : 3,
    "transport" : "1",
    "neighborhood" : "0",
    "houseRules" : "There are not house rules",
    "elevator" : true,
    "heating" : true,
    "kitchen" : true,
    "parking" : true,
    "tv" : true,
    "wifi" : true,
    "ac" : true,
    "smoking" : "Yes",
    "pets" : "Yes",
    "events" : "No"
  }
}

```

SAMPLE RESPONSE 2:

```

{
  "homes" : [ ]
}

```

-----

PUBLIC FOR ALL:

POST (with url parameters as filters) (more filters coming soon)

/api/public/homes/more?type={Private Room | Shared Room | Full

House}&maxPrice={double}&wifi={boolean}&elevator={boolean}&heating={boolean}&kitchen={boolean}&parking={boolean}&tv={boolean}&ac={boolean}

SAMPLE REQUEST:

localhost:8080/api/public/homes/more?type=Shared  
Room&maxPrice=200.0&wifi=true&elevator=true&heating=true&kitchen=true&parking=true&  
tv=true&ac=true

#### ΣΗΜΑΝΤΙΚΟ:

1. Πρέπει να δώσεις και στο body το json που έλαβες από το προηγούμενο endpoint προκειμένου το backend να επεξεργαστεί την υπάρχων λίστα
2. Τα φίλτρα δεν είναι ανάγκη να είναι όλα στο link. Δηλαδή σε αυτό το endpoint μπορείς να βάλεις μόνο πχ φίλτρο για τα mixPrice και Wifi, όλα τα υπόλοιπα δεν είναι υποχρεωτικά  
Παράδειγμα: localhost:8080/api/public/homes/more?maxPrice=200.0&wifi=true

SAMPLE RESPONSE 1:  
(As the above response)

SAMPLE RESPONSE 2:

```
{  
  "homes" : []  
}
```

-----  
PUBLIC FOR ALL:

GET (get home photo)  
localhost:8080/api/public/home/{id}/image

SAMPLE REQUEST:  
localhost:8080/api/public/home/1/image

SAMPLE RESPONSE:  
The Photo :D

-----  
PUBLIC FOR ALL:

POST (post user photo)  
localhost:8080/api/public/user/{id}/image

SAMPLE REQUEST:  
localhost:8080/api/public/user/1/image

Body Request form-data  
with KEY imagefile and VALUE the actual photo

SAMPLE RESPONSE:  
redirect:/user/1/show

-----  
VISIBLE FOR ALL AUTHENTICATED:

GET (get user photo)  
localhost:8080/api/secure/user/{id}/image

SAMPLE REQUEST:  
localhost:8080/api/secure/user/1/image

SAMPLE RESPONSE:  
The User Photo :D

-----  
ADMINS ONLY:  
GET (get all details for a home)  
/api/admin/export/home/{id}/details?format=xml

NOTE: format=xml or format=json

SAMPLE REQUEST:  
localhost:8080/api/admin/export/home/1/details?format=xml

SAMPLE RESPONSE 1:  
{  
 "message" : "Choose format"  
}

SAMPLE RESPONSE 2:  
[  
 {  
  
 }  
]

-----  
ADMINS ONLY:  
GET (get all details for all homes)  
localhost:8080/api/admin/export/homes/details?format=json

NOTE: format=xml or format=json

SAMPLE REQUEST:  
localhost:8080/api/admin/export/homes/details?format=json

SAMPLE RESPONSE 1:  
{  
 "message" : "Choose format"  
}

SAMPLE RESPONSE 2:

```
[
  {
  }
]
```

-----  
VISIBLE FOR ALL AUTHENTICATED:

POST (BOOKING)

localhost:8080/api/secure/home/book

SAMPLE REQUEST:

localhost:8080/api/secure/home/book

```
{
  //h diafora tou panw me tou katw einai oti edw den vazeis reservationId
  "bookedHomeId" : 1,
  "bookedDate" : "2030-03-01",
  "leaveDate" : "2030-04-14",
  "isBooked" : 1,
  "userIdBooked" : 4,
  "userNameBooked" : "user",
  "hostReviewStars" : 0,
  "hostReviewDescription" : null,
  "homeReviewStars" : 0,
  "homeReviewDescription" : null
}
```

SAMPLE RESPONSE 1:

```
{
  "reservationId" : 2,
  "bookedHomeId" : 1,
  "bookedDate" : "2030-03-01",
  "leaveDate" : "2030-04-14",
  "isBooked" : 1,
  "userIdBooked" : 4,
  "userNameBooked" : "user",
  "hostReviewStars" : 0,
  "hostReviewDescription" : null,
  "homeReviewStars" : 0,
  "homeReviewDescription" : null
}
```

SAMPLE RESPONSE 2:

```
{"Status": "Error booking the room"}
```

-----

VISIBLE FOR ALL AUTHENTICATED:

PUT (Update Booking)

localhost:8080/api/secure/home/book

SAMPLE REQUEST:

localhost:8080/api/secure/home/book

```
{
  "reservationId" : 2,
  "bookedHomeId" : 1,
  "bookedDate" : "2030-03-01",
  "leaveDate" : "2030-04-14",
  "isBooked" : 1,
  "userIdBooked" : 4,
  "userNameBooked" : "user",
  "hostReviewStars" : 2,
  "hostReviewDescription" : "evala review",
  "homeReviewStars" : 0,
  "homeReviewDescription" : null
}
```

SAMPLE RESPONSE 1:

```
{
  "reservationId" : 2,
  "bookedHomeId" : 1,
  "bookedDate" : "2030-03-01",
  "leaveDate" : "2030-04-14",
  "isBooked" : 1,
  "userIdBooked" : 4,
  "userNameBooked" : "user",
  "hostReviewStars" : 2,
  "hostReviewDescription" : "evala review",
  "homeReviewStars" : 0,
  "homeReviewDescription" : null
}
```

SAMPLE RESPONSE 2:

```
{"Status": "Error updating booking"}
```

-----  
PUBLIC FOR ALL:

GET REVIEWS

localhost:8080/api/public/home/{id}/reviews

SAMPLE REQUEST:

localhost:8080/api/public/home/1/reviews

SAMPLE RESPONSE:

```
{
  "totalReviews" : 2,
  "average" : 3.0,
  "reviews" : [ 2, 4 ]
}
```

-----  
VISIBLE FOR ALL AUTHENTICATED:

POST (New Message)

localhost:8080/api/secure/message/send

SAMPLE REQUEST:

localhost:8080/api/secure/message/send

In Body Request:

```
{
  "senderId" : 1,
  "receiverId" : 4,
  "message" : "Hello World",
  "read" : false
}
```

SAMPLE RESPONSE 1:

```
{
  "messageId" : 2,
  "senderId" : 1,
  "receiverId" : 4,
  "message" : "Hello World",
  "read" : false,
  "creationTimestamp" : "2020-08-31 02:02 π.μ. EEST"
}
```

SAMPLE RESPONSE 2:

```
{"Status": "Error sending the message"}
```

-----  
VISIBLE FOR ALL AUTHENTICATED:

GET (get all messages from user)

localhost:8080/api/secure/messages/receivedfrom/{id}

SAMPLE REQUEST:

localhost:8080/api/secure/messages/receivedfrom/1

SAMPLE RESPONSE:

```
[ {
```

```
"messageId" : 1,
"senderId" : 1,
"receiverId" : 4,
"message" : "Hello Spyro",
"read" : false,
"creationTimestamp" : "2020-08-31 01:57 π.μ. EEST"
}, {
  "messageId" : 2,
  "senderId" : 1,
  "receiverId" : 4,
  "message" : "Hello Taki",
  "read" : false,
  "creationTimestamp" : "2020-08-31 01:59 π.μ. EEST"
}, {
  "messageId" : 3,
  "senderId" : 1,
  "receiverId" : 4,
  "message" : "Hello Mitso",
  "read" : false,
  "creationTimestamp" : "2020-08-31 02:02 π.μ. EEST"
}, {
  "messageId" : 4,
  "senderId" : 1,
  "receiverId" : 4,
  "message" : "Hello World",
  "read" : false,
  "creationTimestamp" : "2020-08-31 02:02 π.μ. EEST"
}]
```

-----

VISIBLE FOR ALL AUTHENTICATED:

DELETE (delete a message)

localhost:8080/api/secure/message/{id}

SAMPLE REQUEST:

localhost:8080/api/secure/message/1

SAMPLE RESPONSE:

{"Status": "Successful Message Deletion"}

-----

VISIBLE FOR ALL AUTHENTICATED:

GET (get history between two users)

localhost:8080/api/secure/messages/sender/{sender}/receiver/{receiver}



SAMPLE REQUEST:

localhost:8080/api/secure/messages/sender/1/receiver/4

SAMPLE RESPONSE:

```
[ {
  "messageId" : 1,
  "senderId" : 1,
  "receiverId" : 4,
  "message" : "Hello user 1",
  "read" : false,
  "creationTimestamp" : "2020-08-31 01:00 π.μ. EEST"
}, {
  "messageId" : 5,
  "senderId" : 4,
  "receiverId" : 1,
  "message" : "Hello admin",
  "read" : false,
  "creationTimestamp" : "2020-08-31 01:30 π.μ. EEST"
}, {
  "messageId" : 2,
  "senderId" : 1,
  "receiverId" : 4,
  "message" : "How are you user?",
  "read" : false,
  "creationTimestamp" : "2020-08-31 02:00 π.μ. EEST"
}, {
  "messageId" : 4,
  "senderId" : 1,
  "receiverId" : 4,
  "message" : "User? Are you here?",
  "read" : false,
  "creationTimestamp" : "2020-08-31 03:00 π.μ. EEST"
}, {
  "messageId" : 6,
  "senderId" : 4,
  "receiverId" : 1,
  "message" : "Sure i am fine and yes i am here",
  "read" : false,
  "creationTimestamp" : "2020-08-31 03:13 π.μ. EEST"
} ]
```

-----  
VISIBLE FOR ALL AUTHENTICATED:

GET (message connections)

localhost:8080/api/secure/messages

SAMPLE REQUEST:

(if i am the user with id=2)

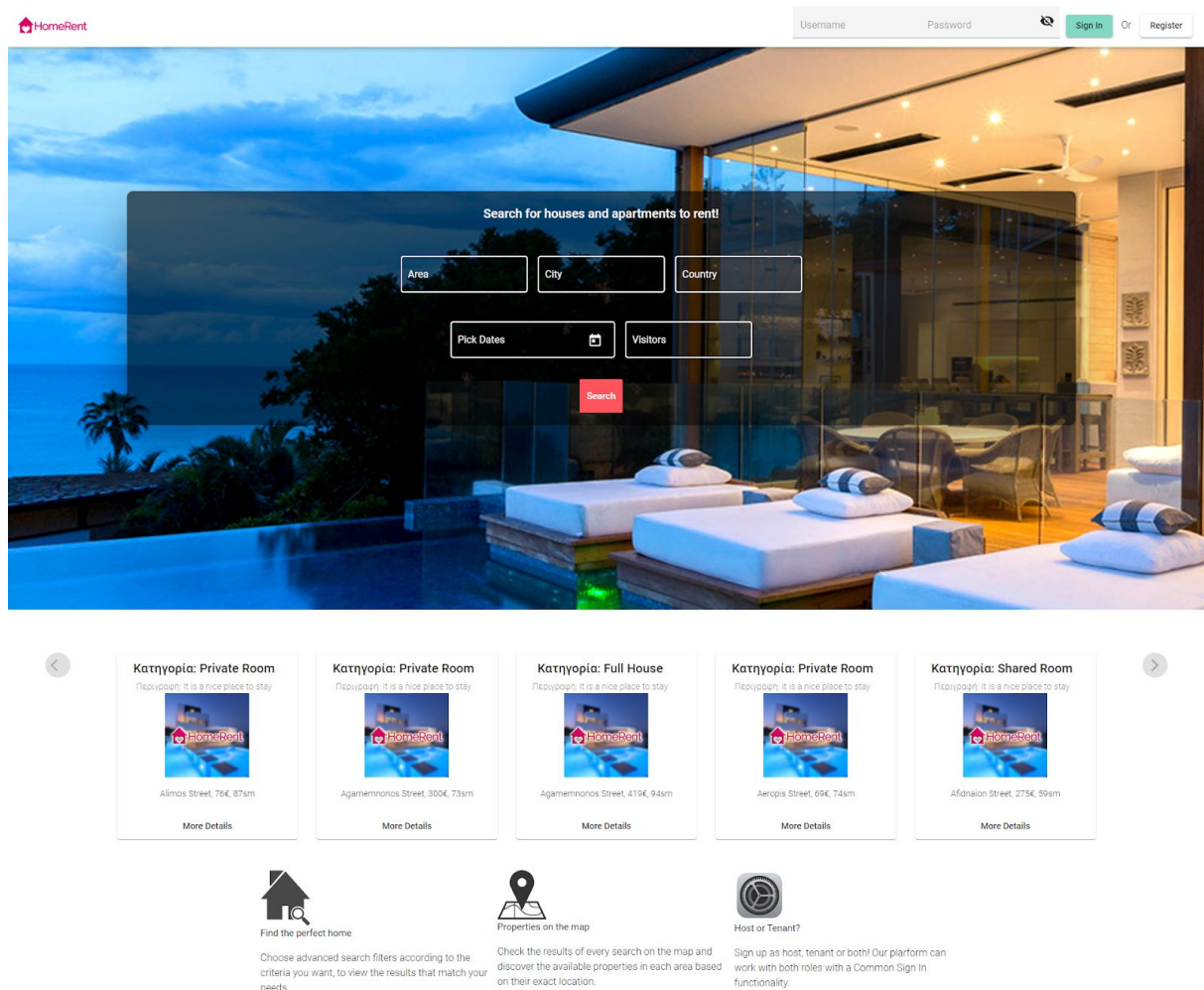
localhost:8080/api/secure/messages

SAMPLE RESPONSE:

```
[{
  "senderId" : 2,
  "receiverId" : 10,
  "receiverName" : "host10"
}]
```

## Frontend Screenshots Samples

### Home Page



## Register Page

HomeRent

Username

Password

Sign In

Or

Register

Register Now for Free!

Username

Password

Password Check

First Name

Last Name

Email

Telephone Number

User Role

Upload Image

0/10

Register

Cancel

Already have an account? Sign In

## Admin Page

HomeRent

HomeRequestsAccountChat

0

EN

Username	Όνομα	Emails
admin	admin	admin
user	User	Sir User
host3	host3	host3
host4	host4	host4
host5	host5	host5
host6	host6	host6
host7	host7	host7
host8	host8	host8
host9	host9	host9
host10	host10	host10

Items per page1 - 10 of 101<>>>

Αιτήματα Χρηστών

Εγκρίνετε ή απορρίψτε τα αιτήματα των χρηστών για τον ρόλο του οικοδεσπότη

Εκκρεμμών

27

αιτήματα χρηστών

Προβολή αιτημάτων

Στοιχεία Εφαρμογής

Δυνατότητα εξαγωγής στοιχείων σε XML ή JSON. Δημιουργία αρχείου με τις πληροφορίες των καταλυμάτων και των χρηστών και αποθήκευση στον υπολογιστή σας.

XMLJSON

## Chat Page

HomeRent

Account

Search

Chat

Chats

host3

host10

2020-09-10 21:59 μ.μ. ΕΕΣΤ

Ενημερωθήκαμε πως ενδιαφέρεσαι για το σπίτι

2020-09-10 22:44 μ.μ. ΕΕΣΤ

Πως μπορώ να βοηθήσω;

2020-09-10 22:47 μ.μ. ΕΕΣΤ

Καλησπέρα σας

2020-09-29 22:07 μ.μ. ΕΕΣΤ

Ναι ενδιαφέρομαι για να αγοράσω το σπίτι

2020-09-29 22:07 μ.μ. ΕΕΣΤ

Τέλεια, θα ήθελες να μάθεις περισσότερες πληροφορίες πάνω σε κάτι;

2020-09-29 22:07 μ.μ. ΕΕΣΤ

Message

send

## Account Page

HomeRent

Account

Search

Chat

Στοιχεία Λογαριασμού

Username

user

Κωδικός

Username

User

Επώνυμο

Surname

Email


user@homerent.com

Τηλέφωνο

+306911111111

Αλλαγή στοιχείων

Φωτογραφία Λογαριασμού



Upload Image

Save

## Search Result Page

HomeRent

Account Search Chat

Filters

Choose extra Filters

Accommodation Type:

Private Room

Max Price:

Max Price

Facilities:

☐ Επιλογή Όλων

☐ A/C

☐ Θέρμανση

☐ WiFi

☐ Ασφάλεια

☐ Κουζίνα


☐ Parking

☐ TV

Apply Filters

Κατηγορία: Private Room

Περιγραφή: luxury villa rental



Voila, 200€, 165 sm

4.5★

Προβολή

Items per page: 10 1 - 1 of 1 < >

Items per page: 10 1 - 1 of 1 < >

## Host Homes Page

HomeRent

Home Account My Places Chat


+

My Account

Log out

Κατηγορία: Full House

Περιγραφή: It is a nice place to stay




Agallanou Street, 284€, 65sm

4.2★

More Details Delete

Κατηγορία: Shared Room

Περιγραφή: It is a nice place to stay




Adrianoupolos Street, 243€, 45sm

3.1★

More Details Delete

Κατηγορία: Shared Room

Περιγραφή: It is a nice place to stay




Aganippis Street, 398€, 33sm

5★

More Details Delete

Κατηγορία: Shared Room

Περιγραφή: It is a nice place to stay




Afaias Square, 403€, 84sm

★

More Details Delete

Κατηγορία: Shared Room

Περιγραφή: It is a nice place to stay




Municipal Unit of Fyli, Municipality of Fyli, Regional Unit of West Attica, Attica, Greece, 398€, 33sm

★

More Details Delete

Κατηγορία: Shared Room

Περιγραφή: It is a nice place to stay




Municipality of Acharnes, Regional Unit of East Attica, Attica, Greece, 398€, 33sm

★

More Details Delete

Κατηγορία: Shared Room

Περιγραφή: It is a nice place to stay




Rodonas, Δήμος Αγρινίου, Flouia Regional Unit, Western Macedonia, Epirus and Western Macedonia, 53200, Greece, 398€, 33sm

★

More Details Delete

Κατηγορία: Shared Room

Περιγραφή: It is a nice place to stay




Kiryoni, Straton, Δήμος Αριστοτέλη, Chalkidiki Regional Unit, Central Macedonia, Macedonia and Thrace, 63083, Greece, 398€, 33sm

★

More Details Delete

Κατηγορία: Shared Room

Περιγραφή: It is a nice place to stay




Moshtica, Municipality of Makedonska Kamenica, Eastern Region, Macedonia, 398€, 33sm

★

More Details Delete

Κατηγορία: Private Room

Περιγραφή: fofsfad




Κοινότητα Μαρτίνου, Δήμος Αοκίων, Phthiotis Regional Unit, Central Greece, Thessaly and Central Greece, Greece, 22€, 53sm

★

More Details Delete

Κατηγορία: Private Room

Περιγραφή: fofsfad




Κοινότητα Μαρτίνου, Δήμος Αοκίων, Phthiotis Regional Unit, Central Greece, Thessaly and Central Greece, Greece, 22€, 53sm

★

More Details Delete

Κατηγορία: Private Room

Περιγραφή: fofsfad



Κοινότητα Μαρτίνου, Δήμος Αοκίων, Phthiotis Regional Unit, Central Greece, Thessaly and Central Greece, Greece, 22€, 53sm

★

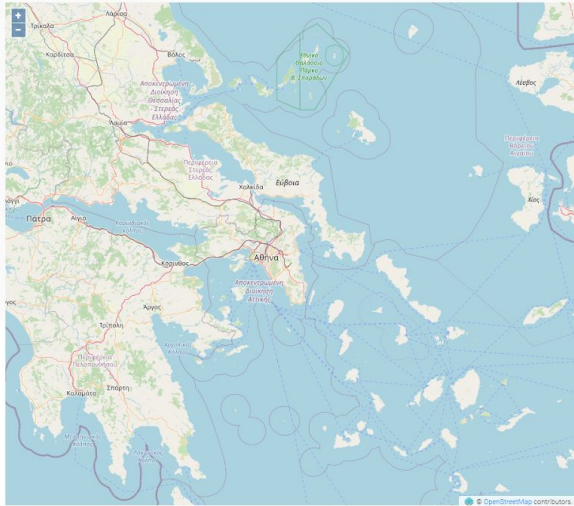
More Details Delete

Items per page: 12 1 - 12 of 12 < >

## House Details Page



[Home](#) [Account](#) [My Places](#) [Chat](#)



### Στοιχεία Ακινήτου

Διεύθυνση:  Ημερομηνία:

Προσφορά:  Κανόνες:

Οδηγίες επισκεπτών:  Περιγραφή:

Άτομα	Τμήμα	T.M	Μήνες	Κατάσταση	Υποβλημένα
3	284	66	9	4	10

Επισκεπτήριο Άτομα:  Ελάχιστες διαμονές:  Είδος Καταλύματος:

Events

☒ A/C ☐ Θέρμανση ☐ WiFi ☐ Ασανσέρ ☒ Κουζίνα ☐ Parking  
☒ TV ☒ Καθίζοντες ☒ Κατοικίδια  
Τροποποίηση Διαγραφή