



Πανεπιστήμιο Πειραιώς
Τμήμα Ψηφιακών Συστημάτων

Δικτυοκεντρικά Πληροφοριακά Συστήματα Απαλλα-
κτική Εργασία Φεβρουαρίου 2022

ΤΕΧΝΙΚΗ ΑΝΑΦΟΡΑ

Ονοματεπώνυμο
ΣΕΜΠΑΣΤΙΑΝ ΑΝΤΡΙΣΑΝ

A.M.
E18008

Ενότητα 1: Θεματολογία Συστήματος

Το σύστημα που υλοποιήθηκε είναι μια ιστοσελίδα τουριστικού γραφείου με αναχωρήσεις από Αθήνα. Σκοπός του συστήματος είναι η αυτοματοποίηση της διαδικασίας αγοράς εισιτηρίων μέσω ενός φιλικού προς τον χρήστη περιβάλλοντος. Εξασφαλίζοντας ένα πακέτο μεταφοράς (αγορά εισιτηρίου) μέσω του συστήματος, ο ενδιαφερόμενος δεν χρειάζεται να μεταβεί σε φυσικό εκδοτήριο, κερδίζοντας χρόνο για την οργάνωση του ταξιδιού του.

Επιπλέον, ο χρήστης έχει την δυνατότητα διατήρησης λίστας αγαπημένων προορισμών καθώς και δημιουργίας κριτικών για αυτούς.

Ενότητα 2: Υλοποίηση Συστήματος

Το πληροφοριακό σύστημα βασίζεται σε REST αρχιτεκτονική. Ο τρόπος λειτουργίας βασίζεται εξ ολοκλήρου στην ανταλλαγή δεδομένων μεταξύ δύο μερών (client και server) μέσω του πρωτοκόλλου HTTP.

Περιγραφή των RESTful Web Services

Όλες βασίζονται σε REST αρχιτεκτονική

User and Admin Log In

Description: Gets username and password from user. Checks if credentials are correct using a select from users. Also, creates and keeps his uid as a global variable (static). If credentials correspond, uses the isAdmin() method to check if he has admin rights.

Endpoint URL: <http://localhost:8080/AAAA/rest/Auth/Login/{username}/{password}>

HTTP Method: GET

Parameters: username and password given

Returns: Login completed or failed

User Register

Description: The user in order to use the system must have account, in order to create one he needs to give some information about him, and create some credentials. This service takes data given by user and inserts them into the users table. Before sending the return message it creates the uid which is used in most services.

Endpoint URL :http://localhost:8080/AAAA/rest/Auth/Register/{first_name}/{last_name}/{username}/{password}/{phone}

HTTP Method: POST

Parameters: first_name, last_name, username, password, phone

User Make a Booking

Description: Creates new booking. Creates new row using the insert into of mysql and passes data given by user and the uid which was set in previous services.

Endpoint URL: http://localhost:8080/AAAA/rest/Booking/MakeBooking/{t_id}/{total_tickets}

HTTP Method: POST

Parameters: as seen in url

Returns: Booking Created!

Admin Add Destination

Description: Insert destination given its name and optionally a phone number

Endpoint URL: http://lo-

calhost:8080/AAAA/rest/DestsUpdate/AddDest/{d_name}/{d_tel}

HTTP Method: POST

Parameters: name and number

Returns: Destination Added

Admin Add Transportation

Description: Insert transportation offer given some information about it provided by the user

Endpoint URL: http://lo-

calhost:8080/AAAA/rest/DestsUpdate/AddTr/{d_name}/{way}

/{price}/{duration}/{month}

HTTP Method: POST

Parameters: name, way, price, duration, month

Returns: Offer created

Admin Delete Transportation/offer

Description: Deletes a transportation offer, it is chosen by the admin.

Endpoint URL: http://localhost:8080/AAAA/rest/DestsUpdate/DelTr/{t_id}

HTTP Method: POST

Parameters: id of the offer

Returns: Transportation offer was deleted

getAllDest()

Description: Method used in many endpoints to provide all destinations

Returns: list of destinations

Client Favorite CRUD

3 endpoints where the user can see (read) his favorite destinations, add (create) new destination to his favorites and remove (delete) from his favorites.

Endpoints URL: `http://localhost:8080/AAAA/rest/Favorites/AddFav/{d_name}`
`http://localhost:8080/AAAA/rest/Favorites/RemoveFav/{d_name}`

Method `getFavs()` is called in the `Favorites.jsp` and returns list of all favorite destinations of the user. Makes use of the global uid in order to know the user.

`getFavs()` is used in both `/AddFav` and `/RemoveFav`.

Also used in both services is the method `isAlreadyFav(d_id)` which is called inside `getFavs` so it provides correct data to the endpoints.

Client Add Review

Description: Create new review. Inserts new record in the review table.

Endpoint URL: `http://localhost:8080/AAAA/rest/Reviews/AddReview/{d_name}/{stars}/{review}`

HTTP Method: POST

Parameters: name of destination, stars (0-5), optional review/comments

Returns: Review added

Admin Delete Review

Description: Choose one of all reviews and delete it. All reviews (select) is provided by the `getAllReviews()` method.

Endpoint URL: `http://localhost:8080/AAAA/rest/Reviews/DeleteReview/{r_id}`

HTTP Method: DELETE

Parameters: review id

Returns: Review was deleted

Client cheapest offers

Description: Takes advantage of getAllTrOptions() method to see the 10 cheapest offers. Join, order by are used in the mysql select query.

Endpoint URL: <http://localhost:8080/AAA/CheapestOptions.jsp>

Returns: list of offers

Note:

Classes Destination.java, Favorite.java, Review.java, Transportation-Option.java act as constructors and are used in get arraylist methods.

Ενότητα 3: Εγχειρίδιο Χρήσης Συστήματος

Για την υλοποίηση του συστήματος χρησιμοποιήθηκαν:

Eclipse IDE 2021-12

Windows 10

Apache-tomcat-9.0.58

Java 1.8.0_231

CSS & HTML & Javascript

MYSQL 8.0

Jersey-bundle-1.19.4.jar

Jsr311-api-1.1.1.jar

mysql-connector-java-8.0.28 (τοποθετήθηκε στο tomcat/lib)

Περιγραφή βάσης δεδομένων:

Η βάση δεδομένων ονομάζεται dikt (δικτυοκεντρικά) και αποτελείται από 6 πίνακες.

Πίνακας users:

Ο κάθε χρήστης περιγράφεται μοναδικά μέσω του users_id. Τα δεδομένα first_name, last_name, username, password, phone δίνονται από τον χρήστη κατά την εγγραφή του σύστημα ενώ η τελευταία στήλη, is_admin παίρνει ως default τιμή 'no' και ξεχωρίζει τον διαχειριστή από τους υπόλοιπους χρήστες.

```
1 • SELECT * FROM dikt.users;
```

users_id	first_name	last_name	username	password	phone	is_admin
4	Seby	Messi	seb1	pass1	6987044828	yes
8	Aggeliki	Mitsakou	agg2	pass2	6987055677	no
9	Marios	Benias	mben	pass	6987055690	no
10	Maria	Andrisan	andr	pass1	6987055638	no

Πίνακας reviews:

Εδώ αποθηκεύονται και ανακτώνται οι κριτικές των προορισμών. Το stars παίρνει τιμές μεταξύ 1 και 5, το review παίρνει default τιμή άδειου string αν δεν επεξεργαστεί από τον χρήστη. Το r_id είναι auto-increment primary key ενώ το d_id και users_id προσδιορίζουν τον προορισμό και τον χρήστη που έκανε την κριτική.

```
1 • SELECT * FROM dikt.reviews;
```

r_id	d_id	users_id	stars	review
13	7	8	4	Family friendly.
14	4	8	1	
15	30	11	5	Best city
NULL	NULL	NULL	NULL	NULL

Πίνακας favorites:

Αυτός ο πίνακας κρατάει μόνο το users_id και το d_id. Ανακτώνται πάντα μαζί καθώς ο συνδυασμός των δύο παραπάνω id προσδιορίζει

μοναδικά κάθε στοιχείο του πίνακα. Δηλαδή primary key του πίνακα είναι και τα δύο.

users_id	d_id
7	5
7	6
7	12
7	13
7	24
7	26

Πίνακας destination:

Το d_name και d_tel αποθηκεύονται και τα δύο ως varchar. Μόνο το d_name είναι υποχρεωτικό αφού το d_tel παίρνει default τιμή άδειου κειμένου αν δεν δοθεί. Το d_id είναι auto-increment primary key.

d_id	d_name	d_tel
9	Ikaria	2114070999
10	Livadia	2254078921
11	Rhodes	2189077354
12	Drama	697056871
13	Samos	2114072891
15	Mani	2189077341
16	Metsovo	6987055683
24	Loutraki	6987055638
26	Arahova	69453569780
27	Kamena...	6975188932
29	Nayplio	
30	Salonica	
31	Zante	

Πίνακας transportation:

Αυτός ο πίνακας αποθηκεύει τις προσφορές ταξιδιού (μετακίνησης). Έχει πρωτεύον κλειδί το t_id, αποθηκεύεται το d_id του προορισμού, τα price και duration είναι double ενώ way και month περιγράφουν την μέθοδο μεταφοράς και τον μήνα για τον οποίο ισχύει η προσφορά αντίστοιχα.

t_id	d_id	way	price	duration	month
6	12	private bus	84	9.2	March
8	6	private boat	183	6.25	July
9	24	limousine	95.3	2.4	November
10	4	bus	8.7	3.05	December
11	5	private bus	49.9	5.55	March
12	6	helicopter	400	1.2	July
14	11	ferry boat	81	13	June
15	11	helicopter	520	3	April
16	29	bus	12	2.25	February
17	12	private jet	315	0.4	May
18	10	bus	3.8	2.1	November
19	30	helicopter	377	0.25	August

Πίνακας bookings:

Κάθε γραμμή αυτού του πίνακα αποτελεί αγορά προσφοράς (εισιτηρίων) από έναν χρήστη. Υπάρχει μοναδικός κωδικός κράτησης b_id καθώς και το id του χρήστη (users_id) και της προσφοράς (t_id). Ακόμη, το total_tickets είναι υποχρεωτικό πεδίο τύπου int και κρατάει τον αριθμό εισιτηρίων/ ατόμων που θα ταξιδέψουν.

b_id	users_id	t_id	total_tickets
1	7	10	3
2	7	18	2
3	7	12	1
4	5	16	10

Εγκατάσταση του συστήματος:

Δημιουργία νέου MySQL connection με user: root και password 18121973

Import την βάση δεδομένων MySQL.

Import το project στον Eclipse IDE.

Δημιουργία tomcat server όπου και θα τρέχει το σύστημα.

Ο κώδικας των client και admin βρίσκονται στο ίδιο project.

Εκκίνηση συστήματος:

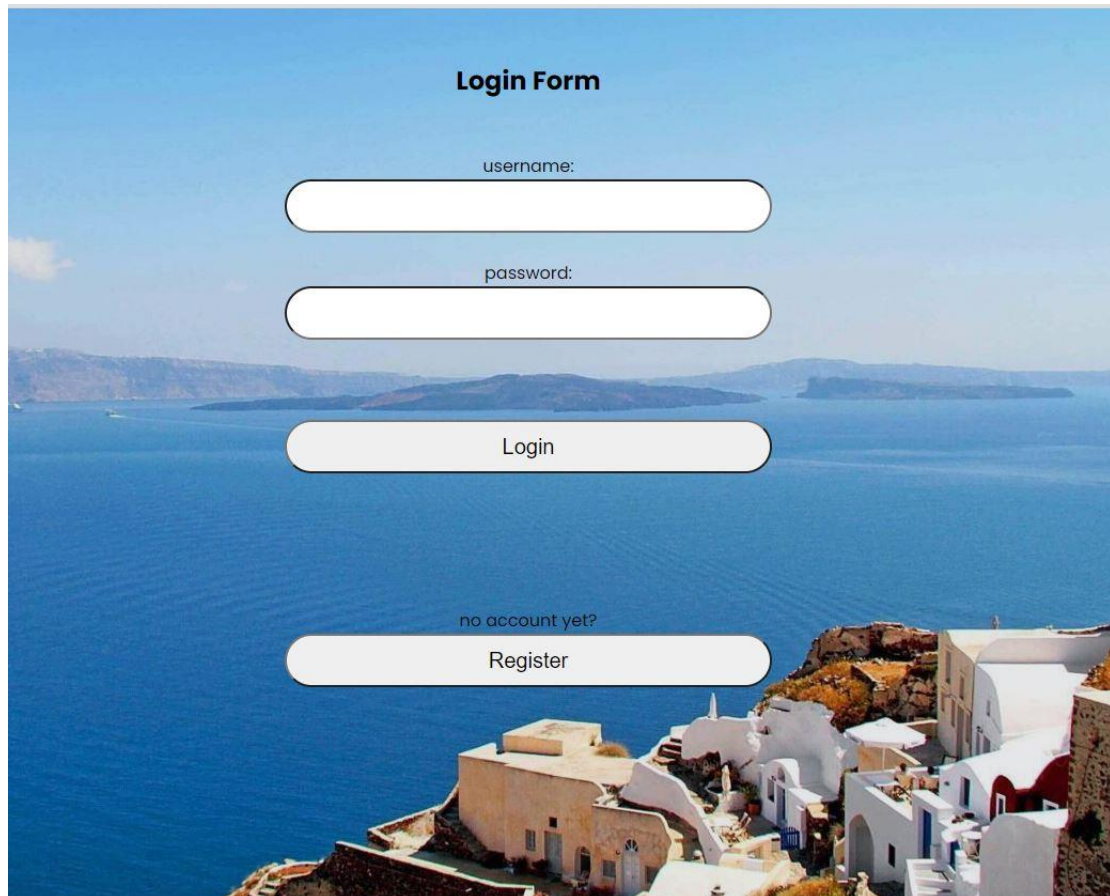
Για την εκκίνηση του project εκτελείται ΠΑΝΤΑ πρώτο το αρχείο LoginPage.jsp. Αν θέλετε να εκτελέσετε το σύστημα μέσω ενός νέου χρήστη υπάρχει αντίστοιχο κουμπί που οδηγεί σε φόρμα εγγραφής. Ανάλογα με την ιδιότητα του χρήστη (διαχειριστής ή απλός) οδηγούμαστε σε αντίστοιχο home page interface.

ΔΙΕΠΑΦΕΣ:

1) Το σύστημα πάντα πρέπει να εκκινεί σε αυτήν την διεπαφή. Ανεξαρτήτου ιδιότητας ο χρήστης πρέπει να δώσει τα στοιχεία που ζητούνται και να κάνει login.

Όταν πατηθεί το login, εκτελείται web service όπου ψάχνει στην βάση δεδομένων το username που δόθηκε, εξετάζει αν ταιριάζει το password και ακόμη ελέγχεται αν είναι διαχειριστής. Αν τα στοιχεία αντιστοιχούν σε χρήστη από την βάση δεδομένων γίνεται σύνδεση και δημιουργείται ένα public static int uid που κρατάει τον users_id του χρήστη που έκανε login.

Αν επιθυμείτε να χρησιμοποιήσετε το σύστημα μέσω ενός νέου χρήστη πατήστε το κουμπί Register.



Login Form

username:

password:

Login

no account yet?

Register

2) Σε περίπτωση που πατηθεί το Register button της παραπάνω διεπαφής, το σύστημα συνεχίζει στο Register.jsp που αποτελείται από μια φόρα εγγραφής.

Το web service που παρέχεται είναι αυτό της εγγραφής σε σύστημα. Εκτελείται insert into query στον πίνακα των χρηστών.

Ο χρήστης που θα δημιουργείται θα είναι πάντα απλός χρήστης, όχι διαχειριστής. Αφού συμπληρωθεί η φόρμα και πατηθεί το Register button, ο χρήστης οδηγείται στο start page για απλούς χρήστες.

Registration Form

first name:

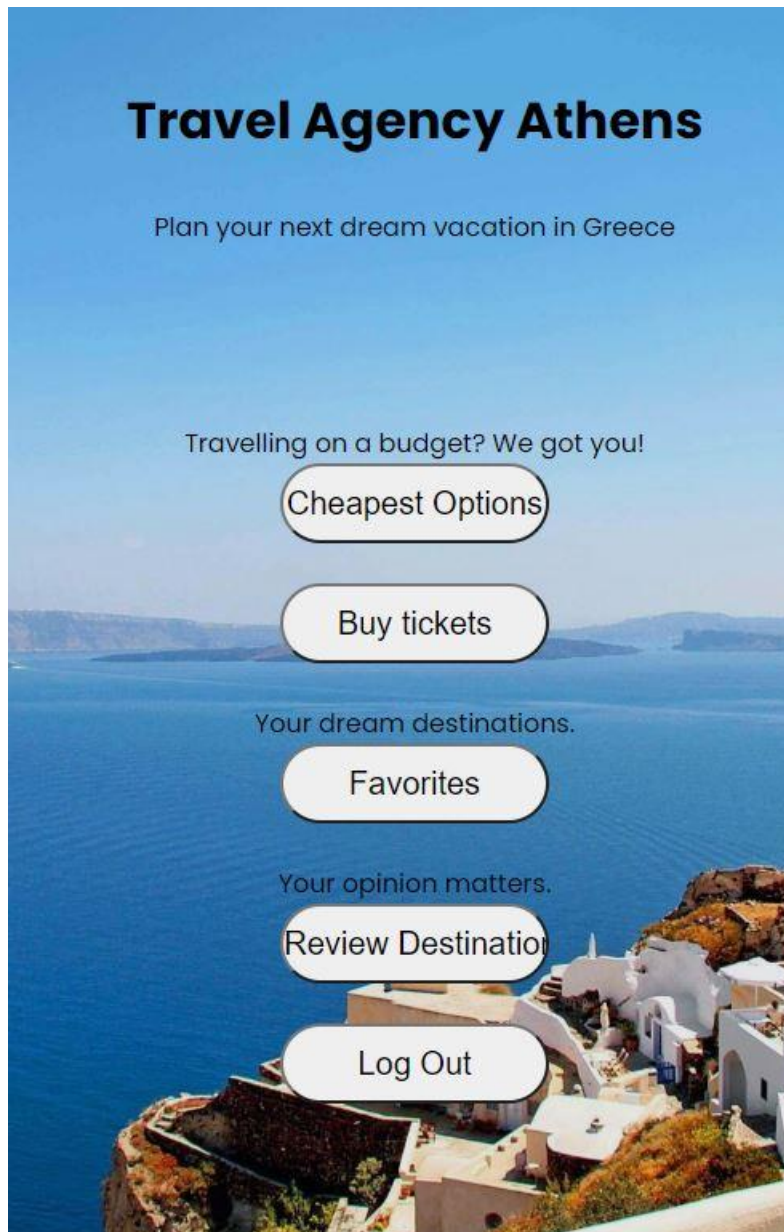
last name:

username:

password:

phone:

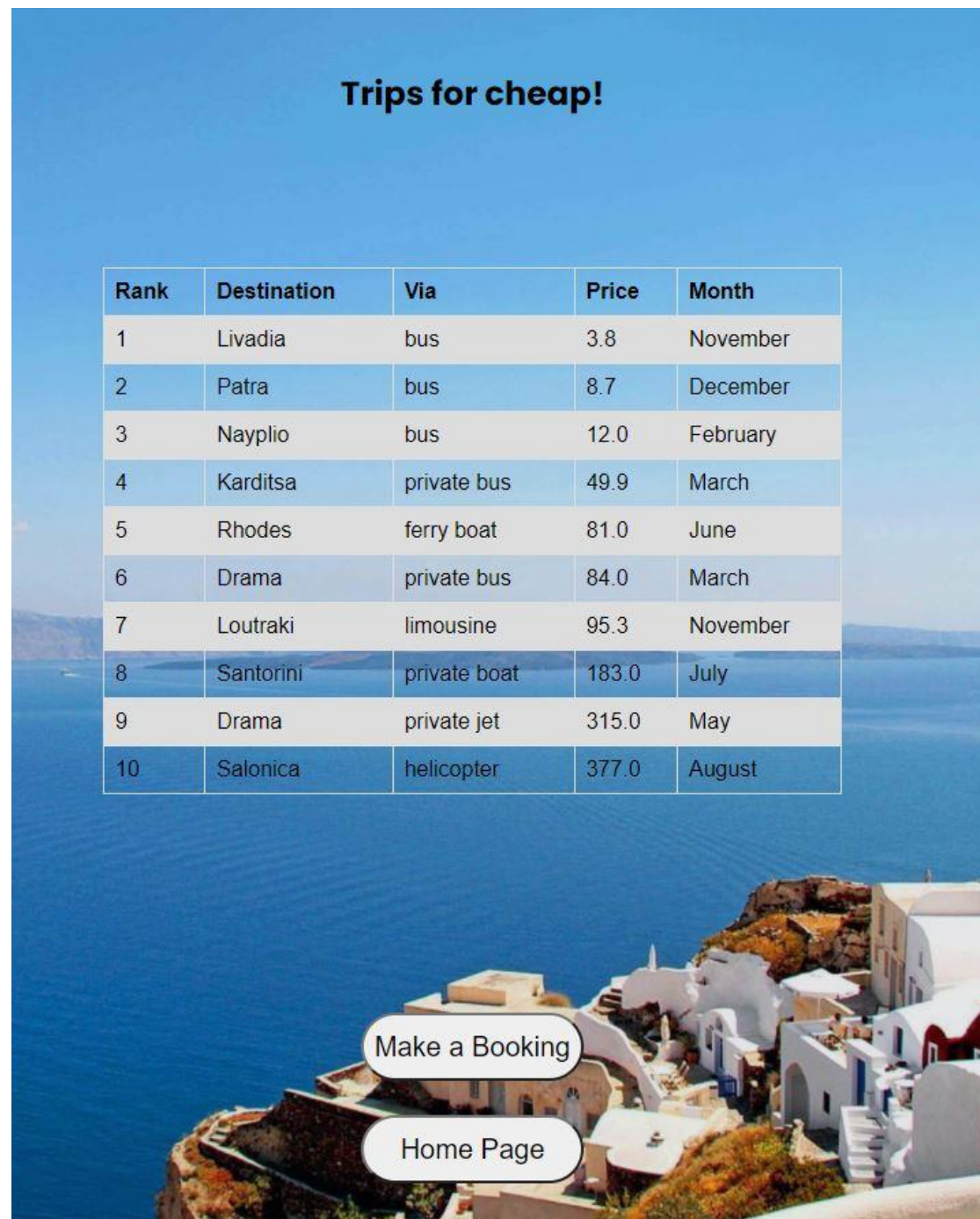
3) Όταν γίνεται επιτυχής σύνδεση απλού χρήστη ή δημιουργία νέου με τους παραπάνω τρόπους, ο χρήστης οδηγείται στο StartPage.jsp που είναι μια διεπαφή όπου μπορεί να επιλέξει μια από τις λειτουργικότητες που επιθυμεί. Αυτό το web service κάνει redirect στις λειτουργικότητες του χρήστη.



4) Πατώντας Cheapest Options στην διεπαφή 3 καλείται web service που εμφανίζει στον χρήστη τις 10 φθηνότερες προσφορές ταξιδιού. Συγκεκριμένα, καλείται η μέθοδος `getAllTrOptions()` που βρίσκεται στο αρχείο `SomeSelects`. Η μέθοδος κάνει `select` στην βάση δεδομένων και επιστρέφει `ArrayList` με τις 10 πιο φθηνές προσφορές ταξιδιού.

Κάθε στοιχείο του `ArrayList` είναι `object` τύπου `TransportationOption` (`TransportationOption.java`).

Η λίστα που επιστρέφεται στην διεπαφή εμφανίζεται με την βοήθεια ενός for loop.



The screenshot displays a web application interface. At the top, the text "Trips for cheap!" is centered in a bold, black font. Below this is a table with five columns: Rank, Destination, Via, Price, and Month. The table contains ten rows of data, each representing a different trip. At the bottom of the interface, there are two buttons: "Make a Booking" and "Home Page". The background of the entire page is a scenic view of a coastal town with white buildings and a blue sea.

Rank	Destination	Via	Price	Month
1	Livadia	bus	3.8	November
2	Patra	bus	8.7	December
3	Nayplio	bus	12.0	February
4	Karditsa	private bus	49.9	March
5	Rhodes	ferry boat	81.0	June
6	Drama	private bus	84.0	March
7	Loutraki	limousine	95.3	November
8	Santorini	private boat	183.0	July
9	Drama	private jet	315.0	May
10	Salonica	helicopter	377.0	August

5) Η διεπαφή MakeBooking.jsp δίνει την δυνατότητα στον χρήστη να αγοράσει εισιστήρια. Πρέπει να πληκτρολογήσει τον κωδικό της προσφοράς, τον αριθμό εισιτηρίων καθώς και έναν κωδικό παγρω με τον

οποίο θα κάνει την αγορά (το paypal code δεν εξετάζεται, η συναλλαγή γίνεται εικονικά μόνο).

Σε αυτήν την διεπαφή δύο web service. Το πρώτο είναι η εμφάνιση πίνακα όλων των προσφορών με παρόμοιο τρόπο που είδαμε και στην διεπαφή φθηνών ταξιδιών. Εδώ χρησιμοποιείται διαφορετική μέθοδος, η getAllOffres() που λειτουργεί παρόμοια με την getAllTrOptions().

Δεύτερο web service είναι το κλείσιμο εισιτηρίων/ αγορά προσφοράς. Όταν πατηθεί το Make Booking button, οι παράμετροι (εκτός του paypal code) στέλνονται στο /MakeBooking/παράμετροι όπου εκτελείται insert into στον πίνακα bookings της βάσης δεδομένων. Το uid λειτουργεί ως global με τιμή για το users_id οπότε απλά το βάζουμε στο query.

Transportation Offers

Offer number	Destination	Travel method	Price (1 person)	Trip duration	Month Available
6	Drama	private bus	84.0	9.2	March
8	Santorini	private boat	183.0	6.25	July
9	Loutraki	limousine	95.3	2.4	November
10	Patra	bus	8.7	3.05	December
11	Karditsa	private bus	49.9	5.55	March
12	Santorini	helicopter	400.0	1.2	July
14	Rhodes	ferry boat	81.0	13.0	June
15	Rhodes	helicopter	520.0	3.0	April
16	Nayplio	bus	12.0	2.25	February
17	Drama	private jet	315.0	0.4	May
18	Livadia	bus	3.8	2.1	November
19	Salonica	helicopter	377.0	0.25	August

Booking Form

offer number:

number of persons:

paypal code:

Make Booking

HomePage

6) Favorites.jsp

Σε αυτό το endpoint υπάρχουν παραπάνω από μια λειτουργικότητες. Το πρώτο είναι η εμφάνιση των αγαπημένων προορισμών του χρήστη. Το δεύτερο είναι η πρόσθεση νέου προορισμού στα αγαπημένα

και η τρίτη λειτουργικότητα η αφαίρεση προορισμού από την λίστα αγαπημένων.

Ο κώδικας των παραπάνω services βρίσκεται στο ManageFavorites.java. Υλοποιήθηκαν οι /AddFav και /RemoveFav που λειτουργούν ως insert into και delete from αντίστοιχα. Επιπλέον, υλοποιήθηκαν οι μέθοδοι getFavs() και isAlreadyFav(id προορισμού). Οι παραπάνω χρησιμοποιούνται μαζί και παραπάνω από μια φορά ώστε εκεί όπου διαλέγουμε προορισμό για να τον προσθέσουμε στα αγαπημένα, να μην εμφανίζει αυτούς που είναι ήδη favorite. Παρόμοια, στο remove destination from favorites εμφανίζει μόνο ήδη αγαπημένους προορισμούς.

My Favorites

Destinations

Aigina

Patra

Lefkada

Mani

Kamena Vourla

Nayplio

Zante

Add a destination to favorites

Arahova ▾

Add to Favorites

Remove destination from favorites:

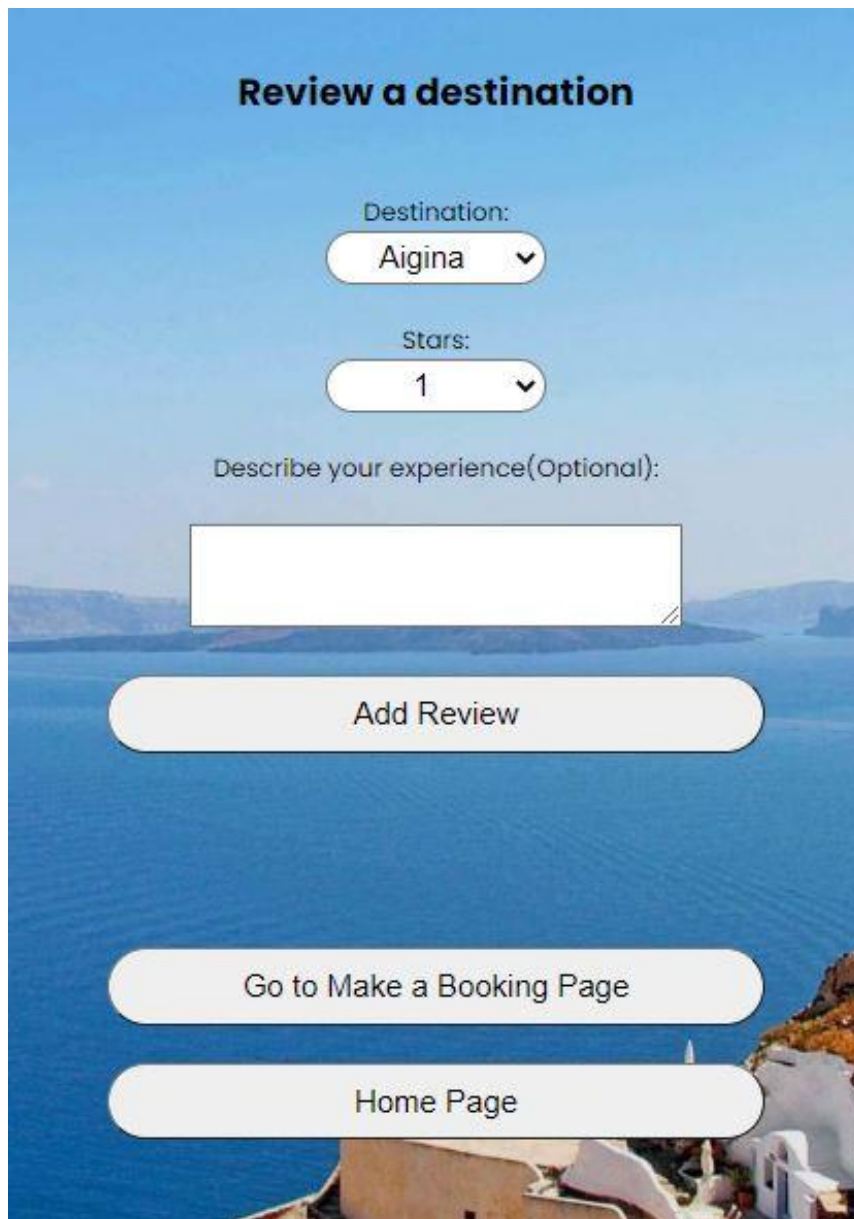
Aigina ▾

Remove Favorite

Go to Make a Booking Page

HomePage

7) Στο Review a destination κύρια λειτουργικότητα είναι η πρόσθεση κριτικής κριτικών σε προορισμούς. Η επιλογή αστεριών είναι αναγκαία, ενώ η πρόσθεση σχολίων όχι.



The screenshot shows a web form titled "Review a destination" set against a background image of a coastal town. The form includes a "Destination:" dropdown menu with "Aigina" selected, a "Stars:" dropdown menu with "1" selected, and a text area for "Describe your experience(Optional):". Below these fields are three buttons: "Add Review", "Go to Make a Booking Page", and "Home Page".

Review a destination

Destination:
Aigina ▼

Stars:
1 ▼

Describe your experience(Optional):

Add Review

Go to Make a Booking Page

Home Page

8) Υπάρχει διαθέσιμο κουμπί για να κάνει ο χρήστης log out. Η λογική του log out είναι να ακυρώνει το global uid θέτοντας του τιμή 0. Στην συνέχεια κάνει redirect στην login interface.

9) Οι admins χρησιμοποιούν την ίδια σελίδα με τους απλούς χρήστες ώστε να κάνουν login. Στο παρασκήνιο, κάνοντας χρήση της μεθόδου `isAdmin(uid)` που βρίσκεται στο `AuthenticationInterface.java` ελέγχεται αν έχει δικαιώματα διαχειριστή. Στην περίπτωση μας είναι admin οπότε κάνει redirect στο home page του διαχειριστή.

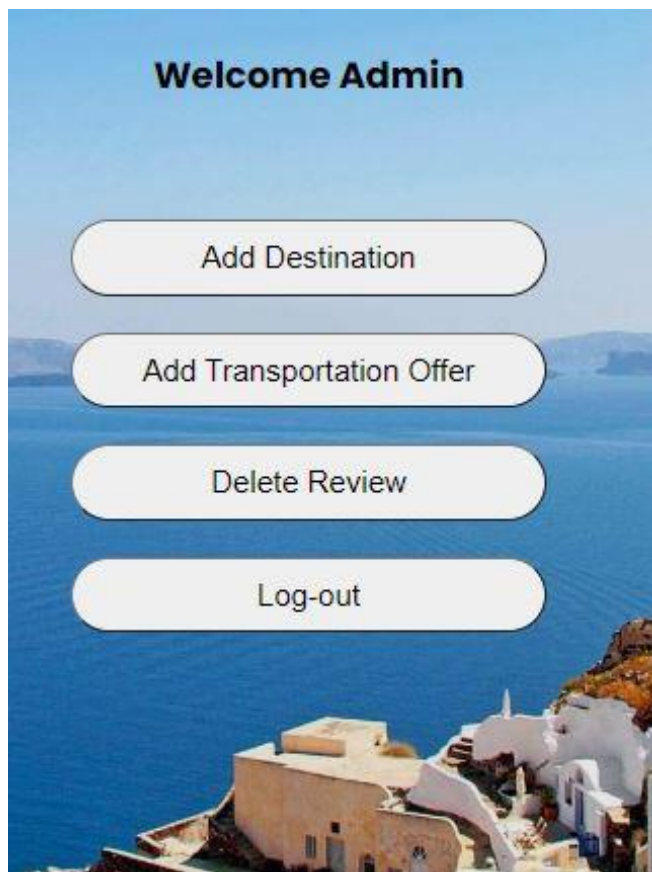
Στοιχεία του μοναδικού admin στην βάση δεδομένων:

Username: seb1

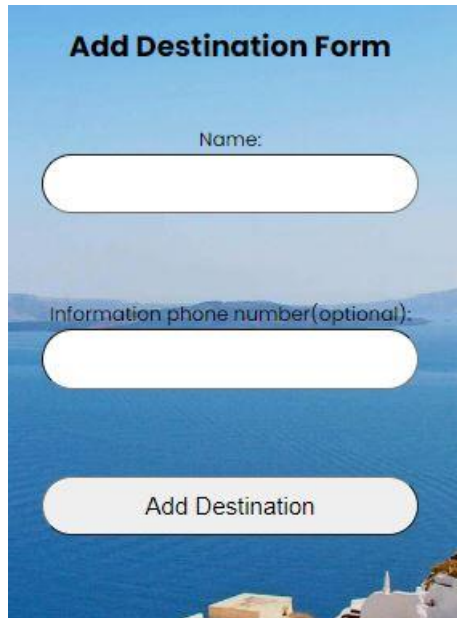
Password: pass1

10)AdminHomePage.jsp

Ο διαχειριστής πατάει την λειτουργικότητα που επιθυμεί και γίνεται το κατάλληλο redirection.



11) Στο AddDestination.jsp γίνεται πρόσθεση (insert into) νέων προορισμών στην βάση δεδομένων. Απαιτείται να πληκτρολογήσει το όνομα ενώ το τηλέφωνο επικοινωνίας είναι προαιρετικό (πχ αν έχει γραφεία η εταιρία σε αυτόν τον προορισμό, προσθέτουμε και τηλέφωνο).



12) Πρόσθεση νέας προσφοράς/ μεθόδου ταξιδιού σε προορισμό. Ο διαχειριστής επιλέγει έναν από όλους τους υπάρχοντες προορισμούς. Οι επιλογές προορισμών εμφανίζονται με την βοήθεια της μεθόδου getAllDest() του DestinationUpdate.java που επαναχρησιμοποιείται σε πολλά σημεία του συστήματος. Στην συνέχεια επιλέγει 1 από τους 8 προεπιλεγμένους τρόπους ταξιδιού. Ομοίως πρέπει να συμπληρωθούν τα υπόλοιπα πεδία.

Add a new transportation offer

Choose a destination:

Aigina ▼

Way of travel:

bus ▼

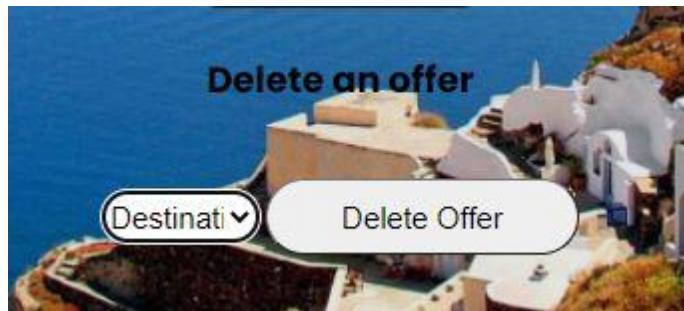
Price(double format):

Duration(hours, double format):

Month: January ▼

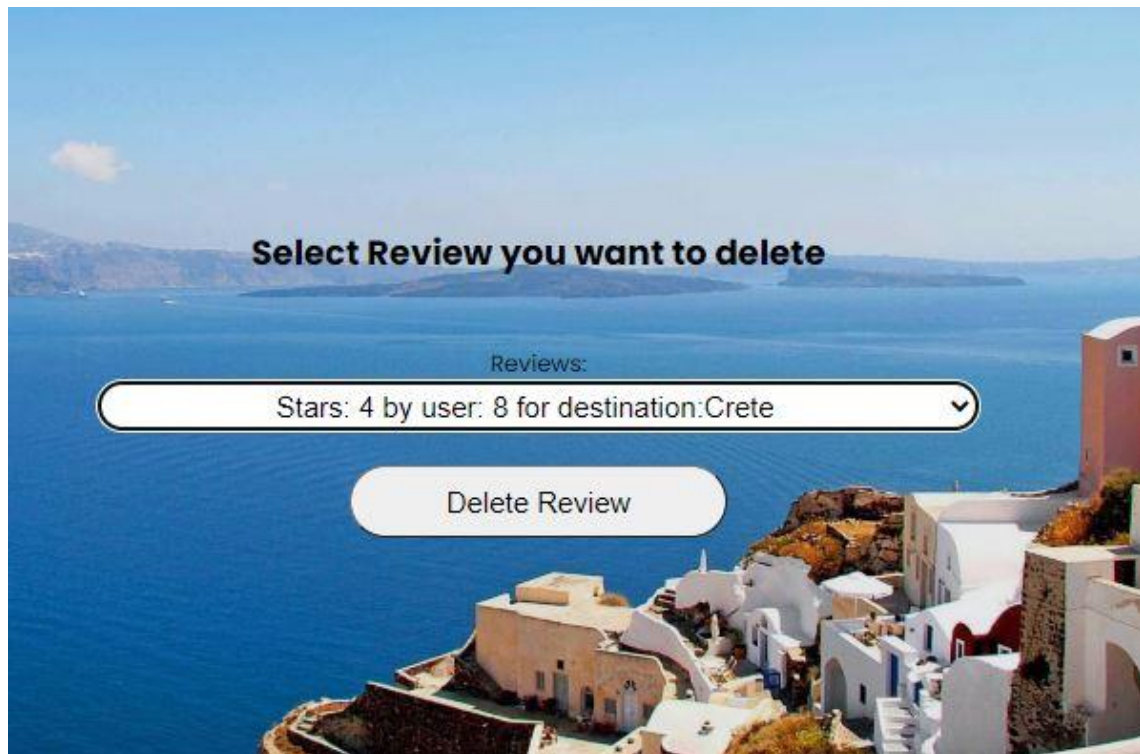
Add Transportation Offer

13) Η διαγραφή προσφορών γίνεται στο ίδιο jsp με την δημιουργία του. Αρχικά, για την εμφάνιση των προσφορών επαναχρησιμοποιείται η μέθοδος getAllTrOptions(), την οποία ξανά είδαμε στην εμφάνιση των φθηνότερων προσφορών (λειτουργία για απλούς χρήστες). Επιλέγουμε μια από τις παραπάνω και πατώντας Delete Offer εκτελείται DELETE FROM transportation όπου t_id = ο κωδικός της προσφοράς που επιλέχτηκε. (χρειάζεται να πατηθεί το αριστερό ώστε να εμφανιστούν όλα).



14)

Βασισμένη στην παραπάνω λογική-αλλά με διαφορετικές κλάσεις και μεθόδους- υλοποιήθηκε endpoint από το οποίο γίνεται να σβήσουμε κριτικές των χρηστών. Μέθοδος getAllServices() που αποθηκεύει object τύπου Review (δημιουργήθηκε constructor).
(πρέπει να πατηθεί για την εμφάνιση όλων των κριτικών)



15) Επαναχρησιμοποίηση του log-out για έξοδο του χρήστη.

Σημείωση:

Υπάρχουν λειτουργικότητες που δεν αναφέρθηκαν οι οποίες βοηθούν στην ομαλή λειτουργία του συστήματος (π.χ. στέλνουν δεδομένα σε μεθόδους οι οποίες έπειτα από ενέργειες, τα εμφανίζουν σε διάφορα endpoints)