

## **Ιστοσελίδα Δισκογραφικής Εταιρείας – Ομάδα 11**

Ανθή Ματθαιοπούλου, 1072732, 4<sup>ο</sup> έτος

Ίων Αθανάσιος Μερκούρη;, 1072741, 4<sup>ο</sup> έτος

### **Περιεχόμενα**

1.Εισαγωγή	2
2.Προσέγγιση του Προβλήματος	2
2.1 Λειτουργίες της Εφαρμογής	2
2.2 Εννοιολογικός Σχεδιασμός	3
3.Υλοποίηση	4
3.1 Front-end	4
3.2 Back-end	7
3.3 Βάση Δεδομένων	8
4. Οδηγίες Χρήσης και Εγκατάστασης	8
5. Χρονοδιάγραμμα	9
6. Αναφορές	10

## 1 Εισαγωγή

Το θέμα που μας ανατέθηκε ήταν η δημιουργία μιας δυναμικής και λειτουργικής ιστοσελίδας για μια Δισκογραφική Εταιρεία. Ονομάσαμε τη σελίδα μας Rio Records και σε αυτή μπορούμε να βρούμε όλους τους συνεργαζόμενους καλλιτέχνες καθώς και τα τραγούδια τους. Έχουμε φροντίσει να καλύψουμε τις βασικές ανάγκες των απλών χρηστών αλλά και των διαχειριστών της ιστοσελίδας. Στην παρούσα αναφορά, λοιπόν, αναλύονται καθαρά οι λειτουργίες της, καθώς και το σκεπτικό και η προσέγγισή μας.

Μπορείτε επίσης να βρείτε όλο τον κώδικα εδώ:

<https://github.com/anthimat11/Record-Label-Project>

## 2 Προσέγγιση του Προβλήματος

### 2.1 Λειτουργίες της Εφαρμογής

Οι απλοί χρήστες (users) έχουν την δυνατότητα να δημιουργήσουν τον προσωπικό τους λογαριασμό και φυσικά να συνδεθούν σε αυτόν. Μπορούν επίσης να δουν όλους τους καλλιτέχνες που εκπροσωπούμε, να ακούσουν τα τραγούδια τους στο YouTube καθώς και να προσθέσουν τα αγαπημένα τους στην προσωπική τους playlist. Τελικά έχουν την δυνατότητα να επικοινωνήσουν μαζί μας μέσω κοινωνικών δικτύων ή ηλεκτρονικού ταχυδρομείου.

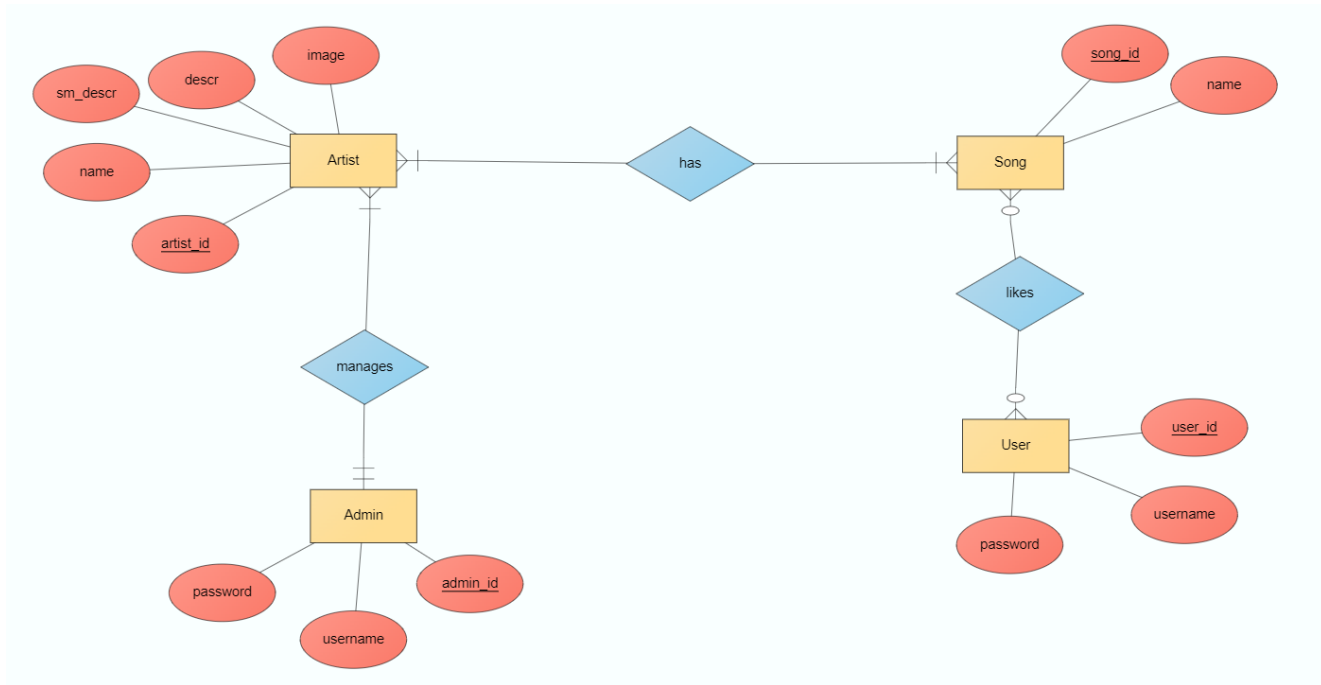
Οι χρήστες-διαχειριστές (admins) της ιστοσελίδας έχουν ξεχωριστούς ειδικούς λογαριασμούς και μπορούν να προστεθούν μόνο χειροκίνητα απευθείας στην βάση δεδομένων μας. Οι admin έχουν τη δυνατότητα φυσικά να προσθέσουν νέους καλλιτέχνες και τραγούδια ή να διαγράψουν κάποια προϋπάρχοντα.

Για να συνδεθείτε χρησιμοποιήστε τα παρακάτω στοιχεία:

- Ως Admin  
Username: admin  
Password: rio123
- Ως User  
Username: anthitsa  
Password: 123

## 2.2 Εννοιολογικός Σχεδιασμός

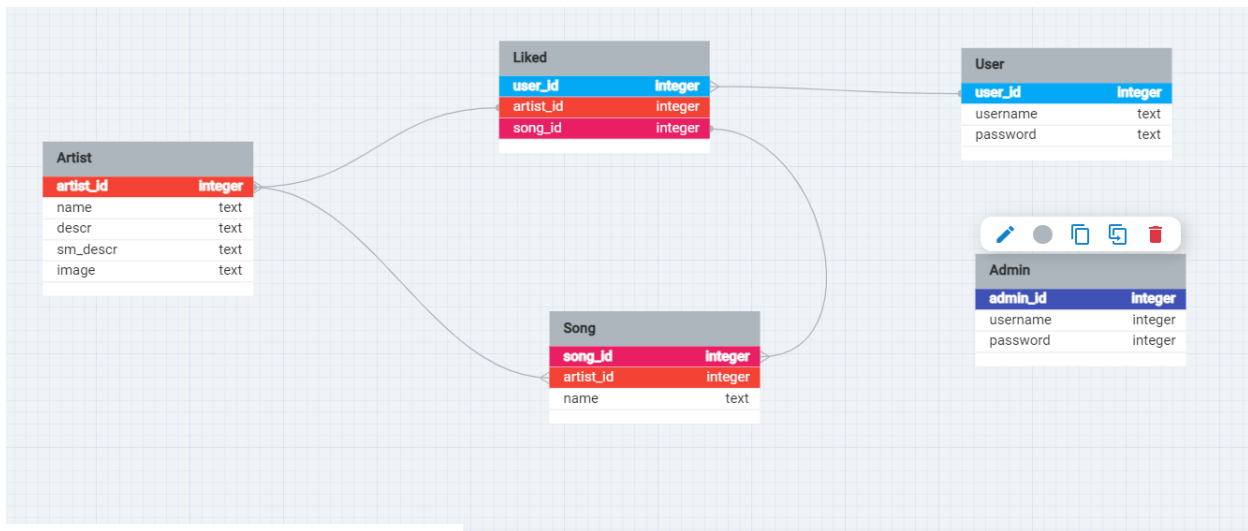
Παρακάτω μπορούμε να δούμε το βασικό εννοιολογικό μοντέλο που δημιουργήσαμε για τον σχεδιασμό της εφαρμογής μας με χρήση του ERDMaker [1]:



Εικόνα 1: Εννοιολογικό διάγραμμα

Είναι το βασικό και απλούστερο μοντέλο για μια εφαρμογή σαν την δική μας. Έχουμε αρχικά την οντότητα του Καλλιτέχνη (Artist) και του Τραγουδιού (Song) και σε αυτές έχουμε συμπεριλάβει τις απολύτως απαραίτητες ιδιότητες (όνομα, περιγραφή κλπ.). Παρατηρούμε επίσης και την οντότητα του Απλού Χρήστη (User) και εκείνη του Διαχειριστή (Admin), φυσικά αυτές αφορούν τους 2 τύπους χρηστών με προσωπικούς λογαριασμούς.

Μετατρέποντας το παραπάνω διάγραμμα σε Σχεσιακό Σχήμα με την βοήθεια του dbdesigner [2]:



Εικόνα 2: Σχεσιακό Σχήμα

Παρατηρούμε έναν παραπάνω πίνακα με όνομα Liked, ο πίνακας αυτός κρατάει στην μνήμη τα τραγούδια που έχει προσθέσει ένας απλός χρήστης στα αγαπημένα του. Με την βοήθεια του πίνακα αυτού είναι δυνατή η δημιουργία προσωποποιημένη Playlist για κάθε χρήστη.

Σημειώνουμε πως δεν κρατάμε στην βάση μας ποιος Διαχειριστής έχει προσθέσει κάθε καλλιτέχνη και κάθε τραγούδι.

### 3 Υλοποίηση

Να σημειώσουμε πως η υλοποίηση του project γινόταν βήμα-βήμα συμβαδίζοντας με τις εβδομαδιαίες διαλέξεις και τα εργαστήρια. Αποφασίσαμε να ακολουθήσουμε την Model View Controller αρχιτεκτονική ώστε τα αρχεία μας να έχουν μια σαφή δομή και οργάνωση.

#### 3.1 Front-end

Τα εργαλεία που χρησιμοποιήσαμε για την δημιουργία του front-end μέρους είναι τα εξής : HTML5[3] που αργότερα διαμορφώθηκε σε Handlebars[4] , CSS3[5] , Bootstrap[6] και JavaScript[7].

Η γραφική διεπαφή της ιστοσελίδας χωρίζεται σε τρία βασικά μέρη. Πρώτα, το header (views > partials) το οποίο περιλαμβάνει το όνομα της εταιρείας μαζί με το logo μας και φυσικά το navigation bar που επιτρέπει στους χρήστες να περιηγούνται στις διάφορες σελίδες.

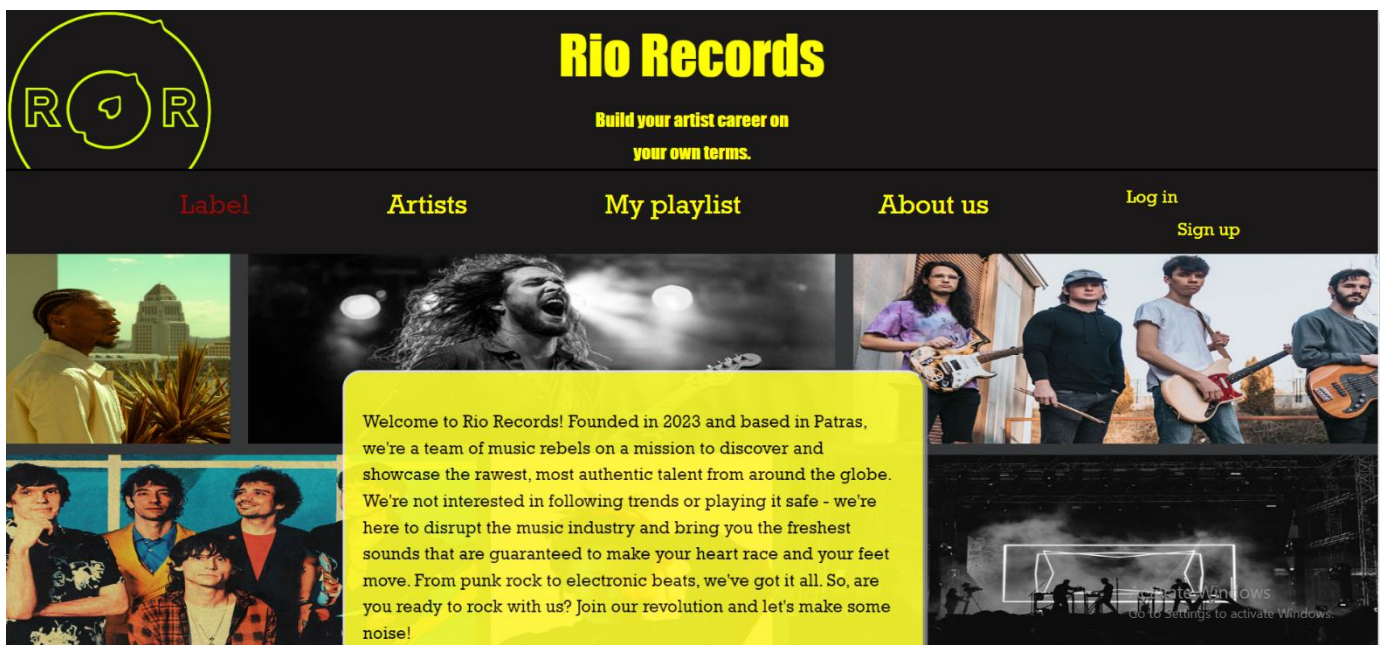
Στην μπάρα αυτή μπορεί κανείς να δει 5 διαφορετικές σελίδες.

- Η Label περιλαμβάνει την κεντρική μας σελίδα που παρουσιάζονται κάποιες πληροφορίες της δισκογραφικής εταιρείας.
- Η Artists παρουσιάζει με μορφή λίστας όλους τους καλλιτέχνες που συνεργάζονται με την Rio Records.
- Στην My Playlist μπορεί ο Απλός Χρήστης, αφού έχει συνδεθεί, να δει την λίστα με όλα τα αγαπημένα του τραγούδια καθώς και να την επεξεργαστεί.
- Στην About εξηγούμε συνοπτικά την λειτουργία της ιστοσελίδας
- Οι Log In/ Sign Up σελίδες επιτρέπουν τη σύνδεση σε λογαριασμό και τη δημιουργία λογαριασμού αντίστοιχα.

Το body, δεύτερο και κύριο μέρος, διαφέρει από σελίδα σε σελίδα καθώς περιλαμβάνει την ουσία της.

Παραδείγματος χάρη, μπορούμε να δούμε τα δύο αυτά μέρη της κεντρικής μας σελίδας:

Εικόνα 3: Αρχική Σελίδα της Rio Records



Οι σελίδες που έχουμε δημιουργήσει είναι οι εξής:

- `label.hbs` : η κύρια σελίδα στην οποία δεν υπάρχει φόρτωση κάποιων δεδομένων,
- `artists.hbs` : φορτώνονται δυναμικά όλοι οι καλλιτέχνες και ανάλογα τον χρήστη μπορούν να διαγραφούν (admin),
- `artistSongs.hbs` : φορτώνονται δυναμικά όλα τα τραγούδια του εκάστοτε καλλιτέχνη και ανάλογα τον χρήστη μπορούν να προστεθούν νέα, να διαγραφούν (admin) ή να προστεθούν σε προσωπική playlist (user),
- `artistForm.hbs` : εδώ έχουν πρόσβαση μόνο οι διαχειριστές, είναι η φόρμα για την προσθήκη νέου καλλιτέχνη στην βάση,
- `myPlaylist.hbs` : έχουν πρόσβαση μόνο οι συνδεδεμένοι απλοί χρήστες, εδώ μπορούν να δουν όλα τα τραγούδια που έχουν προσθέσει στην λίστα τους,
- `about.hbs` : η σελίδα που εξηγεί περιεκτικά το project αυτό,
- `contact.hbs` : η σελίδα που περιέχει τα βασικά στοιχεία επικοινωνίας της ιστοσελίδας, ο σύνδεσμος της σελίδας αυτής βρίσκεται στον footer,
- `login.hbs` : η σελίδα για σύνδεση των χρηστών στον λογαριασμό τους,
- `signup.hbs` : η σελίδα για την δημιουργία νέων λογαριασμών.

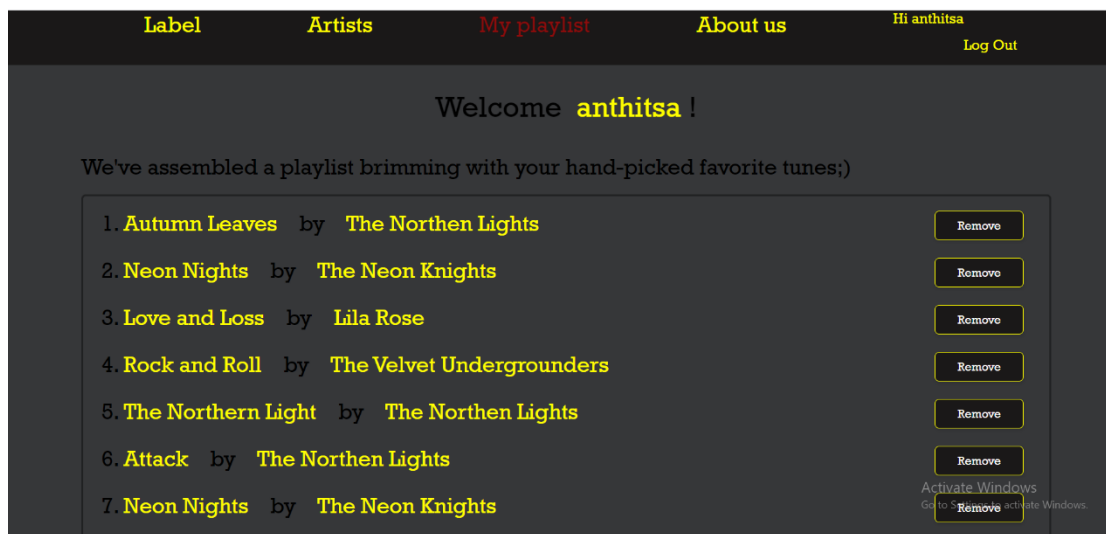
Τελευταία μέρος είναι ο footer (views > partials), ο οποίος περιέχει κάποια βασικά στοιχεία για την εταιρεία όπως τοποθεσία, τα κοινωνικά δίκτυα καθώς και τη σελίδα Contact Us με τα στοιχεία επικοινωνίας.

Μπορούμε να βρούμε τον `hbs` κώδικα για όλες τις σελίδες στον φάκελο views ενώ την CSS καθώς και τις εικόνες στον public φάκελο.

Μερικά ακόμα στιγμιότυπα φαίνονται εδώ:

Εικόνα 4: Φόρμα προσθήκης νέου καλλιτέχνη

The screenshot shows a web application interface with a dark theme. At the top, there is a navigation bar with the following elements: 'Label', 'Artists', 'My playlist', 'About us', and a user profile section for 'Hi admin' with a 'Log Out' link. The main content area features a form titled 'Add new Artist:'. The form has three input fields: 'Name' with the placeholder 'Enter the name of the new artist', 'Description' with the placeholder 'Enter a small description of the artist', and 'Image' with the placeholder 'Enter the url of the artist's image'. Below these fields is a black 'Add' button. In the bottom right corner of the page, there is a Windows watermark that says 'Activate Windows Go to Settings to activate Windows.'



Εικόνα 5: Προσωπική λίστα τραγουδιών του χρήστη anthitsa

### 3.2 Back-end

Τα εργαλεία που χρησιμοποιήσαμε για την δημιουργία του back-end μέρους είναι τα εξής :

Node.js[8], Express[9] και φυσικά JavaScript.

Ακολουθώντας τα μαθήματα κάθε εβδομάδα και με την βοήθεια των εργαστηρίων, καταφέραμε να υλοποιήσουμε όλες τις βασικές λειτουργίες της ιστοσελίδας μας.

Αρχικά, δημιουργήσαμε όλες τις διαδρομές των σελίδων. Μπορούμε να βρούμε όλες τις διασυνδέσεις στον φάκελο routes. Το αρχείο index.js αποτελεί τον κύριο κόμβο των διαδρομών αυτών.

Έπειτα, καθώς δεν γνωρίζαμε ακόμα πως να δουλέψουμε με μια πραγματική βάση δεδομένων, δημιουργήσαμε ένα αρχείο JavaScript το οποίο περιείχε πίνακες που χρησιμοποιήσαμε ως dummy database. Με την βοήθεια του αρχείου αυτού ξεκινήσαμε να δουλεύουμε τον φάκελο controllers. Στον φάκελο αυτό μπορούμε να βρούμε ξεχωριστά αρχεία για κάθε σελίδα που ουσιαστικά επιτελούν τις λειτουργίες. Για παράδειγμα, φόρτωση όλων των καλλιτεχνών, σύνδεση και δημιουργία λογαριασμού, προσθήκη νέου καλλιτέχνη κλπ. Για να ελέγξετε κάποια λειτουργία αρκεί να ανοίξετε τον αντίστοιχο αρχείο.

Όταν εξοικειωθήκαμε με την χρήση βάσης δεδομένων στο εργαστήριο, την προσαρμόσαμε στις παραπάνω ανάγκες.

Τελικά, η υλοποίηση του server ολοκληρώθηκε με την δημιουργία συνεδριών (sessions) και των cookies για κάθε χρήστη που συνδέεται στον λογαριασμό του. Αφού δημιουργηθεί μια επιτυχής σύνδεση μπορούμε τότε να ορίσουμε τις μεταβλητές:

- req.session.loggedUsername
- req.session.loggedUserId
- req.session.UserType (admin or user)

Με αυτόν τον τρόπο μπορούμε να βλέπουμε στο session τα χαρακτηριστικά του συνδεδεμένου χρήστη. Τα sessions έχουν χρόνο ζωής  $2000 * 60 * 60$  milliseconds και αποθηκεύονται ξεχωριστά στο αρχείο `model>sessions>sessions.sqlite`.

### 3.3 Βάση Δεδομένων

Υλοποιήσαμε την βάση μας με την γνωστή ήδη από το προηγούμενο εξάμηνο τεχνολογία SQLite. Η παραγωγή των δεδομένων έγινε με ένα μικρό αρχείο κώδικα `dbInit.js` το οποίο αρχικοποιεί την βάση με τα πρώτα της δεδομένα, κυρίως καλλιτέχνες και τραγούδια. Με τον τρόπο αυτό προσθέσαμε και τον λογαριασμό ενός διαχειριστή. Τέτοιου είδους λογαριασμοί μόνο χειροκίνητα μπορούν να εισαχθούν.

Σημειώνουμε πως όλοι οι καλλιτέχνες δεν είναι πραγματικά πρόσωπα. Αυτοί, τα τραγούδια καθώς και οι περιγραφές τους έχουν επινοηθεί με την βοήθεια του ChatGPT [10].

Όλα τα επιπλέον δεδομένα, τα εισάγαμε κατευθείαν από την ιστοσελίδα. Σημειώνεται, πως τους κωδικούς όλων των χρηστών δεν του περνάμε απευθείας στην βάση αλλά τους κωδικοποιούμε με την βοήθεια του εργαλείου `bcrypt` [11] της Express.

Στον φάκελο `model` μπορούμε να εντοπίσουμε το αρχείο `model.js` το οποίο περιέχει όλες τις συναρτήσεις που απαιτούνται για την επικοινωνία με την βάση δεδομένων.

## 4 Οδηγίες Χρήσης και Εγκατάστασης



Η ιστοσελίδα δεν βρίσκεται σε δημόσιο url οπότε για να την τρέξετε θα χρειαστεί να εκτελέσετε τα παρακάτω βήματα στο terminal του φακέλου της εργασίας:

- npm install hbs
- npm install bcrypt
- npm install express
- npm install express-session
- npm install express-handlebars
- npm install connect-sqlite3

Αφού κατεβάσατε τα πακέτα αυτά τρέξτε ένα τελευταίο npm install και τέλος npm start και ακολουθήστε τον σύνδεσμο [HTTP://LOCALHOST:3000/](http://localhost:3000/).

## 5 Χρονοδιάγραμμα

Όπως ήδη αναφέραμε η πρόοδος της εφαρμογής μας συμβάδιζε με τις διαλέξεις και τα εργαστήρια. Παρακάτω φαίνεται ένα ενδεικτικό χρονοδιάγραμμα:

Περίοδος	Στάδιο
Μάρτιος	Εννοιολογικός Σχεδιασμός
Απρίλιος	Ανάπτυξη του front-end της εφαρμογής
Μάιος	Ανάπτυξη back-end, routes, controllers, database

## 6 Αναφορές

- [1] ERD maker, <https://hci.ece.upatras.gr/erdmaker/designer>
- [2] Database Designer, <https://erd.dbdesigner.net/>
- [3] Html5 MDN Web Docs, <https://developer.mozilla.org/en-US/docs/Glossary/HTML5>
- [4] Handlebars, <https://handlebarsjs.com/>
- [5] CSS3 MDN Web Docs, <https://developer.mozilla.org/en-US/docs/Web/CSS/Reference>
- [6] Bootstrap, <https://getbootstrap.com/>
- [7] JavaScript MDN Web Docs, <https://developer.mozilla.org/en-US/docs/Web/JavaScript>
- [8] Node.js, <https://nodejs.org/en>
- [9] Express npm, <https://www.npmjs.com/>
- [10] ChatGPT, <https://chat.openai.com/>
- [11] bcrypt npm, <https://www.npmjs.com/package/bcrypt>