

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ, ΕΚΠΑ  
Τεχνολογίες εφαρμογών διαδικτύου.

Σοφία Μερόπη Κατσαούνη AM: 1115201800070

Δαμιανός Σκαρτσίλας AM: 1115201800180

## Περιεχόμενα

Εισαγωγή.....	3
Τι χρησιμοποιήθηκε στην εργασία.....	3
Πως κάνω launch την σελίδα.....	3-4
Επεξήγηση αρχείων frontend.....	4-5
Επεξήγηση αρχείων backend.....	5
Ελλείψεις.....	5
Επίλογος.....	5-6

## Εισαγωγή

Στόχος της εργασίας είναι η ανάπτυξη μιας επαγγελματικής εφαρμογής δημοπρασιών όπου οι χρήστες αλληλεπιδρούν σε πραγματικό χρόνο. Ο καθένας αναλαμβάνει έναν ρόλο από τον οποίο προκύπτουν οι ανάλογες δυνατότητες χρήσης της εφαρμογής από τον χρήστη.

Στα επόμενα περιεχόμενα θα δοθούν κατα κύριο λόγο λεπτομέρειες υλοποίησης, τρόποι οργάνωσης παραδοχές / ελλείψεις που έχει η εργασία μας. Κάθε αρχείο είναι εμπλουτισμένο με σχόλια έτσι ώστε ο κώδικας να είναι πιο κατανοητός σε τρίτο άτομο.

email επικοινωνίας για οποιαδήποτε απορία:

Σοφία, [sdi1800070@di.uoa.gr](mailto:sdi1800070@di.uoa.gr)

Δαμιανός, [sdi1800180@di.uoa.gr](mailto:sdi1800180@di.uoa.gr)

## Τι χρησιμοποιήθηκε στην εργασία

Για το frontend χρησιμοποιήθηκε react 18.2.0 και για το backend χρησιμοποιήθηκε Spring Boot 2.7.2. Η βάση δεδομένων είναι υλοποιημένη με SQL. Το λειτουργικό σύστημα στο οποίο έγινε ανάπτυξη της εργασίας είναι windows 10. Το backend αναπτύχθηκε με IntelliJIdea και το frontend αναπτύχθηκε με visual studio code. Παρατηρούσαμε τις αλλαγές στην βάση δεδομένων μέσω MySQL 8.0 Command Line Client.

## Πως κάνω launch την σελίδα

Αρχικά πρέπει να σιγουρευτεί κανείς πως

1. υπάρχει το schema με όνομα auctionweb στο MySQL monitor. Δημιουργείται με την εντολή **create database auctionweb;** στο command line

client. (Έχουμε στον φάκελο sql ένα αρχείο με όνομα auctionWebsite.sql που έχει όλους τους πίνακες που θα υπάρχουν μέσα στο schema και μπορεί κανείς να το τρέξει με την εντολή **source <absolute path to auctionWebsite.sql>** για παράδειγμα:

**source C:\Users\sofia\Desktop\auction-website\project\sql\auctionWebsite.sql**  
όμως δεν είναι απαραίτητο αυτό το βήμα εφόσον φτιάχνει τα tables αυτόματα το Spring Boot.)

2. Στο application.properties αρχείο που βρίσκεται στον φάκελο project\backend\src\main\resources θα πρέπει να δοθεί ο κωδικός που έχετε δώσει εσείς στην mysql όταν την κατεβάσατε, έτσι ώστε να έχει πρόσβαση στη βάση δεδομένων.

3. Εφόσον το backend αναπτύχθηκε σε IntelliJ, ανοίγοντας τον φάκελο και αφού φορτώσουν όλες οι κατάλληλες εξαρτήσεις, απλα πατάμε το **RUN** του editor και το backend δεσμεύει την διεύθυνση localhost:8080.

4. Εφόσον το frontend αναπτύχθηκε σε visual studio code, απλα ανοίγοντας το terminal και ενώ είμαστε στον φάκελο frontend του project, πατάμε την εντολή **npm install react-scripts --save --force** (για να κατεβουν τα node-modules) και στην συνέχεια την εντολή **npm start**. Δεσμεύεται η διεύθυνση localhost:3000 η οποία ανοίγει αυτόματα και στον browser που της επιλογής μας.

5. Για την εισαγωγή mock δεδομένων και ενός admin στην ιστοσελίδα είναι απαραίτητο να τρέξετε την εντολή **source <absolute path to data.sql file>** π.χ.  
**source C:\Users\sofia\Desktop\auction-website\project\sql\data.sql**

!!!! Για την σύνδεση στην ιστοσελίδα ο κωδικός του admin είναι **letmein** και ο κωδικός όλων των υπόλοιπων χρηστών είναι **letuserin** !!!!

## Επεξήγηση αρχείων frontend

Το αρχείο App.js του φακέλου frontend λειτουργεί σαν main function. Εκεί υπάρχουν όλα τα Routes απο τις σελίδες. Στον φάκελο components υπάρχουν τα components του frontend και καθέναν απο αυτά τα αρχεία αντιπροσωπεύει μια διαφορετική σελίδα του site. Ο φάκελος service λειτουργεί σαν "συνδέτης"

του frontend με το backend καθώς κάνει τα http calls για κάθε διεύθυνση του backend και συνήθως οι συναρτήσεις του φακέλου αυτού λαμβάνουν ένα JSON αντικείμενο σαν απάντηση. Κάθε αρχείο περιλαμβάνει σχόλια που εξηγούν ακριβώς τι συμβαίνει σε κάθε σημείο του κώδικα.

## **Επεξήγηση αρχείων backend**

Τα αρχεία χωρίζονται σε 7 φακέλους + το αρχείο της main συνάρτησης του project. Εκτός από την κλασική δομή ενός backend project σε spring boot (model, repository(Repo), controller) έχουμε και τους φακέλους exception, req, res και security. Τα αρχεία του exception φακέλου είναι για να εμφανίζονται exception όταν έχουμε λογικά λάθη στο πρόγραμμα, οι φάκελοι req , res και security είναι όλοι χρήσιμοι για το jwt authentication του χρήστη όταν κάνει εγγραφή ή σύνδεση στην πλατφόρμα. Το refresh token διαρκεί 10 μέρες. Η κάθε δημοπρασία μένει ενεργή για 5 μόλις λεπτά.

Για την δομή δεδομένων όπως αναφέρθηκε παραπάνω υπάρχει ο φάκελος sql που περιέχει τα αρχεία auctionWebsite.sql και data.sql και είναι απαραίτητη η χρήση μόνο του data.sql αρχείου.

## **Ελλειψεις**

Δεν υλοποιήθηκαν τα 2 τελευταία ερωτήματα δηλαδή η ανταλλαγή μηνυμάτων μεταξύ χρηστών και η εξαγωγή των δημοπρασιών σε xml και json μορφή.

## **Επίλογος**

Σε γενικές γραμμές ήταν ένα ενδιαφέρον πρότζεκτ που μας προσέφερε πολλές χρήσιμες γνώσεις σχετικά με τις πρόσφατες τεχνολογίες του διαδικτύου, Αυτό που μας δυσκόλεψε περισσότερο από όλα ήταν σίγουρα το jwt authentication των χρηστών, ιδιαιτέρως το backend κομμάτι του. Επίσης μας δυσκόλεψε αρκετά το bonus ερώτημα και γι αυτό αποφασίσαμε να δώσουμε γραπτή εξέταση. Σχετικά με τα 2 τελευταία ερωτήματα απλά δεν υπήρξε σωστή οργάνωση χρόνου.