

Εμφάνιση Tickets

Σκοπός της άσκησης είναι η δημιουργία μίας web-based εφαρμογής με την οποία ο χρήστης να μπορεί να έχει πρόσβαση σε λίστα με τα τελευταία tickets. Ο χρήστης της εφαρμογής, μέσα από έναν web-browser, θα έχει πρόσβαση στη λίστα των tickets.

Κάθε ticket περιγράφεται από τα παρακάτω πεδία:

- Id: μοναδικό ticket id
- Name: η περιγραφή του ticket.
- StartDate: ημερομηνία έναρξης
- DueDate: ημερομηνία λήξης
- Duration: διάρκεια σε ώρες
- Status: το status του ticket:
 - Open=0,
 - InProgress=1,
 - Finished=2,
- Assignee: ο άνθρωπος στον οποίο έχει ανατεθεί το ticket
- TaxId: το ΑΦΜ του ανθρώπου στον οποίο έχει ανατεθεί το ticket

Λειτουργικότητα

Η πρώτη οθόνη της εφαρμογής θα είναι μία login page στην οποία ο χρήστης θα εισάγει username και password. Σε περίπτωση επιτυχίας ο χρήστης θα μεταφέρεται στην κύρια οθόνη της εφαρμογής. Σε περίπτωση αποτυχίας θα πρέπει να εμφανίζεται σε αυτόν ανάλογο μήνυμα.

Στη κύρια οθόνη της εφαρμογής ο χρήστης θα μπορεί να έχει πρόσβαση στη λίστα των tickets σε μορφή καρτέλας σύμφωνα με το παρακάτω υπόδειγμα καρτέλας:

Ticket Name

Id: 5
Start date: 2022/5/12
Due date: 2022/6/15
Assignee: Nick Pappas
Status: In Progress
Tax Id: 011111112

Κάθε καρτέλα θα αντιστοιχεί σε ένα ticket. Σε κάθε καρτέλα, το χρώμα της περιγραφής του ticket, καθώς και το χρώμα του αριστερού περιθωρίου αυτής θα καθορίζονται από το status:

- Open: red
- InProgress: blue
- Finished: green

Για κάθε ΑΦΜ θα γίνεται έλεγχος της ορθότητας του, βάση του αλγορίθμου ορθότητας του ΑΦΜ, και σε περίπτωση αποτυχίας του ελέγχου η τιμή του ΑΦΜ θα πρέπει να έχει κόκκινο χρώμα.

Αλγόριθμο ορθότητας ΑΦΜ θα βρείτε στο:

<https://tatief.wordpress.com/2008/12/29/%CE%B1%CE%BB%CE%B3%CF%8C%CF%81%CE%B9%CE%B8%CE%BC%CE%BF%CF%82-%CF%84%CE%BF%CF%85-%CE%B1%CF%86%CE%BC-%CE%AD%CE%BB%CE%B5%CE%B3%CF%87%CE%BF%CF%82-%CE%BF%CF%81%CE%B8%CF%8C%CF%84%CE%B7%CF%84%CE%B1%CF%82/>

ή σε άλλα web sites

Ο χρήστης, πατώντας σε κάποιο κουμπί στη πάνω δεξιά γωνία της οθόνης, θα μπορεί να μεταβάλλει τον τρόπο εμφάνισης της λίστας των tickets από καρτέλες σε μορφή grid και αντίστροφα.

Ο χρήστης, κάνοντας διπλό click πάνω στη περιγραφή ενός ticket θα μεταβάλλει το status του ticket σε «Finished». Η συγκεκριμένη μεταβολή θα γίνεται με κλήση στο server (περιγράφεται παρακάτω) και σε περίπτωση επιτυχούς αλλαγής στο server θα μεταβάλλεται και το status στον client (λεκτικό status και χρώματα).

Η κύρια σελίδα της εφαρμογής θα πρέπει να είναι responsive: Σε μεγάλη οθόνη οι καρτέλες των tickets θα εμφανίζονται ανά γραμμές, και κάθε γραμμή θα περιέχει όσα tickets μπορούν να χωρέσουν σε αυτή. Αντίθετα σε μικρές οθόνες κινητών τα tickets θα εμφανίζονται το ένα κάτω από το άλλο.

Τεχνολογίες

Οι τεχνολογίες που θα πρέπει να χρησιμοποιηθούν στην υλοποίηση της άσκησης είναι: **React with javascript, HTML, Css**. Οποιαδήποτε υλοποίηση η οποία δε θα στηρίζεται στις παραπάνω τεχνολογίες θα θεωρείται άκυρη.

Μπορείτε να χρησιμοποιήσετε οποιοδήποτε library, add-on, template ή third-party component κτλ που βασίζεται στις παραπάνω τεχνολογίες και θεωρείτε ότι μπορεί να προσθέσει μεγαλύτερη αξία στη προσπάθειά σας.

Rest Server

Η λήψη όλων των δεδομένων που απαιτούνται από την εφαρμογή (λίστα των tickets, login, κτλ) θα γίνονται με χρήση rest calls σε συγκεκριμένο service.

Service Base url: <https://frontendtest.unixfor.gr>

Κλήσεις:

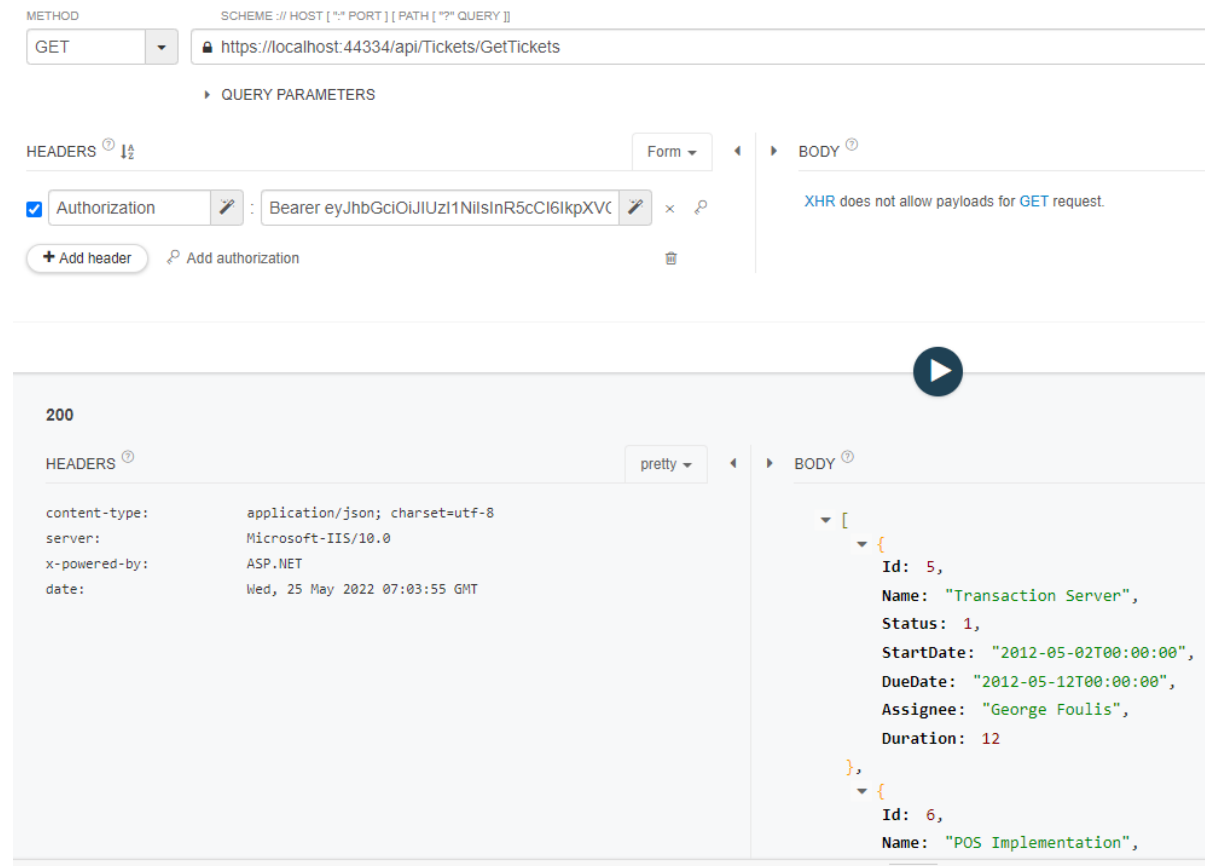
Κλήση	url	Περιγραφή
login	/api/Tickets/login	Login χρήστη με χρήση username και password
Get Tickets	/api/Tickets/GetTickets	Λήψη των τελευταίων tickets
Update Status	/api/Tickets/UpdateStatus	Μεταβολή του status ενός ticket. Γίνεται post το id του ticket και το νέο status.

Η κλήση login είναι ανώνυμη, και μεταξύ άλλων επιστρέφει ένα jwt token. Οι υπόλοιπες κλήσεις δεν είναι ανώνυμες και απαιτείται να εισάγεται σε αυτές Authorization header με τιμή:

Bearer <token>

Όπου <token> το jwt token που επέστρεψε η login.

Ακολουθεί παράδειγμα κλήσης της Get Tickets από κάποιον free rest client:



Στο παραπάνω παράδειγμα το token eyJhbGc... έχει ληφθεί από την κλήση login που έχει προηγηθεί.

Επιπλέον πληροφορίες για τα μοντέλα που γίνονται post στο server καθώς και για τα μοντέλα που επιστρέφει ο server για κάθε κλήση θα βρείτε στη σελίδα του swagger:

<https://frontendtest.unixfor.gr/swagger/index.html>

Θα εκτιμηθούν ιδιαίτερα :

- ο καλά δομημένος και τεκμηριωμένος κώδικας (OOP principles, not smell code, comments, κτλ) .
- η σωστή χρήση components και services
- η χρήση dependency injection.

Παραδοτέα

Θα πρέπει να παραδώσετε τα παρακάτω:

- Τον πλήρες κώδικα που υλοποιήσατε
- Σύντομη αναφορά των τεχνολογιών που χρησιμοποιήσατε καθώς και οδηγίες εγκατάστασης
- Άλλες οδηγίες που εσείς θεωρείται ότι θα βοηθήσουν στην καλύτερη αξιολόγηση της δουλειά σας.

Η Unixfor δεσμεύεται ότι τα παραπάνω παραδοτέα θα είναι εμπιστευτικά και θα χρησιμοποιηθούν μόνο για την αξιολόγηση του συγγραφέα.

Καλή επιτυχία !!!!