

Έκθεση ομαδικής εργασίας – Ομάδα 7

Εφαρμογή αναφοράς βλαβών σε υποδομές

Οδυσσέας Κοπακάκης Μπέλμπας

Τμήμα Ηλεκτρολόγων Μηχανικών και Τεχνολογίας Υπολογιστών, Πανεπιστήμιο Πατρών, up1072653@upnet.gr

Ιωάννης Νικολάου

Τμήμα Ηλεκτρολόγων Μηχανικών και Τεχνολογίας Υπολογιστών, Πανεπιστήμιο Πατρών, up1072681@upnet.gr

Στην παρούσα αναφορά περιγράφεται και αναλύεται το πρόβλημα που πραγματεύεται η εργασία μας, το οποίο είναι η δημιουργία μιας εφαρμογής αναφοράς βλαβών σε υποδομές στα πλαίσια του μαθήματος Προγραμματισμός Διαδικτύου (ECE_ΓΚ802) 2022-2023. Στη συνέχεια ακολουθούν στοιχεία της λύσης μας μαζί με τα αποτελέσματα που προέκυψαν. Ακόμα αναγράφονται η μεθοδολογία με την οποία προσεγγίστηκε το συγκεκριμένο θέμα, η αξιολόγηση της τελικής λύσης και το χρονοδιάγραμμα με όλες τα βασικά υποέργα της εργασίας. Τέλος υπάρχει παράρτημα με οδηγίες εγκατάστασης, παραδείγματα χρήσης του προγράμματος και διάφοροι χρήσιμοι σύνδεσμοι.

1 ΕΙΣΑΓΩΓΗ

Η διαδικτυακή πλατφόρμα DamageTrack έρχεται να αλλάξει τον τρόπο με τον οποίο αναφέρουμε και ελέγχουμε βλάβες σε υποδομές, βελτιώνοντας την ασφάλεια και ενισχύοντας την αποτελεσματικότητα των επιδιορθωτικών διεργασιών σε όλη την Ελλάδα. Μέσω της ιστοσελίδας αυτής οι χρήστες θα μπορούν να αναφέρουν τις ζημιές που παρατηρούν σε εξωτερικούς χώρους και δυσκολεύουν την καθημερινότητά τους με αποτέλεσμα να ανατίθενται σε αξιόλογες εταιρείες για την ταχύτερη επιδιόρθωσή τους. Οι χρήστες θα έχουν τη δυνατότητα να επισημαίνουν με ψήφους έργα που βρίσκονται κοντά στην περιοχή τους με αποτέλεσμα να γνωρίζει ο διαχειριστής της εταιρείας και οι εταιρείες σε ποια έργα να δώσουν τη μεγαλύτερη προτεραιότητα.

1.1 ΜΕΘΟΔΟΛΟΓΙΑ

Η παρούσα εργασία χωρίστηκε στις εξής φάσεις:

- Φάση Α: Ανάλυση Απαιτήσεων
- Φάση Β: Σχεδίαση Διεπαφών
- Φάση Γ: Προγραμματισμός Εξυπηρετητή
- Φάση Δ: Αξιολόγηση Ιστοσελίδας

1.2 ΧΡΟΝΟΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ

Ο χρονοπρογραμματισμός της συγκεκριμένης εργασίας μπορεί να φανεί εκτενώς από το github repository [1]. Από 18-21 Μαρτίου ασχοληθήκαμε με την ανάλυση των απαιτήσεων της εργασίας και κατασκευάσαμε το ER διάγραμμα της βάσης

δεδομένων μας. Από 24 Μαρτίου έως 3 Απριλίου ασχοληθήκαμε με τη κατασκευή των διεπαφών με HTML CSS Bootstrap και JavaScript. Χωρίσαμε τις 4 διεπαφές μεταξύ μας, ο Οδυσσέας ασχολήθηκε με τη φόρμα υποβολής αναφοράς και το admin dashboard κι ο Ιωάννης ασχολήθηκε με την αρχική σελίδα και τη σελίδα εγγραφής/σύνδεσης. Από τις 5-15 Μαΐου ασχοληθήκαμε με τον προγραμματισμό του server και την επικοινωνία της εφαρμογής με τη βάση δεδομένων. Ο καθένας μετέτρεψε τις διεπαφές που είχε αναλάβει σε handlebars κι επίσης δημιούργησε τα κατάλληλα routes για την εμφάνιση των διεπαφών αυτών με την κατάλληλη λογική κι επικοινωνία με τη βάση δεδομένων. Από τις 22-28 Μαΐου προστέθηκαν τα sessions στη βάση δεδομένων, έγιναν hash οι κωδικοί των χρηστών κι ανέβηκε η σελίδα στο Platform as a Service (PaaS) heroku ώστε να μπορεί να φιλοξενείται και να είναι προσβάσιμη από οποιονδήποτε χρήστη [2].

2 ΦΑΣΗ Α

2.1 Ανάλυση Απαιτήσεων

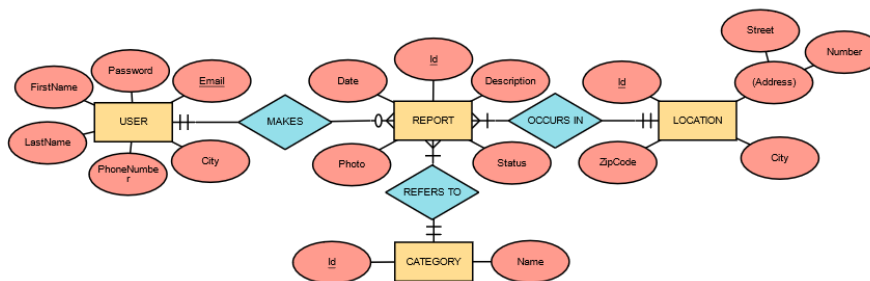
Η ανάλυση των απαιτήσεων της ιστοσελίδας έγινε με βάση το μοντέλο PACT [3]. Το People, Activities, Context και Technology (PACT) μοντέλο είναι ένα χρήσιμο framework για το σχεδιασμό εφαρμογών που αλλάζουν συμπεριφορά. Με βάση τα αρχικά αυτά μπορούμε να αναλύσουμε περαιτέρω την ιστοσελίδα μας και να κατανοήσουμε καλύτερα το πρόβλημα με το οποίο ερχόμαστε αντιμέτωποι.

- **People:** Η ιστοσελίδα αναφοράς βλαβών σε υποδομές είναι σχεδιασμένη για πολίτες της Ελλάδας που έρχονται αντιμέτωποι με τέτοιου είδους βλάβες στις περιοχές τους. Η ιστοσελίδα θα πρέπει να είναι φιλική ως προς τον χρήστη και διαθέσιμη για όλα τα ηλικιακά group και τεχνολογικά υπόβαθρα. Επίσης η ιστοσελίδα αυτή προορίζεται και για έναν διαχειριστή ο οποίος θα μπορεί να ενημερώνεται γι' αυτές τις βλάβες και να καθορίζει στη συνέχεια τον κατάλληλο τρόπο για την αντιμετώπισή τους.
- **Activities:** Η κύρια ενέργεια της ιστοσελίδας είναι η αναφορά μιας βλάβης σε υποδομή. Η ιστοσελίδα πρέπει να σχεδιαστεί έτσι ώστε οι χρήστες να μπορούν εύκολα και γρήγορα να αναφέρουν ζημιές σε υποδομές από τις περιοχές τους. Η δευτερεύουσα ενέργεια είναι η διαχείριση των βλαβών από τον διαχειριστή της σελίδας που θα μπορεί να αλλάζει την κατάσταση της προόδου όσον αφορά την ολοκλήρωση της επισκευής της βλάβης δίνοντας πάντα προτεραιότητα στις ανάγκες των περισσότερων χρηστών.
- **Context:** Η ιστοσελίδα πρέπει να σχεδιαστεί ώστε να μπορεί να χρησιμοποιηθεί σε διάφορα σενάρια. Οι χρήστες θα πρέπει να μπορούν να την χρησιμοποιήσουν όταν περπατάνε, βρίσκονται σε κάποιο εξωτερικό χώρο - μαγαζί ή ακόμα χρησιμοποιούν τα μέσα μαζικής μεταφοράς. Η ιστοσελίδα πρέπει να υλοποιηθεί με σκοπό να εισχωρήσει στην καθημερινότητα του χρήστη χωρίς να προκαλεί προβλήματα.
- **Technology:** Η ιστοσελίδα πρέπει να χρησιμοποιεί κατάλληλες τεχνολογίες ώστε να κάνει τη διαδικασία υποβολής της βλάβης εύκολη και βολική. Πρέπει να περιέχει διάφορα χαρακτηριστικά όπως, υπηρεσίες με βάση την τοποθεσία ώστε να αναγνωρίζει που βρίσκεται ο εκάστοτε χρήστης, τη δυνατότητα να αναρτώνται φωτογραφίες από τη βλάβη κι ενημερώσεις σε πραγματικό χρόνο για την κατάσταση της αναφερόμενης ζημιάς.

2.2 Καταγραφή μοντέλου δεδομένων και δοσοληψιών

2.2.1 Καταγραφή μοντέλου δεδομένων

Για την καταγραφή του μοντέλου δεδομένων έγινε ανάλυση του προβλήματος που μας δόθηκε ώστε να κατανοήσουμε τον μικρόκοσμό του. Χρησιμοποιήθηκε το μοντέλο ERD ώστε να γίνει πιο κατανοητή η σχέση μεταξύ των οντοτήτων που υπάρχουν στο πρόβλημά μας και η αλληλεπίδραση μεταξύ τους.



Σχήμα 1: Διάγραμμα Οντοτήτων – Συσχετίσεων [4]

2.2.2 Καταγραφή δοσοληψιών

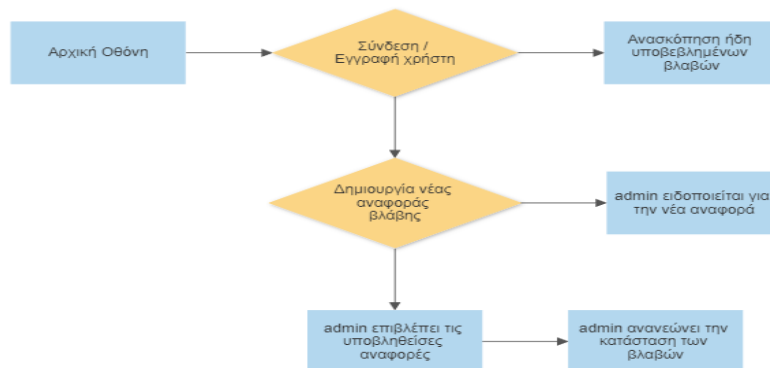
Οι δοσοληψίες είναι λογικές μονάδες εργασίας που αποτελούνται από μία ή περισσότερες λειτουργίες βάσης δεδομένων και εκτελούνται ως μια συνολική λειτουργία με σκοπό να διασφαλίζουν τη συνέπεια κι ακεραιότητα των δεδομένων ακόμα και σε περίπτωση σφάλματος ή λάθους. Παρακάτω βρίσκονται μερικές από τις πιο βασικές δοσοληψίες της ιστοσελίδας.

- CRUD (Create, Read, Update, Delete) λειτουργίες όσον αφορά το χρήστη και τις αναφορές. Αυτό σημαίνει ότι ο κάθε χρήστης θα μπορεί να δημιουργεί, να διαβάζει, να επεξεργάζεται και να διαγράφει αναφορές. Κάθε αναφορά βέβαια ανήκει σε έναν μόνο χρήστη οπότε μόνο αυτός που τη δημιούργησε θα μπορεί να εκτελεί όλες αυτές τις δοσοληψίες εκτός από το διάβασμα το οποίο θα μπορεί να εκτελεί κι οποιοσδήποτε άλλος χρήστης.
- Update της κατάστασης μιας συγκεκριμένης αναφοράς από το διαχειριστή της ιστοσελίδας που μπορεί να βρίσκεται στις καταστάσεις «εκκρεμής», «σε εξέλιξη» και «επιλύθηκε».
- Create στατιστικών από το διαχειριστή για τη συχνότητα και κατανομή των διαφόρων ειδών βλαβών που έχουν αναφερθεί, καθώς και την πρόοδο των αναφερόμενων ζημιών όσο αυτή βρίσκεται υπό επίλυση.
- Ειδοποιήσεις σε χρήστες για την κατάσταση των βλαβών που έχουν αναφέρει όταν ο διαχειριστής αλλάξει την κατάστασή τους.

2.3 Δημιουργία χάρτη πλοήγησης / σεναρίων χρήσης

2.3.1 Χάρτης Πλοήγησης

Για την καταγραφή των σεναρίων χρήσης της ιστοσελίδας δημιουργήθηκε το διάγραμμα ενός χάρτη πλοήγησης, μέσω του οποίου αποτυπώνονται η σειρά των ενεργειών που πραγματοποιούνται στην εφαρμογή από την στιγμή της επίσκεψης στην αρχική σελίδα.



Σχήμα 2: Χάρτης Πλοήγησης

2.3.2 Καταγραφή σεναρίων χρήσης

Τα σενάρια χρήσης προσφέρουν μια σαφή εικόνα των δυνατοτήτων και του σκοπού της ιστοσελίδας. Παρακάτω παρουσιάζεται η αναλυτική τους καταγραφή.

- Σύνδεση / Εγγραφή: Κάθε χρήστης της σελίδας μπορεί να δημιουργήσει τον δικό του λογαριασμό υποβάλλοντας τα απαραίτητα στοιχεία. Μετά την εγγραφή του έχει την δυνατότητα να συνδέεται με τον προσωπικό του λογαριασμό.
- Ο χρήστης έχει την δυνατότητα να δημιουργήσει μια νέα αναφορά βλάβης, στην οποία παραθέτει τις λεπτομέρειες για την βλάβη καθώς και φωτογραφικό υλικό. Επιπροσθέτως, μετά την σύνδεσή του, έχει πρόσβαση στις αναφορές που έχει δημιουργήσει ο ίδιος, καθώς και σε κάθε άλλη αναφορά ώστε να επισημάνει την σοβαρότητα της επιδιόρθωσης της βλάβης, αν το επιθυμεί.
- Ο admin ενημερώνεται, μέσω ενός notification system, για την κάθε καινούρια αναφορά βλάβης. Είναι ικανός να επιβλέψει κάθε υποβληθείσα βλάβη, να ανανεώσει την κατάστασή της και να παρακολουθήσει τα στατιστικά της ιστοσελίδας.

3 ΦΑΣΗ Β

3.1 Σχεδίαση Διεπαφών

3.1.1 Διαμόρφωση Σελίδων

Οι τεχνολογίες που χρησιμοποιήθηκαν για τη δημιουργία των διεπαφών είναι οι html, css, bootstrap και javascript. Χρησιμοποιήθηκε επίσης το AI discord bot Midjourney [5] για τη δημιουργία φωτογραφιών σχετικών με το θέμα της ιστοσελίδας. Κάθε σελίδα του ιστότοπου χωρίζεται σε τρία μέρη, την κεφαλίδα (header), την κύρια περιοχή (main) και το

υποσέλιδο (footer). Η κεφαλίδα μπορεί να έχει τρεις διαφορετικές μορφές αναλόγως ποιος χρήστης περιηγείται τη συγκεκριμένη χρονική στιγμή στην ιστοσελίδα.

Ένας χρήστης που δεν έχει δημιουργήσει λογαριασμό στον ιστότοπο μπορεί να έχει πρόσβαση μόνο στην αρχική σελίδα και στην κεφαλίδα εμφανίζεται μόνο το όνομα του ιστότοπου που αν το πατήσει ο χρήστης θα πραγματοποιηθεί refresh της σελίδας.

DamageTrack

Φωτογραφία 1: Όψη κεφαλίδας αρχικής σελίδας από την πλευρά μη αυθεντικοποιημένου χρήστη

Ένας χρήστης που έχει δημιουργήσει λογαριασμό στον ιστότοπο μπορεί να έχει πρόσβαση στην αρχική σελίδα και στην σελίδα υποβολής αναφοράς. Εφόσον συνδεθεί και αυθεντικοποιηθεί ο λογαριασμός του χρήστη, πλέον στην κεφαλίδα της αρχικής σελίδα θα προστεθεί μια νέα επιλογή υποβολής αναφορά που αν την πατήσει ο χρήστης θα μεταφερθεί στη σελίδα υποβολής αναφοράς. Σ' εκείνη τη σελίδα όμως η κεφαλίδα εμφανίζεται όπως στη Φωτογραφία 1.

DamageTrack

 **Submit Report**

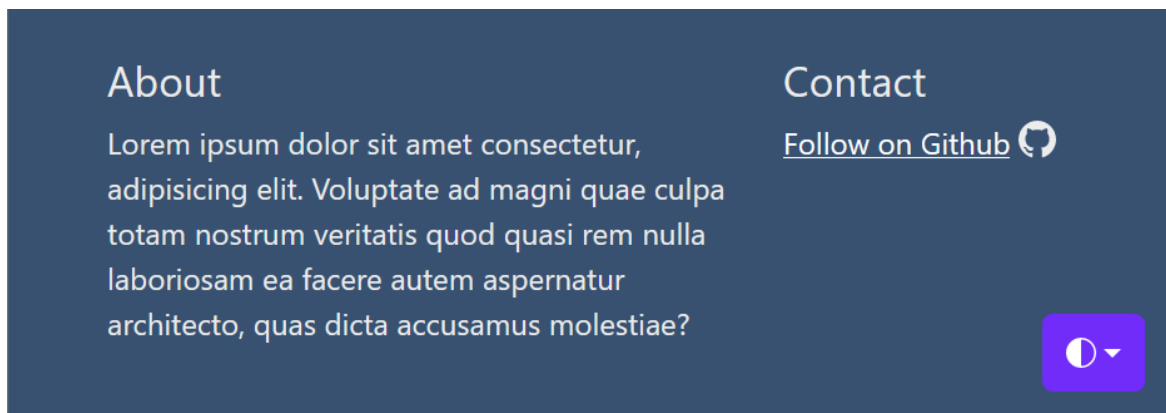
Φωτογραφία 2: Όψη κεφαλίδας αρχικής σελίδας από την πλευρά αυθεντικοποιημένου χρήστη

Ο διαχειριστής της σελίδας (admin) μπορεί να έχει πρόσβαση μόνο στην ειδική σελίδα διαχείρισης αναφορών (admin dashboard). Εφόσον συνδεθεί και αυθεντικοποιηθεί ο λογαριασμός του admin, πλέον η κεφαλίδα έχει τελείως διαφορετική όψη όπου περιλαμβάνει αριστερά της μια επιλογή για refresh του admin dashboard και δεξιά της μια επιλογή Logout για να αποσυνδεθεί ο διαχειριστής.

Dashboard Logout

Φωτογραφία 3: Όψη κεφαλίδας αρχικής σελίδας από την πλευρά του διαχειριστή

Το υποσέλιδο παραμένει ίδιο ανεξάρτητα σε ποια σελίδα βρίσκεται οποιοσδήποτε χρήστης περιηγείται στον ιστότοπο. Αποτελείται από ένα μικρό κείμενο σχετικά με τον ιστότοπο αυτό, ένα link του github repository κι ένα κουμπί που μπορεί να αλλάξει το χρώμα της σελίδας από dark σε light mode ή να σεβαστεί τη ρύθμιση συστήματος του περιηγούμενου χρήστη.



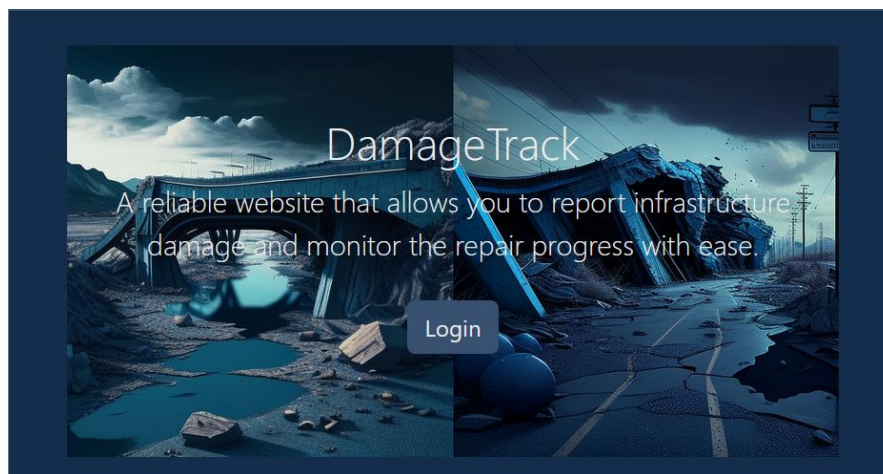
Φωτογραφία 4: Όψη υποσέλιδου

Τέλος απομένει η κύρια περιοχή κάθε σελίδας που διαφέρει ανάλογα με το ποιος είναι ο περιηγούμενος χρήστης και σε ποια σελίδα βρίσκεται τη συγκεκριμένη χρονική στιγμή. Η κάθε κύρια περιοχή θα παρουσιαστεί εκτεταμένα παρακάτω.

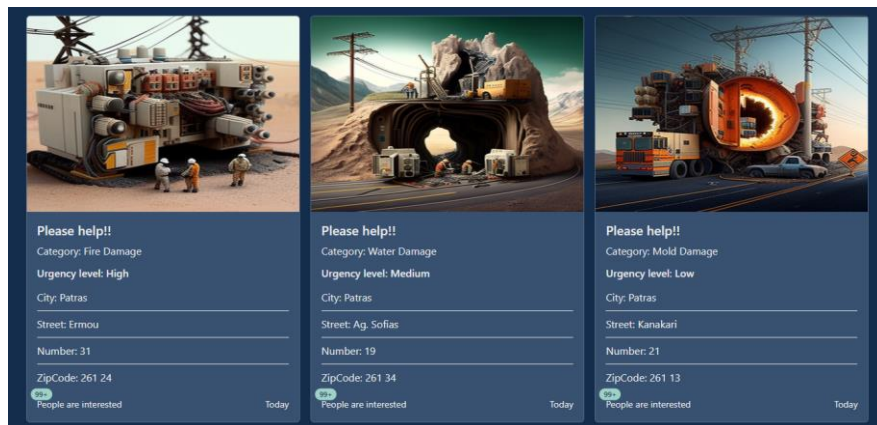
3.1.2 Αρχική Σελίδα

Η κύρια περιοχή της αρχικής σελίδας είναι διαφορετική αναλόγως αν είναι συνδεδεμένος κάποιος χρήστης στον ιστότοπο ή όχι. Αν δεν έχει συνδεθεί κάποιος χρήστης η κύρια περιοχή αποτελείται από τον τίτλο του ιστότοπου μαζί με μια μικρή περιγραφή του κι ακριβώς από κάτω ένα κουμπί με το οποίο ο χρήστης μπορεί να συνδεθεί. Πιο κάτω υπάρχουν μερικά παραδείγματα τυπικών αναφορών όπου μπορεί ο χρήστης πριν δημιουργήσει κάποιον λογαριασμό να έχει μια ιδέα για τη μορφή των αναφορών που ενδέχεται να ανεβάσει.

Η κάθε αναφορά περιέχει σημαντικά στοιχεία για τη βλάβη όπως την περιγραφή, την κατηγορία, αν επείγει η επίλυσή της, την τοποθεσία της, πότε δημιουργήθηκε και πόσοι άνθρωποι ενδιαφέρονται για τη συγκεκριμένη βλάβη.



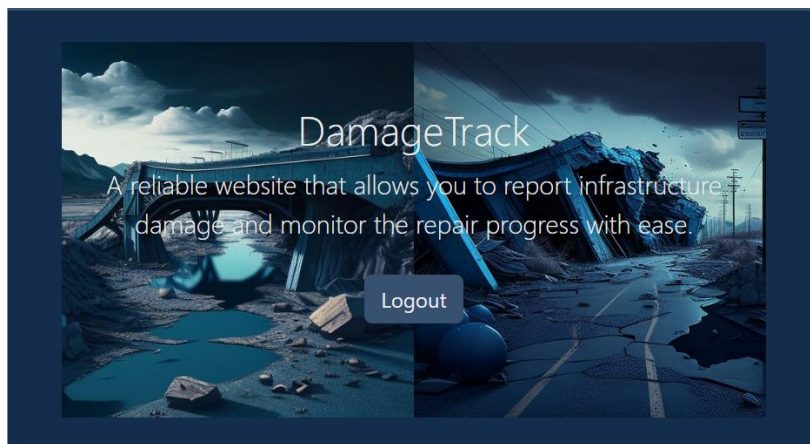
Φωτογραφία 5: Κύρια περιοχή αρχικής σελίδας (Α) από την πλευρά μη αυθεντικοποιημένου χρήστη



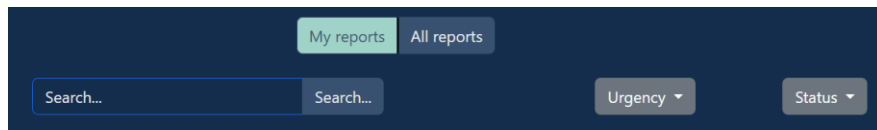
Φωτογραφία 6: Κύρια περιοχή αρχικής σελίδας (B) από την πλευρά μη αυθεντικοποιημένου χρήστη

Όταν ένας χρήστης συνδεθεί στη σελίδα και αυθεντικοποιηθεί τότε η κύρια περιοχή της αρχικής σελίδα έχει μια διαφορετική μορφή. Αρχικά στο πάνω μέρος της κύριας περιοχής εφόσον ο χρήστης έχει συνδεθεί πλέον αναγράφει Logout αντί για Login. Στη συνέχεια εμφανίζονται δύο κουμπιά που ο χρήστης μπορεί να δει τις αναφορές που έχει υποβάλει αλλά και τις αναφορές που έχουν υποβάλει και όλοι οι άλλοι εγγεγραμμένοι χρήστες της σελίδας. Ακόμα υπάρχουν φίλτρα κι ένα search box για αναζήτηση συγκεκριμένων αναφορών από του συνδεδεμένου χρήστη αλλά και μεταξύ όλων των εγγεγραμμένων χρηστών.

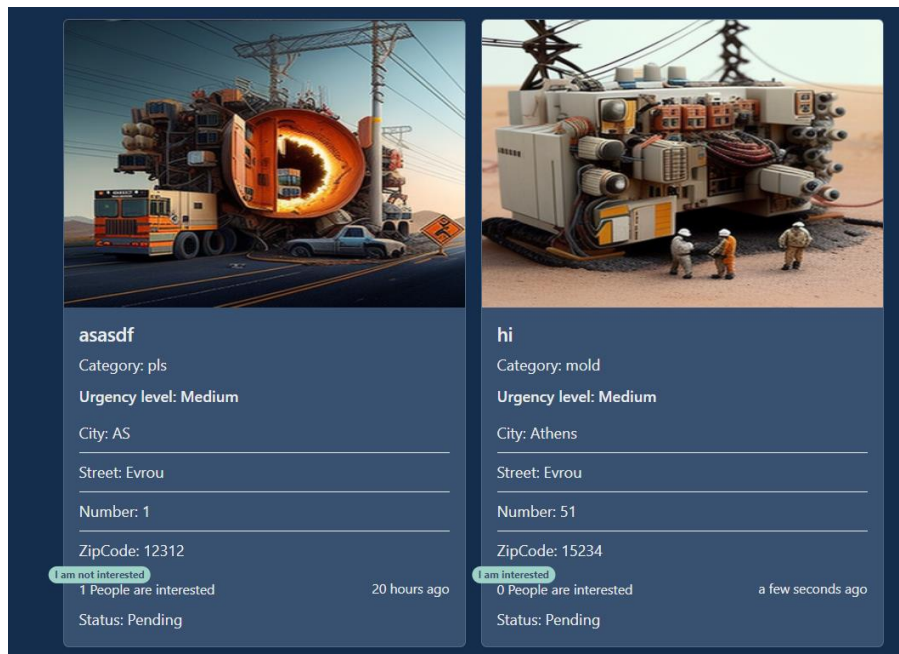
Επιπροσθέτως όταν ο χρήστης έχει επιλέξει να δει όλες τις αναφορές εμφανίζονται κουμπιά με τα οποία μπορεί να δηλώσει το ενδιαφέρον του ή όχι για ορισμένες βλάβες με σκοπό να αυξήσει την προτεραιότητα τους αν τον ενδιαφέρουν. Στο κάτω μέρος της κύριας περιοχής υπάρχει και pagination ώστε να μπορεί ο χρήστης να περιηγηθεί σε όλες τις διαθέσιμες αναφορές αναλόγως τα φίλτρα που έχει επιλέξει ώστε να μην εμφανίζονται όλες μαζί, κάτι που συμβάλλει στην πιο γρήγορη ανταπόκριση της ιστοσελίδας και τη διευκόλυνση της περιήγησης του χρήστη.



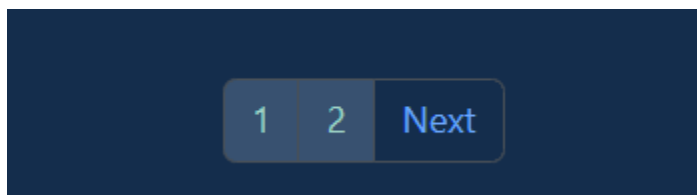
Φωτογραφία 7: Κύρια περιοχή αρχικής σελίδας (A) από την πλευρά αυθεντικοποιημένου χρήστη



Φωτογραφία 8: Κύρια περιοχή αρχικής σελίδας (Β) από την πλευρά αυθεντικοποιημένου χρήστη



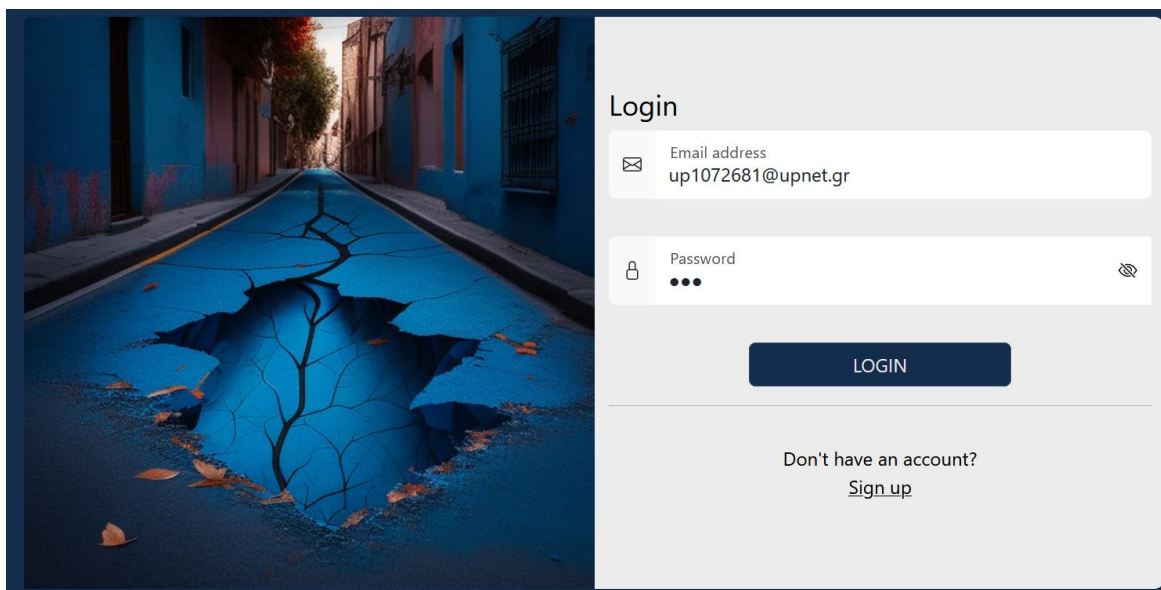
Φωτογραφία 9: Κύρια περιοχή αρχικής σελίδας (Γ) από την πλευρά αυθεντικοποιημένου χρήστη



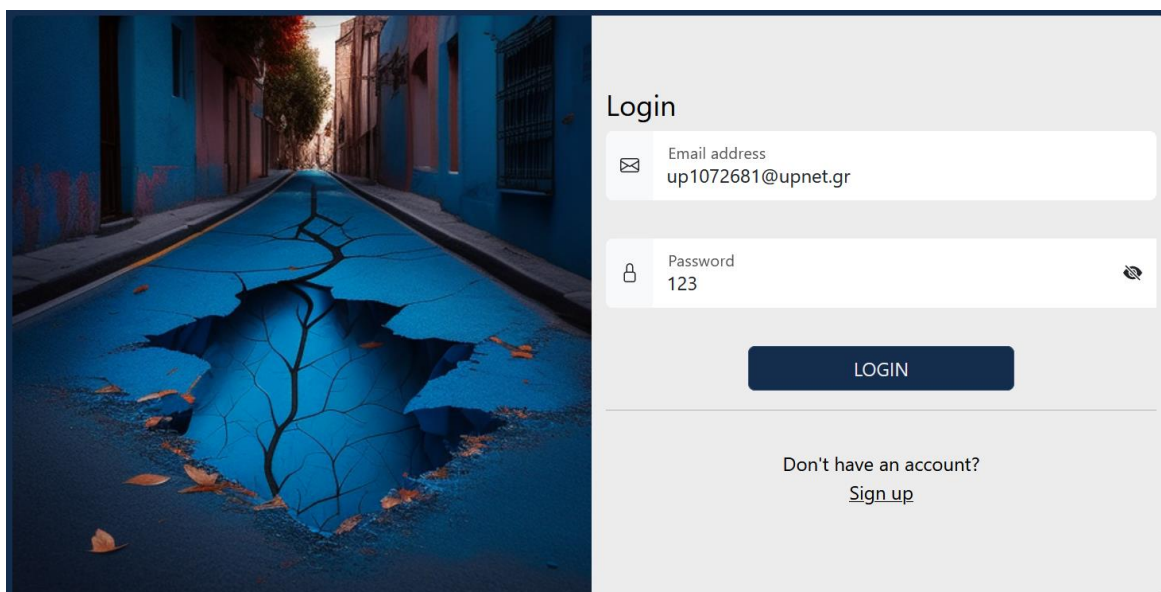
Φωτογραφία 10: Κύρια περιοχή αρχικής σελίδας (Δ) από την πλευρά αυθεντικοποιημένου χρήστη

3.1.3 Σελίδα Εγγραφής/Σύνδεσης Χρήστη

Στη σελίδα αυτή μπορεί ένας χρήστης να δημιουργήσει έναν λογαριασμό για να έχει πρόσβαση στις υπηρεσίες του ιστότοπου αλλιώς αν έχει ήδη μπορεί να πραγματοποιήσει σύνδεση. Στη σελίδα σύνδεσης χρειάζεται να βάλει μόνο το email και τον κωδικό του, επίσης υπάρχει ένα κουμπί που μπορεί να δει τον κωδικό του την ώρα που τον πληκτρολογεί.



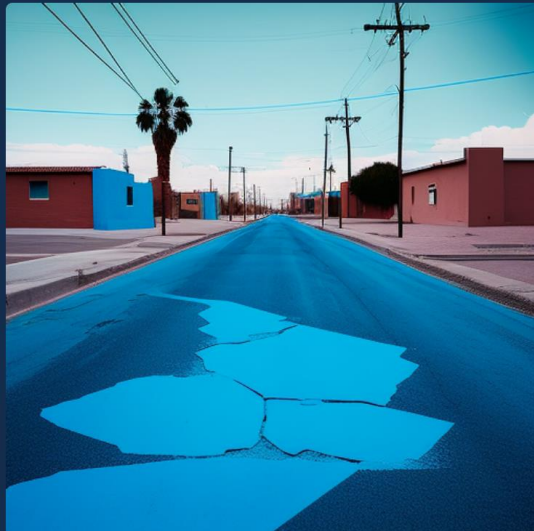
Φωτογραφία 11: Όψη σελίδας σύνδεσης χρήστη με κρυμμένα τα ψηφία του κωδικού



Φωτογραφία 12: Όψη σελίδας σύνδεσης χρήστη με φανερά τα ψηφία του κωδικού

Στη σελίδα εγγραφής ο χρήστης χρειάζεται να αποδώσει περισσότερα στοιχεία για να δημιουργήσει το λογαριασμό του στην ιστοσελίδα. Η διαδικασία εγγραφής χωρίζεται σε 3 μέρη. Στο πρώτο μέρος πρέπει να αποδώσει το email και τον κωδικό πρόσβασής του, στο δεύτερο μέρος πρέπει να αποδώσει το ονοματεπώνυμό του και στο τρίτο μέρος πρέπει να

αποδώσει το τηλέφωνο και τον τόπο διαμονής του. Εφόσον έχει αποδώσει τους σωστούς τύπους δεδομένων για κάθε πεδίο κατά την υποβολή της φόρμας, τότε ο λογαριασμός του δημιουργείται επιτυχώς.



Sign Up

1

2


3

Next

[Back](#)

Already have an account?
[Login](#)

Φωτογραφία 13: Όψη σελίδας εγγραφής χρήστη μέρος (Α)



Sign Up

✓

2

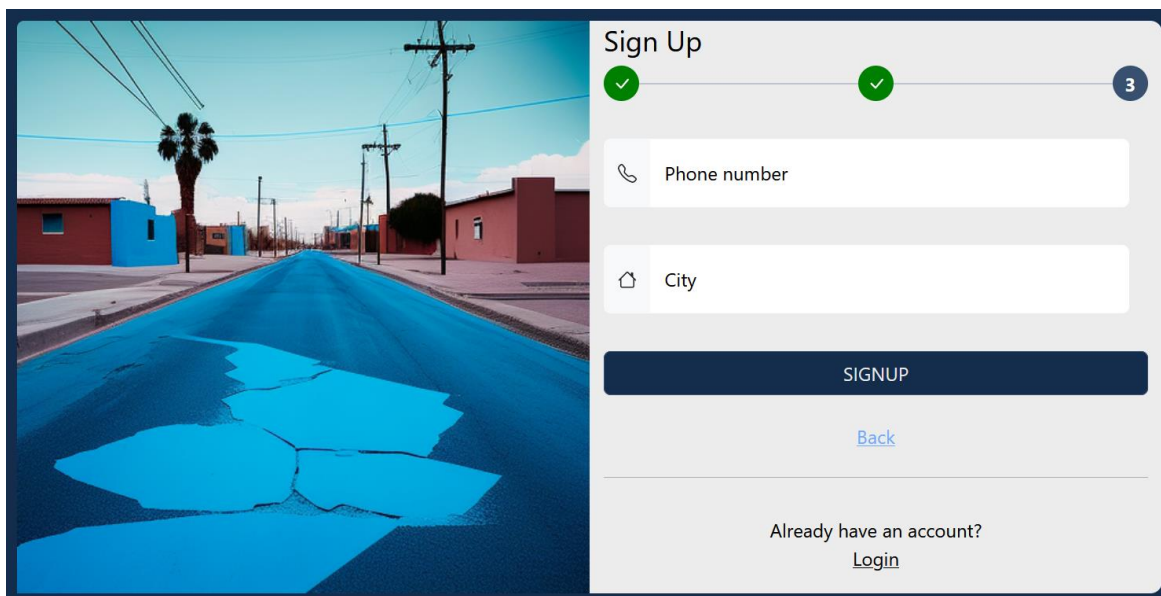
3

Next

[Back](#)

Already have an account?
[Login](#)

Φωτογραφία 14: Όψη σελίδας εγγραφής χρήστη μέρος (Β)

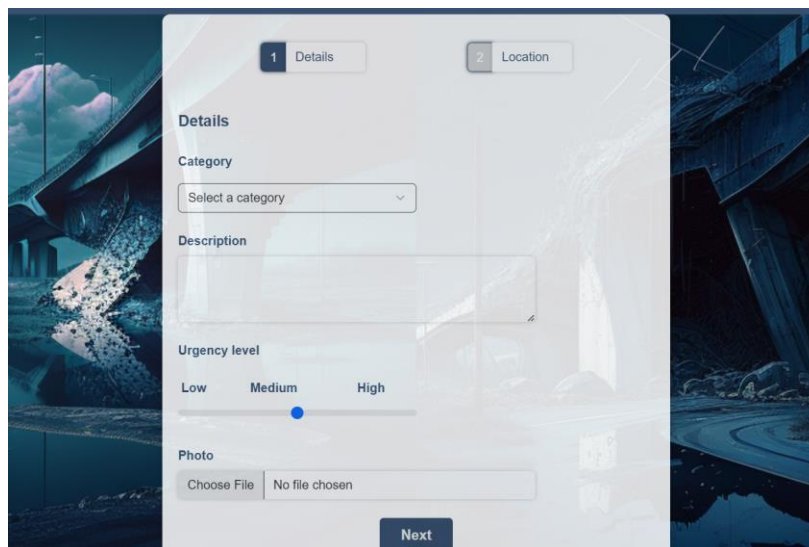


Φωτογραφία 15: Όψη σελίδας εγγραφής χρήστη μέρος (Γ)

3.1.4 Σελίδα Υποβολής Αναφοράς Βλάβης

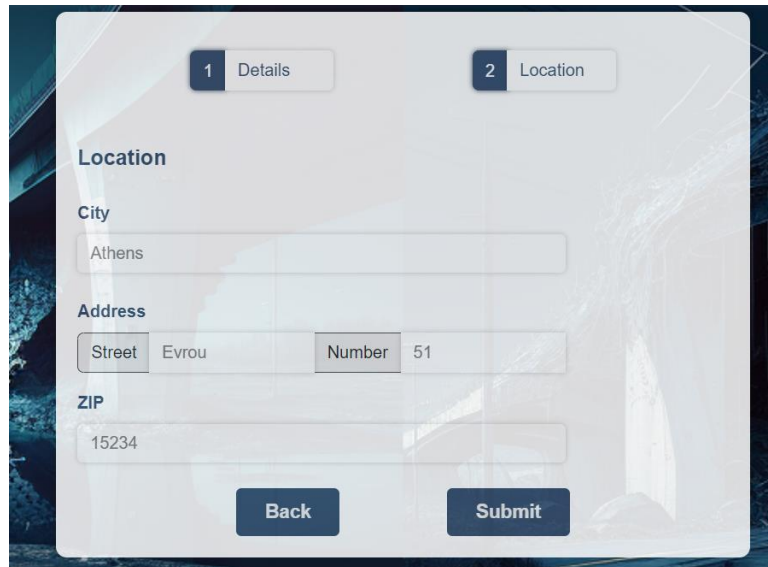
Στη σελίδα αυτή μπορεί ένας συνδεδεμένος χρήστης να υποβάλλει μια νέα βλάβη, αφού πρώτα παραθέσει τα απαραίτητα στοιχεία για την περιγραφή της.

Η διαδικασία υποβολής χωρίζεται σε 2 μέρη. Στο πρώτο μέρος πρέπει να αποδώσει την κατηγορία στην οποία ανήκει η βλάβη, την περιγραφή της, τον βαθμό σοβαρότητας καθώς και μια φωτογραφία.



Φωτογραφία 16: Όψη φόρμας για υποβολή βλάβης μέρος (Α)

Το δεύτερο μέρος αφορά την τοποθεσία στην οποία παρατηρήθηκε η βλάβη. Πιο συγκεκριμένα, ζητείται το όνομα της πόλης, τα στοιχεία του δρόμου(οδός, αριθμός) αλλά και ο ταχυδρομικός κώδικας της περιοχής.



Φωτογραφία 17: Όψη φόρμας για υποβολή βλάβης μέρος (Α)

Εφόσον έχει αποδώσει τους σωστούς τύπους δεδομένων για κάθε πεδίο κατά την υποβολή της φόρμας, τότε η αναφορά του υποβάλλεται επιτυχώς και ο διαχειριστής της σελίδας ενημερώνεται για την νέα βλάβη μέσω ενός email, το οποίο περιέχει όλα τα στοιχεία.

New Damage Report

A new damage report has been uploaded.

Report Details:

- **Category:** fire
- **Description:** HELP
- **City:** Athens
- **Street Name:** Ermou
- **Street Number:** 12
- **ZIP Code:** 14234
- **Urgency:** Medium

← Απάντηση

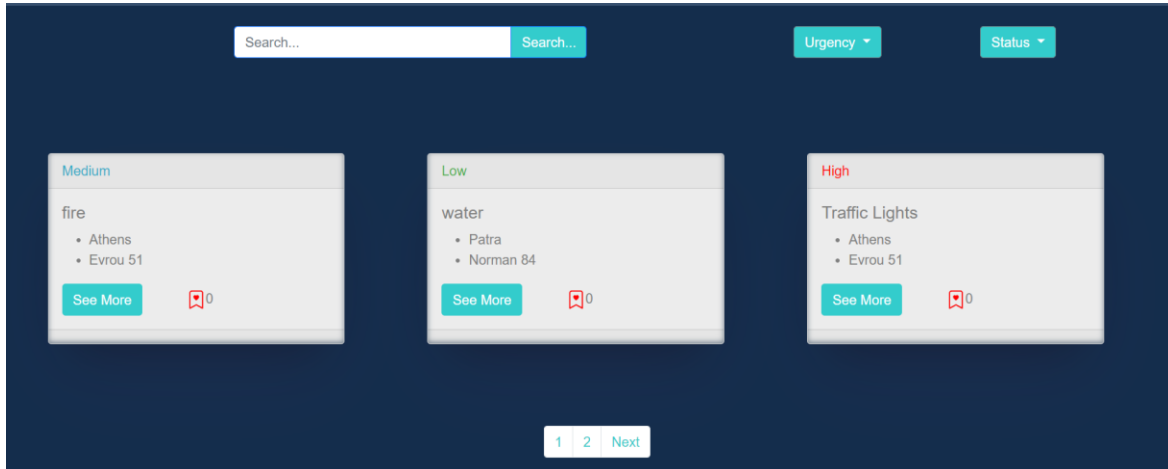
→ Προώθηση

Φωτογραφία 18: Όψη ενημερωτικού email με τα στοιχεία της νέας βλάβης

3.1.5 Σελίδα Διαχειριστή

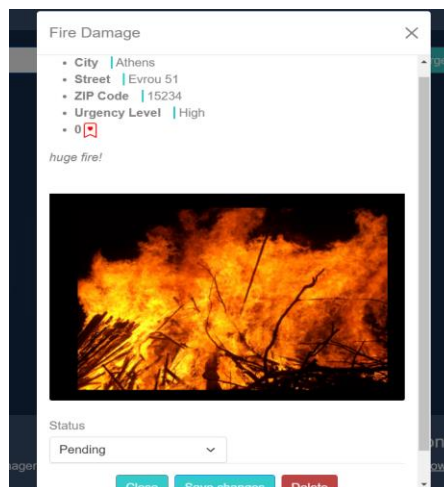
Στη σελίδα αυτή μεταβαίνει ο διαχειριστής της ιστοσελίδας αφού συνδεθεί με τα προσωπικά του στοιχεία. Αρχικά εμφανίζονται όλες οι υποβληθείσες βλάβες με σειρά προτεραιότητας ανάλογα με τον αριθμό των χρηστών που έχουν επισημάνει την σοβαρότητά τους

Η σελίδα περιλαμβάνει φίλτρα κι ένα search box για να μπορέσει ο διαχειριστής να αναζητήσει συγκεκριμένες βλάβες, ανάλογα με την κατάσταση τους ή τον βαθμό σοβαρότητά τους. Επιπλέον, στην σελίδα έχει υλοποιηθεί pagination ώστε να επιταχυνθεί και να απλοποιηθεί η περιήγηση του διαχειριστή.



Φωτογραφία 19: Όψη σελίδας διαχειριστή

Ο διαχειριστής μπορεί να επιλέξει μια αναφορά πατώντας το κουμπί 'see more'. Κατόπιν εμφανίζεται ένα modal με αναλυτικές πληροφορίες για την βλάβη. Δίνεται η δυνατότητα διαγραφής της αναφοράς αλλά και ανανέωσης της κατάστασης της.



Φωτογραφία 20: Όψη modal με τις πληροφορίες της βλάβης

Κάθε φορά που ο διαχειριστής ανανεώνει την κατάσταση μίας βλάβης, στέλνεται ενημερωτικό email στον χρήστη που έχει δημοσιεύσει την συγκεκριμένη βλάβη.

Damage Report Update

Your uploaded report with id: 6473310aa698a70a30a5d2ed has changed to Ongoing

Report Details:

- **Category:** fire
- **Description:** huge fire!
- **City:** Athens
- **Street Name:** Evrou
- **Street Number:** 51
- **ZIP Code:** 15234
- **Urgency:** High
- **Status:** Ongoing

← Απάντηση

→ Προώθηση

Φωτογραφία 21: Όψη ενημερωτικού email με την ανανεωμένη κατάσταση της βλάβης

4 ΦΑΣΗ Γ

4.1 Προγραμματισμός εξυπηρετητή

4.1.1 Τεχνολογίες

Για τον προγραμματισμό του εξυπηρετητή χρησιμοποιήθηκε η γλώσσα προγραμματισμού JavaScript μαζί με το JavaScript Runtime Environment Node.js. Για την πιο εύκολη και κατανοητή υλοποίηση του προγραμματισμού του εξυπηρετητή χρησιμοποιήθηκε η βιβλιοθήκη Express της JavaScript που βοηθάει στο routing των διαφόρων endpoints της ιστοσελίδας ώστε να γίνεται πάντα η κατάλληλη επεξεργασία των αιτημάτων από τον server και να επιστρέφεται η σωστή απάντηση πίσω στον χρήστη.

Χρησιμοποιήθηκε επίσης η template engine Handlebars όπου όταν ο χρήστης εκτελούσε κάποιο αίτημα στον server, ο δεύτερος είχε την υποχρέωση να δημιουργήσει δυναμικά τη σελίδα HTML μαζί με τα απαραίτητα αρχεία CSS και JavaScript και να την επιστρέψει πίσω στον χρήστη. Η Handlebars επιτρέπει να χρησιμοποιήσουμε λογική if statements, for loops κι ακόμα επιτρέπει τη δημιουργία helper functions για πιο σύνθετα expressions.

Για τη μόνιμη αποθήκευση των δεδομένων χρησιμοποιήθηκε η μη σχεσιακή βάση δεδομένων (MongoDB) και ένας Object Relational Mapper, η βιβλιοθήκη Mongoose της Javascript, που επιτρέπει την πιο εύκολη διαχείριση της επικοινωνίας μεταξύ του προγράμματος και της βάσης δεδομένων.

Οι κωδικοί κάθε χρήστη αποθηκεύονται κρυπτογραφημένοι στη βάση με τη βοήθεια της βιβλιοθήκης bcrypt της JavaScript. Επίσης αποθηκεύονται συνεδρίες για κάθε χρήστη στη MongoDB ώστε να μη χρειάζεται σε κάθε αίτημα προς τον server να αποστέλλει συνέχεια το email και τον κωδικό του με τη βοήθεια της βιβλιοθήκης express-session.

4.1.2 Αρχιτεκτονική Model – View – Controller (MVC) [6]

Για τον προγραμματισμό του εξυπηρετητή χρησιμοποιήθηκε η Αρχιτεκτονική – Design Pattern που ονομάζεται MVC. Στο μοντέλο αυτό η εφαρμογή διαιρείται σε τρία διασυνδεδεμένα μέρη ώστε να διαχωριστεί η παρουσίαση της πληροφορίας στον χρήστη από την μορφή που έχει αποθηκευτεί στο σύστημα. Το κύριο μέρος του μοντέλου είναι το αντικείμενο Model το οποίο διαχειρίζεται την ανάκτηση/αποθήκευση των δεδομένων στο σύστημα. Το αντικείμενο View χρησιμοποιείται μόνο για να παρουσιάζεται η πληροφορία στον χρήστη (π.χ. με γραφικό τρόπο). Το τρίτο μέρος είναι ο Controller ο οποίος δέχεται την είσοδο και στέλνει εντολές στο αντικείμενο Model και στο View.

5 ΦΑΣΗ Α

5.1 Αξιολόγηση

Η αξιολόγηση της ιστοσελίδας μας ήταν μια λεπτομερής και πολύπλευρη διαδικασία με σκοπό την αποφυγή λειτουργικών προβλημάτων και την περεταίρω βελτίωση της.

Η ιστοσελίδα δοκιμάστηκε σε διαφορετικές οθόνες και συσκευές, ώστε να επιβεβαιώσουμε ότι είναι πλήρως responsive και ότι δεν παρατηρούνται λειτουργικές και αισθητικές διαφορές όταν μεταβάλλεται το μέγεθος της οθόνης. Επιπροσθέτως, σε όλες τις φόρμες δοκιμάσαμε διαφορετικά αλλά και λανθασμένα δεδομένα για να ελέγξουμε αν παρουσιάζονται ανεπιθύμητα αποτελέσματα που αποτρέπουν την ορθή ροή λειτουργίας. Τέλος, όταν η ιστοσελίδα ανέβηκε στο Heroku, δώσαμε το link σε φίλους και συμφοιτητές μας με σκοπό να τεστάrouμε την ομαλή λειτουργία της και να λάβουμε το απαραίτητο feedback.

ΠΑΡΑΡΤΗΜΑ Α

Οδηγίες Εγκατάστασης

Αρχικά πρέπει να κάνει κάποιος clone το repository εκτελώντας την εντολή ``git clone https://github.com/johnnikolaou/DamageTrack`` σε ένα terminal. Έπειτα πρέπει να δημιουργήσει ένα αρχείο `.env` στο οποίο θα οριστούν μερικές μεταβλητές περιβάλλοντος (αλλιώς μπορεί να της θέσει κατευθείαν στο σύστημα του). Οι μεταβλητές περιβάλλοντος που θα χρειαστούμε είναι:

- `PORT = 3000` (και για τους δύο παρακάτω τρόπους)
- `HOST = localhost` (μόνο για τον δεύτερο τρόπο)
- `DB_URL = "mongodb://localhost:27017/testDB"` (μόνο για τον δεύτερο τρόπο)

Ο πιο εύκολος τρόπος για να εγκαταστήσει κάποιος την ιστοσελίδα είναι μέσω του Docker μέσω του αρχείου που έχει συμπεριληφθεί στο `github docker-compose.yml`. Έχοντας ανοιχτό ένα terminal θα εκτελέσει την εντολή ``docker-compose up --build`` και θα αρχίσουν να εκτελούνται 2 containers ένα με τη node στο οποίο θα τρέχει ο server κι ένα άλλο container στο οποίο θα τρέχει ο mongod server. Έτσι μετά πηγαίνοντας στο url ``http://localhost:3000`` θα μπορεί να έχει πρόσβαση στην εφαρμογή.

Αν δεν έχει εγκατεστημένο το docker τότε πρέπει να έχει εγκατεστημένη την node v18.16 και μια βάση δεδομένων mongo όπου θα χρειαστεί να έχει το connection string για να συνδεθεί το πρόγραμμα στη βάση δεδομένων. Αν έχει εγκατεστημένη τη mongo στον υπολογιστή του ο χρήστης χρησιμοποιεί την παραπάνω μεταβλητή περιβάλλοντος αλλιώς αν χρησιμοποιεί το mongo atlas πρέπει να πάρει το connection string από εκεί και να το θέσει ως τιμή στην παραπάνω μεταβλητή περιβάλλοντος DB_URL. Στη συνέχεια πρέπει να τρέξει την εντολή `npm install` ώστε να εγκατασταθούν όλα τα απαραίτητα dependencies της εφαρμογής. Τέλος πρέπει να τρέξει την εντολή `npm start` ώστε να αρχίσει ο server να ακούει για αιτήματα. Η εφαρμογή είναι διαθέσιμη στο `http://localhost:3000`.

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] <https://github.com/johnnikolaou/DamageTrack>
- [2] <https://damagetrack.herokuapp.com/>
- [3] [https://en.wikipedia.org/wiki/PACT_\(interaction_design\)](https://en.wikipedia.org/wiki/PACT_(interaction_design))
- [4] <https://erdmaker.com/>
- [5] <https://www.midjourney.com/home/>
- [6] <https://el.wikipedia.org/wiki/Model-view-controller>