Εφαρμογή Αναφοράς Βλαβών σε Κτήρια του Πανεπιστημίου Πατρών

Ομάδα 9

Ιωάννης Νικόλαος Θεοδώρου 1083831

Ηλεκτρολόγων Μηχανικών και Τεχνολογίας Υπολογιστών, Πανεπιστήμιο Πατρών, up1083831@upnet.gr

Δημήτριος Γκουνέλας 1084009

Ηλεκτρολόγων Μηχανικών και Τεχνολογίας Υπολογιστών, Πανεπιστήμιο Πατρών, up1084009@upnet.gr

Η βασική ιδέα της εφαρμογής αυτής, είναι οι χρήστες να μπορούν εύκολα, γρήγορα και περιεκτικά, να δηλώσουν μία βλάβη που αυτοί έχουν παρατηρήσει σε κάποιο κτίριο του Πανεπιστημίου Πατρών. Το ενδιαφέρον που παρουσίασε η υλοποίηση αυτής της εφαρμογής είναι η χρήση της, καθώς προορίζεται για mobile use και επιπλέον ότι η ίδια η εφαρμογή λειτουργεί σαν υπηρεσία του Πανεπιστημίου.

1. εισαγωγη

Το θέμα που μας ανατέθηκε για το ομαδικό πρότζεκτ ήταν η δημιουργία μιας ‘Εφαρμογής Αναφοράς Βλαβών σε Υποδομές’. Λόγω της γενικότητας αυτού του θέματος ζητήσαμε καθοδήγηση από τον κ. Αβούρη, ο οποίος, μας έδωσε σαν ένα παράδειγμα μιας τέτοιας εφαρμογής, την αναφορά βλαβών στο Πανεπιστήμιό μας. Αυτή η ιδέα μας άρεσε ιδιαίτερα, αφενός επειδή δεν υπάρχει μια τέτοια εφαρμογή για το Πανεπιστήμιό μας και αφετέρου γιατί η υλοποίηση μιας εφαρμογής αυτού του τύπου θα βελτίωνε την εμπειρία των φοιτητών, των καθηγητών και των υπόλοιπων εργαζόμενων του Πανεπιστημίου.

* 1. Αρχικοί Στόχοι

Οι στόχοι που είχαμε για την εφαρμογή πριν ακόμα αρχίσουμε να σχεδιάζουμε το user interface της, ήταν να δίνουμε στους χρήστες την ελευθερία να δηλώσουν μια βλάβη με τον τρόπο που θέλουν και χωρίς απαραίτητα οι ίδιοι να είναι μέλη του Πανεπιστημίου.

* 1. Σχεδιασμός

Για τον αρχικό σχεδιασμό της εφαρμογής, χρησιμοποιήθηκε το [Figma](https://www.figma.com/design/wOT01axs2imzT9wcudB8eZ/Project_Team_9?node-id=0-1). Ξεκινήσαμε από την μορφή της σελίδας που δίνει στον χρήστη την επιλογή του τρόπου δήλωσης της βλάβης και ακολούθησαν οι υπόλοιπες. Οι σελίδες που σχεδιάστηκαν είναι οι εξής:

* Home Page
* Sign-in / Sign-out
* Home Page after Sign-in
* Selection of Malfunction Location
* User Profile
* Map
* School Selection
* Forms

Το πρόβλημά του αρχικού σχεδιασμού, όμως, ήταν η άγνοιά μας ως προς την πλατφόρμα εστίασης, δηλαδή για mobile ή pc και ως αποτέλεσμα, τα wireframes που σχεδιάστηκαν είχαν μια υβρίδια μορφή mobile-pc. Μετά την ενδιάμεση παρουσίαση και με την καθοδήγηση των καθηγητών, αποφασίσαμε να εστιάσουμε στην mobile μορφή. Η απόφαση αυτή, βέβαια, έφερε νέα προβλήματα, όπως τον περιορισμένο χώρο με τον οποίο είχαμε να δουλέψουμε καθώς και με τα χρώματα της εφαρμογής, αφού το σκούρο θέμα που είχαμε επιλέξει στην αρχή, ήταν ακατάλληλο για μια mobile εφαρμογή που θα χρησιμοποιείται κυρίως την ημέρα όπου υπάρχει μια μεγάλη πιθανότητα ο οποιοσδήποτε χρήστης να βρίσκεται σε εξωτερικό χώρο.

1. Υλοποίηση

Η υλοποίηση της εφαρμογής επιτεύχθηκε με την χρήση: NodeJS, Express-handlebars, CSS, HTML, JS και sqlite3.

* 1. Μεθοδολογία

Η μεθοδολογία που ακολουθήσαμε για να λύσουμε τα προβλήματα που αναφέρθηκαν προηγουμένως και για να ξεκινήσει η υλοποίηση της εφαρμογής ήταν αρχικά να χωρίσουμε την εφαρμογή σε πέντε ενότητες. Η πρώτη ήταν το home page, η δεύτερη ήταν η επιλογή τοποθεσίας μέσω του τμήματος και ότι σχετίζεται με αυτό, η τρίτη η επιλογή τοποθεσίας με βάση τον χάρτη και ότι σχετίζεται με αυτόν, η τέταρτη ήταν η επιλογή τοποθεσίας με βάση το κτήριο και πέμπτη η αυθεντικοποίηση μαζί με το user profile.

Σε πρώτο στάδιο, ασχοληθήκαμε με το user interface (HTML, CSS, JavaScript) των τεσσάρων πρώτων, εφόσον δεν ήταν δυνατή η υλοποίηση του user profile λόγω απουσίας γνώσης. Σε δεύτερο στάδιο, κάναμε τις συνδέσεις των βάσεων δεδομένων με τις ενότητές μας. Τέλος, συμπεριλαμβάνοντας τα sessions, δημιουργήσαμε το user profile μαζί με τις ενέργειες που μπορεί να κάνει και προσθέσαμε, ακόμη, τον admin ο οποίος έχει συγκεκριμένες αρμοδιότητες.

Καθ’ όλα τα στάδια, αν και οι υποχρεώσεις μας χωρίστηκαν εξίσου, είχαμε καθημερινή επαφή, συζήτηση και ανταλλαγή ιδεών πάνω στο τμήμα που ασχολούταν ο καθένας με σκοπό την βελτίωση της εφαρμογής και την επιβεβαίωση ότι βρισκόμαστε θεματικά στο ίδιο μονοπάτι.

* 1. Δεδομένα

Τα δεδομένα που χρησιμοποιήσαμε στην εφαρμογή μας, είναι τα ονόματα των διάφορων προπτυχιακών σπουδών που παρέχει το Πανεπιστήμιο Πατρών, τα ονόματα των σχολών, τα κτίρια που βρίσκονται εντός του Πανεπιστημίου και η επιλογές που δίνουμε στον χρήστη για τον χαρακτηρισμό της βλάβης. Για τα ονόματα των τμημάτων, των σχολών και των εικόνων που αναφέρονται σε αυτές τις σχολές συμβουλευτήκαμε την σελίδα του Πανεπιστημίου Πατρών [upatras.gr](https://www.upatras.gr/education/undergraduate-studies/). Για τα ονόματα των κτιρίων και την χρήση τους συμβουλευτήκαμε την [my.upatras.gr](https://my.upatras.gr/buildings/). Τέλος, για τις επιλογές χαρακτηρισμού δημιουργήσαμε δικά μας δεδομένα.

* 1. Σχεδιασμός Βάσης Δεδομένων

Για τον σχεδιασμό της βάσης χρησιμοποιήσαμε SQLite λόγω της μεγαλύτερης οικειότητας μας. Δημιουργήσαμε το αρχείο(.sql) το οποίο το ονομάσαμε database.sql στο οποίο αποθηκεύσαμε όλα τα tables της βάσης μας και με το Javascript populateDb.js, δημιουργείται ολόκληρη η βάση μας στο αρχείο database.sqlite. Για τον σχεδιασμό είχαμε σαν ένα σημείο αναφοράς το ERD που είχαμε σχεδιάσει για την ενδιάμεση παρουσίαση και προσθέταμε, διορθώναμε ή αφαιρούσαμε ορισμένα πράγματα με σκοπό την δημιουργία ενός ορθού τελικού Σχεσιακού Μοντέλου Οντοτήτων (ERD) που ανταποκρίνεται στις ανάγκες μας.

* + 1. ERD

Στην εικόνα 1 φαίνεται το ERD που προέκυψε με την πλήρη υλοποίηση της εφαρμογής.

Εικόνα που περιέχει διάγραμμα, κύκλος, στιγμιότυπο οθόνης, σχεδίαση

Περιγραφή που δημιουργήθηκε αυτόματαΕικόνα 3.1: Σχεσιακό Μοντέλο Οντοτήτων (ERD)

* + 1. *Σχήμα Βάσης Δεδομένων*

Στην εικόνα 2 φαίνεται το σχήμα της βάσης δεδομένων σχεδιασμένο στην εφαρμογή [DB designer](https://www.dbdesigner.net/).

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, διάγραμμα, αριθμός

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 3.2: Σχήμα Βάσης Δεδομένων της εφαρμογής

* 1. Κύριες Ενέργειες
* Σχεδιασμός στο Figma – Θεοδώρου, Γκουνέλας
* Σχεδιασμός και Κατασκευή Βάσης Δεδομένων - Θεοδώρου, Γκουνέλας
* Deployment στο Heroku – Θεοδώρου
* Home page – Θεοδώρου, Γκουνέλας
* Επιλογή τρόπου δήλωσης βλάβης – Θεοδώρου, Γκουνέλας
* Επιλογή με χάρτη – Γκουνέλας
* Επιλογή με τμήμα – Θεοδώρου
* Επιλογή με κτήριο – Θεοδώρου, Γκουνέλας
* Φόρμες – Θεοδώρου, Γκουνέλας
* Authentication – Θεοδώρου, Γκουνέλας
* User/Admin profile – Θεοδώρου, Γκουνέλας
  1. Χρονοδιάγραμμα

Πίνακας 1: Χρονοδιάγραμμα

|  |  |  |
| --- | --- | --- |
| Εβδομάδα | | Ενέργειες |
| 5 | 18/3 | Σχεδιασμός στο Figma |
| 6 | 25/3 | Υλοποίηση πρωτοτύπων επιλεγμένων σελίδων σε HTML/CSS |
| 7 | 1/4 | Ενδιάμεση παρουσίαση |
| 8 | 8/4 | Ολοκλήρωση των περισσότερων σελίδων σε HTML/CSS |
| 9 | 15/4 | Σύνδεση με server μέσω node.js και μετατροπή των HTML σε Handlebars |
| 10 | 22/4 | Σύνδεση με βάση δεδομένων |
| 11 | 13/5 | Εφαρμογή του authentication |
| 12 | 20/5 | Deployment στο Heroku, συγγραφή αναφοράς, ολοκλήρωση του κώδικα |

* 1. Αξιολόγηση

Τα κριτήρια επιτυχίας που χρησιμοποιήσαμε στο πρότζεκτ μας ήταν:

* η σωστή εμφάνιση των προ υπαρχόντων δεδομένων αλλά και των εισαγόμενων
* η σωστή λειτουργία της αυθεντικοποίησης
* η σωστή λειτουργία του χάρτη σε κινητές συσκευές
* η δυνατότητα παροχής διαφόρων ενεργειών στους χρήστες και admins
* να είναι επαρκής για την πλήρη περιγραφή μιας βλάβης
* κατάλληλα χρώματα για εξωτερική χρήση
* η ομορφιά της εφαρμογής

1. οδηγιες εγκαταστασης και χρησησ
   1. Οδηγίες εγκατάστασης
      1. Εγκατάσταση της Βάσεις Δεδομένων

Για να δημιουργήσουμε την βάση μας, αρκεί να ανοίξουμε το terminal και να μεταφερθούμε στο path .../model/sqlite τρέχοντας την εντολή:

cd path/to/model/sqlite

και στην συνέχεια τρέχουμε την εντολή:

node populateDb.js

* + 1. Εγκατάσταση όλων των απαραίτητων dependencies

Για την εγκατάσταση των απαραίτητων dependencies της εφαρμογής, αρκεί να πάμε στο terminal του root directory και να τρέξουμε την εντολή:

npm install

Τέλος, τρέχουμε την εντολή:

npm run watch

και η εφαρμογή τρέχει πλέον στο <http://localhost:3000/>

* 1. Οδηγίες χρήσης

Η εφαρμογή έχει δομηθεί ώστε να χρησιμοποιείται από έναν απλό χρήστη και να ελέγχεται από έναν διαχειριστή(admin). Οι δυνατότητες που παρέχονται στον καθένα εξηγούνται αναλυτικά παρακάτω.

* + 1. Διαχειριστής

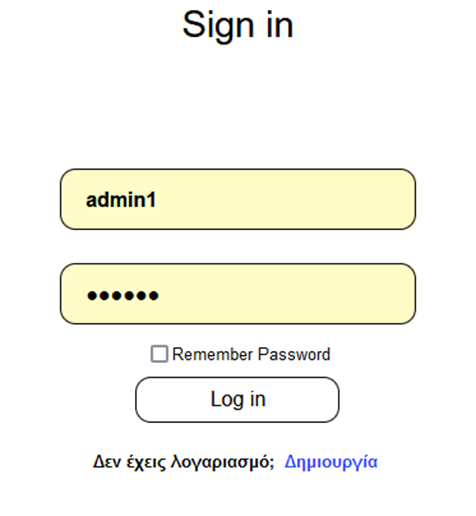
Αρχικά, τα στοιχεία αυθεντικοποίησης των διαχειριστών εισάγονται στην βάση με την δημιουργία της και δεν υπάρχει η δυνατότητα δημιουργίας νέου admin εντός της εφαρμογής. Συνεπώς μόνο οι διαχειριστές του server μπορούν να δημιουργήσουν νέο λογαριασμό admin. Οι κωδικοί των admin εισάγονται στην βάση μας κρυπτογραφημένοι και οι υπάρχοντες λογαριασμοί admin είναι οι εξής:

1. Username: admin1

Password: admin1

1. Username: admin2

Password: admin2

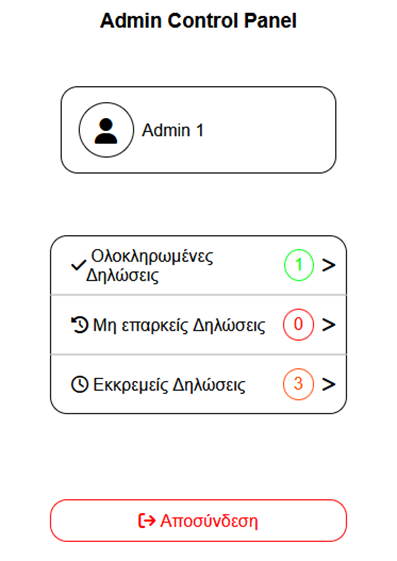


Εικόνα 3.1: Σύνδεση του διαχειριστή

Ο διαχειριστής με την αυθεντικοποίησή του, οδηγείται στην σελίδα admin, στην οποία μπορεί να δει όλες τις καταχωρημένες δηλώσεις των απλών user. Οι δηλώσεις αυτές χωρίζονται σε 3 μεγάλες κατηγορίες:

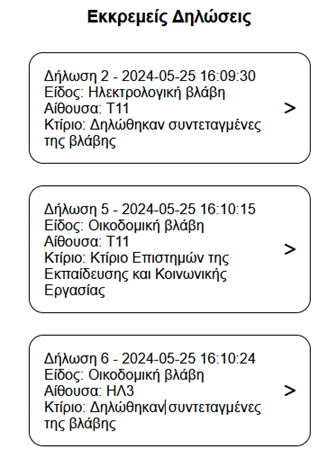
* Ολοκληρωμένες δηλώσεις
* Μη επαρκείς δηλώσεις
* Εκκρεμείς δηλώσεις

Επιπλέον, μπορεί να κάνει και αποσύνδεση πατώντας το κουμπί ‘Αποσύνδεση’ που φαίνεται στο κάτω μέρος της σελίδας.



Εικόνα 3.2: Σελίδα διαχείρισης των καταχωρημένων δηλώσεων

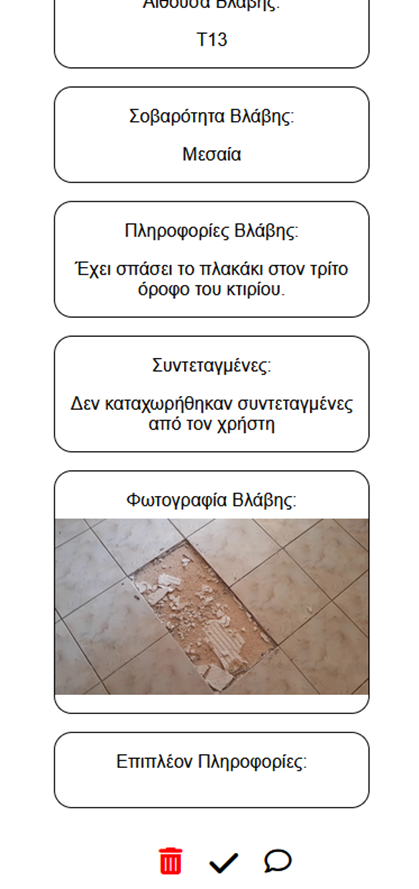
Όταν ο χρήστης κάνει μια καινούργια δήλωση, την αίτηση θα μπορεί ο admin να την δει στις εκκρεμείς δηλώσεις.



Εικόνα 3.3: Σελίδα προεπισκόπησης και αλλαγής κατάστασης της δήλωσης

Όταν ο admin πατήσει πάνω της, θα του εμφανιστούν όλα τα καταχωρημένα από τον χρήστη στοιχεία της δήλωσης. Εκεί ο admin έχει τρεις δυνατότητες:

1. Να την μετακινήσει στις ολοκληρωμένες
2. Να της προσθέσει κάποιο σχόλιο, σε περίπτωση που ενδεχομένως τα στοιχεία είναι ανεπαρκής για την αντιμετώπιση της βλάβης.
3. Να την διαγράψει οριστικά.



Εικόνα 3.4: Σελίδα προεπισκόπησης και αλλαγής κατάστασης της δήλωσης

Όσον αφορά την πρώτη περίπτωση, ο admin βλέπει την δήλωση, τα στοιχεία της είναι επαρκή και η βλάβη που περιγράφεται στην δήλωση αντιμετωπίζεται επιτυχώς. Τότε ο admin μεταφέρει την δήλωση στις ολοκληρωμένες. Επιπλέον, στην περίπτωση αυτή σκεφτήκαμε και το ενδεχόμενο ο admin να πατήσει κατά λάθος την ολοκλήρωση της αίτησης οπότε έχουμε προσθέσει την δυνατότητα στον admin να μπορεί να την επαναφέρει από τις ολοκληρωμένες στις μη ολοκληρωμένες τα πρώτα 20 λεπτά από την στιγμή μεταφοράς της.

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά, αριθμός

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 3.5: Σελίδα προσθήκης σχολίου του διαχειριστή

Όπως φαίνεται παραπάνω, την πρώτη δήλωση μπορεί να την επαναφέρει στις μη ολοκληρωμένες επειδή δεν έχουν περάσει ακόμα 20 λεπτά από την στιγμή της αλλαγής κατάστασης, ενώ την δεύτερη δήλωση δεν μπορεί να της αλλάξει κατάσταση αφού έχει περάσει το χρονικό αυτό όριο.

Για την δεύτερη περίπτωση, ο admin βλέπει την δήλωση, τα στοιχεία της δεν είναι αρκετά επαρκής για την αντιμετώπιση της δηλωμένης βλάβης είτε γιατί έχουν δηλωθεί λάθος συντεταγμένες είτε για οποιονδήποτε λόγο ο admin θεωρεί ότι χρειάζεται περισσότερες πληροφορίες για την αντιμετώπιση της βλάβης. Συνεπώς, ο admin έχει την δυνατότητα να γράψει ένα Σχόλιο διαχειριστή και η δήλωση να πάει κατευθείαν στις μη επαρκείς δηλώσεις στις οποίες ο χρήστης θα πρέπει να κάνει αλλαγές(θα αναλυθεί στον χρήστη).

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, σχεδίαση

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 3.6: Σελίδα προσθήκης σχολίου του διαχειριστή

Να σημειωθεί ότι ο admin μπορεί στις μη επαρκείς δηλώσεις να κάνει αλλαγές στο σχόλιο του οποιαδήποτε στιγμή πατώντας το κουμπί edit που φαίνεται στα δεξιά της παρακάτω εικόνας.

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 3.7: Σελίδα προεπισκόπησης των μη επαρκών δηλώσεων από τον admin

Για την τρίτη περίπτωση, η οποία είναι για ειδικές περιπτώσεις, ο admin μπορεί να διαγράψει εντελώς την δήλωση(με popup επιβεβαίωσης για να εξασφαλίσουμε να μην γίνει κατά λάθος). Θεωρήσαμε ότι πρέπει ο admin να έχει την δυνατότητα να διαγράφει κιόλας κάποια αίτηση η οποία να μην αρμόζει τις προδιαγραφές μιας και η εφαρμογή προορίζεται για ευρεία χρήση.

* + 1. Απλός χρήστης(user)

Όσον αφορά τον απλό χρήστη, μπορεί ο καθένας να δημιουργήσει λογαριασμό με ένα unique Όνομα χρήστη(username), ένα unique email και ένα password.

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά, σχεδίαση

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 3.8: Αρχική σελίδα

Αφού δημιουργήσει τον λογαριασμό του θα πρέπει να συνδεθεί στον λογαριασμό του με το email και τον κωδικό του και στην συνέχεια θα οδηγηθεί στο home page. Στο home page έχει δυο δυνατότητες ο χρήστης:

1. Δήλωση νέας βλάβης
2. Προβολή προφίλ του χρήστη

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά, σχεδίαση

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 3.9: Κεντρική σελίδα του χρήστη

* + - 1. Δήλωση νέας βλάβης

Όσον αφορά την πρώτη περίπτωση, ο χρήστης όταν πατήσει το κουμπί ‘Δήλωση νέας βλάβης’, θα οδηγηθεί στην σελίδα location που θα επιλέξει τον τρόπο με τον οποίο θέλει να επιλέξει την τοποθεσία της βλάβης. Ο χρήστης έχει τρείς επιλογές:

1. Επιλογή με βάση το όνομα του τμήματος
2. Επιλογή με βάση το όνομα του κτιρίου
3. Επιλογή με βάση την τοποθεσία στον χάρτη

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά, λογότυπο

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 3.10: Σελίδα επιλογής τοποθεσίας της βλάβης

Στην πρώτη περίπτωση ο χρήστης θα οδηγηθεί στην σελίδα school στην οποία θα έχει να διαλέξει αρχικά την Σχολή:

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά, αριθμός

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 3.11: Σελίδα επιλογής σχολής

και έπειτα από τα όλα τμήματα της σχολής αυτής, να επιλέξει το τμήμα που βρίσκεται η βλάβη.

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 3.12: Σελίδα επιλογής τμήματος με βάση την σχολή που επιλέχθηκε

Στην συνέχεια, αφού επιλέξει και το τμήμα, θα οδηγηθεί τελικά στην δήλωση στην οποία θα έχει να συμπληρώσει τα εξής:

* Όνομα κτιρίου (υποχρεωτικό)
* Όνομα αίθουσας (υποχρεωτικό)
* Είδος βλάβης (υποχρεωτικό)
* Σοβαρότητα βλάβης (υποχρεωτικό)
* Περιγραφή βλάβης (προαιρετικό)
* Επιλογή φωτογραφίας της βλάβης (υποχρεωτικό)
* Πρόσθετα σχόλια (προαιρετικό)

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά, αριθμός

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 3.13: Σελίδα δήλωσης της βλάβης με βάση το τμήμα που επιλέχθηκε

Θεωρήσαμε ότι πρέπει να ξέρει ο admin το όνομα κτιρίου, όνομα αίθουσας, είδος της βλάβης, σοβαρότητα της βλάβης και την φωτογραφίας της βλάβης. Όσον αφορά την φωτογραφία αρχικά είπαμε να είναι προαιρετική, ωστόσο ζυγίσαμε τα θετικά και τα αρνητικά και το πόσο πιο εύκολα θα μπορεί να βρεθεί η κάθε βλάβη με μόνο μια απλή φωτογραφία της, άρα κατά συνέπεια προς διευκόλυνση του admin καταλήξαμε ότι θα πρέπει να είναι και αυτή υποχρεωτική

Να σημειωθεί, ότι ο χρήστης ανάλογα με το τι τμήμα θα επιλέξει, στις επιλογές του κτιρίου και αίθουσας θα του εμφανίζονται τα αντίστοιχα κτίρια και αίθουσες της συγκεκριμένης σχολής για διευκόλυνση, ενώ έχει την δυνατότητα να γράψει ο ίδιος τελικά ότι θέλει αυτός σε περίπτωση που δεν εμφανίζεται το κτίριο ή η αίθουσα που θέλει. Για παράδειγμα αν ο χρήστης επιλέξει προηγουμένως ως το τμήμα με την βλάβη το ΗΜΤΥ:

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά, γραμμή

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 3.14: Παράδειγμα επιλογής ως Τμήμα το ΗΜΤΥ

Αν ο χρήστης επιλέξει την επιλογή με βάση το όνομα κτιρίου (σε περιπτώσεις που ο ίδιος δεν γνωρίζει το όνομα του τμήματος ή της σχολής) θα οδηγηθεί κατευθείαν στην δήλωση στην οποία θα έχει να συμπληρώσει τα ίδια με την πρώτη δήλωση με την διαφορά ότι στην περίπτωση αυτή προς διευκόλυνση του χρήστη θα του εμφανίζονται τα κτίρια και οι αίθουσες όλου του Πανεπιστημίου Πατρών ωστόσο ο ίδιος μπορεί και πάλι να μην επιλέξει κάποιο από αυτά αλλά να πληκτρολογήσει κάποιο άλλο.

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά, γραμμή

Περιγραφή που δημιουργήθηκε αυτόματαΕικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά, αριθμός

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 3.15: Σελίδα δήλωσης της βλάβης έχοντας επιλέξει με βάση το όνομα κτιρίου

Αν ο χρήστης επιλέξει με βάση την τοποθεσία στον χάρτη θα οδηγηθεί στην σελίδα επιλογής συντεταγμένων:

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, χάρτης, διάγραμμα

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 3.16: Σελίδα επιλογής συντεταγμένων

Στην οποία ο χρήστης μπορεί να επιλέξει είτε την ακριβής τοποθεσία που βρίσκεται εκείνη την στιγμή πατώντας το κουμπί ‘Η θέση μου’( εδώ να σημειωθεί ότι αν ο χρήστης θέλει να έχει αυτή την επιλογή καλό θα ήταν να τρέξει την εφαρμογή από mozilla, chrome ή edge καθώς το safari δεν παρέχει αυτή την δυνατότητα ) είτε να σύρει την οθόνη στην περιοχή που θέλει και να πατήσει επιλογή τοποθεσίας για να καταχωρηθούν οι συντεταγμένες του σημείου και να οδηγηθεί στην επόμενη σελίδα της δήλωσης της βλάβης:

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά, αριθμός

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 3.17: Σελίδα δήλωσης βλάβης από επιλογή τοποθεσίας με χάρτη

Η μόνη διαφορά που υπάρχει σε αυτή την δήλωση σε σχέση με τις προηγούμενες είναι ότι δεν υπάρχει καν η επιλογή κτιρίου αφού ουσιαστικά ο χρήστης έχει επιλέξει τις ακριβείς συντεταγμένες, ωστόσο θεωρήσαμε καλό να πρέπει να βάλει την αίθουσα/χώρο τον οποίο βρίσκεται η βλάβη. Και σε αυτή την περίπτωση οι επιλογές αιθουσών είναι από όλο το Πανεπιστήμιο προς διευκόλυνση του χρήστη ενώ και πάλι μπορεί να πληκτρολογήσει ότι θέλει ο ίδιος σε περίπτωση που βρίσκεται σε κάποια άλλη αίθουσα/χώρος.

* + - 1. Προφίλ Χρήστη

Αν ο χρήστης πατήσει το κουμπί που βρίσκεται στο πάνω μέρος της οθόνης του βρισκόμενος στην Αρχική σελίδα:

Εικόνα που περιέχει σκίτσο/σχέδιο, σχεδίαση, εικονογράφηση

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 3.18: Κουμπί για είσοδο στο προφίλ του χρήστη

Θα οδηγηθεί στο Προφίλ Χρήστη:

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά, αριθμός

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 3.19: Προφίλ χρήστη

Εδώ ο χρήστης μπορεί να βλέπει ποιες δηλώσεις του έχουν ολοκληρωθεί, ποιές είναι μη επαρκείς και ποιες δεν έχουν ακόμα ολοκληρωθεί. Επιπλέον πατώντας το κουμπί ‘Δημιουργία νέας δήλωσης’ θα ξεκινήσει την δημιουργία μιας νέας δήλωσης από την αρχή, ενώ πατώντας το κουμπί ‘Αποσύνδεση’ θα αποσυνδεθεί από τον λογαριασμό του και θα οδηγηθεί στην αρχική σελίδα:

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 3.20: Αρχική σελίδα

Ο χρήστης πατώντας στις ‘Εκκρεμείς Δηλώσεις’ οδηγείται στην σελίδα:

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά, αριθμός

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 3.21: Σελίδα προεπισκόπησης των εκκρεμών δηλώσεων του χρήστη

Στην οποία οδηγούνται κατευθειάν οι δηλώσεις του μετά την υποβολή τους και ο χρήστης περιμένει τον διαχειριστή να την δει και να την επεξεργαστεί αναλόγως. Όπως φαίνεται παραπάνω, ο χρήστης όταν κάνει μια καινούργια δήλωση έχει χρονικό όριο 20 λεπτά στα οποία μπορεί να την επεξεργαστεί ή ακόμα και να την διαγράψει εντελώς, ωστόσο μετά το πέρας του χρονικού ορίου ο χρήστης δεν θα μπορεί πλέον να κάνει το οτιδήποτε στην αίτηση.

Αν ο χρήστης θέλει να την επεξεργαστεί πριν το χρονικό αυτό ωράριο πατάει το κουμπί επεξεργασία και οδηγείται στην επεξεργασια της δήλωσης:

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά, αριθμός

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 3.22: Σελίδα επεξεργασίας της δήλωσης του χρήστη

Όπως φαίνεται και στην εικόνα κατά την επεξεργασία θα εμφανιστούν και οι συντεταγμένες που είχαν καταχωρηθεί από τον χρήστη(αν δεν ειχε καταχωρήσει θα του εμφανιστεί το μήνυμα δεν καταχωρήθηκαν συντεταγμένες όπως φαίνεται παραπάνω) και ο χρήστης πατώντας το κόκκινο εικονίδιο δίπλα από τις συντεταγμένες μπορεί να ανανεώσει ή να προσθέσει και συντεταγμένες από τον χάρτη.

Πατώντας να δει τις μη επαρκείς του δηλώσεις ο χρήστης οδηγείται στην σελίδα:

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά, αριθμός

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 3.23: Σελίδα προεπισκόπησης των μη επαρκών δηλώσεων του χρήστη

Στην οποία ο χρήστης βλέπει τα σχόλια που έχει προσθέσει ο διαχειριστής και πατώντας το κουμπί που φαίνεται στα δεξιά οδηγείται στην σελίδα επεξεργασίας της δήλωσης στην οποία την επεξεργάζεται με βάση τις οδηγίες που του έδωσε ο διαχειριστής:

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά, αριθμός

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 3.24: Σελίδα επεξεργασίας της δήλωσης του χρήστη

Όπως αναφέρθηκε και πριν ο χρήστης όταν επεξεργάζεται την δήλωση του μπορεί να δει και τις συντεταγμένες που καταχώρησε και να τις ανανεώσει.

Τέλος, αν ο χρήστης πατήσει το ‘Ολοκληρωμένες Δηλώσεις’ θα οδηγηθεί στην σελίδα:

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 3.25: Σελίδα επισκόπησης των ολοκληρωμένων δηλώσεων

Στην οποία ο χρήστης βλέπει ποιες από τις δηλώσεις του ολοκληρώθηκαν, ωστόσο δεν μπορεί να κάνει σε αυτές κάποια περαιτέρω ενέργεια.

* + 1. About us

Βρίσκεται τόσο στην Αρχική σελίδα της εφαρμογής όσο και στην αρχική σελίδα του χρήστη και μπορεί να την επισκεφτεί και κάποιος χωρίς λογαριασμό.

Εικόνα που περιέχει κείμενο, γραμματοσειρά, στιγμιότυπο οθόνης, γραμμή

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 3.26: About us

Πατώντας εκεί ο χρήστης θα οδηγηθεί στην σελίδα:

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 3.27: Σελίδα About us

Εκεί φαίνεται ο κύριος σκοπός της εφαρμογής μας με λίγα λόγια και μερικές πληροφορίες για τους δημιουργούς της.

1. Παραδειγματα χρησης του προγραμματος

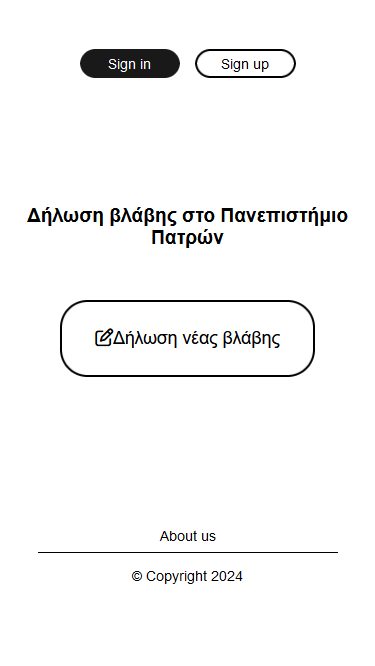
Σε αυτή την ενότητα θα παρουσιαστούν δύο παραδείγματα χρήσης της εφαρμογής μας. Το πρώτο είναι η δήλωση μιας βλάβης από έναν φοιτητή και η δεύτερη είναι η ενέργεια του admin αφού η βλάβη αναγνωριστεί και επιδιορθωθεί.

* 1. Χρήση του προγράμματος από έναν φοιτητή

Ο φοιτητής δημιουργεί έναν λογαριασμό, αυθεντικοποιείται και συνεχίζει προς τη δήλωση της βλάβης.

* + 1. Εντοπισμός βλάβης – Πρώτη χρήση της εφαρμογής

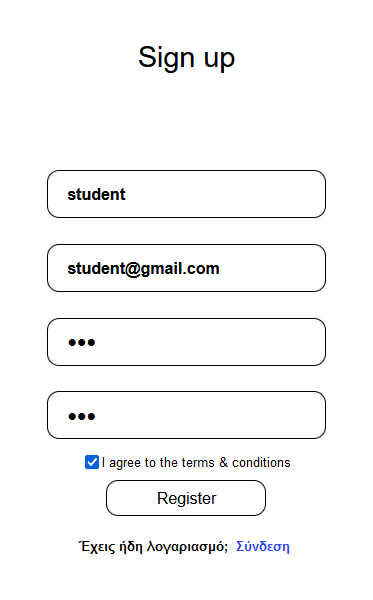
Ο φοιτητής αφού εντοπίσει μια βλάβη εισέρχεται στην εφαρμογή για πρώτη φορά



Εικόνα 4.1: Home page εφαρμογής

* + 1. Sign-up

Ο φοιτητής χρησιμοποιεί για πρώτη φορά την εφαρμογή, επομένως δημιουργεί τον λογαριασμό του για να έχει πρόσβαση στις δυνατότητες της εφαρμογής



Εικόνα 4.2: Sign-up της εφαρμογής

* + 1. Sign-in

Ο φοιτητής, αφού έχει δημιουργήσει πλέον τον λογαριασμό του, κάνει sign-in στην εφαρμογή.

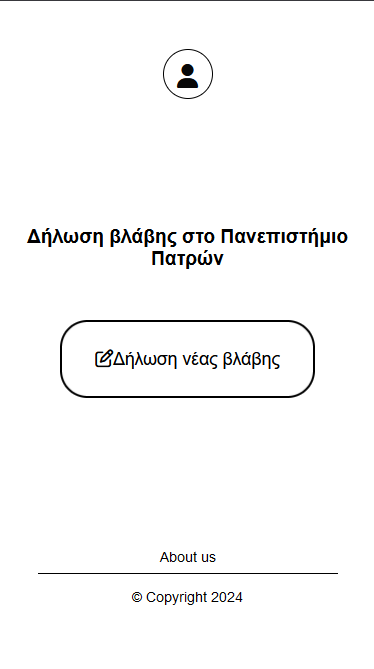
Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά, γραμμή

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 4.3: Sign-in εφαρμογής

* + 1. Δήλωση νέας βλάβης

Όταν κάνει την αυθεντικοποίηση, η εφαρμογή τον επαναφέρει στο home-page, αλλά αυτή την φορά έχει πρόσβαση στο profile του και στην δήλωση βλάβης.



Εικόνα 4.4: Home Page μετά την αυθεντικοποίηση

* + 1. Επιλογή τρόπου δήλωσης της βλάβης

Πατώντας το κουμπί δήλωση νέας βλάβης, του εμφανίζονται τρεις επιλογές.



Εικόνα 4.5: Επιλογή τοποθεσίας βλάβης με βάση την επιλογή του χρήστη

* + 1. Επιλογή τοποθεσίας με βάση τον χάρτη

Επιλέγει το τρίτο κουμπί και ο χρήστης οδηγείται σε έναν χάρτη, ο οποίος έχει ως κέντρο το Πανεπιστήμιο Πατρών. Επιλέγοντας την θέση του, δηλαδή το σημείο που βρίσκεται η βλάβη, το κέντρο του χάρτη γίνεται η θέση του.

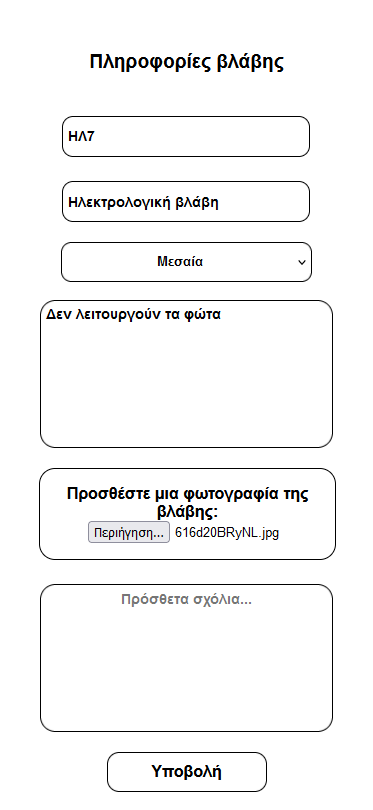
Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, διάγραμμα, γραμματοσειρά

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 4.6: Επιλογή Τοποθεσίας βλάβης

* + 1. Έλεγχος Καταγραφής της βλάβης

Όταν επιλέξει την τοποθεσία της βλάβης, οδηγείται σε μια φόρμα, στην οποία μπορεί να βάλει πληροφορίες για τη βλάβη που θα συμβάλουν στην επίλυσή της.



Εικόνα 4.7: Φόρμα δήλωσης πληροφοριών της βλάβης

* + 1. Επιστροφή στο Home page

Μετά την υποβολή της φόρμας, ο φοιτητής οδηγείται στο Home page και ένα pop-up τον ενημερώνει για την επιτυχή καταχώρηση της βλάβης.



Εικόνα 4.8: Επιτυχής καταχώρηση της βλάβης

* 1. Χρήση του προγράμματος από τον admin

Ο admin έχει ειδικές αρμοδιότητες όπως το να σημειώσει την ολοκλήρωση καθώς και την ανεπάρκεια μια δήλωσης σχολιάζοντάς την.

* + 1. Admin profile

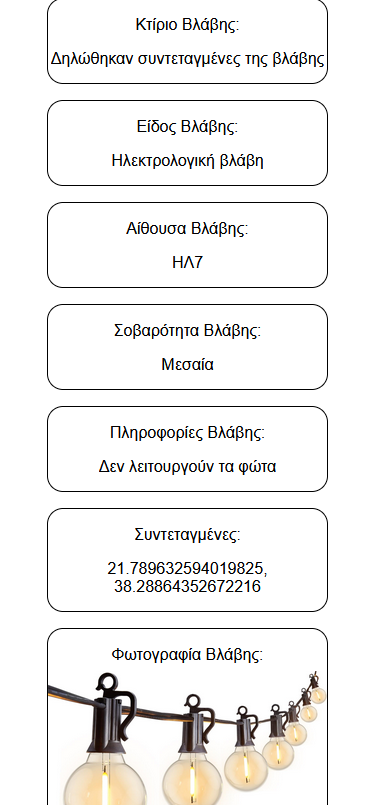
Ο admin κάνει σύνδεση με έναν ειδικό λογαριασμό και εισέρχεται στο profile του όπου μπορεί να δει αν υπάρχει κάποια βλάβη που εκκρεμεί από οποιονδήποτε χρήστη.



Εικόνα 4.9: Admin profile

* + 1. Έλεγχος της δήλωσης

O admin επιλέγει την νέα εκκρεμή δήλωση για να αντλήσει τις πληροφορίες. Αφού ελέγξει ότι τηρούνται ορισμένοι κανόνες, εκτελεί τις κατάλληλες ενέργειες για την επιδιόρθωσή της.



Εικόνα 4.10: Οι πληροφορίες της δήλωσης του φοιτητή

* + 1. Ολοκλήρωση της διαδικασίας

Ο admin, αφού επιδιορθωθεί η βλάβη, επιλέγει την ολοκλήρωση της δήλωσης.



Εικόνα 4.11: Επιλογές του admin για μια εκκρεμή βλάβη

1. ΒΙΒΛΙΟΓΡΑΦΙΑ ΚΑΙ ΚΩΔΙΚΑΣ
   1. Βιβλιογραφία

Για την συγγραφή του κώδικα χρησιμοποιήθηκαν αρχεία από το eclass του μαθήματος, Node.js, Express.js, Handlebars, MDN Web Docs, HTML, Google Maps Javascript API, JavaScript

* 1. Κώδικας

Ο κώδικάς μας βρίσκεται στο [GitHub](https://github.com/JohnTheo02/Malfunction-Report-Project-Web-Programming).

Την εφαρμογή μας μπορείτε να την βρείτε στο public [URL](https://malfunctionreport-26370e429b90.herokuapp.com/).