



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΛΟΠΟΝΝΗΣΟΥ
ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ
ΣΧΕΔΙΑΣΜΟΣ ΚΑΙ ΑΝΑΠΤΥΞΗ
ΔΙΑΔΙΚΤΥΑΚΗΣ ΠΛΑΤΦΟΡΜΑΣ
ΝΗΟΓΝΩΜΟΝΩΝ ΓΙΑ ΕΚΔΟΣΗ
ΠΙΣΤΟΠΟΙΗΤΙΚΩΝ ΣΕ ΠΛΟΙΑ

ΑΝΤΩΝΙΟΥ ΠΑΠΑ (ΑΜ 2497)
ΓΕΩΡΓΙΟΥ ΖΗΣΙΜΑΤΟΥ (ΑΜ 2503)
(πρώην Τμήματος ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ, ΤΕΙ ΔΥΤ. ΕΛΛΑΔΑΣ)

ΕΠΟΠΤΕΥΩΝ ΚΑΘΗΓΗΤΗΣ: Δρ. ΑΘΑΝΑΣΙΟΣ ΚΟΥΤΡΑΣ

ΠΑΤΡΑ, 2022

[Αυτή η σελίδα είναι κενή]

ΥΠΕΥΘΥΝΗ ΔΗΛΩΣΗ ΠΕΡΙ ΜΗ ΛΟΓΟΚΛΟΠΗΣ

Βεβαιώνω ότι είμαι συγγραφέας αυτής της εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της, είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, έχω αναφέρει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Ακόμα δηλώνω ότι αυτή η γραπτή εργασία προετοιμάστηκε από εμένα προσωπικά και αποκλειστικά και ειδικά για την συγκεκριμένη πτυχιακή εργασία και ότι θα αναλάβω πλήρως τις συνέπειες εάν η εργασία αυτή αποδειχθεί ότι δεν μου ανήκει.

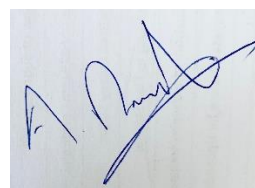
ΟΝΟΜΑΤΕΠΩΝΥΜΟ ΦΟΙΤΗΤΗ 1

ΑΜ

ΥΠΟΓΡΑΦΗ

ΠΑΠΑΣ ΑΝΤΩΝΗΣ

2497



ΟΝΟΜΑΤΕΠΩΝΥΜΟ ΦΟΙΤΗΤΗ 2

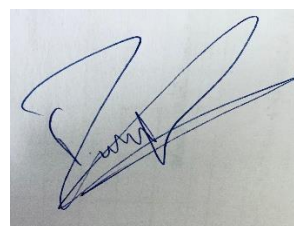
ΑΜ

ΥΠΟΓΡΑΦΗ

(σε περίπτωση που είναι απαραίτητο)

ΖΗΣΙΜΑΤΟΣ ΓΕΩΡΓΙΟΣ

2503



[Αυτή η σελίδα είναι κενή]

ΕΥΧΑΡΙΣΤΙΕΣ

Θα θέλαμε να ευχαριστήσουμε πολύ τον επιβλέποντα καθηγητή μας, κ. Α. Κούτρα για την αποδοχή και εκπόνηση της πρότασης μας, ενός αναγκαίου πλέον θέματος στον τομέα της ναυτιλίας, εν καιρώ που τα πάντα εξελίσσονται και γίνονται πλέον ψηφιακά και πάνω απ' όλα ευχαριστούμε όλους όσους μας υποστήριξαν κυρίως ψυχολογικά, τις οικογένειες μας οι οποίες συνέβαλαν στο να πετύχουμε τους στόχους μας, υπέμειναν και μας καθοδήγησαν καθ' όλη την διάρκεια των σπουδών και της εκπόνησης της πτυχιακής εργασίας μας σε αυτούς τους δύσκολους καιρούς.

Η παρούσα πτυχιακή εργασία αφιερώνεται αναμεταξύ μας, με απώτερο σκοπό την ένδειξη μιας γλυκιάς ανάμνησης καθώς πλησιάζουμε προς το τέλος αυτού του κύκλου των σπουδών και στην αληθοφανή μας φιλία η οποία δημιουργήθηκε στο περιβάλλον των μαθημάτων και δείχνει να κρατάει χρόνια.

[Αυτή η σελίδα είναι κενή]

ΠΡΟΛΟΓΟΣ

Με την πάροδο των χρόνων και την ραγδαία εξέλιξη της τεχνολογίας, παρατηρείται ότι οι καθημερινές ανάγκες των ανθρώπων αλλάζουν και μαζί αλλάζει και ο τρόπος που αυτοί τις αντιμετωπίζουν. Η τεχνολογία εισχωρεί ολοένα και περισσότερο τόσο στην προσωπική όσο και στην επαγγελματική μας ζωή με αποτέλεσμα να γίνεται ένα αναπόσπαστο κομμάτι της κοινωνίας μας, προσφέροντας μας απλόχερα καινοτομίες, λύσεις, ευελιξία, παραγωγικότητα και πολλά ακόμα εργαλεία που με σωστή αξιοποίηση μπορούν όχι μόνο να μας διευκολύνουν στην καθημερινότητα μας αλλά να οδηγήσουν και τον πολιτισμό μας χιλιάδες βήματα μπροστά.

[Αυτή η σελίδα είναι κενή]

ΠΕΡΙΛΗΨΗ

Στο πλαίσιο της πτυχιακής μας εργασίας υλοποιήσαμε μια διαδικτυακή εφαρμογή για την επίλυση προβλημάτων γραφειοκρατίας στους Νηογνώμονες. Η εφαρμογή έχει ως σκοπό την μηχανογράφηση ενός οργανισμού που ασχολείται με την πιστοποίηση και ασφάλεια του πλοίου.

Σημαντικό πρόβλημα στην λειτουργία των οργανισμών αλλά και των πλοίων είναι ο υπερβολικός μεγάλος αριθμός εγγράφων που παρουσιάζει τη πραγματική κατάσταση ενός πλοίου , όπως σχέδια , μελέτες και για τον λόγο αυτό, προτείνουμε στην παρούσα εργασία την δημιουργία ενός ολοκληρωμένου συστήματος διαχείρισης. Ευελπιστούμε η εφαρμογή μας να λύσει προβλήματα όπως η υπερβολική κατανάλωση χρόνου, χαρτιού και κόπου των εργαζομένων και να διαχειρίζονται τις διαδικτυακές αιτήσεις ενδιαφέροντος από πελάτες, για τις εξειδικευμένες υπηρεσίες που παρέχει η εκάστοτε εταιρεία και να ενημερώνουν την εξέλιξη της πορείας της κάθε αίτησης.

Πιο συγκεκριμένα, η διαδικτυακή αυτή πλατφόρμα έχει ως απώτερο στόχο να απλοποιήσει όλες τις διαδικασίες σε σύγκριση με τους τρόπους που ξέρουμε ως σήμερα, καθώς η προσπάθεια που γίνεται είναι για να καλυφθούν όλα τα στάδια μιας διαδικτυακής αίτησης, μέχρι και το τελικό στάδιο της έκδοσης πιστοποιητικού εξ ολοκλήρου ηλεκτρονικά.

Η διαδικτυακή αυτή πλατφόρμα αναπτύχθηκε χρησιμοποιώντας εργαλεία και τεχνολογίες που βρίσκονται στο προσκήνιο και σε μεγάλη ζήτηση στην αγορά εργασίας, προσφέροντας μας την ευελιξία και τις καινοτομίες που φέρνουν μαζί τους και ταυτόχρονα δίνοντας μας εφόδια τα οποία θα μπορέσουμε να αξιοποιήσουμε ενεργά μετά το πέρας της εκπόνησης της πτυχιακής μας.

Η εργασία αυτή δομείται σε τρία μέρη. Αρχικά, γίνεται μια εισαγωγή στο Νηογνώμονα ως οργανισμό. Εν συνεχεία, παρουσιάζονται τα εργαλεία κατασκευής. Τέλος, παρουσιάζεται με εικονική αναπαράσταση η δημιουργία της πλατφόρμας αλλά και η τελική της μορφή.

ABSTRACT

As part of our dissertation, we implemented an online application for solving bureaucratic problems in the Classification Society. The application aims to computerize an organization that deals with the certification and safety of the ship.

An important problem in the operation of organizations and ships is the excessive number of documents that present the real condition of a ship, such as plans, studies and for this reason, we propose in this work the creation of an integrated management system. We hope our application solves problems such as excessive consumption of time, paper and effort of employees and to manage online requests for interest from customers, for the specialized services provided by each company and to inform the progress of each application.

More specifically, this online platform has the **goal** of simplifying all the procedures compared to the ways we know today, as the effort is made to cover all the stages of an online application, up to the final stage of issuing a certificate entirely electronically.

This online platform was developed using tools and technologies that are in the forefront and in great demand in the labor market, offering us the flexibility and innovations they bring with them and at the same time giving us supplies that we will be able to actively use after its development. our degree.

This work is structured in three parts. Initially, an introduction is made to the Registry as an organism. Next, the construction tools are presented. Finally, the creation of the platform and its final form are presented with a virtual representation.

[Αυτή η σελίδα είναι κενή]

ΠΕΡΙΕΧΟΜΕΝΑ

ΕΥΧΑΡΙΣΤΙΕΣ	v
ΠΡΟΛΟΓΟΣ	vii
ΠΕΡΙΛΗΨΗ	ix
ABSTRACT	x
ΠΕΡΙΕΧΟΜΕΝΑ	xii
ΕΥΡΕΤΗΡΙΟ ΕΙΚΟΝΩΝ	xv
ΕΙΣΑΓΩΓΗ	xx
1 ΝΗΟΓΝΩΜΟΝΑΣ ΚΑΙ ΕΤΑΙΡΕΙΑ	23
1.1 ΝΗΟΓΝΩΜΟΝΑΣ.....	23
1.1.1 Βασικές Πληροφορίες σχετικά με τον σημαντικό ρόλο των Νηογνώμωνων στον Ναυτιλιακό Τομέα.	23
1.1.2 Πόσοι νηογνώμονες υπάρχουν;	23
1.1.3 Είναι υποχρεωτική η εκδοση πιστοποιητικού Νηογνώμονα για τα πλοία; 24	
1.1.4 Τι είναι το πιστοποιητικό κλάσης;	24
1.1.5 Ποιος είναι ο ρόλος του νηογνώμονα;	24
1.2 ΙΣΤΟΡΙΑ.....	24
1.3 ΓΡΑΦΕΙΟΚΡΑΤΙΑ, ΚΑΘΥΣΤΕΡΗΣΗ ΚΑΙ ΛΑΘΗ.....	25
1.4 ΑΠΟΔΟΤΙΚΟΤΕΡΗ ΚΑΙ ΠΑΡΑΓΩΓΙΚΟΤΕΡΗ ΛΥΣΗ.....	30
2 ΤΕΧΝΟΛΟΓΙΕΣ ΚΑΙ ΕΡΓΑΛΕΙΑ	31
2.1 ΤΕΧΝΟΛΟΓΙΕΣ ΣΤΟ ΣΗΜΕΡΑ	31
2.2 ΕΡΓΑΛΕΙΑ.....	31
2.2.1 Task Management - Trello	32
2.2.2 UI/UX Design - Figma	33
2.2.3 Virtualization - Docker	34
2.2.4 Versioning- Git + Github	39
2.2.5 Code Editor/IDE – Visual Studio Code	42
2.2.6 Package Manager – NPM	44
2.2.7 Database - MongoDB	45
2.2.8 Server- NodeJS	47
2.2.9 UI Development	49

3	Η ΥΛΟΠΟΙΗΣΗ ΤΗΣ ΕΦΑΡΜΟΓΗΣ.....	57
3.1	ΣΧΕΔΙΑΣΜΟΣ ΕΦΑΡΜΟΓΗΣ	57
3.2	ΑΝΑΠΤΥΞΗ ΕΦΑΡΜΟΓΗΣ	58
3.2.1	ΓΡΑΦΙΚΟ ΠΕΡΙΒΑΛΛΟΝ	59
3.2.2	ΥΠΗΡΕΣΙΑ/ΔΙΑΚΟΜΙΣΤΗΣ	78
3.2.3	ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ.....	81
3.3	ΕΝΩΣΗ ΚΑΙ ΑΠΟΤΕΛΕΣΜΑ.....	84
	ΣΤΟΧΟΙ ΚΑΙ ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ.....	85
	ΣΥΜΠΕΡΑΣΜΑΤΑ	87
	ΑΝΑΦΟΡΕΣ	88

[Αυτή η σελίδα είναι κενή]

ΕΥΡΕΤΗΡΙΟ ΕΙΚΟΝΩΝ

Εικόνα 1 Πίνακας Ποής.....	27
Εικόνα 2 User Case.....	28
Εικόνα 3 User Case Web Platform	29
Εικόνα 4 Trello	33
Εικόνα 5 Figma.....	34
Εικόνα 6 Docker	34
Εικόνα 7 Docker Architecture	36
Εικόνα 8 Docker Components	37
Εικόνα 9 Docker Basic Taxonomy	38
Εικόνα 10 Git.....	39
Εικόνα 11 Git for Professionals	40
Εικόνα 12 Git Interface.....	42
Εικόνα 13 Visual Studio Code.....	42
Εικόνα 14 NPM	44
Εικόνα 15 NPM Diagram	45
Εικόνα 16 MongoDB Atlas Interface	46
Εικόνα 17 MongoDB Collections.....	47
Εικόνα 18 NodeJS.....	48
Εικόνα 19 NodeJS Pros.....	49
Εικόνα 20 Basic Frontend Technologies	50
Εικόνα 21 Bootstrap.....	50
Εικόνα 22 Bootstrap v5.....	52
Εικόνα 23 JavaScript	54
Εικόνα 24 React Properties.....	56
Εικόνα 25 React Use by Companies.....	56
Εικόνα 26 Figma Design.....	57
Εικόνα 27 Flow Diagram.....	58
Εικόνα 28 Web App Homepage	59
Εικόνα 29 Homepage Menu	59
Εικόνα 30 Survey Tracking	60

Εικόνα 31 Login Page.....	60
Εικόνα 32 Users	61
Εικόνα 33 Users' Roles.....	62
Εικόνα 34 Dashboard.....	62
Εικόνα 35 Menu.....	63
Εικόνα 36 Profile	64
Εικόνα 37 Edit Profile.....	64
Εικόνα 38 Users	65
Εικόνα 39 Edit User.....	65
Εικόνα 40 User Role	66
Εικόνα 41 Register New User.....	66
Εικόνα 42 New Request.....	67
Εικόνα 43 Request Main Particulars.....	67
Εικόνα 44 New or Existing Vessel	68
Εικόνα 45 Existing Vessel	68
Εικόνα 46 Services Selection.....	69
Εικόνα 47 Request Status Selection.....	69
Εικόνα 48 Under Process Requests	70
Εικόνα 49 Materialized Requests	70
Εικόνα 50 Cancelled Requests.....	71
Εικόνα 51 Request Info.....	71
Εικόνα 52 Edit Request.....	72
Εικόνα 53 In Progress Cases.....	72
Εικόνα 54 Case Info.....	73
Εικόνα 55 Case Additional Info.....	73
Εικόνα 56 Edit Case.....	74
Εικόνα 57 Case – Status of Application	74
Εικόνα 58 Completed Cases	75
Εικόνα 59 App Notifications	75
Εικόνα 60 Survey Status	75
Εικόνα 61 Vessel Info.....	76
Εικόνα 62 Vessel Additional Info.....	76
Εικόνα 63 Vessel Certificate.....	77

Εικόνα 64 Edit Certificate.....	77
Εικόνα 65 Manager Role Interface	78
Εικόνα 66 Client Role Interface.....	78
Εικόνα 67 Service router paths	79
Εικόνα 68 Service Controllers	80
Εικόνα 69 Service GET Requests	80
Εικόνα 70 Service DB Models.....	81
Εικόνα 71 SQL vs NoSQL.....	82
Εικόνα 72 DB Diagrams	82
Εικόνα 73 MongoDB Atlas Main Interface	83
Εικόνα 74 Database Collections	83
Εικόνα 75 Request DB Example	84

[Αυτή η σελίδα είναι κενή]

[Αυτή η σελίδα είναι κενή]

ΕΙΣΑΓΩΓΗ

Μετά από έρευνα, παρατηρήθηκε ότι ενώ πολλές εταιρείες και οργανισμοί φρόντιζαν να συμβαδίζουν με τις εξελίξεις, ένας τομέας που είναι άρρητα συνδεδεμένος με την Ελλάδα και την γεωγραφική της θέση, έχει καθυστερήσει την επένδυση και ανάπτυξη σε ένα τέτοιο απαραίτητο πλέον εργαλείο. Ο τομέας αυτός είναι της Ναυτιλίας.

Όπως και στην αυτοκινητοβιομηχανία, έτσι και στην Ναυτιλία απαιτείται η συμμόρφωση ενός πλοίου με τις απαιτήσεις, εκδίδοντας πιστοποιητικό κλάσης πλοίων και εισάγοντας το στο μητρώο πλοίων της εταιρείας. Η συμμόρφωση αυτή επιβεβαιώνεται από έναν Νηογνώμονα οργανισμό.

Ο Νηογνώμονας είναι ο Οργανισμός που πιστοποιεί το πλοίο σε σχέση με την ασφάλεια τόσο του πλοίου ως κατασκευής όσο και των επιβαίνόντων του. Κάθε πλοίο ανεξαρτήτου μεγέθους αριθμό πληρώματος και επιβατών ή έτους κατασκευής χρειάζεται έναν Νηογνώμονα που να το πιστοποιήσει σε σχέση πάντα με την σημαία που φέρει. Αν ένα πλοίο για παράδειγμα δεν έχει πιστοποιητικό κλάσης δεν μπορεί να έχει και ασφάλεια. Είναι απαραίτητη προϋπόθεση για τις ασφαλιστικές εταιρίες το πλοίο να πιστοποιείται από έναν αναγνωρισμένο Νηογνώμονα.

Οι κανόνες του εκάστοτε Νηογνώμονα απαιτούν κάθε ταξινομημένο σκάφος να υποβάλλεται σε περιοδικές επιθεωρήσεις καθ' όλη τη διάρκεια ζωής του για να διαπιστώνεται κατά πόσο διατηρείται σύμφωνα με τα πρότυπα Κλάσης του εκάστοτε πλοίου. Οι επιθεωρήσεις κλάσης βασίζονται σε έναν τακτικό πενταετή κύκλο που περιλαμβάνει:

- Ετήσιες Επιθεωρήσεις
- Μια ενδιάμεση επιθεώρηση ανά μια ολοκληρωμένη Ετήσια
- Ειδική επιθεώρηση, συμπεριλαμβανομένης της επιθεώρησης ξηράς ή της υποβρύχιας επιθεώρησης αντί της ξηράς επιθεώρησης δύο φορές σε μια περίοδο πέντε ετών

Η ανάθεση κλάσης βασίζεται στα χαρακτηριστικά του πλοίου και παρατηρήσεων του, που υποδηλώνουν τον βαθμό εμπιστοσύνης που αξίζει το κάθε πλοίο, ανατίθενται μετά την ικανοποιητική ολοκλήρωση των επιθεωρήσεων που διενεργήθηκαν από τους επιθεωρητές της εταιρείας και την ολοκλήρωση της επαλήθευσης της συμμόρφωσης με τους κανόνες του εκάστοτε Νηογνώμονα. Οι παρατηρήσεις κλάσης καλύπτουν τόσο τις υποχρεωτικές όσο και τις προαιρετικές απαιτήσεις και αναφέρουν την σημαντικότητα του προβλήματος και συνήθως απαιτούν την άμεση ενέργεια για την διόρθωση τους για την σωστή λειτουργία του πλοίου με τον μικρότερο πιθανό κίνδυνο. Ωστόσο σπάνιο φαινόμενο είναι η απαγόρευση πλεύσης ενός πλοίου όσες παρατηρήσεις και να έχει καθώς το πιστοποιητικό δίνεται στον καπετάνιο με απώτερο σκοπό οι παρατηρήσεις που έχουν αναφερθεί πάνω στο πιστοποιητικό να έχουν διορθωθεί μέχρι την διεξαγωγή της επόμενης επιθεώρησης. Κατασκευαστικά στοιχεία και τεχνικά χαρακτηριστικά του πλοίου που εισέρχεται σε κλάση καταγράφονται στο

πιστοποιητικό κλάσης και σε άλλα έγγραφα που εκδίδονται από τον εκάστοτε Νηογνώμονα.

Οι επιθεωρητές ωστόσο παρακολουθούν επίσης επισκευές ή/ και τροποποιήσεις για να διαπιστώσουν ότι οι εργασίες που εκτελούνται επαναφέρουν τη μονάδα σε μια κατάσταση που συμμορφώνεται με τους κανόνες του εκάστοτε Νηογνώμονα.

Οι κανόνες και οι κανονισμοί του εκάστοτε Νηογνώμονα προέρχονται από τις αρχές της ναυτικής αρχιτεκτονικής, των συμβατικών απαιτήσεων του Διεθνούς Οργανισμού Ναυτιλίας (International Maritime Organization IMO) θαλάσσιας μηχανικής, των εγκυκλίων (Maritime Safety Committee MSC), των ενοποιημένων ερμηνειών του ΟΣΒΑ, στην ανατροφοδότηση υπηρεσιών και στα ερευνητικά ευρήματα. Επιπλέον, η κοινωνία των Νηογνομόνων έχει αναπτύξει εγχειρίδια καθοδήγησης και οδηγιών για τις θαλάσσιες επιθεωρήσεις για τη συμμόρφωση με την κλάση και διάφορες νομικές απαιτήσεις. Ωστόσο οι απαιτήσεις σχεδιασμού, κατασκευής, το πεδίο εφαρμογής των δοκιμών και των ερευνών καθορίζονται από τους κανόνες της κλάσης και κατασκευής πλοίων.

[Αυτή η σελίδα είναι κενή]

1 ΝΗΟΓΝΩΜΟΝΑΣ ΚΑΙ ΕΤΑΙΡΕΙΑ

1.1 ΝΗΟΓΝΩΜΟΝΑΣ

Ο νηογνώμονας είναι ένας μη κυβερνητικός οργανισμός που θεσπίζει και διατηρεί τεχνικά πρότυπα για την κατασκευή και λειτουργία πλοίων και υπεράκτιων κατασκευών. Οι νηογνώμονες πιστοποιούν ότι η ναυπήγηση ενός σκάφους συμμορφώνεται με τα σχετικά πρότυπα και διενεργούν τακτικές επιθεωρήσεις σε υπηρεσία για να διασφαλίζουν τη συνεχή συμμόρφωση με τα πρότυπα. Επί του παρόντος, περισσότεροι από 50 οργανισμοί περιγράφουν τις δραστηριότητές τους ως οργανισμοί Νηογνώμονες.

1.1.1 Βασικές Πληροφορίες σχετικά με τον σημαντικό ρόλο των Νηογνώμωνων στον Ναυτιλιακό Τομέα.

Οι νηογνώμονες θέτουν τεχνικούς κανόνες με βάση την εμπειρία και την έρευνα, επιβεβαιώνουν ότι τα σχέδια και οι υπολογισμοί πληρούν αυτούς τους κανόνες, τα πλοία και οι κατασκευές ερευνούν κατά τη διαδικασία κατασκευής και θέσης σε λειτουργία και περιοδικά επιθεωρούν τα πλοία για να διασφαλίζουν ότι συνεχίζουν να πληρούν τους κανόνες. Οι νηογνώμονες είναι επίσης υπεύθυνοι για την ταξινόμηση των πλατφορμών πετρελαίου, άλλων υπεράκτιων κατασκευών και υποβρυχίων. Αυτή η διαδικασία επιθεώρησης καλύπτει κινητήρες ντίζελ, σημαντικές αντλίες πλοίων και άλλα ζωτικά μηχανήματα.

Οι Νηογνώμονες λειτουργούν με συμβάσεις. Είναι Οργανισμοί εξουσιοδοτημένοι από τις σημαίες ανα τον κόσμο. Αν η σημαία είναι Ελληνική η νομοθεσία που ακολουθεί είναι βάση εγκυκλίων. Αν είναι ξένη οι σημαία ακολουθεί ξένες συμβάσεις όπως International Convention for the Safety of Life at Sea (SOLAS), International Convention for the Prevention of Pollution from Ships (MARPOL), LOAD LINE βάσει του Διεθνούς Οργανισμού Ναυτιλίας (International Maritime Organization IMO). Οι συμβάσεις ελληνικές και διεθνείς τέθηκαν σε εφαρμογή ανα τα χρόνια μετά από κάποιο πολύ μεγάλο συμβάν πχ ναυάγιο.

1.1.2 Πόσοι νηογνώμονες υπάρχουν;

Σήμερα υπάρχουν περισσότεροι από πενήντα νηογνώμονες στον κόσμο εκ των οποίων μόνο οι δεκατρείς είναι μέλη της Διεθνούς Ένωσης Νηογνώμωνων.

Η Διεθνής Ένωση Νηογνώμωνων, Δ.Ε.Ν., (International Association of Classification Societies – IACS) ή αλλιώς IACS(International Association of Classification Societies) με έδρα το Λονδίνο, απαρτίζεται από δεκατρείς (13) νηογνώμονες οι οποίοι είναι οι μόνοι διεθνώς ανεγνωρισμένοι νηογνώμονες στον κόσμο ενώ στο σύνολο υπάρχουν περισσότεροι από πενήντα.

Οι μεγαλύτεροι από αυτούς είναι Ο Αγγλικός Νηογνώμονας, (LR) [Lloyd's Register], Ο Αμερικανικός Νηογνώμονας, (ABS) [American Bureau of Shipping], ο Γαλλικός Νηογνώμονας, (BV) [Bureau Veritas] και ο Ιαπωνικός Νηογνώμονας, (NK) [Nippon Kaiji Kyokai] (Class NK).

1.1.3 Είναι υποχρεωτική η έκδοση πιστοποιητικού Νηογνώμονα για τα πλοία;

Τα κράτη σημαίας απαιτούν τα πλοία να σχεδιάζονται, να κατασκευάζονται και να συντηρούνται σύμφωνα με τα πρότυπα των νηογνωμόνων. Αυτό καθιστά το "Πιστοποιητικό κλάσης" νόμιμο πιστοποιητικό που το θεσπίζει και υποχρεωτικό.

1.1.4 Τι είναι το πιστοποιητικό κλάσης;

Πιστοποιητικό κλάσης είναι Έγγραφο που εκδίδεται από οργανισμό νηογνώμονα που επιβεβαιώνει την κατηγορία στην οποία είναι νηολογημένο ένα σκάφος.

1.1.5 Ποιος είναι ο ρόλος του νηογνώμονα;

Σκοπός ενός νηογνώμονα είναι η παροχή υπηρεσιών ταξινόμησης και καταστατικών υπηρεσιών και βοήθειας στη ναυτιλιακή βιομηχανία και τους ρυθμιστικούς φορείς όσον αφορά την ασφάλεια στη θάλασσα και την πρόληψη της ρύπανσης, με βάση τη συσσώρευση ναυτικών γνώσεων και τεχνολογίας.

1.2 ΙΣΤΟΡΙΑ

Στο δεύτερο μισό του 18ου αιώνα, οι έμποροι, οι πλοιοκτήτες και οι καπετάνιοι του Λονδίνου συχνά συγκεντρώνονταν στην καφετέρια του Edward Lloyds για κουτσομπολιό και έκαναν συμφωνίες, συμπεριλαμβανομένου του διαμοιρασμού των κινδύνων και των ανταμοιβών μεμονωμένων ταξιδιών. Δεν άργησε να συνειδητοποιήσει ότι οι ασφαλιστές χρειαζόνταν έναν τρόπο αξιολόγησης της ποιότητας των πλοίων που τους ζητήθηκε να ασφαλίσουν. Το 1760 ιδρύθηκε η Εταιρεία Νηολογίου — ο πρώτος νηογνώμονας και αυτός που στη συνέχεια θα γινόταν Νηολόγιο της Lloyd's — για να δημοσιεύει ετήσιο νηολόγιο πλοίων. Η δημοσίευση αυτή επιχείρησε να ταξινομήσει την κατάσταση του κύτους και του εξοπλισμού του πλοίου. Εκείνη την εποχή, έγινε μια προσπάθεια να ταξινομηθούν οι συνθήκες κάθε πλοίου σε ετήσια βάση.

1.3 ΓΡΑΦΕΙΟΚΡΑΤΙΑ, ΚΑΘΥΣΤΕΡΗΣΗ ΚΑΙ ΛΑΘΗ

Μια σημαντική καινοτομία που έφερε η άνοδος της τεχνολογίας ήταν η μηχανογράφηση. Μια μεσαίου μεγέθους εταιρεία με χιλιάδες πελάτες, διαφορετικές υπηρεσίες και προσωπικό, είχε να αντιμετωπίσει στην καθημερινότητα της όλη τη διαχείριση, αρχειοθέτηση και γραφειοκρατία με ελάχιστα μέσα και εργαλεία. Εργαλεία όπως τα χειρόγραφα βιβλία, αποδείξεις, μπλοκάκια σύντομα αντικαταστάθηκαν από ένα πιο εξελιγμένο, ηλεκτρονικό πλέον σύστημα που γλίτωσε χώρο, χρόνο αλλά και κόπο στις εταιρείες και βοήθησε καταλυτικά στην παραγωγικότητα και εξέλιξη τους. Με γνώμονα την φιλική εμπειρία χρήσης, τα αυτοματοποιημένα συστήματα καταχωρήσεις και διαχειρίσεις είναι ειδικά φτιαγμένα και διαμορφωμένα για τις ανάγκες της κάθε εταιρείας, λύνοντας έτσι προβλήματα που μπορούσαν να δυσκολέψουν το έργο των υπαλλήλων και την οργάνωση των εταιριών.

Ως τώρα η διαδικασία για μια αίτηση ενδιαφέροντος από κάποιον πελάτη για να επιθεωρήσει το πλοίο του με τον οργανισμό του νηογνώμονα μέχρι και την έκδοση του πιστοποιητικού γίνεται ως εξής.

- Γίνεται κάποια επικοινωνία είτε μέσω τηλεφώνου είτε μέσω email που αιτείται ο πελάτης, ο οποίος ενδιαφέρεται να επιθεωρήσει το σκάφος/πλοίο του. Αντίστοιχα ο υπεύθυνος που είχε τηλεφωνική επικοινωνία δημιουργεί εσωτερικό email, ώστε να δημιουργηθεί email αίτησης .
- Εν συνεχεία το email τυπώνεται ώστε να προχωρήσει από την γραμματεία στο πρωτόκολλο.
- Του παραχωρείται αριθμός πρωτοκόλλου με σφραγίδα, ώστε να παρακολουθηθεί η πορεία αίτησης ενδιαφέροντος, σε περίπτωση που κάποιος το αναζητήσει.
- Προχωράει στο αρμόδιο τμήμα όπου επικοινωνεί με τον πελάτη για να του στείλει προηγούμενα πιστοποιητικά μελέτες και σχέδια κατασκευής του πλοίου που να αναγραφούν βασικές τεχνικές πληροφορίες για την ιστορία κατασκευής του πλοίου. Όταν ληφθούν και αυτά τυπώνονται και συμπληρώνεται στα έγγραφα της αίτησης.
- Αναλαμβάνει κάποιος εξειδικευμένος τεχνικός να αξιολογήσει εάν το πλοίο πληροί κάποιες πολύ βασικές προϋποθέσεις για να εισαχθεί στον Νηογνώμονα και εφόσον παραχωρηθεί η έγκριση με το αντίστοιχο έγγραφο το οποίο είναι για εσωτερική ενημέρωση, προχωράει στο οικονομικό τμήμα.
- Στο οικονομικό τμήμα επικοινωνούν με τον πελάτη μέσω email για το κόστος των υπηρεσιών και εφόσον συμφωνήσουν τότε η αίτηση ξεκινάει να γίνεται υπόθεση πλέον καθώς του δίνεται νέος αριθμός πρωτοκόλλου για να εισαχθεί στον στόλο του Οργανισμού.
Σε αυτό το σημείο ανοίγεται φάκελος και αρχειοθετούνται όλα τα τυπωμένα έγγραφα μέσα και ο φάκελος αριθμείται κατά αύξοντα αριθμό για να ευκολύνει την εύρεσή του.
- Αμέσως ενημερώνεται ο αντίστοιχος επιθεωρητής που πληροί τα αντίστοιχα κριτήρια για να επιθεωρήσει το εκάστοτε πλοίο για την ημέρα και το σημείο που θα γίνει η επιθεώρηση. Η επικοινωνία μεταξύ του Οργανισμού και του επιθεωρητή επίσης τυπώνεται για αρχειοθετείται στο φάκελο.

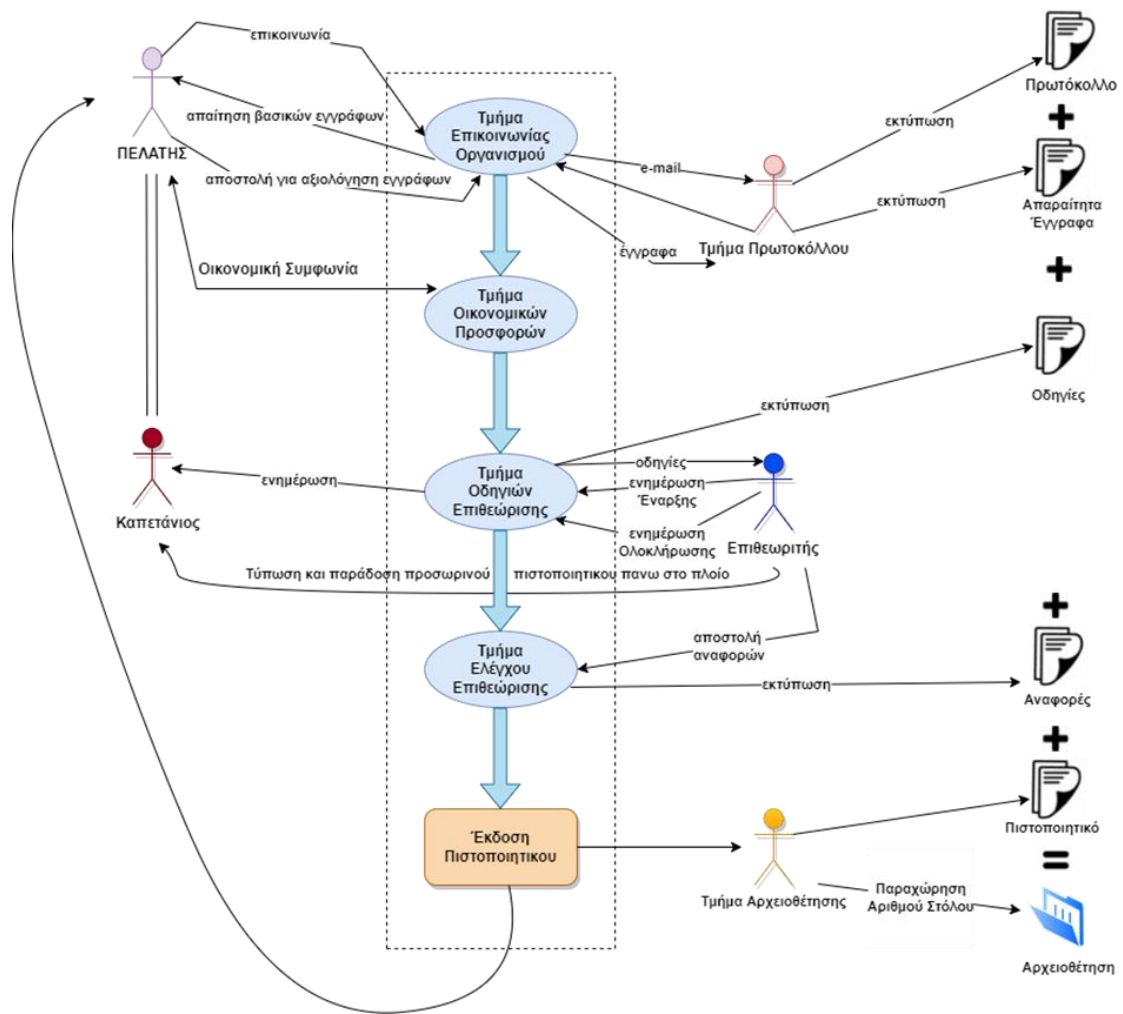
- Παράλληλα ο πελάτης σε περίπτωση που δεν είναι ο ίδιος καπετάνιος του πλοίου ενημερώνει τον καπετάνιο για το πότε και που θα γίνει η επιθεώρηση.
- Το αρμόδιο τμήμα αναλαμβάνει να καθοδηγήσει τον επιθεωρητή με οδηγίες για το πλοίο και ποιες ενέργειες θα ακολουθήσει. Όταν σταλθούν οι οδηγίες αυτές τυπώνονται και αρχειοθετούνται στο φάκελο.
- Ο επιθεωρητής ενημερώνει τον Οργανισμό ότι επρόκειτο να διεξάγει κανονικά την επιθεώρηση και επικοινωνεί ξανά με τον Νηογνώμονα αφού ολοκληρωθεί η επιθεώρηση καθώς μπορεί η επιθεώρησή να διαρκεί κάποιες μέρες ανάλογα το μέγεθος, τον τύπο του πλοίου και τα ευρήματά/παρατηρήσεις που θα βρει ο επιθεωρητής .
- Κατά την ολοκλήρωση της επιθεώρησης ο επιθεωρητής εκδίδει πάνω στο σκάφος το πιστοποιητικό και το δίνει στον καπετάνιο, καθώς στην πίσω σελίδα αναγράφονται οι παρατηρήσεις εάν υπάρχουν . Ο απώτερος σκοπός είναι το πλοίο να έχει συμμορφωθεί σε τυχόν παρατηρήσεις κατά την επομένη επιθεώρησή του.
- Ο επιθεωρητής μέσα σε ένα εύλογο διάστημα στέλνει κάποιες φόρμες και αναφορές που αξιολογούν την εμπειρία του κατά την διεξαγωγή της επιθεώρησης πάνω στο πλοίο μαζί με τα ευρήματά του.
- Μόλις ληφθούν από τον Οργανισμό τυπώνονται και όλος ο φάκελος πάει στο τμήμα του consideration. Εκεί ερευνούν αν όλα όσα αναφέρει ο επιθεωρητής είναι αληθές και ότι όλα διεξάχθηκαν με την απαραίτητη ασφάλεια καθώς ελέγχονται οι υπηρεσίες του επιθεωρητή εάν έγιναν σωστά και με βάση τις οδηγίες που του δόθηκαν. Εφόσον όλα αυτά ελεγχθούν, αρχειοθετούνται όλα μαζί στον φάκελο καθώς ταυτόχρονα εκδίδεται το πιστοποιητικό με την σφραγίδα του Οργανισμού και αποστέλλεται στον πελάτη /διαχειριστή/πλοιοκτήτη. Έπειτα το σκάφος αποτελεί πλέον στόλο του Οργανισμού, του παραχωρείται και ένα νούμερο στόλου όπου αναγράφεται πάνω στον φάκελο για ακόμα ευκολότερη εύρεση. Ο φάκελος αποθηκεύεται στο αρχείο και παραμένει εκεί ως που κάποιος το αναζητήσει.

ΜΕ ΜΙΑ ΜΑΤΙΑ ΣΤΟΝ ΠΑΡΑΚΑΤΩ ΠΙΝΑΚΑ Η ΡΟΗ :

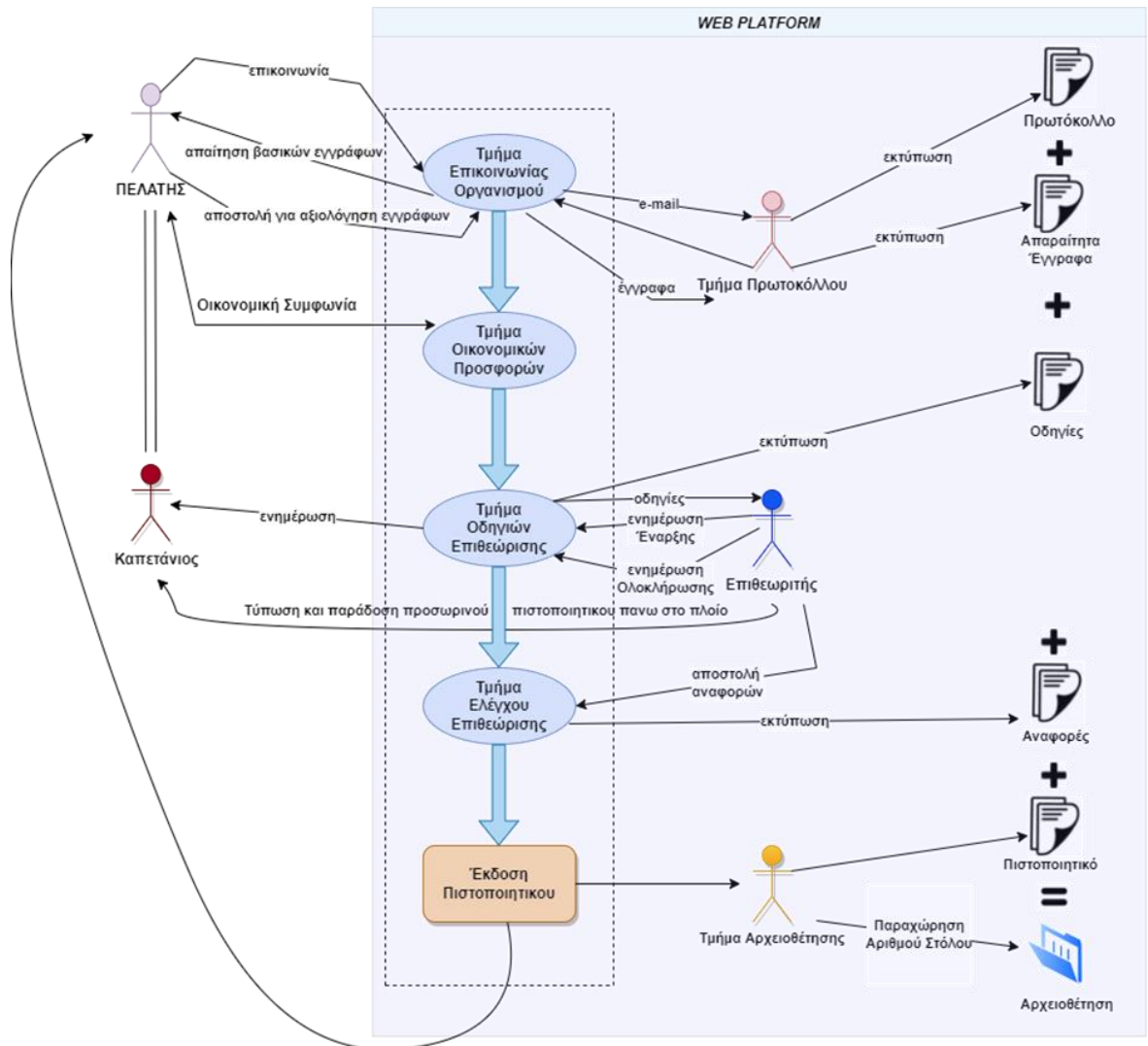


Εικόνα 1 Πίνακας Ροής

Παρακάτω παρουσιάζουμε ένα user case σενάριο με την υφιστάμενη αλληλεπίδραση μεταξύ των τμημάτων και των ενεργειών που γίνονται από την στιγμή που ένας πελάτης εκδηλώνει το ενδιαφέρον του για κάποια επιθεώρηση και ύστερα ένα αντίστοιχο διάγραμμα user case όπου φαίνεται σε ποιες ενέργειες της εταιρείας θέλουμε να καλύψουμε για την επιτάχυνση και την διευκόλυνση των διαδικασιών με την δημιουργία της πλατφόρμας μας.



Εικόνα 2 User Case



Εικόνα 3 User Case Web Platform

1.4 ΑΠΟΔΟΤΙΚΟΤΕΡΗ ΚΑΙ ΠΑΡΑΓΩΓΙΚΟΤΕΡΗ ΛΥΣΗ

Εν μέσω της πλατφόρμας που σχεδιάσαμε , οι χρήστες , οι υπάλληλοι και οι πελάτες θα έχουν μια πιο άμεση και πιο αποδοτικότερη / παραγωγικότερη διαδικασία στην αίτηση για κάποιο πιστοποιητικό καθώς και ο πελάτης θα ενημερώνεται για την πορεία της αίτησης του και ο χρήστης θα δουλεύει σε ένα πιο εκσυγχρονισμένο περιβάλλον το οποίο θα του αποφέρει χρόνο και απόδοση.

Καθώς τα βήματα θα είναι λιγότερα, η διαδικασία θα είναι μια και θα χρειάζεται απλά μικρές επεμβάσεις. Οποτε δεν θα χρειάζεται πλέον να τυπώνονται τόσα έγγραφα και να πιάνουν χώρο στο αρχείο με φάκελους καθώς όλα θα είναι ψηφιακά με τις βασικές πληροφορίες που χρειάζονται. Με την χρήση της πλατφόρμας εξοικονομούμε καταρχάς μελανιά , χαρτιά , εκτυπωτές , φάκελους , χώρο στο αρχείο , την γραφειοκρατία , τα βιβλία πρωτόκολλου , τα ταμπελάκια με τα ονόματα φακέλων και πολλά άλλα , καθώς έχουμε άμεση ενημέρωση μέσω της πλατφόρμας για την κάθε αίτηση και τη κάθε υπόθεση , καλύτερη οργάνωση , λιγότερα λάθη , πιο άμεση εύρεση και πολύ περισσότερη παραγωγικότητα καθώς η διαδικασίες γίνονται πολύ πιο άπλες .Τα πρωτοκολλά δημιουργούνται αυτόματα και ο πελάτης ενημερώνεται για την αίτηση/υπόθεση του μέσω του tracking.

2 ΤΕΧΝΟΛΟΓΙΕΣ ΚΑΙ ΕΡΓΑΛΕΙΑ

2.1 ΤΕΧΝΟΛΟΓΙΕΣ ΣΤΟ ΣΗΜΕΡΑ

Ως **Τεχνολογία** ορίζεται το άθροισμα των τεχνικών, των δεξιοτήτων, των μεθόδων ή/και των διαδικασιών που χρησιμοποιούνται και αξιοποιούνται κατά την παραγωγή αγαθών, προϊόντων ή υπηρεσιών για την βελτίωση της ζωής του ανθρώπου, ή για την επίτευξη βασικών στόχων κατά την επιστημονική έρευνα.

Σημαντικός μας στόχος ήταν να χρησιμοποιήσουμε αυτές τις τεχνικές, δεξιότητες, μεθόδους και διαδικασίες που όμως να ανταποκρίνονται στις σημερινές απαιτήσεις και να μας προμηθεύουν με όλα τα εφόδια και τις ευκολίες που έχει επιφέρει η συνολική εξέλιξη του ανθρώπου μέχρι σήμερα. Έτσι, βασιζόμενοι σε καλά εδραιωμένες τεχνολογίες, θέσαμε τις βάσεις και χτίσαμε από πάνω προσθέτοντας καινοτόμες τεχνολογίες και μέσα από αυτό το συνονθύλευμα πραγματοποιήθηκε η δικιά μας ιδέα

2.2 ΕΡΓΑΛΕΙΑ

Ένα **προγραμματιστικό εργαλείο** είναι ένα πρόγραμμα υπολογιστή ή μια εφαρμογή που χρησιμοποιούν οι προγραμματιστές για να δημιουργήσουν, αποσφαλματώσουν ή να συντηρήσουν άλλα προγράμματα και εφαρμογές. Ο όρος αναφέρεται συνήθως σε σχετικά απλά προγράμματα που μπορούν να συνδυαστούν μαζί για την επίτευξη κάποιου αποτελέσματος, όπως ανάλογα κάποιος τεχνίτης θα χρησιμοποιούσε πολλά εργαλεία για να δημιουργήσει ένα φυσικό αντικείμενο.

Η ιστορία των εργαλείων προγραμματισμού άρχισε με τους πρώτους υπολογιστές στις αρχές της δεκαετίας 1950 όταν πρωτοεμφανίστηκαν εργαλεία όπως οι συνδέτες (linkers), οι φορτωτές και τα προγράμματα ελέγχου. Η έλευση του Unix στις αρχές της δεκαετίας του 1970 έφερε περισσότερα εργαλεία, όπως το grep, το awk και το make, τα οποία ήταν ευέλικτα σχεδιασμένα ώστε να συνεργάζονται μεταξύ τους μέσω διασωλήνωσης.

Τα εργαλεία ήταν αρχικά απλά και "ελαφρά". Μερικά όμως που χρησιμοποιούνται πιο συχνά ενσωματώνονται σε ολοκληρωμένα περιβάλλοντα ανάπτυξης (IDEs). Σε αυτά τα περιβάλλοντα η εργασία του προγραμματιστή μπορεί να γίνει πιο γρήγορα και εύκολα αλλά χάνεται κάτι από την ευελιξία και την δύναμη των καθαυτό εργαλείων.

Ο διαχωρισμός μεταξύ εργαλείων και εφαρμογών είναι ασαφής. Για παράδειγμα πολλοί προγραμματιστές χρησιμοποιούν βάσεις δεδομένων σαν εργαλεία. Για πολλά χρόνια τα εργαλεία computer-assisted software engineering (CASE) ήταν διαδεδωμένα αλλά η δημιουργία πραγματικά επιτυχημένων εργαλείων αποδείχτηκε δύσκολη. Κατά μια έννοια τα εργαλεία CASE έδιναν έμφαση στο σχεδιασμό όπως η UML, αν και τα πιο επιτυχημένα από αυτά είναι ολοκληρωμένα περιβάλλοντα ανάπτυξης.

Η δυνατότητα να χρησιμοποιεί μια πληθώρα εργαλείων παραγωγικά θεωρείται καλή πρακτική από τους προγραμματιστές και τους μηχανικούς λογισμικού.

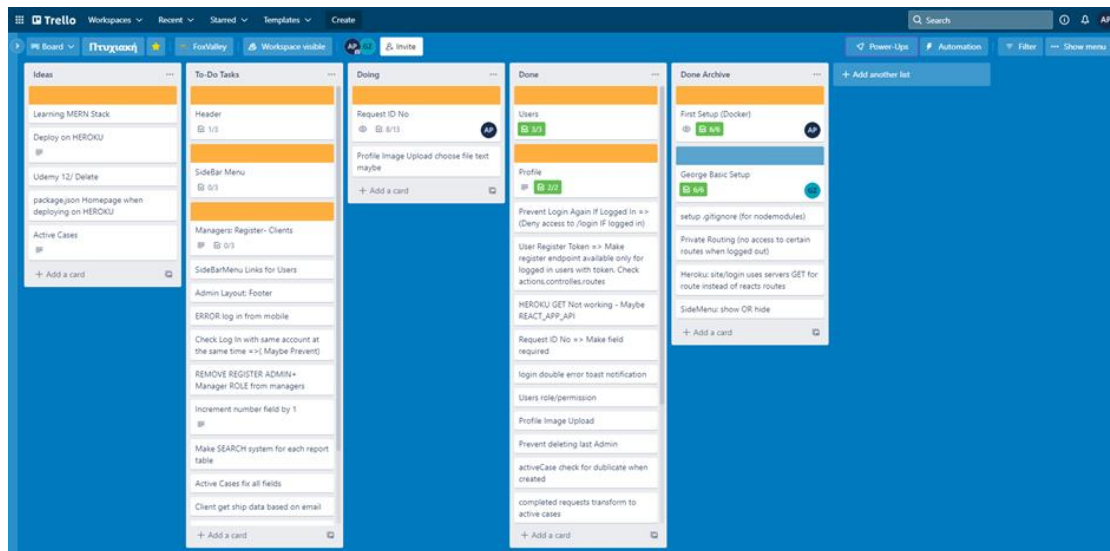
2.2.1 Task Management - Trello

Η διαχείριση εργασιών είναι η διαδικασία διαχείρισης μιας εργασίας κατά τη διάρκεια του κύκλου ζωής της. Περιλαμβάνει σχεδιασμό, δοκιμή, παρακολούθηση και αναφορά. Η διαχείριση εργασιών μπορεί να βοηθήσει είτε μεμονωμένα άτομα να πετύχουν στόχους είτε ομάδες ατόμων να συνεργαστούν και να μοιραστούν τη γνώση για την επίτευξη συλλογικών στόχων. Οι εργασίες διαφοροποιούνται επίσης ανάλογα με την πολυπλοκότητα, από χαμηλή σε υψηλή.

Η αποτελεσματική διαχείριση εργασιών απαιτεί τη διαχείριση όλων των πτυχών μιας εργασίας, συμπεριλαμβανομένης της κατάστασης, της προτεραιότητας, του χρόνου, των αναθέσεων ανθρώπινων και οικονομικών πόρων, της επανάληψης, της εξάρτησης, των ειδοποιήσεων και ούτω καθεξής. Αυτά μπορούν να συγκεντρωθούν σε γενικές γραμμές στις βασικές δραστηριότητες της διαχείρισης εργασιών.

Με γνώμονα τα παραπάνω, προχωρήσαμε στην χρήση του **Trello** ως εργαλείο γενικής διαχείρισης των εργασιών και καθηκόντων μας. Οι χρήστες του συγκεκριμένου εργαλείου μπορούν να δημιουργήσουν τους πίνακες εργασιών τους με διαφορετικές στήλες και να μετακινήσουν τις εργασίες μεταξύ τους. Συνήθως οι στήλες περιλαμβάνουν καταστάσεις εργασιών όπως: Εκκρεμότητα, Σε εξέλιξη, Έγινε.

Το εργαλείο μπορεί να χρησιμοποιηθεί για προσωπικούς και επαγγελματικούς σκοπούς, συμπεριλαμβανομένης της διαχείρισης ακινήτων, της διαχείρισης έργων λογισμικού, των σχολικών πινάκων ανακοινώσεων, του προγραμματισμού μαθημάτων, της λογιστικής, του σχεδιασμού ιστοσελίδων, των τυχερών παιχνιδιών και της διαχείρισης υποθέσεων δικηγορικού γραφείου.



Εικόνα 4 Trello

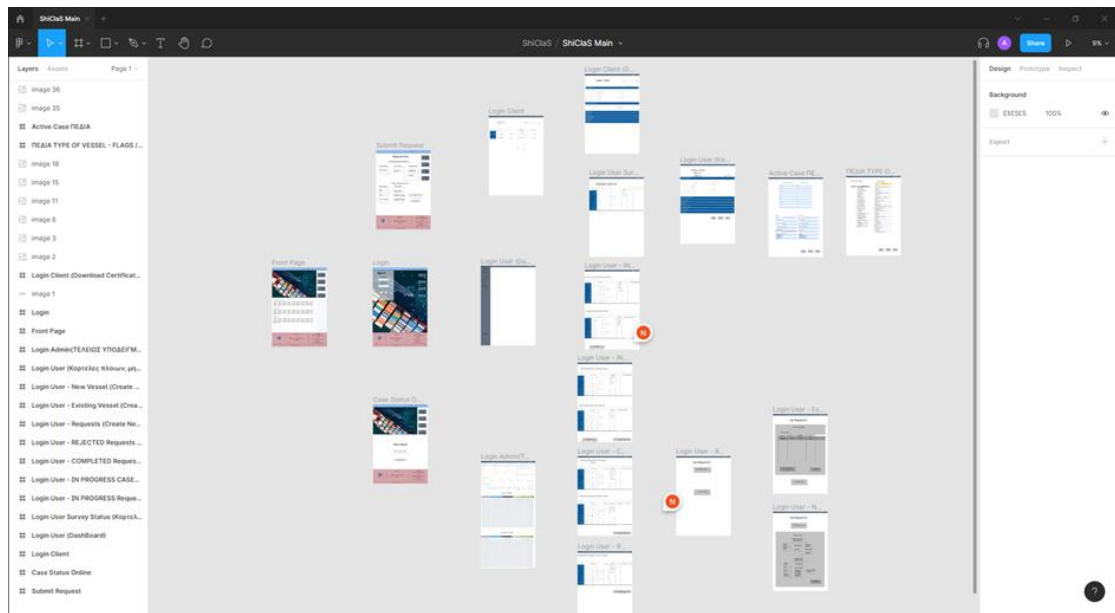
Στην παραπάνω εικόνα θα δείτε τον τρόπο που σημειώναμε εργασίες και προβλήματα στο αρχικό στάδιο της εφαρμογής ώστε να ξέρουμε ανα πάσα στιγμή τι έχει υλοποιηθεί και τι χρήζει προτεραιότητας.

2.2.2 UI/UX Design - Figma

Το Figma είναι ένα εργαλείο επεξεργασίας διανυσματικών γραφικών και πρωτοτύπων. Χρησιμοποιείται για τον σχεδιασμό του περιβάλλοντος χρήστη και της εμπειρίας του χρήστη, βοηθώντας έτσι τα υπόλοιπα μέλη της ομάδας να προβούν στην γρηγορότερη και σωστότερη ανάπτυξη της εφαρμογής.

Η σχεδίαση μιας εφαρμογής πριν την υλοποίηση της προσφέρει καλύτερο και αποδοτικότερο τρόπο επικοινωνίας ανάμεσα στα μέλη της ομάδας και ένα συνολικά πιο ολοκληρωμένο αποτέλεσμα πάντα με γνώμονα τις ανάγκες του χρήστη.

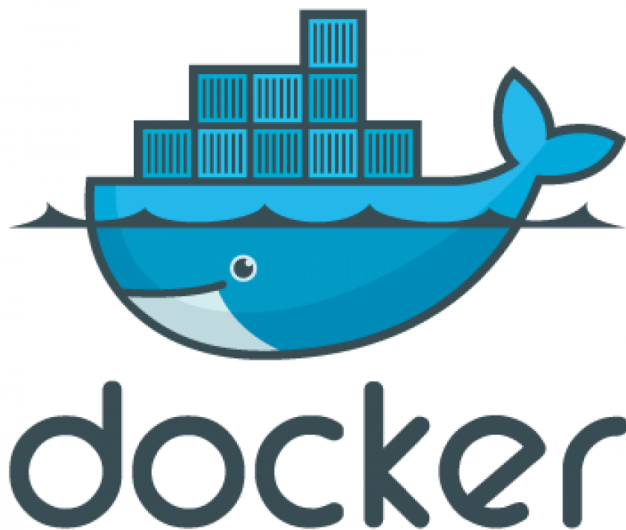
Το σύνολο χαρακτηριστικών του Figma εστιάζει στη χρήση, στη διεπαφή χρήστη και στο σχεδιασμό εμπειρίας χρήστη, με έμφαση στη συνεργασία σε πραγματικό χρόνο. Έτσι και εμείς, χρησιμοποιώντας το συγκεκριμένο εργαλείο, είχαμε μια ταυτόχρονη και σε πραγματικό χρόνο εικόνα για το πως θέλουμε να φαίνεται η εφαρμογή μας, το οποίο μας βοήθησε αφάνταστα κατά την διάρκεια της υλοποίησης καθώς εστιάζαμε στο πως θα πραγματοποιήσουμε ένα χαρακτηριστικό καλύτερο και αποδοτικότερο και όχι στο ποίο θα είναι αυτό μιας και το είχαμε ήδη κανονίσει και προσχεδιάσει.



Εικόνα 5 Figma

2.2.3 Virtualization - Docker

Το Docker είναι ένα σύνολο προϊόντων πλατφόρμας ως υπηρεσίας (PaaS) που χρησιμοποιούν



Εικόνα 6 Docker

εικονικοποίηση σε επίπεδο λειτουργικού συστήματος για την παράδοση λογισμικού σε πακέτα που ονομάζονται κοντέινερ. Η υπηρεσία έχει τόσο δωρεάν όσο και premium επίπεδα. Το λογισμικό που φιλοξενεί τα κοντέινερ ονομάζεται Docker Engine. Ξεκίνησε για πρώτη φορά το 2013 και αναπτύχθηκε από την Docker, Inc.

Το Docker μπορεί να συσκευάσει μια εφαρμογή και τις εξαρτήσεις της σε ένα

εικονικό κοντέινερ που μπορεί να εκτελεστεί σε οποιοδήποτε υπολογιστή Linux, Windows ή macOS. Αυτό δίνει τη δυνατότητα στην εφαρμογή να εκτελείται σε διάφορες τοποθεσίες, όπως εντός της εγκατάστασης, σε δημόσιο ή ιδιωτικό cloud.

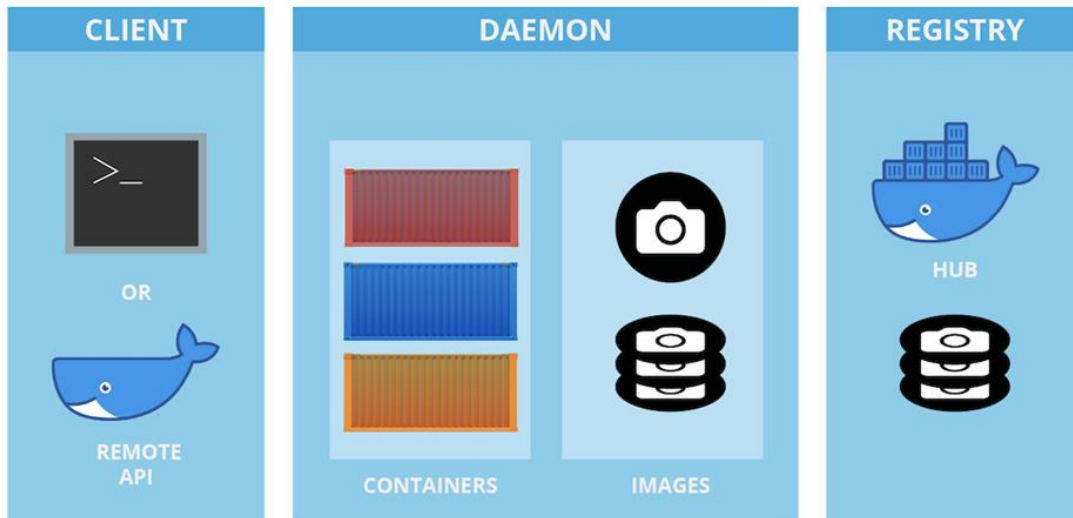
Όταν εκτελείται σε Linux, το Docker χρησιμοποιεί τις δυνατότητες απομόνωσης πόρων του πυρήνα Linux (όπως cgroups και χώρους ονομάτων πυρήνα) και ένα σύστημα αρχείων με δυνατότητα ένωσης (όπως το OverlayFS) για να επιτρέψει στα κοντέινερ να εκτελούνται μέσα σε μία μόνο παρουσία Linux, αποφεύγοντας τα γενικά έξοδα εκκίνησης και συντήρησης εικονικών μηχανών. Το Docker στο macOS χρησιμοποιεί μια εικονική μηχανή Linux για την εκτέλεση των κοντέινερ.

Επειδή τα κοντέινερ Docker είναι ελαφριά, ένας διακομιστής ή εικονική μηχανή μπορεί να εκτελεί πολλά κοντέινερ ταυτόχρονα. Μια ανάλυση του 2018 διαπίστωσε ότι μια τυπική περίπτωση χρήσης Docker περιλαμβάνει τη λειτουργία οκτώ κοντέινερ ανά κεντρικό υπολογιστή και ότι το ένα τέταρτο των οργανισμών που αναλύθηκαν εκτελούν 18 ή περισσότερα ανά κεντρικό υπολογιστή.

Η υποστήριξη του πυρήνα του Linux για χώρους ονομάτων ως επί το πλείστον απομονώνει την άποψη μιας εφαρμογής για το λειτουργικό περιβάλλον, συμπεριλαμβανομένων των δέντρων διεργασιών, του δικτύου, των αναγνωριστικών χρηστών και των προσαρτημένων συστημάτων αρχείων, ενώ οι ομάδες c του πυρήνα παρέχουν περιορισμό πόρων για τη μνήμη και την CPU. Από την έκδοση 0.9, το Docker περιλαμβάνει το δικό του στοιχείο (που ονομάζεται "libcontainer") για τη χρήση διευκολύνσεων εικονικοποίησης που παρέχονται απευθείας από τον πυρήνα του Linux, επιπλέον της χρήσης αφηρημένης διεπαφής εικονικοποίησης μέσω libvirt, LXC και systemd-nspawn.

Το Docker εφαρμόζει ένα API υψηλού επιπέδου για να παρέχει ελαφριά κοντέινερ που εκτελούν διαδικασίες μεμονωμένα. Τα κοντέινερ Docker είναι τυπικές διεργασίες, επομένως είναι δυνατή η χρήση χαρακτηριστικών του πυρήνα για την παρακολούθηση της εκτέλεσής τους -- συμπεριλαμβανομένης για παράδειγμα της χρήσης εργαλείων όπως το strace για την παρατήρηση και τη διαμεσολάβηση στις κλήσεις συστήματος.

Docker Architecture



Εικόνα 7 Docker Architecture

Το λογισμικό Docker ως προσφορά υπηρεσίας αποτελείται από τρία στοιχεία:

Λογισμικό: Ο δαίμονας Docker, που ονομάζεται `dockerd`, είναι μια επίμονη διαδικασία που διαχειρίζεται δοχεία Docker και χειρίζεται αντικείμενα κοντέινερ. Ο δαίμονας ακούει για αιτήματα που αποστέλλονται μέσω του Docker Engine API. Το πρόγραμμα πελάτη Docker, που ονομάζεται `docker`, παρέχει μια διεπαφή γραμμής εντολών (CLI), που επιτρέπει στους χρήστες να αλληλεπιδρούν με τους δαίμονες του Docker.

Αντικείμενα: Τα αντικείμενα Docker είναι διάφορες οντότητες που χρησιμοποιούνται για τη συναρμολόγηση μιας εφαρμογής στο Docker. Οι κύριες κατηγορίες αντικειμένων Docker είναι εικόνες, κοντέινερ και υπηρεσίες.

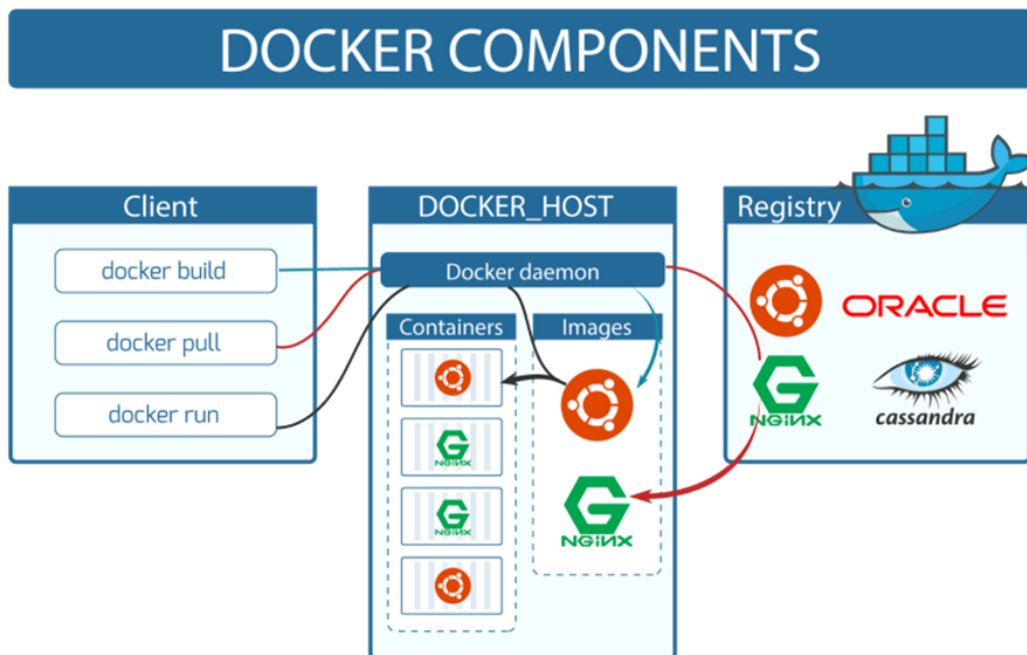
Ένα κοντέινερ Docker είναι ένα τυποποιημένο, ενθυλακωμένο περιβάλλον που εκτελεί εφαρμογές. Η διαχείριση ενός κοντέινερ γίνεται με χρήση του Docker API ή CLI.

Μια εικόνα Docker είναι ένα πρότυπο μόνο για ανάγνωση που χρησιμοποιείται για τη δημιουργία κοντέινερ. Οι εικόνες χρησιμοποιούνται για την αποθήκευση και αποστολή εφαρμογών.

Μια υπηρεσία Docker επιτρέπει στα κοντέινερ να κλιμακώνονται σε πολλούς δαίμονες Docker. Το αποτέλεσμα είναι γνωστό ως σμήνος, ένα σύνολο συνεργαζόμενων δαιμόνων που επικοινωνούν μέσω του Docker API.

Μητρώα: Το μητρώο Docker είναι ένα αποθετήριο για εικόνες Docker. Τα προγράμματα-πελάτες Docker συνδέονται σε μητρώα για λήψη ("έλξη") εικόνων για χρήση ή για αποστολή ("push") εικόνων που έχουν δημιουργήσει. Τα μητρώα μπορεί να είναι δημόσια ή ιδιωτικά. Δύο κύρια δημόσια μητρώα είναι το Docker Hub και το Docker Cloud. Το Docker Hub είναι το προεπιλεγμένο μητρώο όπου το Docker

αναζητά εικόνες. Τα μητρώα Docker επιτρέπουν επίσης τη δημιουργία ειδοποιήσεων με βάση συμβάντα.

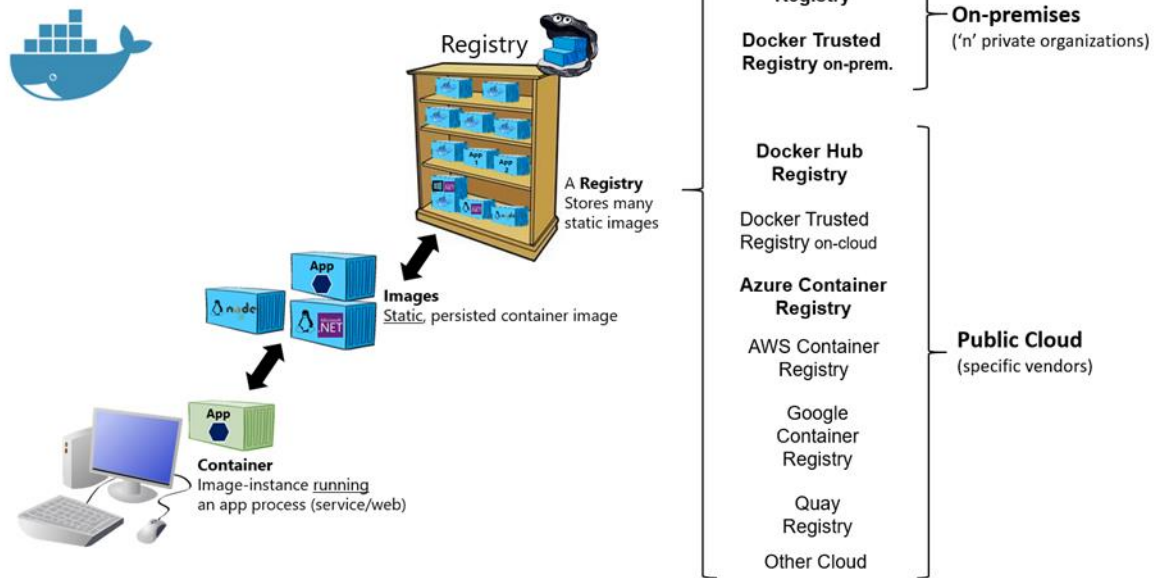


Εικόνα 8 Docker Components

Εργαλεία

Το Docker Compose είναι ένα εργαλείο για τον καθορισμό και την εκτέλεση εφαρμογών Docker πολλαπλών κοντέινερ. Χρησιμοποιεί αρχεία YAML για τη διαμόρφωση των υπηρεσιών της εφαρμογής και εκτελεί τη διαδικασία δημιουργίας και εκκίνησης όλων των κοντέινερ με μία μόνο εντολή. Το βοηθητικό πρόγραμμα docker-compose CLI επιτρέπει στους χρήστες να εκτελούν εντολές σε πολλά κοντέινερ ταυτόχρονα, για παράδειγμα, δημιουργία εικόνων, κλιμάκωση κοντέινερ, εκτέλεση κοντέινερ που έχουν σταματήσει και πολλά άλλα. Εντολές που σχετίζονται με χειρισμό εικόνας ή επιλογές αλληλεπίδρασης με το χρήστη, δεν είναι σχετικές στο Docker Compose επειδή απευθύνονται σε ένα κοντέινερ. Το αρχείο docker-compose.yml χρησιμοποιείται για τον καθορισμό των υπηρεσιών μιας εφαρμογής και περιλαμβάνει διάφορες επιλογές διαμόρφωσης. Για παράδειγμα, η επιλογή δημιουργίας ορίζει επιλογές διαμόρφωσης όπως η διαδρομή Dockerfile, η επιλογή εντολής επιτρέπει σε κάποιον να παρακάμψει τις προεπιλεγμένες εντολές Docker και πολλά άλλα. Η πρώτη δημόσια έκδοση beta του Docker Compose (έκδοση 0.0.1) κυκλοφόρησε στις 21 Δεκεμβρίου 2013. Η πρώτη έκδοση έτοιμη για παραγωγή (1.0) έγινε διαθέσιμη στις 16 Οκτωβρίου 2014.

Basic taxonomy in Docker



Εικόνα 9 Docker Basic Taxonomy

Το Docker Swarm παρέχει λειτουργία εγγενούς ομαδοποίησης για κοντέινερ Docker, η οποία μετατρέπει μια ομάδα μηχανών Docker σε μια ενιαία εικονική μηχανή Docker. Στο Docker 1.12 και νεότερη έκδοση, η λειτουργία Swarm είναι ενσωματωμένη στο Docker Engine. Το βοηθητικό πρόγραμμα `docker swarm CLI` επιτρέπει στους χρήστες να εκτελούν κοντέινερ Swarm, να δημιουργούν διακριτικά εντοπισμού, να καταγράφουν κόμβους στο σύμπλεγμα και πολλά άλλα. Το βοηθητικό πρόγραμμα `docker node CLI` επιτρέπει στους χρήστες να εκτελούν διάφορες εντολές για τη διαχείριση κόμβων σε ένα σμήνος, για παράδειγμα, καταχώριση των κόμβων σε ένα σμήνος, ενημέρωση κόμβων και κατάργηση κόμβων από το σμήνος. Ο Docker διαχειρίζεται τα σμήνη χρησιμοποιώντας τον αλγόριθμο συναίνεσης Raft. Σύμφωνα με τον Raft, για να εκτελεστεί μια ενημέρωση, η πλειονότητα των κόμβων Swarm πρέπει να συμφωνήσουν σχετικά με την ενημέρωση.

Το Docker Volume διευκολύνει την ανεξάρτητη διατήρηση των δεδομένων, επιτρέποντας στα δεδομένα να παραμένουν ακόμα και μετά τη διαγραφή ή την εκ νέου δημιουργία του κοντέινερ.

2.2.4 Versioning- Git + Github



Εικόνα 10 Git

Το Git (/git/) είναι ένα σύστημα ελέγχου εκδόσεων (λέγεται και σύστημα ελέγχου αναθεωρήσεων ή σύστημα ελέγχου πηγαίου κώδικα) με έμφαση στην ταχύτητα, στην ακεραιότητα των δεδομένων και στην υποστήριξη για καταναμημένες μη γραμμικές ροές εργασίας. Το Git σχεδιάστηκε και αναπτύχθηκε αρχικά από τον Λίνους Τόρβαλντς για τη ανάπτυξη του πυρήνα Linux το 2005 και έχει γίνει από τότε το πιο διαδεδομένο σύστημα ελέγχου εκδόσεων για ανάπτυξη λογισμικού.

Όπως τα περισσότερα άλλα καταναμημένα συστήματα ελέγχου εκδόσεων/αναθεωρήσεων και αντίθετα με τα περισσότερα συστήματα πελάτη-διακομιστή, κάθε κατάλογος εργασίας του Git είναι ένα ολοκληρωμένο αποθετήριο λογισμικού με πλήρες ιστορικό και δυνατότητες πλήρους παρακολούθησης της έκδοσης, ανεξάρτητα από την πρόσβαση δικτύου ή ενός κεντρικού διακομιστή. Όπως ο πυρήνας Linux, το Git είναι Ελεύθερο λογισμικό που διανέμεται κάτω από τους όρους της έκδοσης 2 της Γενικής Άδειας Δημόσιας Χρήσης GNU.

Η σχεδίαση του Git εμπνεύστηκε από τα BitKeeper και Monotone. Το Git σχεδιάστηκε αρχικά ως μια μηχανή συστήματος ελέγχου σε έκδοση χαμηλού επιπέδου πάνω στην οποία άλλοι μπορούν να γράψουν προγράμματα προσκηνίου, όπως Cogito ή StGIT. Το έργο του πυρήνα του Git έχει γίνει από τότε μια πλήρης έκδοση συστήματος ελέγχου άμεσα χρησιμοποιήσιμη. Αν και επηρεάστηκε έντονα από το BitKeeper, ο Τόρβαλντς προσπάθησε εκούσια να αποφύγει συμβατικές προσεγγίσεις, που οδηγούν σε έναν μοναδικό σχεδιασμό.



Εικόνα 11 Git for Professionals

Η ανάπτυξη του Git ξεκίνησε αφού πολλοί προγραμματιστές του πυρήνα Linux σταμάτησαν την πρόσβαση στο BitKeeper, ένα ιδιοταγές (δηλαδή μη ανοικτού κώδικα) σύστημα ελέγχου εκδόσεων που χρησιμοποιούνταν προηγουμένως για τη διατήρηση του έργου. Ο κάτοχος των πνευματικών δικαιωμάτων του BitKeeper, Larry McVoy, είχε αποσύρει την ελεύθερη χρήση του προϊόντος αφού ισχυρίστηκε ότι ο Andrew Tridgell είχε εφαρμόσει αντίστροφη μηχανική στα πρωτόκολλα του BitKeeper.

Ο Τόρβαλντς ήθελε ένα κατανεμημένο σύστημα που θα μπορούσε να χρησιμοποιηθεί σαν το BitKeeper, αλλά κανένα από τα διαθέσιμα ελεύθερα συστήματα δεν κάλυπτε τις ανάγκες του, ιδιαίτερα στην απόδοση. Ο Τόρβαλντς πήρε ένα παράδειγμα ενός συστήματος SCM που χρειαζόταν 30 δευτερόλεπτα για να εφαρμόσει μια διόρθωση και να ενημερώσει όλα τα συνδεδεμένα μεταδεδομένα και σημείωσε ότι αυτό δεν μπορούσε να ανταποκριθεί στις ανάγκες της ανάπτυξης του πυρήνα Linux, όπου ο συγχρονισμός με τους συνεργάτες του μπορούσε να απαιτήσει 250 τέτοιες ενέργειες τη φορά. Ήθελε η διόρθωση να κρατήσει 3 δευτερόλεπτα, και είχε αρκετά άλλα κριτήρια σχεδίασης κατά νου:

- να πάρει ταυτόχρονο σύστημα εκδόσεων (CVS) ως ένα παράδειγμα του τι να «μην» κάνει· αν υπήρχε αμφιβολία, να γίνει η ακριβώς αντίθετη απόφαση
- να υποστηρίξει μια διανεμόμενη, παρόμοια με το BitKeeper ροή εργασίας
- να έχει πολύ ισχυρές εγγυήσεις ενάντια σε αλλοίωση, είτε τυχαία είτε κακόβουλη.

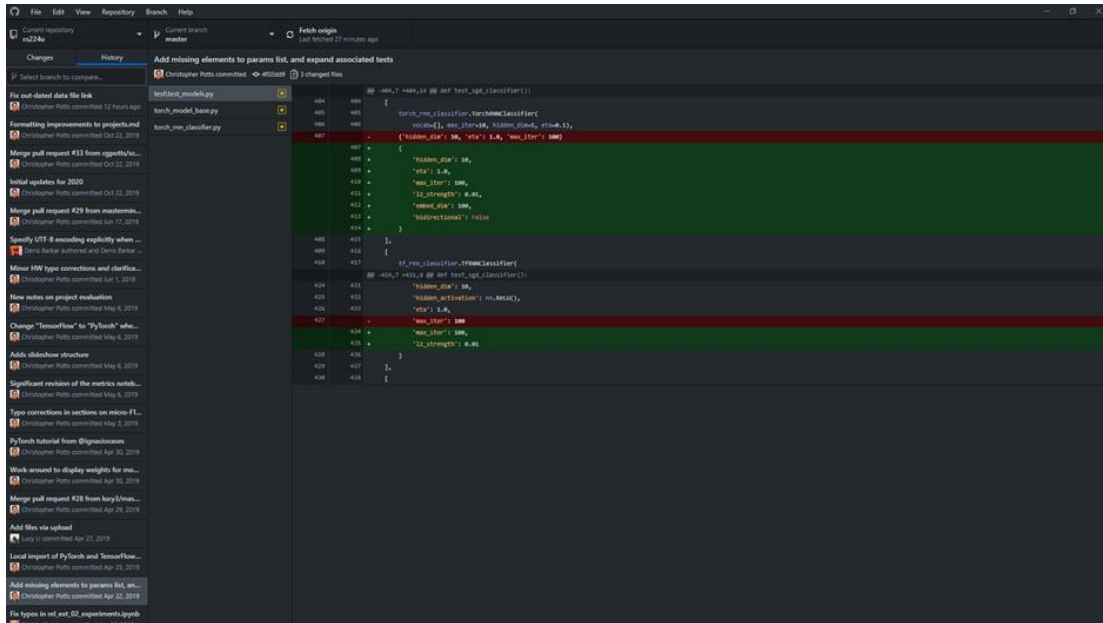
Αυτά τα τρία κριτήρια απάλειψαν κάθε σύστημα ελέγχου εκδόσεων που υπήρχε τότε, εκτός από το Monotone. Εξετάζοντας την απόδοσή του, το απέρριψε και αυτό. Έτσι αμέσως μετά την έκδοση του πυρήνα Linux 2.6.12-rc2, ο Τόρβαλντς άρχισε να γράφει ένα δικό του.

Ο Τόρβαλντς δικαιολόγησε το όνομα git, που στην βρετανική αργκό σημαίνει κάτι σαν «ανεπιθύμητος», λέγοντας: «Είμαι ένα φαντασμένο άτομο και ονομάζω όλα τα έργα μου σύμφωνα με τον εαυτό μου. Πρώτα ήταν το Λίνουξ, τώρα το git». Άλλωστε και η σελίδα του εγχειριδίου (man page) περιγράφει το git ως «τον ηλίθιο εντοπιστή περιεχομένου».

GitHub

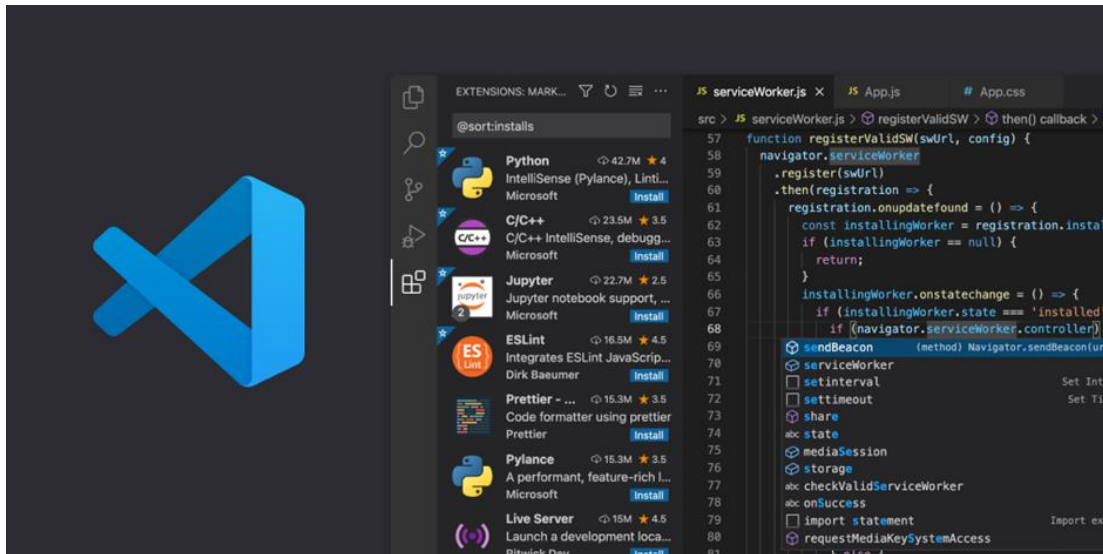
Η GitHub, Inc. είναι πάροχος φιλοξενίας Διαδικτύου για ανάπτυξη λογισμικού και έλεγχο έκδοσης με χρήση του Git. Προσφέρει τη λειτουργία καταμεμημένου ελέγχου έκδοσης και διαχείρισης πηγαίου κώδικα (SCM) του Git, καθώς και τις δικές του δυνατότητες. Παρέχει έλεγχο πρόσβασης και πολλές δυνατότητες συνεργασίας, όπως παρακολούθηση σφαλμάτων, αιτήματα δυνατοτήτων, διαχείριση εργασιών, συνεχή ενοποίηση και wiki για κάθε έργο. Με έδρα την Καλιφόρνια, είναι θυγατρική της Microsoft από το 2018.

Χρησιμοποιείται συνήθως για τη φιλοξενία έργων ανοιχτού κώδικα. Από τον Νοέμβριο του 2021, το GitHub αναφέρει ότι είχε πάνω από 73 εκατομμύρια προγραμματιστές και περισσότερα από 200 εκατομμύρια αποθετήρια (συμπεριλαμβανομένων τουλάχιστον 28 εκατομμυρίων δημόσιων αποθετηρίων). Είναι ο μεγαλύτερος κεντρικός υπολογιστής πηγαίου κώδικα από τον Νοέμβριο του 2021.



Εικόνα 12 Git Interface

2.2.5 Code Editor/IDE – Visual Studio Code



Εικόνα 13 Visual Studio Code

Το Visual Studio Code είναι ένα πρόγραμμα επεξεργασίας πηγαίου κώδικα που δημιουργήθηκε από τη Microsoft για Windows, Linux και macOS. Οι δυνατότητες περιλαμβάνουν υποστήριξη για εντοπισμό σφαλμάτων, επισήμανση σύνταξης, έξυπνη συμπλήρωση κώδικα, αποσπάσματα, ανακατασκευή κώδικα και ενσωματωμένο Git. Οι χρήστες μπορούν να αλλάξουν το θέμα, τις συντομεύσεις πληκτρολογίου, τις προτιμήσεις και να εγκαταστήσουν επεκτάσεις που προσθέτουν επιπλέον λειτουργίες.

Το Visual Studio Code ανακοινώθηκε για πρώτη φορά στις 29 Απριλίου 2015, από τη Microsoft στο συνέδριο Build 2015. Μια έκδοση προεπισκόπησης κυκλοφόρησε λίγο αργότερα.

Στις 18 Νοεμβρίου 2015, η πηγή του Visual Studio Code κυκλοφόρησε υπό την άδεια MIT και έγινε διαθέσιμη στο GitHub. Ανακοινώθηκε επίσης η υποστήριξη επέκτασης. Στις 14 Απριλίου 2016, το Visual Studio Code αποφοίτησε από το στάδιο δημόσιας προεπισκόπησης και κυκλοφόρησε στον Ιστό. Η Microsoft έχει κυκλοφορήσει το μεγαλύτερο μέρος του πηγαίου κώδικα του Visual Studio Code στο GitHub υπό την επιτρεπτή άδεια MIT, ενώ οι εκδόσεις από τη Microsoft είναι ιδιόκτητο δωρεάν λογισμικό.

Στο Stack Overflow 2021 Developer Survey, το Visual Studio Code κατατάχθηκε ως το πιο δημοφιλές εργαλείο περιβάλλοντος προγραμματιστών, με το 70% των 82.000 ερωτηθέντων να αναφέρει ότι το χρησιμοποιεί.

Το Visual Studio Code είναι ένα πρόγραμμα επεξεργασίας πηγαίου κώδικα που μπορεί να χρησιμοποιηθεί με μια ποικιλία γλωσσών προγραμματισμού, όπως Java, JavaScript, Go, Node.js, Python και C++. Βασίζεται στο πλαίσιο Electron, που χρησιμοποιείται για την ανάπτυξη εφαρμογών Web Node.js που εκτελούνται στη μηχανή διάταξης Blink. Το Visual Studio Code χρησιμοποιεί το ίδιο στοιχείο επεξεργασίας (με την κωδική ονομασία "Monaco") που χρησιμοποιείται στο Azure DevOps (παλαιότερα ονομαζόταν Visual Studio Online και Visual Studio Team Services).

Αντί για σύστημα έργου, επιτρέπει στους χρήστες να ανοίγουν έναν ή περισσότερους καταλόγους, οι οποίοι στη συνέχεια μπορούν να αποθηκευτούν σε χώρους εργασίας για μελλοντική επαναχρησιμοποίηση. Αυτό του επιτρέπει να λειτουργεί ως γλωσσοαγνωστικός επεξεργαστής κώδικα για οποιαδήποτε γλώσσα. Υποστηρίζει έναν αριθμό γλωσσών προγραμματισμού και ένα σύνολο χαρακτηριστικών που διαφέρουν ανά γλώσσα. Τα ανεπιθύμητα αρχεία και φάκελοι μπορούν να εξαιρεθούν από το δέντρο του έργου μέσω των ρυθμίσεων. Πολλές λειτουργίες του κώδικα του Visual Studio δεν εκτίθενται μέσω των μενού ή της διεπαφής χρήστη, αλλά μπορούν να προσπελαστούν μέσω της παλέτας εντολών.

Ο κώδικας του Visual Studio μπορεί να επεκταθεί μέσω επεκτάσεων, που διατίθενται μέσω ενός κεντρικού αποθετηρίου. Αυτό περιλαμβάνει προσθήκες στο πρόγραμμα επεξεργασίας και υποστήριξη γλώσσας. Ένα αξιοσημείωτο χαρακτηριστικό είναι η δυνατότητα δημιουργίας επεκτάσεων που προσθέτουν υποστήριξη για νέες γλώσσες, θέματα και προγράμματα εντοπισμού σφαλμάτων, εκτελούν στατική ανάλυση κώδικα και προσθέτουν γραμμές κώδικα χρησιμοποιώντας το Πρωτόκολλο Διακομιστή Γλωσσών.

Το Visual Studio Code περιλαμβάνει πολλαπλές επεκτάσεις για FTP, επιτρέποντας στο λογισμικό να χρησιμοποιηθεί ως δωρεάν εναλλακτική για την ανάπτυξη Ιστού. Ο κώδικας μπορεί να συγχρονιστεί μεταξύ του επεξεργαστή και του διακομιστή, χωρίς λήψη επιπλέον λογισμικού.

Το Visual Studio Code επιτρέπει στους χρήστες να ορίσουν την κωδικοσελίδα στην οποία αποθηκεύεται το ενεργό έγγραφο, τον χαρακτήρα νέας γραμμής και τη γλώσσα προγραμματισμού του ενεργού εγγράφου. Αυτό επιτρέπει τη χρήση του σε

οποιαδήποτε πλατφόρμα, σε οποιαδήποτε τοπική ρύθμιση και για οποιαδήποτε δεδομένη γλώσσα προγραμματισμού.

2.2.6 Package Manager – NPM



Εικόνα 14 NPM

Το npm είναι ένας διαχειριστής πακέτων για τη γλώσσα προγραμματισμού JavaScript που διατηρείται από την npm, Inc.

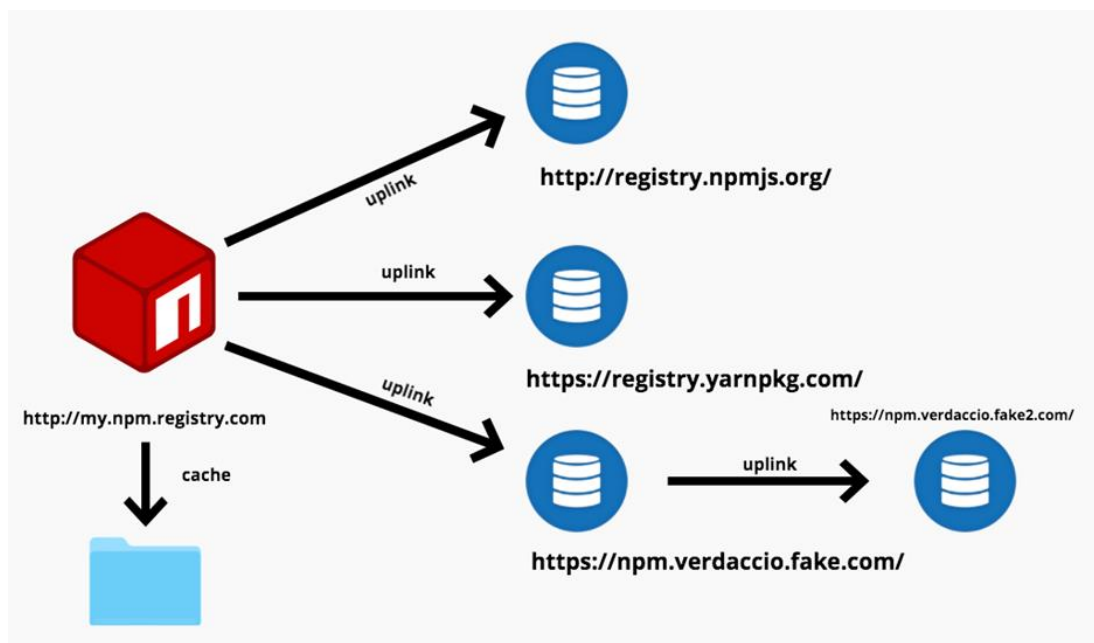
Το npm είναι ο προεπιλεγμένος διαχειριστής πακέτων για το περιβάλλον χρόνου εκτέλεσης JavaScript Node.js. Αποτελείται από έναν πελάτη γραμμής εντολών, που ονομάζεται επίσης npm, και μια ηλεκτρονική βάση δεδομένων δημόσιων και ιδιωτικών πακέτων επί πληρωμή, που ονομάζεται μητρώο npm. Η πρόσβαση στο μητρώο γίνεται μέσω του πελάτη και τα διαθέσιμα πακέτα μπορούν να περιηγηθούν και να αναζητηθούν μέσω του ιστότοπου npm. Ο διαχειριστής πακέτων και το μητρώο διαχειρίζονται από την npm, Inc.

Το npm περιλαμβάνεται ως προτεινόμενη δυνατότητα στο πρόγραμμα εγκατάστασης Node.js. Το npm αποτελείται από έναν πελάτη γραμμής εντολών που αλληλεπιδρά με ένα απομακρυσμένο μητρώο. Επιτρέπει στους χρήστες να καταναλώνουν και να διανέμουν λειτουργικές μονάδες JavaScript που είναι διαθέσιμες στο μητρώο. Τα πακέτα στο μητρώο είναι σε μορφή CommonJS και περιλαμβάνουν ένα αρχείο μεταδεδομένων σε μορφή JSON. Πάνω από 1,3 εκατομμύρια πακέτα είναι διαθέσιμα στο κύριο μητρώο npm. Το μητρώο δεν διαθέτει διαδικασία ελέγχου για υποβολή, πράγμα που σημαίνει ότι τα πακέτα που βρίσκονται εκεί μπορεί να είναι δυνητικά χαμηλής ποιότητας, ανασφαλή ή κακόβουλα. Αντίθετα, το npm βασίζεται σε αναφορές χρηστών για την κατάργηση πακέτων σε περίπτωση που παραβιάζουν πολιτικές επειδή είναι χαμηλής ποιότητας, ανασφαλής ή κακόβουλα. Το npm εκθέτει στατιστικά στοιχεία, συμπεριλαμβανομένου του αριθμού των λήψεων και του αριθμού των εξαρτώμενων πακέτων για να βοηθήσει τους προγραμματιστές να κρίνουν την ποιότητα των πακέτων.

Στην έκδοση npm 6, η δυνατότητα ελέγχου εισήχθη για να βοηθήσει τους προγραμματιστές να εντοπίσουν και να διορθώσουν ευπάθειες ασφαλείας στα

εγκατεστημένα πακέτα. Η πηγή των τρωτών σημείων ασφαλείας ελήφθη από αναφορές που βρέθηκαν στην Πλατφόρμα Ασφάλειας Κόμβων (NSP) και έχει ενσωματωθεί στο npm από την εξαγορά του NSP από την npm.

Το npm μπορεί να διαχειρίζεται πακέτα που είναι τοπικές εξαρτήσεις ενός συγκεκριμένου έργου, καθώς και καθολικά εγκατεστημένα εργαλεία JavaScript. Όταν χρησιμοποιείται ως διαχειριστής εξαρτήσεων για ένα τοπικό έργο, το npm μπορεί να εγκαταστήσει, σε μία εντολή, όλες τις εξαρτήσεις ενός έργου μέσω του αρχείου `package.json`. Στο αρχείο `package.json`, κάθε εξάρτηση μπορεί να καθορίσει μια σειρά από έγκυρες εκδόσεις χρησιμοποιώντας το σχήμα σημασιολογικής έκδοσης εκδόσεων, επιτρέποντας στους προγραμματιστές να ενημερώνουν αυτόματα τα πακέτα τους ενώ ταυτόχρονα αποφεύγουν ανεπιθύμητες αλλαγές. Το npm παρέχει επίσης εργαλεία έκδοσης `bumping` για προγραμματιστές για να επισημάνουν τα πακέτα τους με μια συγκεκριμένη έκδοση. Το npm παρέχει επίσης το αρχείο `package-lock.json` που έχει την καταχώρηση της ακριβούς έκδοσης που χρησιμοποιείται από το έργο μετά την αξιολόγηση της σημασιολογικής έκδοσης στο `package.json`.



Εικόνα 15 NPM Diagram

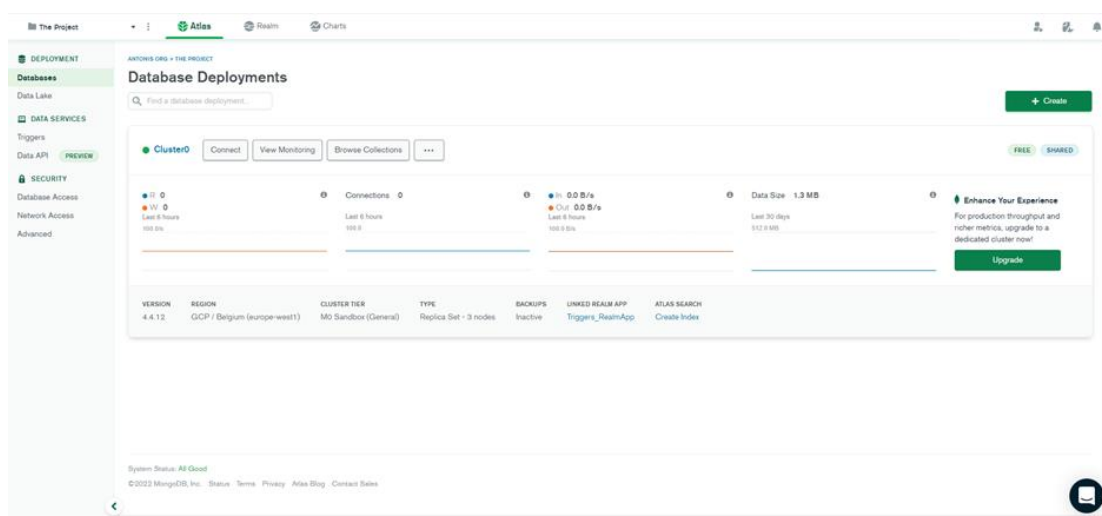
2.2.7 Database - MongoDB

Μία βάση δεδομένων, είναι ουσιαστικά μία συλλογή δεδομένων κατάλληλα αποθηκευμένων σε έναν υπολογιστή προκειμένου να μπορούν να προσπελαστούν, διαχειριστούν και ενημερωθούν με ευκολία από το άτομο που τις διαχειρίζεται. Οι ενέργειες αυτές πάνω στα δεδομένα γίνονται μέσα από τα Συστήματα Διαχείρισης Βάσεων Δεδομένων (DataBase Management Systems - DBMS), τα οποία έχουν κατάλληλους μηχανισμούς για να προσπελαίνουν οργανωμένες δομές δεδομένων και να τις επεξεργάζονται χωρίς απώλειες και κίνδυνο παραποίησης ή φθοράς των δεδομένων.

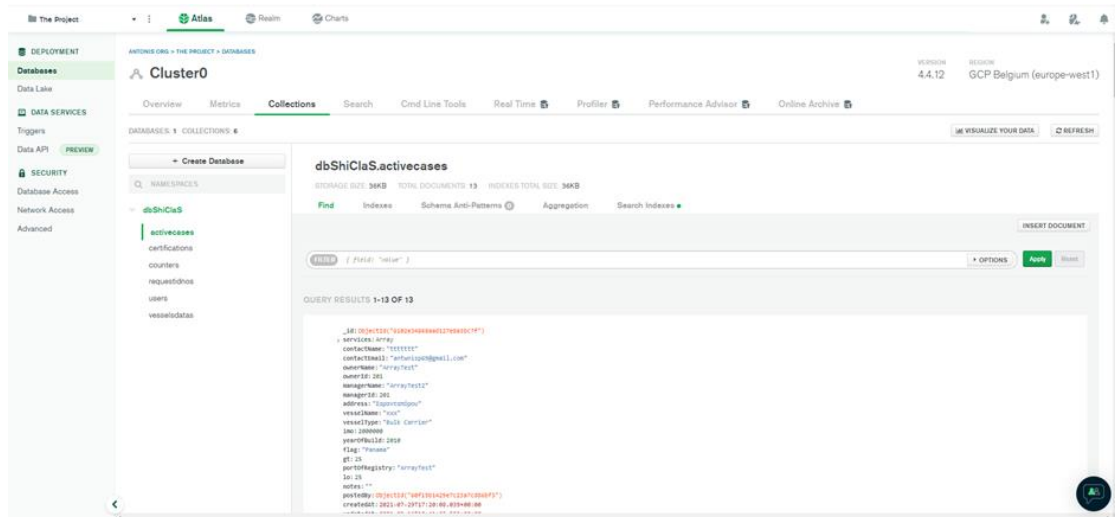
Η MongoDB είναι μία γλώσσα προγραμματισμού για Συστήματα Διαχείρισης Βάσεων Δεδομένων (DBMS). Προκειμένου τα DBMS να μπορούν να έχουν κάποιους κοινούς άξονες αναφοράς έχει αναπτυχθεί η γλώσσα SQL (Structured Query Language) ή οποία επιτρέπει μέσα από ένα σύστημα εντολών, την προσπέλαση των δομημένων δεδομένων που περιέχει ένα DBMS. Κάθε DBMS ουσιαστικά δέχεται και εκτελεί ένα σεντ εντολών SQL για να διαχειριστεί τα δεδομένα του. Η MySQL είναι μία διανομή (έκδοση) της γλώσσας SQL και ένα MySQL DBMS είναι ένα σύστημα που δέχεται εντολές της MySQL. Η MySQL αποτελεί ελεύθερο λογισμικό που μπορεί να χρησιμοποιήσει κανείς ερασιτεχνικά ή επαγγελματικά. Αντίστοιχα με την MySQL υπάρχουν και άλλα συστήματα δωρεάν ή με πληρωμή. Από τα πιο γνωστά DBMS είναι ο SQL Server της Microsoft, ο Oracle SQL server, η DB2 κ.ά. Επομένως μία βάση δεδομένων σε MySQL είναι ένας χώρος που έχει διαμορφωθεί κατάλληλα προκειμένου να αποθηκεύσουμε τα δεδομένα των ιστοσελίδων μας με σχεσιακό τρόπο.

Τι είναι το MongoDB Atlas;

Το MongoDB Atlas είναι μια υπηρεσία βάσης δεδομένων πολλαπλών νέφους από τους ίδιους ανθρώπους που δημιουργούν το MongoDB. Η Atlas απλοποιεί την ανάπτυξη και τη διαχείριση των βάσεων δεδομένων σας, ενώ προσφέρει την ευελιξία που χρειάζεται για τη δημιουργία ανθεκτικών και αποδοτικών παγκόσμιων εφαρμογών στους παρόχους cloud της επιλογής σας.



Εικόνα 16 MongoDB Atlas Interface



Εικόνα 17 MongoDB Collections

2.2.8 Server- NodeJS

Ο Server ή αλλιώς εξυπηρετητής ή διακομιστής, στην πιο απλή του μορφή είναι ένας ηλεκτρονικός υπολογιστής που τρέχει κατάλληλο λογισμικό ώστε να εξυπηρετεί τους χρήστες που συνδέονται με αυτόν για κάποιο σκοπό. Ανάλογα με τον σκοπό, ο server τρέχει τις κατάλληλες υπηρεσίες με την κατάλληλη ονομασία.

Συγκεκριμένα, αν ο server εξυπηρετεί ιστοσελίδες, ονομάζεται web server και τρέχει λογισμικό κατάλληλο για να εξυπηρετεί τα αρχεία με τα οποία είναι κατασκευασμένη μία ιστοσελίδα. Ενώ, αν ο server εξυπηρετεί e-mail λογαριασμούς ώστε οι χρήστες να έχουν το e-mail τους σε αυτόν, τότε θα είναι e-mail server και θα τρέχει κατάλληλο λογισμικό για αποστολή και λήψη e-mail μηνυμάτων.

Στην πιο απλή του μορφή ένας server είναι ένας ηλεκτρονικός υπολογιστής, ο οποίος τρέχει κατάλληλο λογισμικό. Στην πραγματικότητα, δεν είναι απλοί υπολογιστές, αλλά, ηλεκτρονικοί υπολογιστές αυξημένων δυνατοτήτων οι οποίοι είναι εγκατεστημένοι σε ειδικούς χώρους που ονομάζονται datacenters.

Το **Node.js** είναι μια πλατφόρμα ανάπτυξης λογισμικού (κυρίως διακομιστών) χτισμένη σε περιβάλλον **Javascript**. Στόχος του **Node** είναι να παρέχει ένα εύκολο τρόπο δημιουργίας κλιμακωτών διαδικτυακών εφαρμογών. Σε αντίθεση με τα περισσότερα σύγχρονα περιβάλλοντα ανάπτυξης εφαρμογών δικτύων μία διεργασία node δεν στηρίζεται στην πολυνηματικότητα αλλά σε ένα μοντέλο ασύγχρονης επικοινωνίας εισόδου/εξόδου.

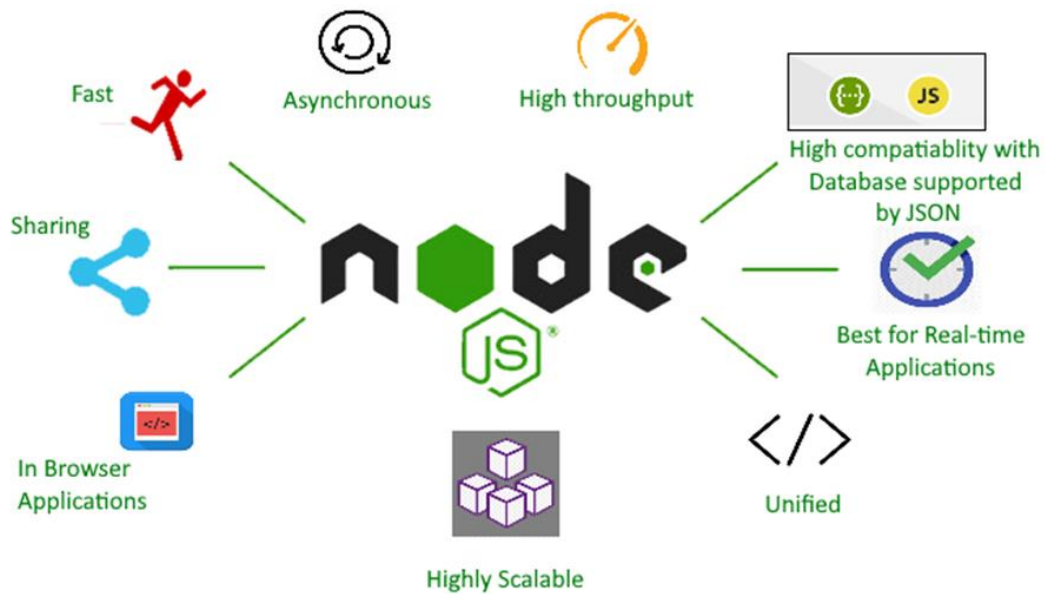
Το **Node** χαρακτηρίζεται από την έμφαση στην ασύγχρονη επικοινωνία μεταξύ των υπολογιστικών πόρων. Αυτό επιτυγχάνεται με την χρήση συμβάντων (events) που προσφέρει η **Javascript** και ονομάζονται **callbacks**. Για παράδειγμα όταν ένας περιηγητής ιστού φορτώσει πλήρως ένα αρχείο, ένας χρήστης πατάει κάποιο κουμπί, ολοκληρώνεται ένα αίτημα **AJAX**, τα συμβάντα αυτά πυροδοτούν ένα συγκεκριμένο callback. Αυτό με την σειρά του επιτρέπει την ροή του κώδικα χωρίς να αφήνει

ανενεργό τον επεξεργαστή προκειμένου να εκτελεστεί μια λειτουργία, όπως μια επιτυχής ανάγνωση αρχείου από τον δίσκο.

Η κοινότητα έχει δημιουργήσει ένα ολόκληρο οικοσύστημα από βιβλιοθήκες που προορίζονται ή είναι συμβατές με το **node**. Ανάμεσά τους εργαλεία που ξεχώρισαν όπως το **node-mysql**, το **Mongodb** και το **Express** παίζουν σημαντικό ρόλο υποστηρίζοντας την ασύγχρονη διάδραση με τις παραδοσιακές και **NoSQL** μεθόδους βάσεων δεδομένων. Αυτό επιτυγχάνεται με την χρήση του node package manager το οποίο επιτρέπει την εγκατάσταση των παραπάνω βιβλιοθηκών. Χρησιμοποιείται συνήθως σε εφαρμογές Chat, Proxy, Http Server καθώς και για παρακολούθηση εφαρμογών και του συστήματος (monitoring).



Εικόνα 18 NodeJS



Εικόνα 19 NodeJS Pros

2.2.9 UI Development

2.2.9.1 HTML5

Τα αρχικά της **HTML** προέρχονται από το αγγλικό ακρωνύμιο των λέξεων **HyperText Markup Language** (γλώσσα μορφοποίηση υπερκειμένου) και είναι η βασική γλώσσα δόμηση σελίδων του World Wide Web. Είναι μία γλώσσα σήμανσης όπου χρησιμοποιείται για να μορφοποιεί κατάλληλα ένα τμήμα κειμένου και να το κάνει να εμφανίζεται καλύτερα. Επιτρέπει την ενσωμάτωση ήχου και εικόνων στις web σελίδες. Αρχικά είχε κατασκευαστεί με σκοπό μόνο την μορφοποίηση κειμένου, αλλά μεγάλωσε και ενσωμάτωσε σχεδιαστικές τεχνικές κ.α. Η γλώσσα χρησιμοποιεί ένα πλήθος από tags για την μορφοποίηση κειμένου, για την δημιουργία συνδέσμων (links) μετάβασης ανάμεσα των σελίδα, για την εισαγωγή εικόνων, ήχου κ.α. Όταν ένας Web Browser ανοίγει ένα αρχείο HTML τα στοιχεία (tags) μεταφράζονται σε κατάλληλα χαρακτηριστικά με αποτελέσματα στην εμφάνιση και στην λειτουργικότητα της συγκεκριμένης σελίδας.



Εικόνα 20 Basic Frontend Technologies

2.2.9.2 CSS / Bootstrap

Η **CSS (Cascading Style Sheets)** – διαδοχικά φύλλα ύφους ή επάλληλα φύλλα ύφους) είναι μια γλώσσα υπολογιστή που ανήκει στην κατηγορία των γλωσσών φύλλων ύφους που χρησιμοποιείται για τον έλεγχο της εμφάνισης ενός εγγράφου που έχει γραφτεί με μια γλώσσα σήμανσης. Χρησιμοποιείται δηλαδή για τον έλεγχο της εμφάνισης ενός εγγράφου που γράφτηκε στις γλώσσες HTML και XHTML, δηλαδή για τον έλεγχο της εμφάνισης μιας ιστοσελίδας και γενικότερα ενός ιστοτόπου. Η CSS είναι μια γλώσσα υπολογιστή προορισμένη να αναπτύσσει στυλιστικά μια ιστοσελίδα δηλαδή να διαμορφώνει περισσότερα χαρακτηριστικά, χρώματα, στοίχιση και δίνει περισσότερες δυνατότητες σε σχέση με την html. Για μια όμορφη και καλοσχεδιασμένη ιστοσελίδα η χρήση της CSS κρίνεται ως απαραίτητη.



Εικόνα 21 Bootstrap

Το **Bootstrap** είναι μια συλλογή εργαλείων ανοιχτού κώδικα (ελεύθερο λογισμικό) για τη δημιουργία ιστοσελίδων και διαδικτυακών εφαρμογών. Περιέχει **HTML** και **CSS** για τις μορφές τυπογραφίας, κουμπιά πλοήγησης και άλλων στοιχείων του περιβάλλοντος, καθώς και προαιρετικές επεκτάσεις **JavaScript**. Είναι το πιο δημοφιλές πρόγραμμα στο GitHub [1] και έχει χρησιμοποιηθεί από τη **NASA** και το **MSNBC**, μεταξύ άλλων.

Το **Bootstrap** έχει σχετικά ελλιπή υποστήριξη για **HTML5** και **CSS**, αλλά είναι συμβατό με όλους τους φυλλομετρητές (browsers). Βασικές πληροφορίες

συμβατότητας των ιστοσελίδων ή εφαρμογές είναι διαθέσιμες για όλες τις συσκευές και τα προγράμματα περιήγησης. Υπάρχει μια έννοια της μερικής συμβατότητας που κάνει τα βασικά στοιχεία μιας ιστοσελίδας που διατίθενται για όλες τις συσκευές και τα προγράμματα περιήγησης. Για παράδειγμα, οι ιδιότητες πάνω στις οποίες θεσπίστηκε το **CSS3** για στρογγυλεμένες γωνίες, κλίσεις και σκιές, χρησιμοποιούνται από το Bootstrap παρά την έλλειψη υποστήριξης από μεγάλα προγράμματα περιήγησης στο Web. Αυτά επεκτείνουν τη λειτουργικότητα του πακέτου εργαλείων, αλλά δεν απαιτούνται για τη χρήση του.

Από την έκδοση 2.0 υποστηρίζει επίσης ανταποκρίσιμο σχεδιασμό (responsive design). Αυτό σημαίνει ότι η διάταξη των ιστοσελίδων προσαρμόζεται δυναμικά, λαμβάνοντας υπόψη τα χαρακτηριστικά της συσκευής που χρησιμοποιείται (PC, tablet, κινητό τηλέφωνο).

Το Bootstrap είναι ανοικτού κώδικα και είναι διαθέσιμο στο GitHub. Οι προγραμματιστές ενθαρρύνονται να συμμετέχουν στο έργο και να κάνουν τη δική τους συνεισφορά στην πλατφόρμα.

Δομή και Λειτουργία

Το Bootstrap είναι σπονδυλωτό και αποτελείται ουσιαστικά από μια σειρά στυλ(stylsheets) που εφαρμόζουν τα διάφορα συστατικά του πακέτου εργαλείων. Ένα στυλ που ονομάζεται bootstrap.less περιλαμβάνει τα συστατικά stylesheets. Οι προγραμματιστές μπορούν να προσαρμόσουν το αρχείο Bootstrap, επιλέγοντας τα στοιχεία που θέλουν να χρησιμοποιήσουν στο έργο τους.

Προσαρμογές είναι δυνατές σε περιορισμένη έκταση μέσω ενός κεντρικού στυλ διαμόρφωσης. Η χρήση γλώσσας στυλ επιτρέπει τη χρήση για μεταβλητές, λειτουργίες και φορείς (operators), ένθετους επιλογείς, γνωστά και ως μείγματα mixin.

Από την έκδοση 2.0, η διαμόρφωση του Bootstrap έχει επίσης μία ειδική επιλογή "Προσαρμογή" στην τεκμηρίωση (documentation). Επιπλέον, ο σχεδιαστής του έργου επιλέγει σε μια φόρμα τα επιθυμητά συστατικά και τα προσαρμόζει, εάν είναι αναγκαίο, σε τιμές διαφόρων εναλλακτικών λύσεων για τις ανάγκες του. Στη συνέχεια δημιουργείται ένα πακέτο που περιλαμβάνει ήδη το προ-χτισμένο CSS στυλ.

Σύστημα πλέγματος(Grid System) και ανταποκρίσιμος σχεδιασμός (responsive design)

Το Bootstrap έρχεται σταθερό με 940 pixel πλάτος. Εναλλακτικά, ο προγραμματιστής μπορεί να χρησιμοποιήσει μία μεταβλητού πλάτους διάταξη. Και για τις δύο περιπτώσεις, η εργαλειοθήκη έχει τέσσερις παραλλαγές και κάθε παραλλαγή ρυθμίζει το πλάτος των στηλών.

Η κατανόηση του CSS στυλ

Το Bootstrap παρέχει ένα σύνολο στυλ που παρέχουν βασικούς ορισμούς στυλ για όλα τα βασικά στοιχεία HTML. Αυτά παρέχουν ενιαία, σύγχρονη εμφάνιση για πίνακες, μορφοποίηση κειμένου, καθώς και στοιχεία μιας φόρμας.

Επαναχρησιμοποιήσιμα συστατικά

Εκτός από τα βασικά HTML στοιχεία, το Bootstrap περιέχει και άλλα στοιχεία περιβάλλοντος που χρησιμοποιούνται συχνά. Αυτά περιλαμβάνουν κουμπιά με προηγμένα χαρακτηριστικά (π.χ. ομαδοποίηση κουμπιών ή drop -down επιλογή, οριζόντιες και κάθετες καρτέλες, πλοήγηση, σελιδοποίηση, κ.λπ.), ετικέτες, προηγμένες τυπογραφικές δυνατότητες, εικονίδια, προειδοποιητικά μηνύματα και μια γραμμή προόδου.

JavaScript στοιχεία

Το Bootstrap έρχεται με πολλά συστατικά JavaScript σε μια μορφή jQuery plugin. Παρέχουν πρόσθετη διεπαφή χρήστη με στοιχεία όπως παράθυρα διαλόγου, επεξηγήσεις, και καρουσέλ. Μπορούν επίσης να επεκτείνουν τη λειτουργικότητα ορισμένων υφιστάμενων στοιχείων της διασύνδεσης, όπως για παράδειγμα μια αυτόματη πλήρη λειτουργία για πεδία εισαγωγής. Στην έκδοση 2.0, υποστηρίζονται τα ακόλουθα JavaScript plugins: Modal, Αναπτυσσόμενο, Scrollspy, Tab, Tooltip, Popover, Alert, Button, Collapse, Carousel και Typeahead.

Διατίθεται επίσης μια υλοποίηση του Bootstrap Twitter που χρησιμοποιεί το Dojo Toolkit. Λέγεται Dojo Bootstrap και είναι στα plugins του Bootstrap Twitter. Χρησιμοποιεί 100% κώδικα Dojo και έχει υποστήριξη για AMD (Asynchronous Module Definition).



Εικόνα 22 Bootstrap v5

2.2.9.3 Javascript

Η **JavaScript** (JS) είναι διερμηνευμένη γλώσσα προγραμματισμού για ηλεκτρονικούς υπολογιστές. Αρχικά αποτέλεσε μέρος της υλοποίησης των φυλλομετρητών Ιστού, ώστε τα σενάρια από την πλευρά του πελάτη (client-side scripts) να μπορούν να επικοινωνούν με τον χρήστη, να ανταλλάσσουν δεδομένα ασύγχρονα και να αλλάζουν δυναμικά το περιεχόμενο του εγγράφου που εμφανίζεται.

Η JavaScript είναι μια γλώσσα σεναρίων που βασίζεται στα πρωτότυπα (prototype-based), είναι δυναμική, με ασθενείς τύπους και έχει συναρτήσεις ως αντικείμενα πρώτης τάξης. Η σύνταξή της είναι επηρεασμένη από τη C. Η JavaScript αντιγράφει πολλά ονόματα και συμβάσεις ονοματοδοσίας από τη Java, αλλά γενικά οι δύο αυτές γλώσσες δε σχετίζονται και έχουν πολύ διαφορετική σημασιολογία. Οι βασικές αρχές σχεδιασμού της JavaScript προέρχονται από τις γλώσσες προγραμματισμού Self και Scheme. Είναι γλώσσα βασισμένη σε διαφορετικά προγραμματιστικά παραδείγματα (multi-paradigm), υποστηρίζοντας αντικειμενοστρεφές, προστακτικό και συναρτησιακό στυλ προγραμματισμού.

Η JavaScript χρησιμοποιείται και σε εφαρμογές εκτός ιστοσελίδων — τέτοια παραδείγματα είναι τα έγγραφα PDF, οι εξειδικευμένοι φυλλομετρητές (site-specific browsers) και οι μικρές εφαρμογές της επιφάνειας εργασίας (desktop widgets). Οι νεότερες εικονικές μηχανές και πλαίσια ανάπτυξης για JavaScript (όπως το Node.js) έχουν επίσης κάνει τη JavaScript πιο δημοφιλή για την ανάπτυξη εφαρμογών Ιστού στην πλευρά του διακομιστή (server-side).

Το πρότυπο της γλώσσας κατά τον οργανισμό τυποποίησης ECMA ονομάζεται ECMAScript.

Η γλώσσα προγραμματισμού JavaScript δημιουργήθηκε αρχικά από τον Brendan Eich της εταιρείας Netscape με την επωνυμία Mocha. Αργότερα, η Mocha μετονομάστηκε σε LiveScript, και τελικά σε JavaScript, κυρίως επειδή η ανάπτυξή της επηρεάστηκε περισσότερο από τη γλώσσα προγραμματισμού Java. LiveScript ήταν το επίσημο όνομα της γλώσσας όταν για πρώτη φορά κυκλοφόρησε στην αγορά σε βήτα (beta) εκδόσεις με το πρόγραμμα περιήγησης στο Web, Netscape Navigator εκδοχή 2.0 τον Σεπτέμβριο του 1995. Η LiveScript μετονομάστηκε σε JavaScript σε μια κοινή ανακοίνωση με την εταιρεία Sun Microsystems στις 4 Δεκεμβρίου, 1995, όταν επεκτάθηκε στην έκδοση του προγράμματος περιήγησης στο Web, Netscape εκδοχή 2.0B3.

Η JavaScript απέκτησε μεγάλη επιτυχία ως γλώσσα στην πλευρά του πελάτη (client-side) για εκτέλεση κώδικα σε ιστοσελίδες, και περιλήφθηκε σε διάφορα προγράμματα περιήγησης στο Web. Κατά συνέπεια, η εταιρεία Microsoft ονόμασε την εφαρμογή της σε JScript για να αποφύγει δύσκολα θέματα εμπορικών σημάτων. JScript πρόσθεσε νέους μεθόδους για να διορθώσει τα Y2K-προβλήματα στην JavaScript, οι οποίοι βασίστηκαν στην java.util.Date τάξη της Java. JScript περιλήφθηκε στο πρόγραμμα Internet Explorer εκδοχή 3.0, το οποίο κυκλοφόρησε τον Αύγουστο του 1996.

Τον Νοέμβριο του 1996, η Netscape ανακοίνωσε ότι είχε υποβάλει τη γλώσσα JavaScript στο Ecma International (μια οργάνωση της τυποποίησης των γλωσσών προγραμματισμού) για εξέταση ως βιομηχανικό πρότυπο, και στη συνέχεια το έργο είχε ως αποτέλεσμα την τυποποιημένη μορφή που ονομάζεται ECMAScript.

Η JavaScript έχει γίνει μία από τις πιο δημοφιλείς γλώσσες προγραμματισμού ηλεκτρονικών υπολογιστών στον Παγκόσμιο Ιστό (Web). Αρχικά, όμως, πολλοί επαγγελματίες προγραμματιστές υποτίμησαν τη γλώσσα διότι το κοινό της ήταν ερασιτέχνες συγγραφείς ιστοσελίδων και όχι επαγγελματίες προγραμματιστές (μαζί με άλλους λόγους). Με τη χρήση της τεχνολογίας Ajax, η JavaScript γλώσσα επέστρεψε

στο προσκήνιο και έφερε πιο επαγγελματική προσοχή προγραμματισμού. Το αποτέλεσμα ήταν ένα καινοτόμο αντίκτυπο στην εξάπλωση των πλαισίων και των βιβλιοθηκών, τη βελτίωση προγραμματισμού με JavaScript, καθώς και αυξημένη χρήση της JavaScript έξω από τα προγράμματα περιήγησης στο Web.

Τον Ιανουάριο του 2009, το έργο CommonJS ιδρύθηκε με στόχο τον καθορισμό ενός κοινού προτύπου βιβλιοθήκης κυρίως για την ανάπτυξη της JavaScript έξω από το πρόγραμμα περιήγησης και μέσα σε άλλες τεχνολογίες (π.χ. server-side).

Η αρχική έκδοση της Javascript βασίστηκε στη σύνταξη στη γλώσσα προγραμματισμού C, αν και έχει εξελιχθεί, ενσωματώνοντας πια χαρακτηριστικά από νεότερες γλώσσες.

Αρχικά χρησιμοποιήθηκε για προγραμματισμό από την πλευρά του πελάτη (client), που ήταν ο φυλλομετρητής (browser) του χρήστη, και χαρακτηρίστηκε σαν client-side γλώσσα προγραμματισμού. Αυτό σημαίνει ότι η επεξεργασία του κώδικα Javascript και η παραγωγή του τελικού περιεχομένου HTML δεν πραγματοποιείται στο διακομιστή, αλλά στο πρόγραμμα περιήγησης των επισκεπτών, ενώ μπορεί να ενσωματωθεί σε στατικές σελίδες HTML. Αντίθετα, άλλες γλώσσες όπως η PHP εκτελούνται στο διακομιστή (server-side γλώσσες προγραμματισμού).

Παρά την ευρεία χρήση της Javascript για συγγραφή προγραμμάτων σε περιβάλλον φυλλομετρητή, από την αρχή χρησιμοποιήθηκε και για τη συγγραφή κώδικα από την πλευρά του διακομιστή, από την ίδια τη Netscape στο προϊόν LiveWire, με μικρή επιτυχία. Η χρήση της Javascript στο διακομιστή εμφανίζεται πάλι σήμερα, με τη διάδοση του Node.js, ενός μοντέλου προγραμματισμού βασισμένο στα γεγονότα (events).

Η Javascript δεν θα πρέπει να συγχέεται με τη Java, που είναι διαφορετική γλώσσα προγραμματισμού και με διαφορετικές εφαρμογές. Η χρήση της λέξης "Java" στο όνομα της γλώσσας έχει περισσότερη σχέση με το προφίλ του προϊόντος που έπρεπε να έχει και λιγότερο με κάποια πιθανή συμβατότητα ή άλλη στενή σχέση με τη Java. Ρόλο σε αυτήν τη σύγχυση έπαιξε και ότι η Java και η Javascript έχουν δεχτεί σημαντικές επιρροές από τη γλώσσα C, ειδικά στο συντακτικό, ενώ είναι και οι δύο αντικειμενοστρεφείς γλώσσες. Τονίζεται ότι ο σωστός τρόπος γραφής της είναι "Javascript" και όχι 'Java script' σαν δύο λέξεις, όπως λανθασμένα γράφεται ορισμένες φορές.



Εικόνα 23 JavaScript

2.2.9.4 *React*

Το React (γνωστό και ως React.js ή ReactJS) είναι μια δωρεάν και ανοιχτού κώδικα βιβλιοθήκη JavaScript front-end για τη δημιουργία διεπαφών χρήστη που βασίζονται σε στοιχεία διεπαφής χρήστη. Διατηρείται από τη Meta (πρώην Facebook) και μια κοινότητα μεμονωμένων προγραμματιστών και εταιρειών. Το React μπορεί να χρησιμοποιηθεί ως βάση για την ανάπτυξη εφαρμογών μιας σελίδας ή για κινητές συσκευές. Ωστόσο, το React ασχολείται μόνο με τη διαχείριση κατάστασης και την απόδοση αυτής της κατάστασης στο DOM, επομένως η δημιουργία εφαρμογών React απαιτεί συνήθως τη χρήση πρόσθετων βιβλιοθηκών για δρομολόγηση, καθώς και ορισμένες λειτουργίες από την πλευρά του πελάτη.

Το React δεν επιχειρεί να παρέχει μια πλήρη "βιβλιοθήκη εφαρμογών". Έχει σχεδιαστεί ειδικά για τη δημιουργία διεπαφών χρήστη και επομένως δεν περιλαμβάνει πολλά από τα εργαλεία που ορισμένοι προγραμματιστές μπορεί να θεωρήσουν απαραίτητα για τη δημιουργία μιας εφαρμογής. Αυτό επιτρέπει την επιλογή όποιων βιβλιοθηκών προτιμά ο προγραμματιστής για την εκτέλεση εργασιών όπως η εκτέλεση πρόσβασης στο δίκτυο ή η τοπική αποθήκευση δεδομένων. Κοινά πρότυπα χρήσης έχουν εμφανιστεί καθώς η βιβλιοθήκη ωριμάζει.

Μονοκατευθυντική ροή δεδομένων

Για να υποστηρίξει την ιδέα του React για μονοκατευθυντική ροή δεδομένων (η οποία μπορεί να έρχεται σε αντίθεση με την αμφίδρομη ροή του AngularJS), η αρχιτεκτονική Flux αναπτύχθηκε ως εναλλακτική λύση στη δημοφιλή αρχιτεκτονική μοντέλο-προβολή-ελεγκτής. Το Flux διαθέτει ενέργειες που αποστέλλονται μέσω ενός κεντρικού διεκπεραιωτή σε ένα κατάσταση και οι αλλαγές στο κατάσταση μεταδίδονται πίσω στην προβολή. Όταν χρησιμοποιείται με το React, αυτή η διάδοση πραγματοποιείται μέσω των ιδιοτήτων του στοιχείου. Από τη σύλληψή του, το Flux έχει αντικατασταθεί από βιβλιοθήκες όπως η Redux και η MobX.

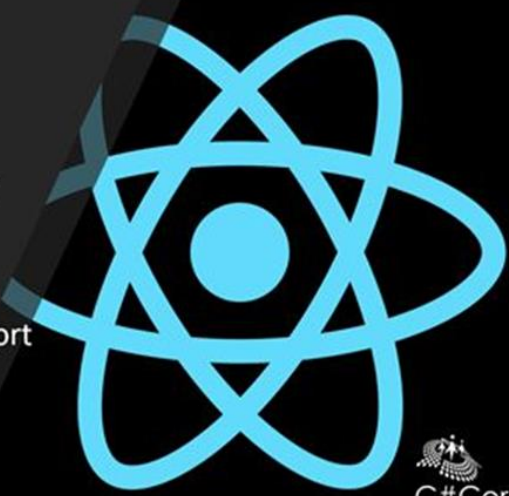
Η ροή μπορεί να θεωρηθεί μια παραλλαγή του προτύπου του παρατηρητή.

Ένα στοιχείο React κάτω από την αρχιτεκτονική Flux δεν θα πρέπει να τροποποιεί απευθείας τυχόν στηρίγματα που του έχουν διαβιβαστεί, αλλά θα πρέπει να μεταβιβάζονται συναρτήσεις επανάκλησης που δημιουργούν ενέργειες που αποστέλλονται από τον διεκπεραιωτή για την τροποποίηση του χώρου αποθήκευσης. Η ενέργεια είναι ένα αντικείμενο του οποίου η ευθύνη είναι να περιγράψει αυτό που έχει λάβει χώρα: για παράδειγμα, μια ενέργεια που περιγράφει έναν χρήστη που "ακολουθεί" έναν άλλο μπορεί να περιέχει ένα αναγνωριστικό χρήστη, ένα αναγνωριστικό χρήστη προορισμού και τον τύπο `USER_FOLLOWED_ANOTHER_USER`. Τα καταστήματα, τα οποία μπορούν να θεωρηθούν ως μοντέλα, μπορούν να αλλάξουν τον εαυτό τους ως απάντηση σε ενέργειες που λαμβάνονται από τον αποστολέα.

Αυτό το μοτίβο μερικές φορές εκφράζεται ως «οι ιδιότητες ρέουν προς τα κάτω, οι ενέργειες ρέουν προς τα πάνω». Πολλές υλοποιήσεις του Flux έχουν δημιουργηθεί από την έναρξή του, ίσως η πιο γνωστή είναι το Redux, το οποίο διαθέτει ένα μόνο κατάστημα, που συχνά αποκαλείται μια ενιαία πηγή αλήθειας.

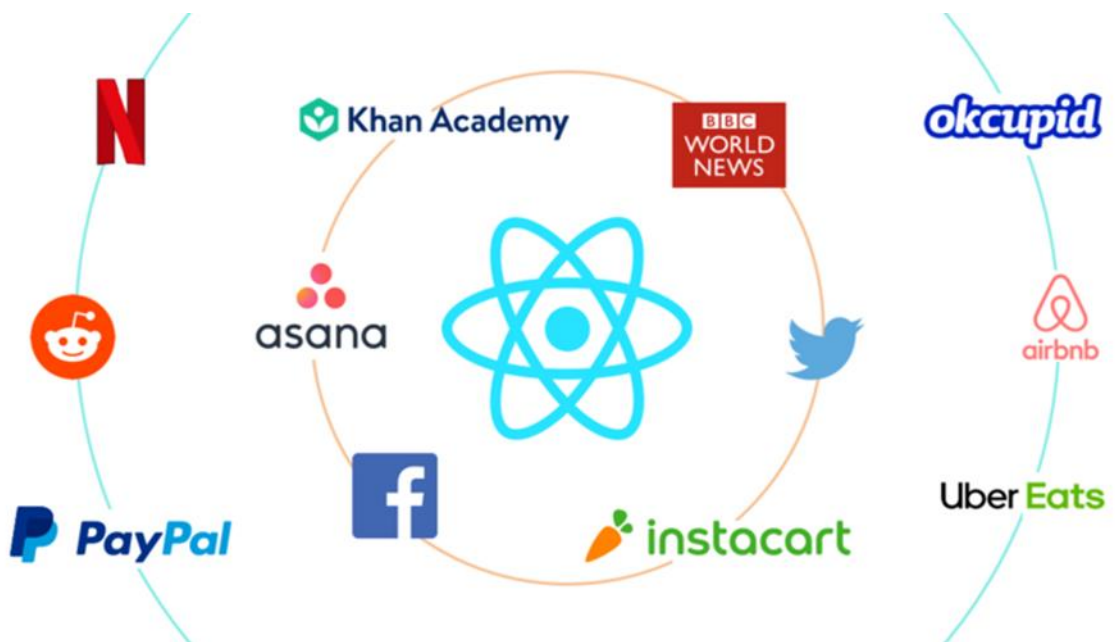
React Properties

1. React is declarative
2. React is simple yet powerful
3. React is component-based
4. React supports server-side
5. React supports mobile support
6. React is extensible



C#Corner

Εικόνα 24 React Properties



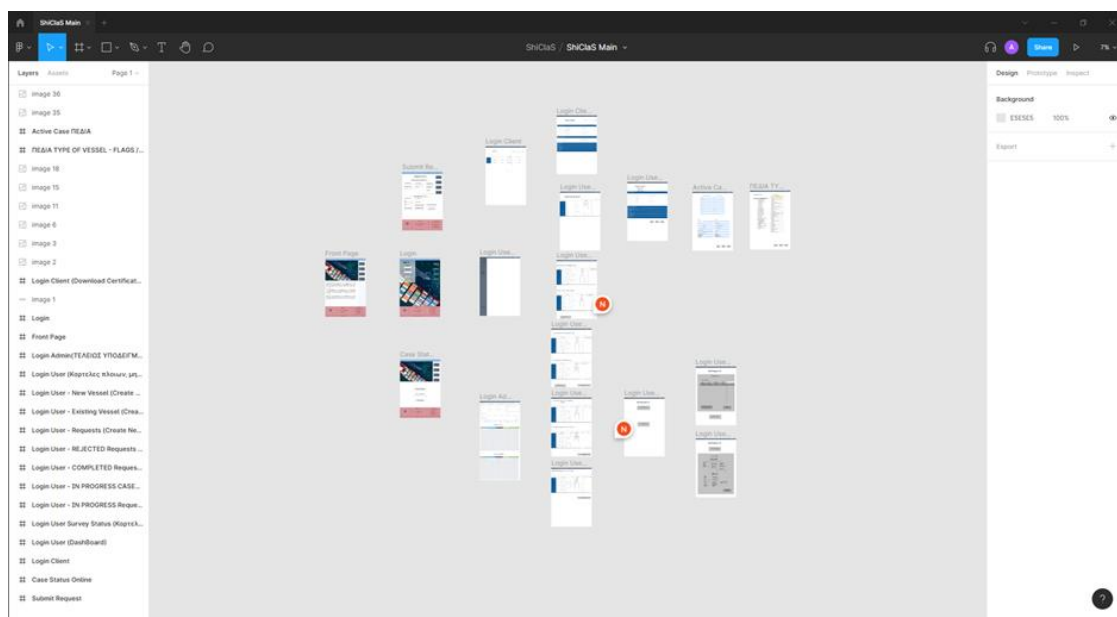
Εικόνα 25 React Use by Companies

3 Η ΥΛΟΠΟΙΗΣΗ ΤΗΣ ΕΦΑΡΜΟΓΗΣ

Η υλοποίηση μιας εφαρμογής μπορεί να διαιρεθεί σε πολλές κατηγορίες που διαφέρουν αναλόγως τα ζητούμενα και τις ανάγκες της εκάστοτε περίπτωσης. Στην ενότητα αυτή θα παρουσιάσουμε την υλοποίηση της δικής μας εφαρμογής, βήμα βήμα, τους στόχους και ότι προβλήματα αντιμετωπίσαμε ώστε να επιτύχουμε την κάλυψη της εφαρμογής σε όλες τις φάσεις του κύκλου ζωής της.

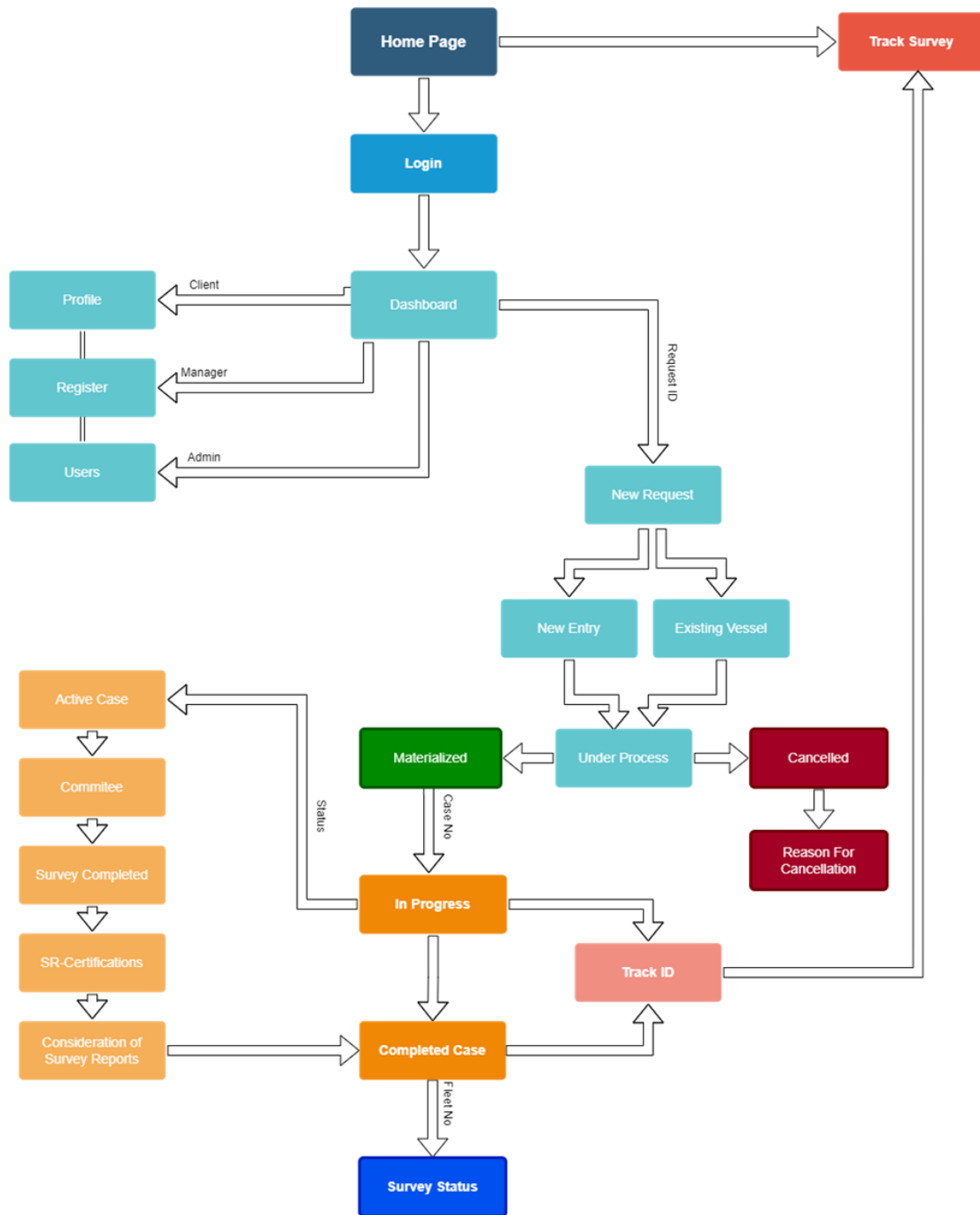
3.1 ΣΧΕΔΙΑΣΜΟΣ ΕΦΑΡΜΟΓΗΣ

Βασικό κομμάτι πριν από κάθε ανάπτυξη είναι ο σχεδιασμός. Σε όλους τους τομείς τις καθημερινότητας μας, οτιδήποτε κατασκευάζεται πρέπει πρώτα να έχει σχεδιαστεί και να έχει αναλυθεί αρκετά ώστε να υπάρχουν αναλυτικά όλες οι περιπτώσεις χρήσεως, τα προβλήματα, τα λάθη που μπορεί να προκύψουν, το τι μπορεί να υλοποιηθεί με τα διαθέσιμα μέσα και το τι όχι. Βέβαια το κομμάτι του σχεδιασμού προϋποθέτει ένα ολόκληρο τμήμα ή και τμήματα σε μικρές και μεγάλες εταιρείες. Εμείς με ότι γνώσεις και μέσα διαθέταμε, προσπαθήσαμε να μην παραλείψουμε αυτο το βήμα και να προσχεδιάσουμε την εφαρμογή μας όσο το δυνατόν καλύτερα και αναλυτικότερα. Παραπάνω αναφέραμε το εργαλείο Figma για UI/UX Design το οποίο και χρησιμοποιήσαμε. Αρκετά εύκολο εργαλείο για να ξεκινήσεις χωρίς να χάνει σε δυνατότητες, που όσο δουλεύεις και εμβαθύνεις θα καταλάβεις ότι μπορεί να κάνει απίστευτα πράγματα για κάθε επαγγελματία εκεί έξω.



Εικόνα 26 Figma Design

Ξεκινήσαμε να σχεδιάζουμε ταυτόχρονα το τι θέλουμε από την εφαρμογή μας να κάνει και το πως θα φαίνεται. Ευτυχώς το Figma έχει απλά εργαλεία με πολλές όμως παραμετροποιήσεις. Έτσι κάναμε μια αρχική σχεδίαση των σελίδων που θα έχει η εφαρμογή και το τι θα περιέχει κάθε σελίδα. Παρακάτω ακολουθεί το διάγραμμα ροής.



Εικόνα 27 Flow Diagram

3.2 ΑΝΑΠΤΥΞΗ ΕΦΑΡΜΟΓΗΣ

Το σημαντικότερο κομμάτι στην αλυσίδα της δημιουργίας ενός προϊόντος, οποιοδήποτε και να είναι αυτό, είναι η ίδια η ανάπτυξη του προϊόντος. Όσο τέλεια και

αν είναι η σχεδίαση του είτε η προώθηση του, αν δεν έχει δοθεί η απαραίτητη προσοχή στην ανάπτυξη του εκάστοτε προϊόντος, ο καταναλωτής/αγοραστής/χρήστης θα δυσκολευτεί αν όχι να το απορρίψει εντελώς. Έτσι με γνώμονα αυτό, προσπαθήσαμε να δώσουμε όλη μας την προσοχή στην ανάπτυξη της εφαρμογής μας και εξελίσσοντας τις γνώσεις μας μέρα με την μέρα για το πως θα την κάνουμε καλύτερη.

3.2.1 ΓΡΑΦΙΚΟ ΠΕΡΙΒΑΛΛΟΝ



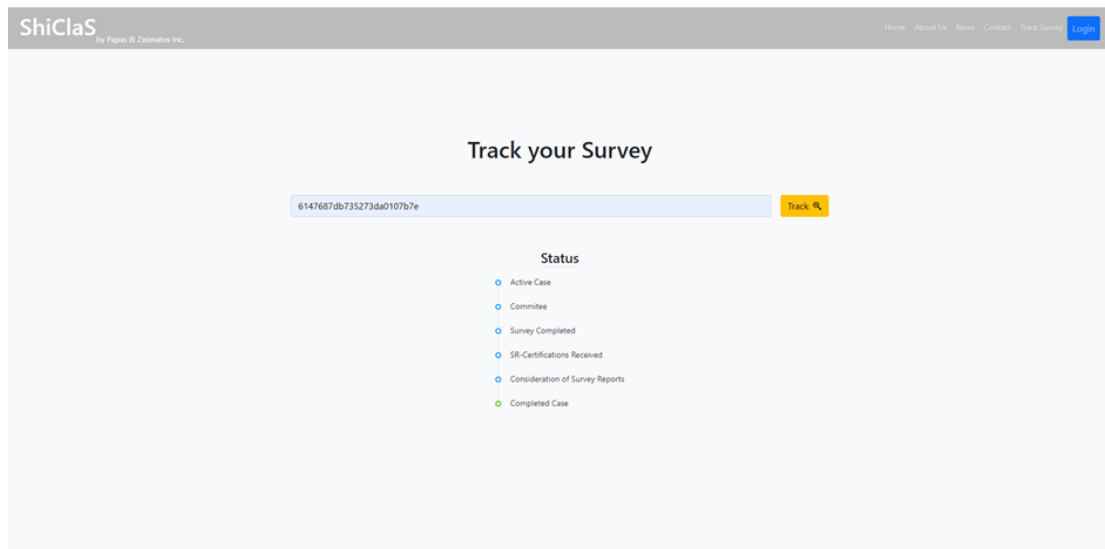
Εικόνα 28 Web App Homepage

Η εφαρμογή μας ξεκινά από την αρχική σελίδα, στην οποία βλέπουμε ένα βίντεο καλωσορίσματος και πάνω δεξιά το βασικό μενού.



Εικόνα 29 Homepage Menu

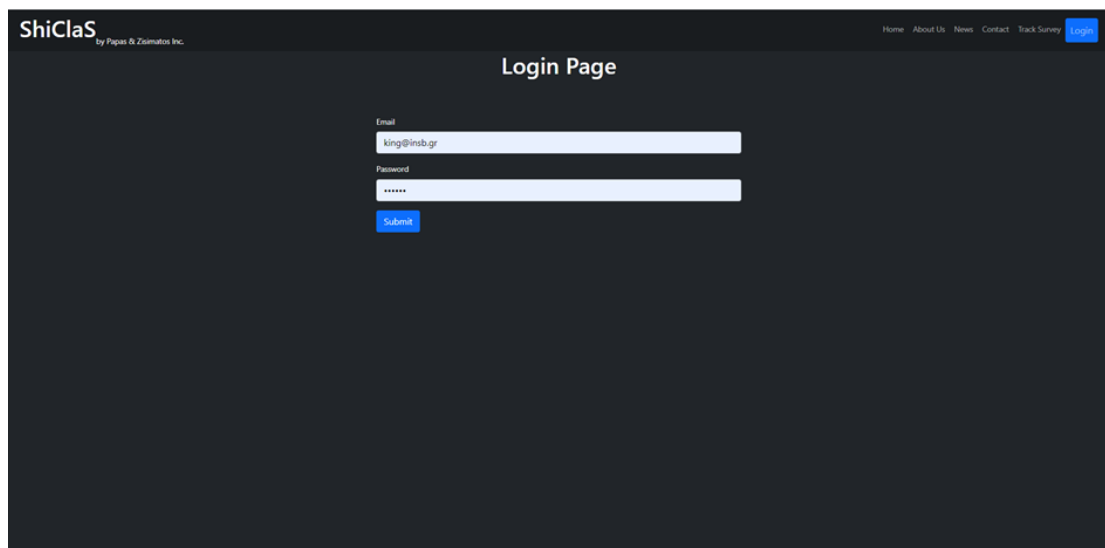
Το μενού περιέχει κάποια βασικά κομμάτια όπως πληροφορίες, νέα, επικοινωνία αλλά κυρίως θα εστιάσουμε στα δύο τελευταία τα οποία είναι βασικά μέρη της εφαρμογής.



Εικόνα 30 Survey Tracking

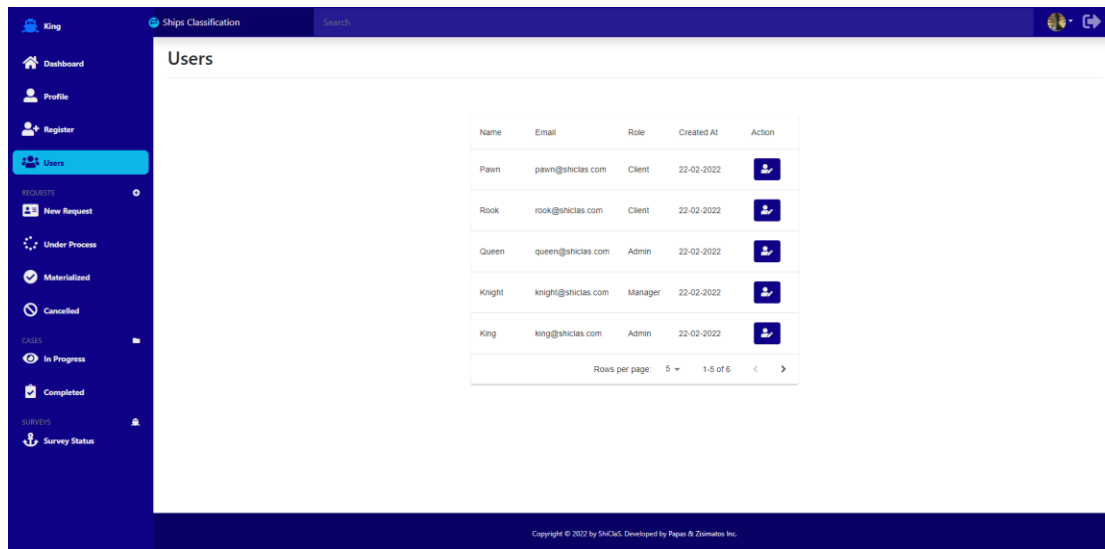
Track Survey: Κύριο χαρακτηριστικό της εφαρμογής είναι το να μπορεί οποιοσδήποτε να ελέγξει την κατάσταση της αίτησης του.

Έτσι του δίνουμε την δυνατότητα μέσα από το tracking σύστημα μας, να βάλει τον κωδικό που του έχει δοθεί όταν έκανε την αίτηση και ανά πάσα στιγμή να δει σε ποιο βήμα βρίσκεται ή αν ολοκληρώθηκε.



Εικόνα 31 Login Page

Login: Εν συνεχεία, το υπόλοιπο κομμάτι της εφαρμογής βρίσκεται πίσω από ένα Authentication System και χρειάζονται τα κατάλληλα στοιχεία για να προχωρήσει ο χρήστης.



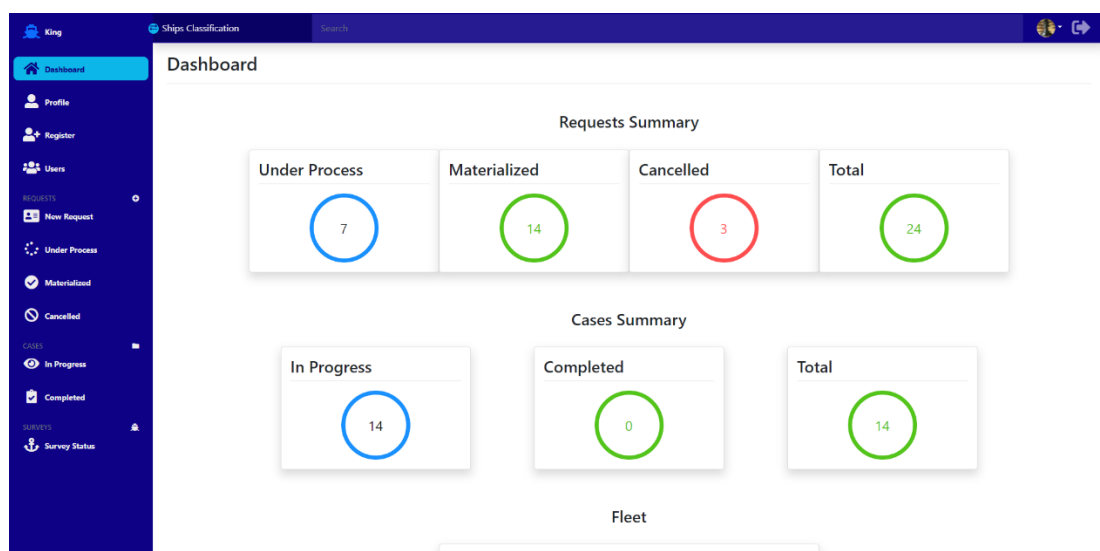
Εικόνα 32 Users

Η εφαρμογή σχεδιάστηκε με γνώμονα τους διαφορετικούς χρήστες οι οποίοι έχουν και διαφορετικές δυνατότητες μέσα σε αυτήν. Τα στοιχεία που θα εισάγει ο χρήστης αντιπροσωπεύουν έναν λογαριασμό με συγκεκριμένο ρόλο και λειτουργίες. Χαρακτηριστικά βλέπουμε πως υπάρχει ένας ρόλος Admin, ένας Manager και εκείνος του Client.

Name	Email	Role	Created At	Action
Pawn	pawn@shiclas.com	Client	22-02-2022	
Rook	rook@shiclas.com	Client	22-02-2022	
Queen	queen@shiclas.com	Admin	22-02-2022	
Knight	knight@shiclas.com	Manager	22-02-2022	
King	king@shiclas.com	Admin	22-02-2022	

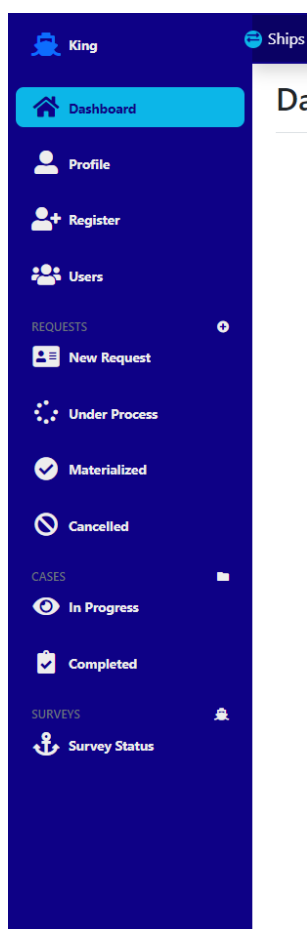
Rows per page: 5 ▾ 1-5 of 6 < >

Εικόνα 33 Users' Roles



Εικόνα 34 Dashboard

Μόλις κάνουμε είσοδο, θα βρεθούμε στο Dashboard, το οποίο περιέχει βασικά στατιστικά και πληροφορίες για το πόσες αιτήσεις έχουμε σε κάθε διαθέσιμη κατάσταση.



Αριστερά θα δούμε το μενού όπου από εδώ μπορεί να περιηγηθεί ο χρήστης σε όλες τις σελίδες της εφαρμογής.

Είναι χωρισμένο σε 4 βασικές κατηγορίες, με το όνομα του εκάστοτε ενεργού χρήστη να φαίνεται στην κορυφή δίπλα από το εικονίδιο του πλοίου.

Έπειτα βλέπουμε 4 μενού που έχουν να κάνουν με τον χρήστη αυτόν καθ' αυτόν όπως το προφίλ του είτε με την διαχείριση των υπόλοιπων χρηστών.

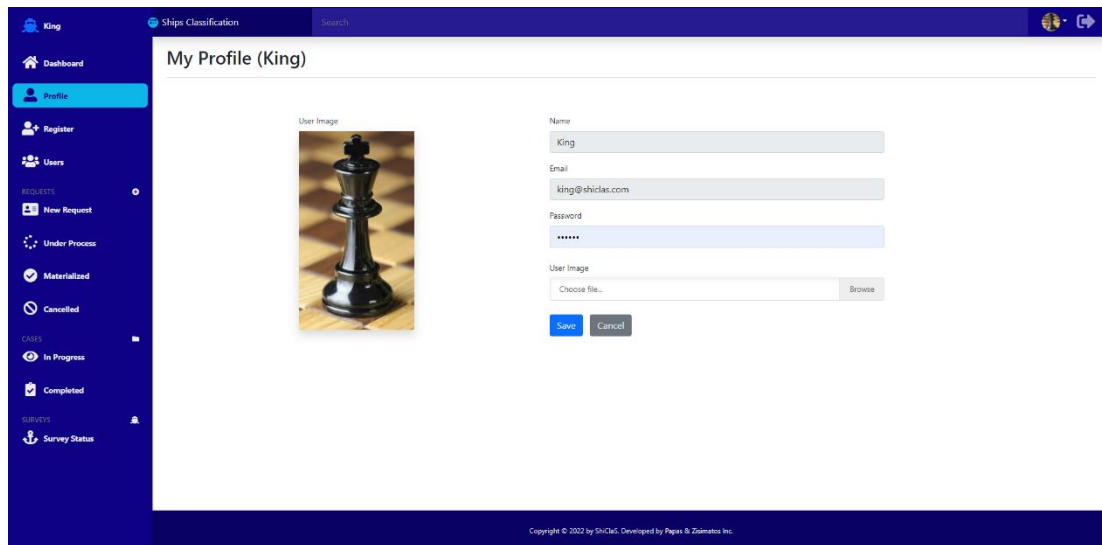
Έπειτα έχουμε τα Requests, τα οποία είναι οι αιτήσεις που έχουν φτάσει στην εταιρεία και βρίσκονται σε αρχικό στάδιο διαπραγματεύσεις.

Μόλις ολοκληρωθούν οι διαπραγματεύσεις, οι αιτήσεις περνάει στα Cases σαν ενεργή υπόθεση πλέον, όπου γίνεται η περισυλλογή όλων των δικαιολογητικών και ότι πληροφοριών χρειάζεται η εταιρεία.

Τέλος, μετά την ολοκλήρωση του Case, γίνεται η έκδοση του πιστοποιητικού και το πλοίο καταχωρείται στα Surveys.

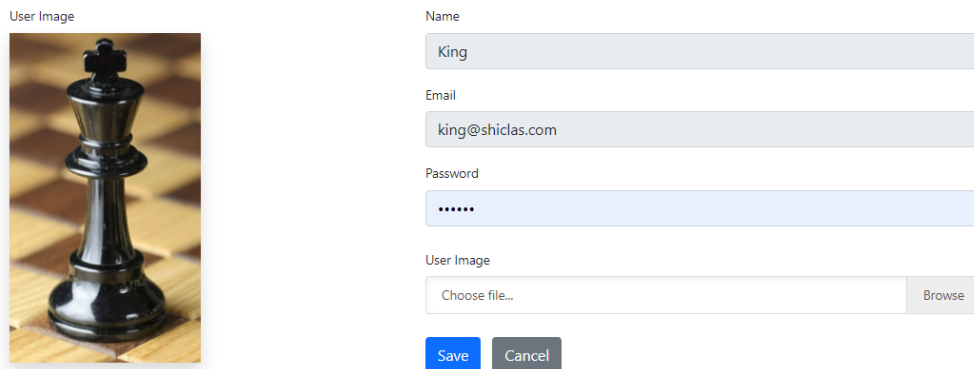
Το μενού σχεδιάστηκε ακολουθώντας την ροή της διαδικασίας.

Εικόνα 35 Menu



Εικόνα 36 Profile

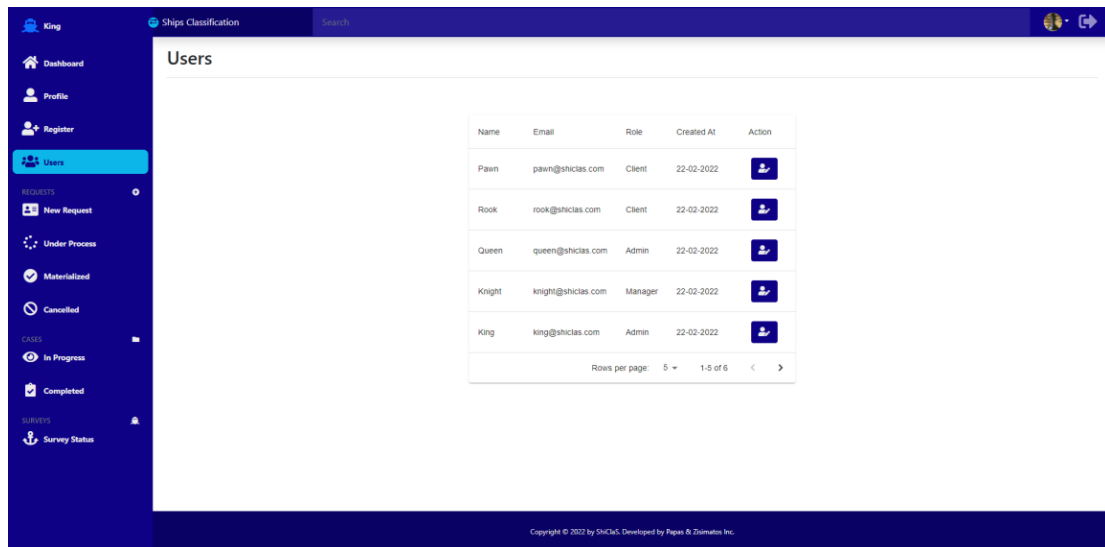
Στο Προφίλ θα βρείτε όλες τις πληροφορίες που έχουν να κάνουν με τον λογαριασμό του χρήστη. Όπως το Ονοματεπώνυμο του, το email του, τον κωδικό του και την προσωπική του φωτογραφία.



Εικόνα 37 Edit Profile

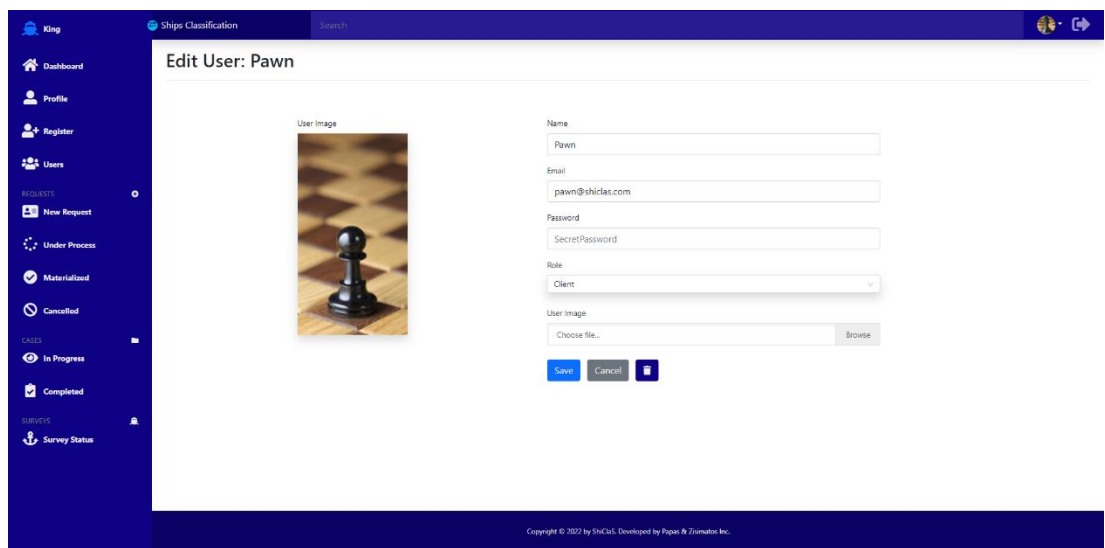
Από εδώ ο κάθε χρήστης μπορεί να αλλάξει μόνο τον κωδικό και την φωτογραφία του, καθώς η αλλαγή στο Ονοματεπώνυμο και το email του θα επηρεάζει την λειτουργία της εφαρμογής.

Σε περίπτωση λάθους, υπάρχει η δυνατότητα αλλαγής αυτών των στοιχείων από τον κάποιον με τα κατάλληλα δικαιώματα όπως ο Admin.



Εικόνα 38 Users

Πηγαίνοντας στην κατηγορία Users, θα δούμε την λίστα με όλους τους εγγεγραμμένους χρήστες. Από εδώ, επιλέγοντας το κουμπί της επεξεργασίας, δίπλα από το χρήστη που επιθυμούμε, θα οδηγηθούμε στα στοιχεία του.



Εικόνα 39 Edit User

Εδώ μπορεί να γίνει τροποποίηση των στοιχείων ενός χρήστη όπως και η αλλαγή του ρόλου του αλλά και η διαγραφή του.


Role

Client

Client

Manager

Admin

Save Cancel 

Εικόνα 40 User Role

Φυσικά, υπάρχει και η σελίδα της εγγραφής όπου μπορεί να δημιουργηθεί ένας νέος χρήστης.

King Ship Classification Search

Register a new user.

Name
Nick

Email
nick@example.com

Password
SecretPassword

Role
Select Role

User Image
Choose file... Browse

Submit

Copyright © 2022 by SHC/SLG. Developed by Paper & Zlatosoft Inc.

Εικόνα 41 Register New User

Επόμενα στη λίστα είναι τα Requests.

Όπως προείπαμε, οι αιτήσεις που έχουν φτάσει στην εταιρεία και βρίσκονται σε αρχικό στάδιο διαπραγματεύσεις περνάνε στο σύστημα σαν ένα νέο αίτημα/request.

The screenshot shows a web application interface for creating a new request. The page title is "New Request". At the top, there is a search bar and two tabs: "New Entry" (selected) and "Existing Vessel". The form is organized into two main sections: "Owner/Manager Information" and "Vessel Main Particulars".

Owner/Manager Information:

- Contact Name: Nick
- Contact Email: nick@example.com
- Owner Name: Nick
- Owner ID: 001
- Manager Name: Nick
- Manager ID: 001
- Address: 1234 Main St

Vessel Main Particulars:

- Vessel Name: Titanic
- Vessel Type: Select Vessel Type (dropdown)
- IMO: 12345678
- Year of Build: 2004

Εικόνα 42 New Request

Στην σελίδα New Request μπορεί ο χρήστης να προσθέσει ένα νέο αίτημα/request με τις βασικές πληροφορίες που χρειάζεται η εταιρεία για να ξεκινήσει την διαδικασία.

The screenshot shows a detailed view of the "Vessel Main Particulars" form. The form contains the following fields:

- Vessel Name: Titanic
- Vessel Type: Select Vessel Type (dropdown)
- IMO: 12345678
- Year of Build: 2004
- Flag: Select Flag (dropdown)
- GT (Gross Tonnage): 25
- Port of Registry: Piraeus
- LO (Length Overall): 10
- Notes: Add additional information (text area)
- Services: Select Services (dropdown)
- Status: Select Status (dropdown)

At the bottom right, there are two buttons: "Clear" and "Create".

Εικόνα 43 Request Main Particulars

Αυτές μπορεί να είναι βασικές πληροφορίες επικοινωνίας με τον διαχειριστή του εκάστοτε πλοίου, όπως και βασικές πληροφορίες που αφορούν το ίδιο το πλοίο αλλά και την ίδια την αίτηση.

New Request

New Entry

Existing Vessel

Εικόνα 44 New or Existing Vessel

Υπάρχει και η δυνατότητα Επιλογής είτε New Entry είτε Existing Vessel από το μενού στην κορυφή της σελίδας. Στο πρώτο γίνεται μια καταχώρηση ενός καινούργιου πλοίου που δεν προϋπάρχει σαν καταχώρηση στην εταιρεία, ενώ στο δεύτερο γίνεται η επιλογή από την λίστα με τα ήδη υπάρχοντα καταχωρημένα στην εταιρεία πλοία.

New Entry Existing Vessel

Select a Vessel

Vessel

- Titanic
- Silver Eagle
- Titanic
- ttS
- DOKIMASTIKO 1

Contact Name: Vasiliopoulos Ioannis

Contact Email: vasilop@gmail.com

Owner Name: Sotiria M

Owner ID: 57321

Manager Name: Vasiliopoulos J

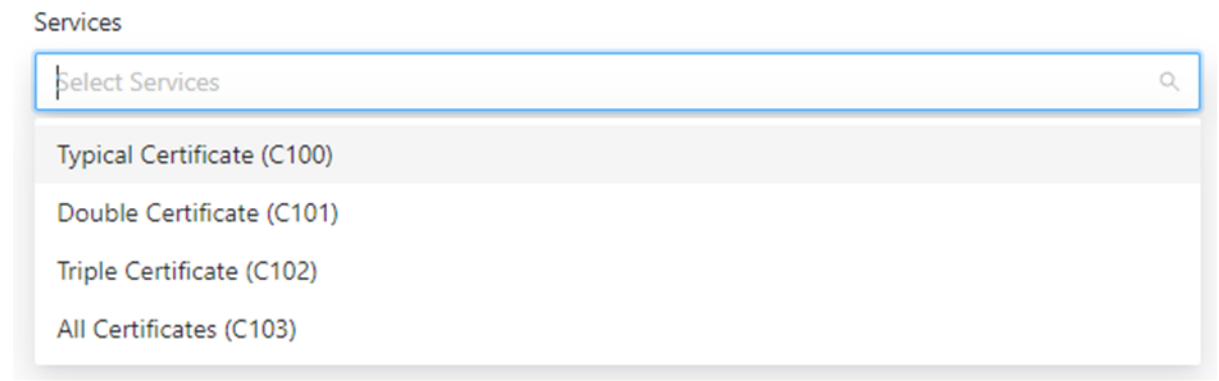
Manager ID: 21321

INSB No.: 1312

Address: Paridos2938 str

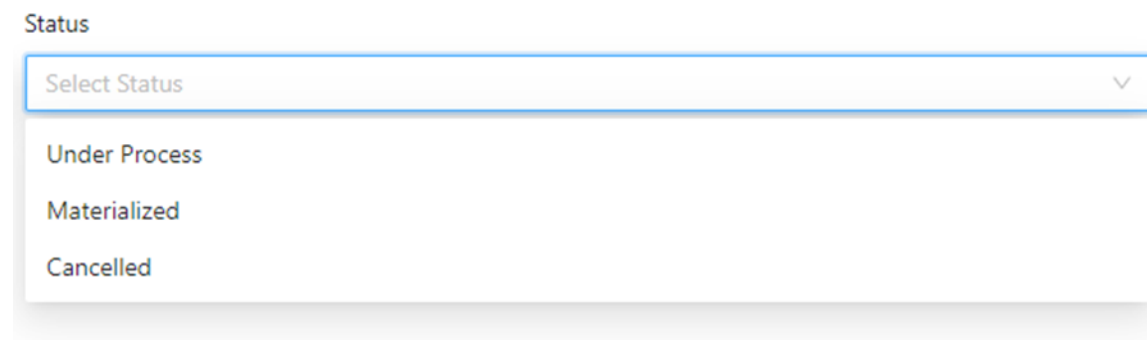
Εικόνα 45 Existing Vessel

Σημαντικό σημείο είναι η επιλογή του είδους του πιστοποιητικού που θέλουμε να εκδοθεί, το οποίο θα το βρούμε στο ανάλογο πεδίο Services.



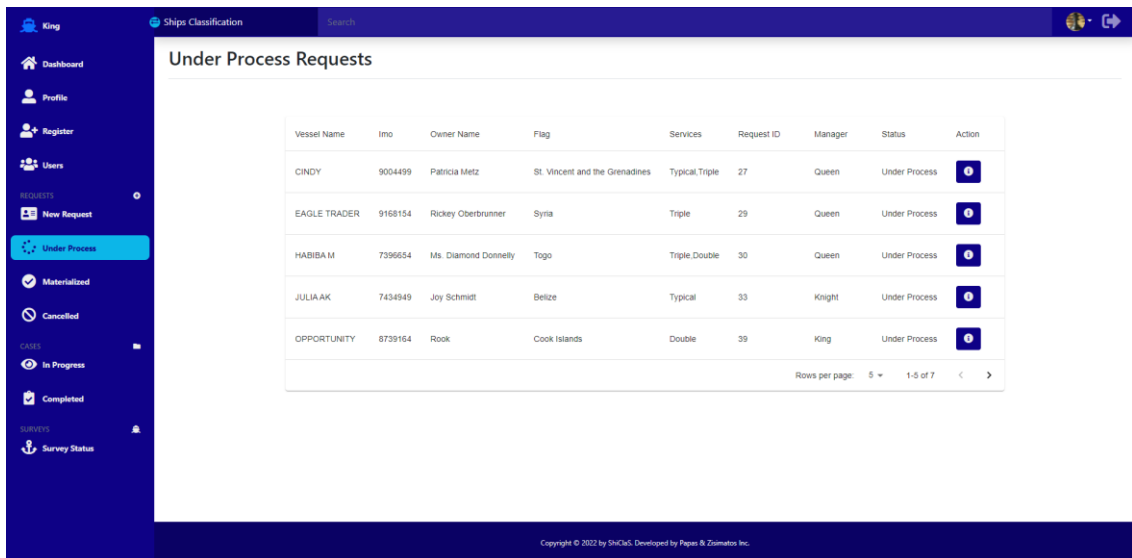
Εικόνα 46 Services Selection

Έπειτα, το αίτημα/request περνάει αυτόματα από το στάδιο με Status Under Process έως ότου γίνει η αλλαγή της κατάσταση σε Materialized είτε Cancelled από τον χρήστη.



Εικόνα 47 Request Status Selection

Ακολούθως, υπάρχουν και οι 3 σελίδες για την κάθε κατηγορία όπου εκεί εμφανίζονται οι αιτήσεις της κάθε μιας, όπου μπορεί να γίνει αναζήτηση και ταξινόμηση αλλά και επεξεργασία μιας αίτησης.

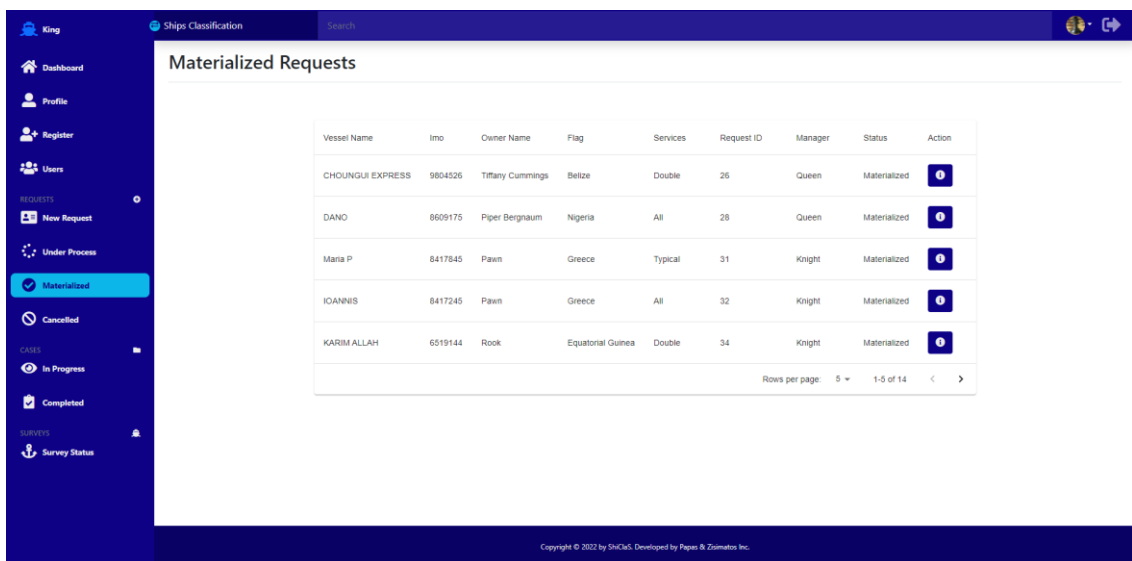


Under Process Requests

Vessel Name	Imo	Owner Name	Flag	Services	Request ID	Manager	Status	Action
CINDY	9004499	Patricia Metz	St. Vincent and the Grenadines	Typical, Triple	27	Queen	Under Process	
EAGLE TRADER	9168154	Rickey Oberbrunner	Syria	Triple	29	Queen	Under Process	
HABIBAM	7396654	Ms. Diamond Donnelly	Togo	Triple, Double	30	Queen	Under Process	
JULIAAK	7434949	Joy Schmidt	Belize	Typical	33	Knight	Under Process	
OPPORTUNITY	8739164	Rook	Cook Islands	Double	39	King	Under Process	

Rows per page: 5 | 1-5 of 7

Εικόνα 48 Under Process Requests



Materialized Requests

Vessel Name	Imo	Owner Name	Flag	Services	Request ID	Manager	Status	Action
CHOUNGUEI EXPRESS	9804526	Tiffany Cummings	Belize	Double	26	Queen	Materialized	
DANO	8609176	Piper Bergnaum	Nigeria	All	28	Queen	Materialized	
Maria P	8417845	Pawn	Greece	Typical	31	Knight	Materialized	
IOANNIS	8417245	Pawn	Greece	All	32	Knight	Materialized	
KARIM ALLAH	6519144	Rook	Equatorial Guinea	Double	34	Knight	Materialized	

Rows per page: 5 | 1-5 of 14

Εικόνα 49 Materialized Requests

Vessel Name	Imo	Owner Name	Flag	Services	Request ID	Manager	Status	Action
LEDRA ENA	7350363	Rook	Barbados	Triple	36	King	Cancelled	
PATHOS	8793768	Savlon Green	Greece	All	41	King	Cancelled	
POSEIDONAS	9500003	Gust Gerhold	Greece	Double	43	Bishop	Cancelled	

Rows per page: 5 - 1-3 of 3

Εικόνα 50 Cancelled Requests

Πατώντας το κουμπί των ενεργειών στην στήλη Actions, θα οδηγηθούμε στην προεπισκόπηση της αίτησης με όλες τις πληροφορίες, ενώ στο κάτω μέρος θα δούμε και 3 κουμπιά επιλογών για την συγκεκριμένη αίτηση.

Το ένα είναι της επεξεργασίας όπου θα οδηγηθούμε στην σελίδα επεξεργασίας της αίτησης, το άλλο είναι οι περισσότερες επιλογές όπου μπορούμε να ανοίξουμε ή να κατεβάσουμε την αίτηση σε pdf μορφή, ενώ το τρίτο είναι της ακύρωσης όπου θα οδηγηθούμε πίσω στην σελίδα με τις αιτήσεις.

Owner/Manager Information		Vessel Main Particulars	
Contact Name	Patricia MEC	Vessel Name	CINDY
Contact Email	Patricia1@medtrical.com	Vessel Type	Container
Owner Name	Patricia MEC	MIC	
Owner ID	2321321	Year of Build	2007
Manager Name	Patricia MEC	Flag	St. Vincent and the Grenadines
Manager ID	2321321	GT (Gross Tonnage)	6230
Address	49600 Nacyni Island	Port of Registry	Belize City
Services	Typical Trade	LO Length Overall	106.42
Notes			
Request ID	27	Status	Under Process

Print PDF
Open PDF
Download PDF

Εικόνα 51 Request Info

Στην Σελίδα της επεξεργασίας, μπορούμε να αλλάξουμε όλα τα στοιχεία της αίτησης, να της αλλάξουμε την κατάσταση που προείπαμε, να την αποθηκεύσουμε ή τέλος να την διαγράψουμε.

Owner/Manager Information

Contact Name	Contact Email
Patricia Metz	Patricia17@hotmail.com
Owner Name	Owner ID
Patricia Metz	2321321
Manager Name	Manager ID
Patricia Metz	2321321
Address	
69680 Jaclyn Island	

Vessel Main Particulars

Vessel Name	Vessel Type
CINDY	Container
IMO	Year of Build
9004499	2007
Flag	GT (Gross Tonnage)
St. Vincent and the Grenadines	6030
Port of Registry	LO (Length Overall)
Belize City	106.42
Notes	
Add additional information	
Services	Status
Typical, Triple	Under Process

[Save](#) [Cancel](#) [Delete](#)

Εικόνα 52 Edit Request

Εν συνεχεία, βλέπουμε τη δεύτερη φάση της αίτησης που αφού πρώτα έχει γίνει Materialized, περνάει αυτόματα σαν Case στην κατηγορία Cases με κατάσταση In Progress.

Vessel Name	Imo	Owner Name	Flag	Services	Fleet Number	Manager	Status	Action
CHOUNGLI EXPRESS	9804526	Tiffany Cummings	Belize	Double	10	Queen	Active Case	
RISING EAGLE	9073672	Maryse Becker	St. Vincent and the Grenadines	Triple	11	Queen	Active Case	
SEA BRIDLE	9047001	Efrain Feil	Belize	Double	12	Queen	Active Case	
Maria P	8417845	Pawn	Greece	Typical	13	Queen	Active Case	
IOANNIS	8417245	Pawn	Greece	All	14	Queen	Active Case	

Rows per page: 5 1-5 of 14

Εικόνα 53 In Progress Cases

Με τον ίδιο τρόπο, έχουμε την σελίδα In Progress, όπου βλέπουμε μέσα όλα τα Cases, τα οποία διαφοροποιούνται από τις προηγούμενες προαναφερθέντες αιτήσεις, με

διαφορετικές καταστάσεις/status. Μπορούμε να δούμε την κατάσταση στην οποία βρίσκεται κάθε Case στην ειδική στήλη Status.

Αυτή η κατάσταση είναι και η κατάσταση η οποία πρέπει να γνωρίζει ο πελάτης για την πορεία της αίτησης του, και αυτή μπορεί να κάνει Track μέσω του Tracking System.

Όπως και προηγουμένως, πατώντας το κουμπί πληροφοριών θα οδηγηθούμε στην σελίδα προεπισκόπησης του εκάστοτε Case και από εκεί έχουμε τις ίδιες δυνατότητες με τα Requests. Δηλαδή, να επεξεργαστούμε, κατεβάζουμε και να διαγράψουμε το Case μας.

Μπαίνοντας στην σελίδα προεπισκόπησης, θα δούμε πως υπάρχει μια επιπλέον πληροφορία πάνω δεξιά. Εκεί βρίσκεται το Tracking ID, που όπως είπαμε χρειάζεται να δοθεί στον πελάτη με σκοπό την παρακολούθηση της πορείας της αίτησης του.

Case for Vessel: CHOUNGUI EXPRESS	
Active Case ID	INSB No
15	10
Contact Name	Vessel Name
Tiffany Cummings	CHOUNGUI EXPRESS
Contact Email	Vessel Type
Tiffany38@yahoo.com	Container
Owner Name	IMO
Tiffany Cummings	9804526
Owner ID	Year of Build
2154623	2018
Manager Name	Flag
Tiffany Cummings	Belize
Manager ID	GT (Gross Tonnage)

Εικόνα 54 Case Info

Application	Country Of Origin
Date Of Application	Date Of Last Follow Up
Reference Of Incoming Message	Reference Of Technical Instructions
Reference Of Outgoing Message	Name Of Attending Surveyor
Consideration of Survey Reports	Considerator
Type Of Service	Date Of Completion
Services	Status Of Application
Double	Active Case

Εικόνα 55 Case Additional Info

Εικόνα 56 Edit Case

Η ιδιαιτερότητα των Cases είναι πως, ενώ στα Requests γινόταν αλλαγή της κατάστασης όταν ολοκληρώνονται οι διαδικασίες διαπραγμάτευσης, εδώ έχουμε αλλαγή της κατάστασης μόλις γίνει η περισυλλογή όλων των δικαιολογητικών που απαιτεί η εταιρεία από τον πελάτη.

Status Of Application

Εικόνα 57 Case – Status of Application

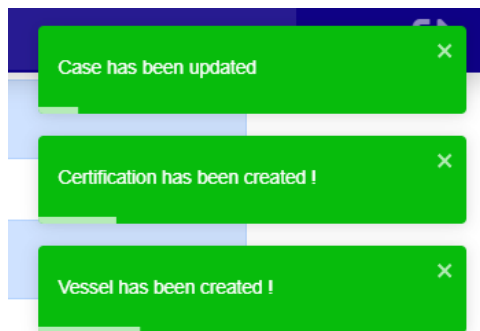
Καθ' αυτό το τρόπο, μόλις η εταιρεία συλλέξει μια κατηγορία δικαιολογητικών, ενημερώνει την κατάσταση ή αλλιώς Status of Application στην επόμενη, περιμένοντας τα υπόλοιπα δικαιολογητικά.

Vessel Name	Imo	Owner Name	Flag	Services	Fleet Number	Manager	Status	Action
IOANNIS	8417245	Pavni	Greece	All	14	Queen	Completed Case	
NEW CHALLENGE	9129627	Rook	Honduras	All	22	King	Completed Case	
ZUZANNA	8203555	Rook	Madagascar	Double	23	King	Completed Case	

Rows per page: 5 | 1-3 of 3

Εικόνα 58 Completed Cases

Όταν το Case πάρει την κατάσταση Completed Case, θα μπει στο αρχείο στην σελίδα Completed.



Σε αυτό το σημείο, γίνεται αυτόματα η καταχώρηση του Πλοίου εφόσον αυτό δεν υπάρχει ήδη καταχωρημένο στην εταιρεία, και επίσης γίνεται αυτόματα και η έκδοση του πιστοποιητικού που ζητήθηκε.

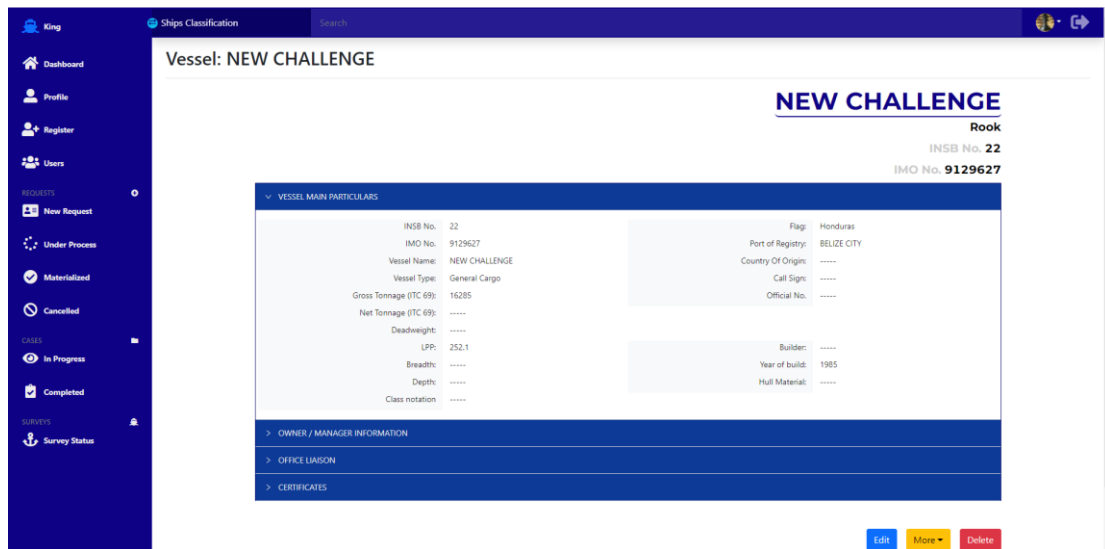
Εικόνα 59 App Notifications

Vessel Name	Imo	Fleet Number	Flag	Manager	Action
ZUZANNA	8203555	23	Madagascar	King	
NEW CHALLENGE	9129627	22	Honduras	King	
IOANNIS	8417245	14	Greece	King	

Rows per page: 5 | 1-3 of 3

Εικόνα 60 Survey Status

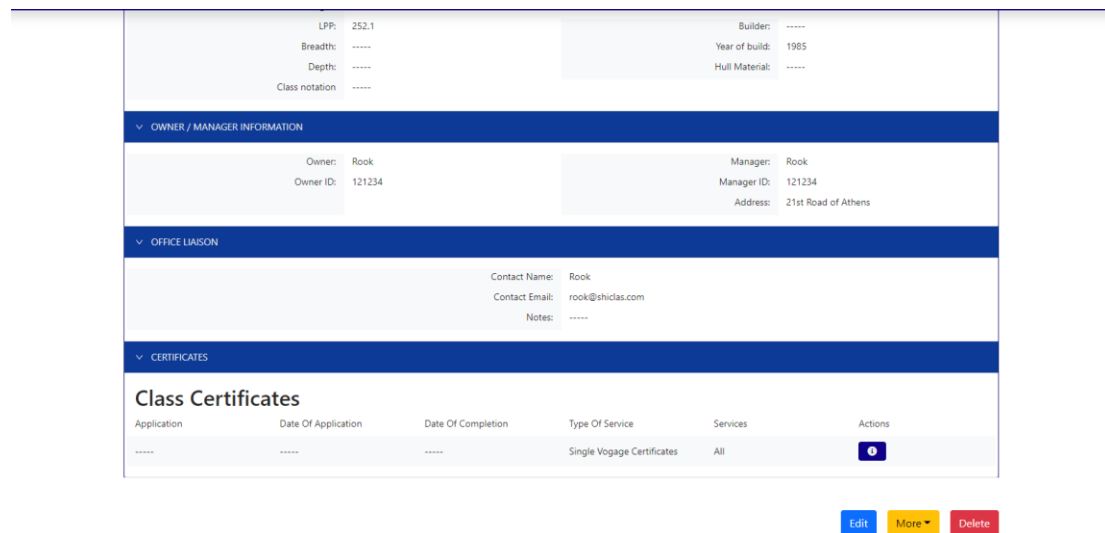
Πηγαίνοντας λοιπόν στην τελευταία μας σελίδα με όνομα Survey Status, θα δούμε όλα τα καταχωρημένα πλοία στην εταιρεία. Εδώ θα πατήσουμε το κουμπί πληροφοριών για να οδηγηθούμε στην προεπισκόπηση του κάθε πλοίου.



Εικόνα 61 Vessel Info

Εδώ θα δούμε την συνολική εικόνα του πλοίου με όλες τις διαθέσιμες πληροφορίες του και με την δυνατότητα επεξεργασίας, κατεβάσματος του pdf ή και διαγραφή του πλοίου.

Ίσως όμως το πιο σημαντικό κομμάτι είναι η τελευταία καρτέλα, όπου θα δούμε όλα τα πιστοποιητικά που έχουν εκδοθεί για το συγκεκριμένο πλοίο, με τις ημερομηνίες του κάθε πιστοποιητικού, αλλά και όλες τις υπόλοιπες πληροφορίες που το αφορούν αφού πρώτα πατήσουμε για μια τελευταία φορά το κουμπί των πληροφοριών του εκάστοτε πιστοποιητικού.



Εικόνα 62 Vessel Additional Info

Certificate: NEW CHALLENGE

Certificate Information

Application	NEW CHALLENGE	Vessel IMO	9129627
Date Of Application	20/02/2022	Date Of Last Follow Up	21/02/2022
Reference Of Incoming Message	-----	Reference Of Technical Instructions	-----
Reference Of Outgoing Message	-----	Name Of Attending Surveyor	-----
Consideration of Survey Reports	Reports Considered	Considerator	-----
Type Of Service	Single Vogage Certificates	Date Of Completion	25/02/2022
Services		Status Of Application	

Εικόνα 63 Vessel Certificate

Edit Certificate

Certificate Information

Application	NEW CHALLENGE	IMO	9129627
Date Of Application	20/02/2022	Date Of Last Follow Up	21/02/2022
Reference Of Incoming Message		Reference Of Technical Instructions	
Reference Of Outgoing Message		Name Of Attending Surveyor	
Consideration of Survey Reports	Reports Considered	Considerator	
Type Of Service	Single Vogage Certificates	Date Of Completion	25/02/2022
Services	All x	Status Of Application	Completed Case

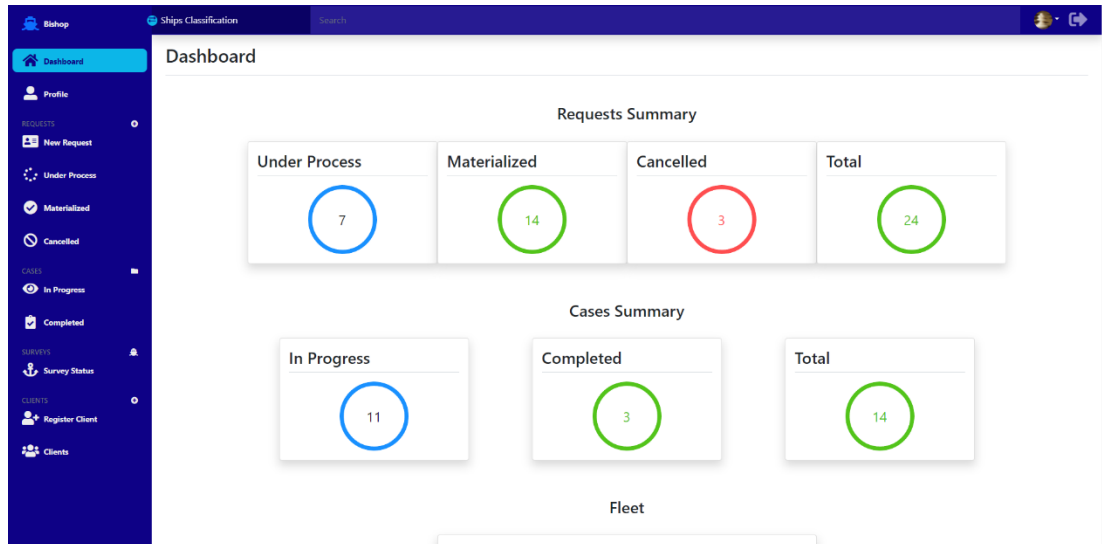
Save Cancel Delete

Εικόνα 64 Edit Certificate

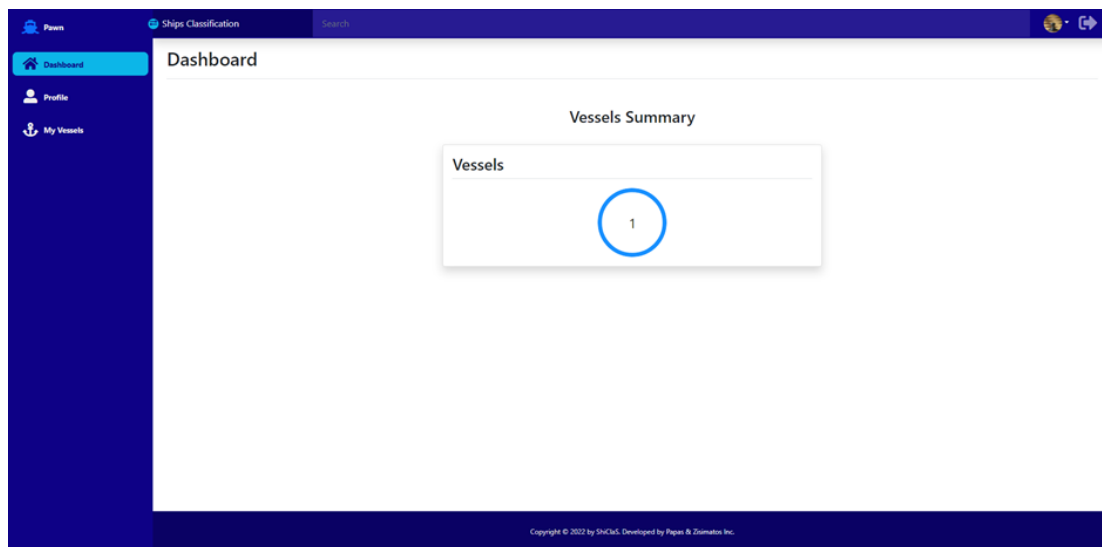
Τέλος, να διευκρινίσουμε πως όλα αυτά τα βήματα τα βλέπαμε από την σκοπιά ενός χρήστη με δικαιώματα Admin.

Αυτό σημαίνει πως είχαμε το εκατό τις εκατό όλων των δυνατοτήτων της εφαρμογής. Αντιστοίχως, οι επόμενες δύο κατηγορίες χρήστη με δικαιώματα είτε Manager είτε Client έρχονται με λιγότερες δυνατότητες για λόγους ασφαλείας. Παραδείγματος χάριν, ο Manager δεν έχει την δυνατότητα να διαγράψει τίποτα από το σύστημα. Αυτό συμβαίνει γιατί πρέπει να υπάρχει ο έλεγχος από επιλεγμένα άτομα για τέτοιες δυνατότητες όπως της διαγραφής, μιας και είναι μια πολύ σπάνια ενέργεια που μπορεί να προκαλέσει αναστάτωση στην λειτουργία μιας εταιρείας. Μια αίτηση για παράδειγμα, μπορεί να γίνει Cancelled από τον Manager μιας και είναι πολύ συνηθισμένο, αλλά το να διαγραφεί τελείως από το σύστημα θα γίνει μόνο σε περίπτωση λάθους.

Έχοντας αυτό υπόψη, ο Client δεν μπορεί να έχει ενέργειες που έχουν να κάνουν με την διαχείριση αιτήσεων, παρά μόνο με την προεπισκόπηση και κατέβασμα δικών του αιτήσεων ή πλοίων.



Εικόνα 65 Manager Role Interface



Εικόνα 66 Client Role Interface

3.2.2 ΥΠΗΡΕΣΙΑ/ΔΙΑΚΟΜΙΣΤΗΣ

Η παρούσα εφαρμογή είναι σύγχρονη και πολυεπίπεδη που τρέχει σε πραγματικό χρόνο, με δυναμικά δεδομένα και πολλαπλούς ταυτόχρονους χρήστες. Για αυτές τις ανάγκες λοιπόν, χρειαζόμασταν κάτι πέρα από μια στατική σελίδα, έναν ζωντανό οργανισμό.

Η Υπηρεσία ή αλλιώς Service, είναι ένας ασταμάτητος server ο οποίος διαχειρίζεται όλες τις συνδέσεις και requests από το περιβάλλον του χρήστη έως και την βάση δεδομένων. Είναι ο συνδεδεμένος κρίκος που κάνει τον χρήστη να αλληλεπιδρά με τον

περιβάλλον και να έχει σε πραγματικό χρόνο αληθινά δεδομένα από την βάση δεδομένων. Διαχειρίζεται επίσης και όλη την ασφάλεια της εφαρμογής αλλά και την αυθεντικότητα των χρηστών.

Με αυτόν τον τρόπο, στήσαμε μια υπηρεσία σε NodeJS και με βιβλιοθήκες της Express. Χρησιμοποιήσαμε Routing System ώστε ο χρήστης μέσω της εφαρμογής, κάνοντας τα κατάλληλα GET-POST-PUT-DELETE Requests να αλληλεπιδρά με την υπηρεσία, δίνοντας της δεδομένα και λαμβάνοντας το αντίστοιχο αποτέλεσμα.

```
1 //Import Libraries
2 import express from 'express';
3 import formidable from 'express-formidable';
4
5 //middleware
6 import {requireSignIn} from '../middlewares';
7
8 //Import Controllers
9 import { request, requestExisting, requests, read, update, remove } from '../controllers/requestIdNo';
10
11 const router = express.Router();
12
13
14 router.post(["/requestidnoexisting",requireSignIn, formidable(), requestExisting]);
15 router.post("/requestidno",requireSignIn, formidable(), request);
16 router.get("/idnorequests",requireSignIn, requests);
17 router.get("/idnorequest/:requestId", read);
18 router.put("/updateidnorequest/:requestId", requireSignIn, formidable(), update);
19 router.delete("/deleteidnorequest/:requestId", requireSignIn, formidable(), remove);
20
21 module.exports = router;
```

Εικόνα 67 Service router paths

Μόλις η υπηρεσία λάβει το Request, θα χρησιμοποιήσει κάποια συνάρτηση που της έχουμε φτιάξει ή αλλιώς Controller. Αυτός ο Controller είναι υπεύθυνος για την ενέργεια που πρέπει να γίνει ανάλογα το Request, και για να μας επιστρέψει πίσω τα σωστά αποτελέσματα.

Παρακάτω βλέπουμε έναν Controller ο οποίος μπορεί να δημιουργήσει (POST) στην βάση, να διαβάσει (GET), να επεξεργαστεί υπάρχοντα δεδομένα (UPDATE) και τέλος να διαγράψει (DELETE).

```
//READ ONE
export const readVessel = async (req, res) => {
  let vessel = await VesselsData.findById(req.params.vessel)
    .select("")
    .exec();
  console.log("Singe Request:", readVessel);
  res.json(vessel);
};

//UPDATE
export const updateVessel = async (req, res) => {
  try {
    let fields = req.fields;

    let data = {...fields}

    let updated = await VesselsData.findByIdAndUpdate(req.params.vessel, data, {new: true,});
    res.json(updated);
  } catch (err) {
    console.log(err);
    res.status(400).send('Vessel Update Failed.Try again. ');
  }
}

//DELETE
export const deleteVessel = async (req, res) => {
  let removed = await VesselsData.findByIdAndDelete(req.params.vessel).exec();
  res.json(removed);
}
```

Εικόνα 68 Service Controllers

```
Server is running on port: 5000
DB Connected
MATCH PASSWORD true
COMPARE PASSWORD IN LOGIN ERR null
POST /app/login 200 804.537 ms - 356
GET /app/readActiveCases 200 215.732 ms - 11272
GET /app/readVessels 200 285.567 ms - 2816
GET /app/idnorequests 200 384.863 ms - 14313
Singe User: [AsyncFunction: read]
GET /app/users/60f15b1429e7c23a7cd86bf3 200 463.756 ms - 500102
GET /app/user/image/60f15b1429e7c23a7cd86bf3 200 1216.884 ms - 140416
█
```

Εικόνα 69 Service GET Requests

Ένα παράδειγμα για το πως φαίνονται τα εισερχόμενα Request στην υπηρεσία μας. Τέλος, η υπηρεσία χρησιμοποιεί διάφορα μοντέλα που της έχουμε δημιουργήσει, ώστε να μπορεί να αλληλεπιδρά με την βάση δεδομένων. Σε αυτά τα μοντέλα χαρακτηρίζονται επακριβώς όλες η λεπτομέρεια που μπορεί να έχει ένας πίνακας στην βάση, όπως αυτός των Πλοίων.

Με αυτόν τον τρόπο γίνεται πιο εύκολη η διαχείριση αλλά και ο έλεγχος ότι όλα τα δεδομένα μας οδηγούντε από και προς την βάση σωστά.

```
server > models > @ vesselsData.js > ...
1 import mongoose from 'mongoose';
2
3 const { Schema } = mongoose;
4 const { ObjectId } = mongoose.Schema;
5
6 const VesselsDataSchema = new Schema({
7
8   insbMo: {
9     type: String,
10    trim: true,
11  },
12  imo: {
13    type: String,
14    trim: true,
15  },
16  vesselName: {
17    type: String,
18    trim: true,
19  },
20  vesselType: {
21    type: String,
22    trim: true,
23  },
24  flag: {
25    type: String,
26    trim: true,
27  },
28  portOfRegistry: {
29    type: String,
30    trim: true,
31  },
32  countryOfOrigin: {
33    type: String,
```

Εικόνα 70 Service DB Models

3.2.3 ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ

Αναφορικά με την βάση δεδομένων, αποφασίσαμε να χρησιμοποιήσουμε Μη Σχεσιακή Βάση Δεδομένων (NoSQL) και συγκεκριμένα την MongoDB η οποία έχει κερδίσει απίστευτο έδαφος τα τελευταία χρόνια σε σχέση με τις μέχρι τώρα διαδεδομένες Σχεσιακές Βάσης όπως η MySQL.

Μια βάση δεδομένων NoSQL (αρχικά αναφερόμενη σε "non-SQL" ή "non-relational") παρέχει έναν μηχανισμό αποθήκευσης και ανάκτησης δεδομένων που μοντελοποιείται με άλλα μέσα από τις σχέσεις πινάκων που χρησιμοποιούνται σε σχεσιακές βάσεις δεδομένων.

Αυτή ακριβώς η μοντελοποίηση βοηθάει τους προγραμματιστές να αποθηκεύουν τα δεδομένα στην βάση σε μορφές που ήδη χρησιμοποιούνται από τις γλώσσες προγραμματισμού την ώρα της ανάπτυξης. Με αυτόν τον τρόπο ο προγραμματιστής γλιτώνει κόπο, χρόνο, χρήμα, γράφει λιγότερες γραμμές κώδικα που αυτό με τη σειρά του οδηγεί σε γρηγορότερους χρόνους ανάπτυξης για όλη την εταιρεία και τέλος μειώνονται σημαντικά τα σφάλματα και κατ επέκταση τα bugs.

Επιπλέον, αυτός ο τύπος βάσης συνήθως έχει πολύ ευέλικτα σχήματα. Ένα ευέλικτο σχήμα επιτρέπει εύκολα αλλαγές στη βάση δεδομένων καθώς αλλάζουν οι απαιτήσεις.

Μπορεί να επαναληφθεί γρήγορα και συνεχώς να ενσωματώνονται νέες δυνατότητες στην εφαρμογής πιο γρήγορα.

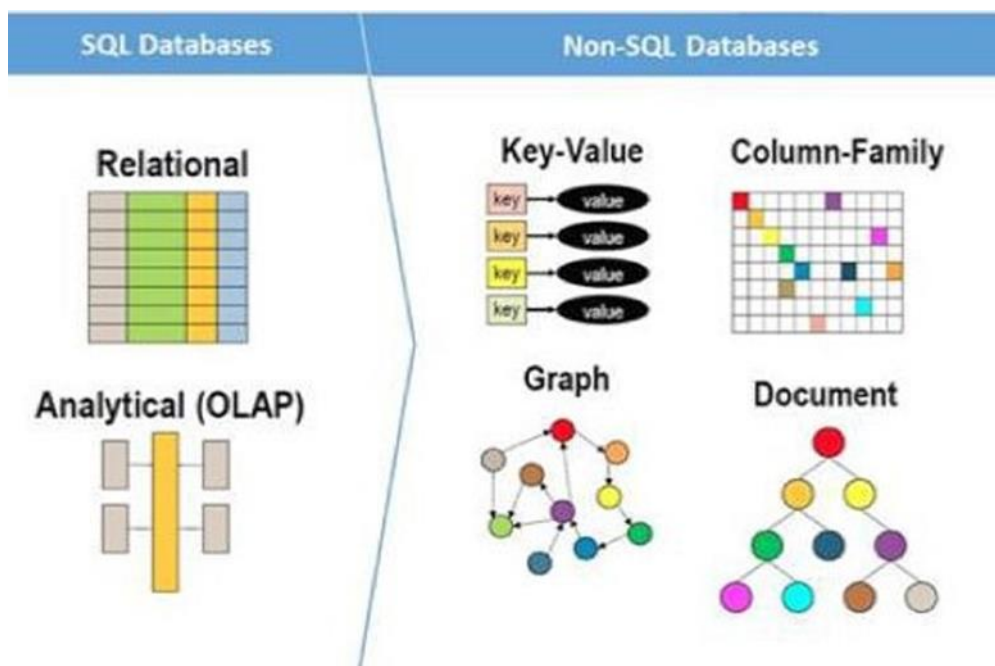
Όλη η επικοινωνία από και προς την βάση, όπως και οι αναζητήσεις σε αυτήν είναι ακόμα γρηγορότερες λόγω της πιο απλοϊκής δομής της.

Τέλος, αυτή ακριβώς η απλοϊκή δομή της είναι πολύ σημαντική και για την μελλοντική επέκταση της εφαρμογής και κατ επέκταση της βάσης, καθώς επιτρέπει την εύκολη επέκταση της βάσης χωρίς να χρειάζονται να ξοδευτούν πολλοί πόροι και χρήμα, αλλά

όταν ανέβουν οι ανάγκες της εφαρμογής να γίνει τότε μια επέκταση του υπάρχοντος συστήματος και όχι μια ολική αναβάθμιση του.

	SQL	NoSQL
Database Type	Relational Databases	Non-relational Databases / Distributed Databases
Structure	Table-based	<ul style="list-style-type: none"> • Key-value pairs • Document-based • Graph databases • Wide-column stores
Scalability	Designed for scaling up vertically by upgrading one expensive custom-built hardware	Designed for scaling out horizontally by using shards to distribute load across multiple commodity (inexpensive) hardware
Strength	<ul style="list-style-type: none"> • Great for highly structured data and don't anticipate changes to the database structure • Working with complex queries and reports 	<ul style="list-style-type: none"> • Pairs well with fast paced, agile development teams • Data consistency and integrity is not top priority • Expecting high transaction load

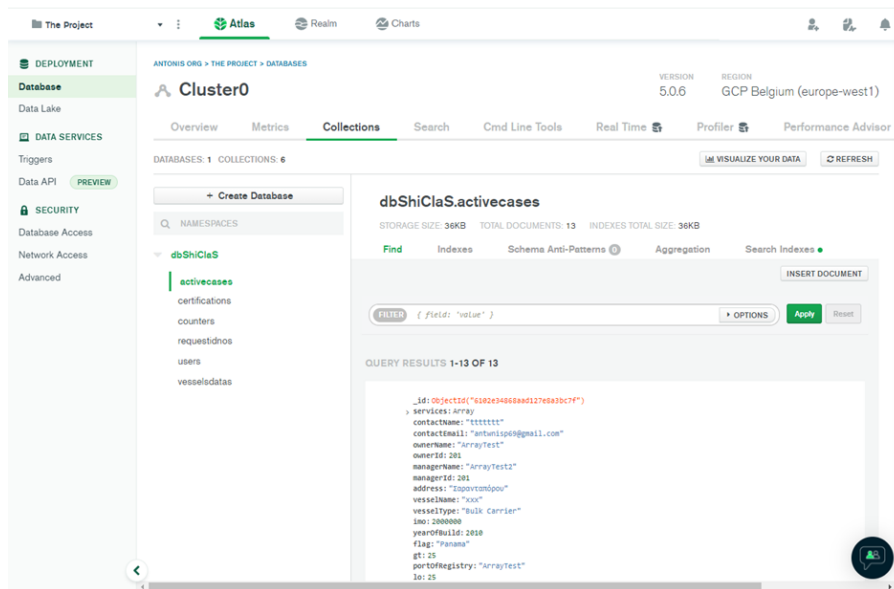
Εικόνα 71 SQL vs NoSQL



Εικόνα 72 DB Diagrams

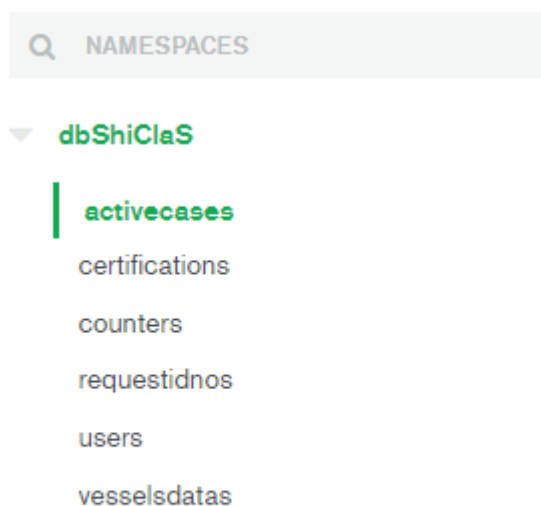
Κατά κύριο λόγο χρησιμοποιήσαμε το MongoDB Atlas, το οποίο είναι ένα Online Σύστημα Διαχείρισης Βάσεων Δεδομένων.

Αυτό μας βοήθησε ώστε να έχουμε ανα πάσα στιγμή ταυτόχρονα τα ίδια δεδομένα, γλιτώνοντας μας έτσι χρόνο αλλά και λιγότερα προβλήματα στην ανάπτυξη της εφαρμογής.



Εικόνα 73 MongoDB Atlas Main Interface

Η βάση αποτελείται από Collections, η οποίες είναι οι συλλογές της εφαρμογής μας, κάτι αντίστοιχο των πινάκων στους Σχεσιακές Βάσης.



Οι συλλογές αυτές χαρακτηρίζουν κάτι συγκεκριμένο μέσα στην εφαρμογή, όπως για παράδειγμα η συλλογή Certifications χαρακτηρίζει τα Πιστοποιητικά των πλοίων αλλά και η συλλογή Users χαρακτηρίζει τους εγγεγραμμένους χρήστες.

Τέλος, η δομή ενός Collection είναι απλή, με καθένα από αυτά να περιέχει τα απαραίτητα στοιχεία και ότι πληροφορίες χρειάζεται για να είναι λειτουργικό.

Εικόνα 74 Database Collections

```
> _id: ObjectId("60f16a7129e7c23a7cd86cd0")
  > services: Array
    > contactName: "Sotirios Papakonstantis"
      contactEmail: "sotirispapako@hotmail.gr"
      ownerName: "Sotirios Papakonstantis"
      ownerId: 9328
      managerName: "Papakonstantis Co"
      managerId: 21387
      address: "Manhattan 2187 , USA"
      vesselName: "Silver Eagle"
      vesselType: "Cargo"
      imo: 875336
      yearOfBuild: 2005
      flag: "FR"
      gt: 213213
      portOfRegistry: "OMGPort"
      lo: 1232131
      notes: "Δεν έχει απαντήσει απο περσι"
      status: "Materialized"
      postedBy: ObjectId("60f15b1429e7c23a7cd86bf3")
      createdAt: 2021-07-16T11:16:01.710+00:00
      updatedAt: 2021-09-08T11:07:30.097+00:00
      __v: 0
      reqId: 17
```

Εικόνα 75 Request DB Example

3.3 ΕΝΩΣΗ ΚΑΙ ΑΠΟΤΕΛΕΣΜΑ

Χρησιμοποιώντας λοιπόν αυτούς τους τρεις πυλώνες της ανάπτυξης μιας εφαρμογής, καταφέραμε και συνδέσαμε με όμορφο αλλά κυρίως λειτουργικό τρόπο τα απαραίτητα κομμάτια, ώστε η εφαρμογή μας να φτάσει σε αυτό το σημείο.

Αυτή τη στιγμή μέσα από την εφαρμογή μπορεί να ολοκληρωθεί η τυπική ροή μιας εταιρείας, από την αρχή μιας αίτησης, στην παρακολούθηση της, έως και την ολοκλήρωση της με την έκδοση του πιστοποιητικού.

Παρακάμπτοντας μεγάλα εμπόδια της γραφειοκρατίας αλλά προσθέτοντας και δυνατότητες που δεν θα μπορούσαν να υπάρχουν αλλιώς, καθιστούν έναν ολοκληρωμένο κύκλο ζωής της εφαρμογής που από εδώ και πέρα μπορεί μόνο να επεκτείνεται, να μεγαλώνει και να εξελίσσεται ραγδαία προς τα πάνω.

ΣΤΟΧΟΙ ΚΑΙ ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ

Οι στόχοι μας τέθηκαν από την αρχή, με βασικότερο όλων η εφαρμογή να έχει μια ολοκληρωμένη ροή από την αρχή μέχρι το τέλος, σύμφωνα με τις ανάγκες μιας εταιρείας νηογνώμονα.

Η λειτουργικότητα ήταν αυτό που μας ενδιέφερε και το κατά πόσο λύνουμε τα προβλήματα που συναντώνται στις υπάρχουσες διαδικασίες.

Παραθέτουμε παρακάτω τους βασικούς στόχους της εφαρμογής μας:

1. Να μπορεί να γίνει μια ολοκληρωμένη διαδικασία στον κύκλο ζωής μιας αίτησης.
2. Απλοποίηση της διαδικασίας για τους χρήστες.
3. Μείωση στο ελάχιστο της αλληλεπίδρασης του χρήστη με την διαδικασία έκδοσης πιστοποιητικού.
4. Προσθήκη και διαχωρισμός χρηστών ανάλογα με τις δυνατότητες και τα δικαιώματα του καθενός.
5. Δυνατότητα παρακολούθησης της πορείας της αίτησης από τον πελάτη.
6. Περιβάλλον ειδικά σχεδιασμένο για τον πελάτη ώστε να παρακολουθεί τα πλοία που έχει στη διαχείριση του.
7. Κατασκευή ενός όμορφου γραφικού περιβάλλοντος.
8. Κατασκευή μιας λειτουργικής υπηρεσίας/server.
9. Κατασκευή μιας λειτουργικής βάσης δεδομένων
10. Άμεση εύρεση κατάστασης υπόθεσης για τους ενδιαφερόμενους χρήστες.
11. Διαχωρισμό υποθέσεων που δεν προχώρησαν σε συμφωνία.
12. Εύκολη αναζήτηση βάση υπευθύνου υπόθεσης, όνομα πλοίου κ.α.
13. Sorting με βάση όνομα πλοίου, σημαίας, IMO κ.α.
14. Αυτοματισμούς και ψηφιοποίηση όλων των περιττών πλέον διαδικασιών που περιέχουν ενημέρωση μεταξύ τμημάτων, πρωτοκόλληση, αρχειοθέτηση.

Μελλοντικές επεκτάσεις

1. Οι πελάτες μέσω της αρχικής ιστοσελίδας μας να καταθέσουν τα request τους ηλεκτρονικά από μονοί τους με μια απλή συμπλήρωση μιας φόρμας με κάποια βασικά στοιχεία του πλοίου που θα ενδιαφέρονταν να κάνουν επιθεώρηση. Καθώς το request θα πέρναγε αυτόματα στην καρτέλα των Request μέσα στην πλατφόρμα χωρίς την αλληλεπίδραση κάποιου υπαλλήλου.
2. Ένα αυτόματο σύστημα ενημέρωσης πελατών για το ποτέ λήγουν τα πιστοποιητικά τους ώστε να κρατάμε ενήμερους τους πελάτες και να μην χρειάζεται οι υπάλληλοι να σπαταλάνε χρόνο σε email-reminders ενώ θα μπορούσαν να χρησιμοποιήσουν το χρόνο τους για κάτι πιο παραγωγικό.
3. Άλλη μια μεγάλη μελλοντική επέκταση που έχουμε σκεφτεί να κάνουμε είναι για τους ίδιους τους επιθεωρητές μέσω της πλατφόρμας να υπάρχει ένας χώρος όπου οι επιθεωρητές θα ενημερώνονται για το πότε έχουν επιθεώρηση, που θα βρίσκεται το πλοίο για την επιθεώρηση και πόσο καιρό θα διαρκέσει καθώς και οι οδηγίες για την επιθεώρηση να βρίσκονται εκεί για να τις κατεβάσουν/διαβάσουν, ώσπου οι επιθεωρητές με την ολοκλήρωση της επιθεώρησης να ενημερώνουν και να καταθέτουν τα απαραίτητα reports μέσω της πλατφόρμας.

ΣΥΜΠΕΡΑΣΜΑΤΑ

Αν και σε αρχικό στάδιο, η εφαρμογή μας πληροί όλους τους αρχικούς μας στόχους αλλά αρκετά και από τα κριτήρια που χρειάζεται μια σημερινή εταιρεία για να την ενσωματώσει στην ήδη υπάρχουσα αλυσίδα της. Έχοντας αυτό κατά νου, η εφαρμογή δεν θεωρείται ένα ολοκληρωμένο έργο, αλλά ένα πρότζεκτ το οποίο τώρα ξεκινά να εξελίσσεται. Υπάρχουν αρκετές δυνατότητες που θα θέλαμε να προσθέσουμε σαν μελλοντικές επεκτάσεις που θα την πάνε ένα βήμα παρακάτω και θα την κάνουν να ξεχωρίσει.

Υπάρχει επιτακτική ανάγκη σε όλους τους κλάδους σήμερα να εκσυγχρονιστούν και να ψηφιοποιηθούν όλες ή έστω κάποιες βασικές εταιρικές τους δραστηριότητες. Ως εκ τούτου, τέτοιου είδους εφαρμογές μπορούν να λύσουν τα χέρια σε μεγάλο μέρος της κοινωνίας μας και να την οδηγήσουν γρηγορότερα σε αυτό που αποκαλούμε σήμερα ψηφιακή εποχή.

Η εργασία αυτή μας βοήθησε στο να μελετήσουμε την διαδικασία μιας επιχείρησης νηογνώμονα εις βάθος και να αποκτήσουμε γνώσεις στην μελέτη και ανάλυση μιας εταιρείας. Τις γνώσεις αυτές μπορούμε να τις χρησιμοποιήσουμε σαν μελλοντικά εργαλεία σε εταιρείες και επιχειρήσεις που τυγχάνουν να έχουν παρόμοιες ανάγκες.

Ελπίζουμε, αυτή η πτυχιακή να γίνει φάρος στην επίτευξη του στόχου μας και να φωτίσει τον δρόμο για την επερχόμενη επαγγελματική μας πορεία.

ΑΝΑΦΟΡΕΣ

- Convention, S. (1980, May 25). *SOLAS Convention*. Ανάκτηση από https://en.wikipedia.org/wiki/SOLAS_Convention
- Dahl, R. (2022, February 10). *Node.js*. Ανάκτηση από <https://en.wikipedia.org/wiki/Node.js>
- Doyle, M. (2009). *Beginning PHP 5.3 (Wrox Programmer to Programmer)*.
- Felke-Morris. (2019). *Basics of Web Design: HTML5 & CSS3, 5th Edition*. Pearson Education.
- freeCodeCamp.org. (2019, April 30). *Learn HTML5 and CSS3 From Scratch - Full Course*. Ανάκτηση από <https://www.youtube.com/watch?v=mU6anWqZJcc>
- Friedlein, A. (2000). *Web Project Management: Delivering Successful Commercial Web Sites*. Morgan Kaufmann.
- IACS. (χ.χ.). *International Association of Classification Societies*. Ανάκτηση από https://en.wikipedia.org/wiki/International_Association_of_Classification_Societies
- ILO. (1919, April 11). *International Labour Organization*. Ανάκτηση από https://en.wikipedia.org/wiki/International_Labour_Organization
- IMO. (1948, March 17). *International Maritime Organization*. Ανάκτηση από https://en.wikipedia.org/wiki/International_Maritime_Organization
- Inc., M. (2021, December 6). *MongoDB*. Ανάκτηση από <https://en.wikipedia.org/wiki/MongoDB>
- Keith, Jeremy and Andrew, Rachel. (χ.χ.). *HTML5 For Web Designers*. 2016: A Book Apart.
- MARPOL. (1983, October 2). *International Convention for the Prevention of Pollution from Ships*. Ανάκτηση από https://en.wikipedia.org/wiki/MARPOL_73/78
- Media, T. (2021, June 09). *Bootstrap 5 Crash Course | Website Build & Deploy*. Ανάκτηση από <https://www.youtube.com/watch?v=4sosXZsdy-s>
- Media, T. (2021, January 18). *React JS Crash Course*. Ανάκτηση από <https://www.youtube.com/watch?v=w7ejDZ8SWv8>
- Meiert, Jens Oliver. (2015). *The Little Book of HTML/CSS Coding Guidelines*.
- Mosh, P. w. (2018, February 22). *Node.js Tutorial for Beginners: Learn Node in 1 Hour*. Ανάκτηση από https://www.youtube.com/watch?v=TIB_eWDSMt4
- Mosh, P. w. (2021, March 30). *Docker Tutorial for Beginners [2021]*. Ανάκτηση από <https://www.youtube.com/watch?v=pTFZFxd4hOI>

- Nixon, R. (2012). *Learning PHP, MySQL, JavaScript, and CSS: A Step-by-Step Guide to Creating Dynamic Websites*. O'Reilly.
- Pfeiffer, Silvia, and Green, Tom. (2015). *Beginning HTML5 Media: Make the most of the new video and audio standards for the Web*. APress.
- Powers, S. (2011). *HTML5 Media Integrating audio and video with the Web*.
- Rhea, J. (2018). *A Beginner's Guide to Learning HTML (and Smacking Zombies Upside the Web Development)*. Undead Institute.
- Simplified, W. D. (2021, September 28). *MongoDB Crash Course*. Ανάκτηση από <https://www.youtube.com/watch?v=ofme2o29ngU>
- SocialNerds. (2019, September 16). *Docker, Τι Είναι & Γιατί το Χρειαζόμαστε, Tutorial*. Ανάκτηση από <https://www.youtube.com/watch?v=9f9AAZfZA78>
- Strauss, Roy and Hogan, Patrick. (2001). *Developing Effective Websites: A Project Manager's Guide*, Focal Press.
- Ullman, L. (2012). *PHP Advanced and Object-Oriented Programming: Visual QuickPro Guide*.
- Walke, J. (2021, March 22). *React (JavaScript library)*. Ανάκτηση από [https://en.wikipedia.org/wiki/React_\(JavaScript_library\)](https://en.wikipedia.org/wiki/React_(JavaScript_library))