**ΧΑΡΟΚΟΠΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ**

ΣΧΟΛΗ ΨΗΦΙΑΚΗΣ ΤΕΧΝΟΛΟΓΙΑΣ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΜΑΤΙΚΗΣ

ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΜΑΤΙΚΗΣ

ΚΑΤΕΥΘΥΝΣΗ 1η : Τεχνολογίες και Εφαρμογές Ιστού

**Τίτλος Εργασίας**

**ΠΡΟΒΟΛΗ ΤΡΟΧΙΩΝ ΚΑΙ ΚΑΤΗΓΟΡΙΑΣ ΔΡΑΣΤΗΡΙΟΤΗΤΑΣ ΠΛΟΙΩΝ ΣΕ ΔΙΑΔΡΑΣΤΙΚΟΥΣ, ΔΙΑΔΙΚΤΥΑΚΟΥΣ ΧΑΡΤΕΣ**

Διπλωματική Εργασία

**Όνομα φοιτητή**

**ΧΑΡΑ ΜΠΟΥΛΟΥΓΑΡΗ**

Αθήνα, Ιούλιος 2021

**ΧΑΡΟΚΟΠΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ**

ΣΧΟΛΗ ΨΗΦΙΑΚΗΣ ΤΕΧΝΟΛΟΓΙΑΣ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΜΑΤΙΚΗΣ

ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΜΑΤΙΚΗΣ

ΚΑΤΕΥΘΥΝΣΗ 1η : Τεχνολογίες και Εφαρμογές Ιστού

**Τριμελής Εξεταστική Επιτροπή**

**Τσερπές Κωνσταντίνος, Επίκουρος Καθηγητής,**

**Τμήμα Πληροφορικής & Τηλεματικής,**

**Χαροκόπειο Πανεπιστήμιο**

**Κουσιουρής Γεώργιος, Επίκουρος Καθηγητής,**

**Τμήμα Πληροφορικής & Τηλεματικής,**

**Χαροκόπειο Πανεπιστήμιο**

**Ξύδης Σωτήριος, Επίκουρος Καθηγητής,**

**Τμήμα Πληροφορικής & Τηλεματικής,**

**Χαροκόπειο Πανεπιστήμιο**

Η Μπουλούγαρη Χαρά,

δηλώνω υπεύθυνα ότι:

1. Είμαι ο κάτοχος των πνευματικών δικαιωμάτων της πρωτότυπης αυτής εργασίας και από όσο γνωρίζω η εργασία μου δε συκοφαντεί πρόσωπα, ούτε προσβάλει τα πνευματικά δικαιώματα τρίτων.
2. Αποδέχομαι ότι η ΒΚΠ μπορεί, χωρίς να αλλάξει το περιεχόμενο της εργασίας μου, να τη διαθέσει σε ηλεκτρονική μορφή μέσα από τη ψηφιακή Βιβλιοθήκη της, να την αντιγράψει σε οποιοδήποτε μέσο ή/και σε οποιοδήποτε μορφότυπο καθώς και να κρατά περισσότερα από ένα αντίγραφα για λόγους συντήρησης και ασφάλειας.

**ΕΥΧΑΡΙΣΤΙΕΣ**

Η παρούσα διπλωματική εργασία υλοποιήθηκε στα πλαίσια του μεταπτυχιακού προγράμματος σπουδών του τμήματος Πληροφορικής και Τηλεματικής στο Χαροκόπειο Πανεπιστήμιο το Νοέμβριο του 2020, αρχές Μαΐου και Ιουνίου του 2021.

Θα ήθελα να ευχαριστήσω όλους όσους με υποστήριξαν καθ’ όλη τη διάρκεια αυτής της εργασίας και κατ’ επέκταση και όλων των σπουδών μου. Πιο συγκεκριμένα θα ήθελα να ευχαριστήσω τον κ. Κωνσταντίνο Τσερπέ, επίκουρο καθηγητή στο Χαροκόπειο Πανεπιστήμιο, που υπήρξε ο επιβλέπων καθηγητής της παρούσας διπλωματικής για όλη την καθοδήγηση, βοήθεια που μου έδωσε με κάθε απίστευτο τρόπο και υποστήριξη που μου παρείχε και χάρης τον οποίο κατάφερα να μην το βάλω κάτω και να φτάσω έως αυτό το σημείο των σπουδών μου με υπομονή και μελέτη. Έτσι μου δόθηκε η ευκαιρία να γνωρίσω και να εξερευνήσω τεχνολογίες που ποτέ δεν μου πέρασε από το μυαλό πως θα κατάφερνα να διαχειριστώ! Επίσης τον ευχαριστώ και για το γεγονός ότι πίστεψε σε μένα όταν εγώ δεν πίστευα και στο ότι αναγνώρισε και εκτίμησε όλη μου την προσπάθεια που κατέβαλα ώστε να καταφέρω να φτάσω έως εδώ, που και πάλι ποτέ μου δεν φανταζόμουν πως θα πετύχαινα στο παρελθόν!

Επίσης, ευχαριστώ από την καρδιά μου τον παππού μου και τη γιαγιά μου για όλη την αγάπη και φροντίδα που μου έδωσαν ως παιδί και βοήθεια που ποτέ δεν μου στέρησαν σε όλα τα χρόνια σπουδών μου, μαθητικών και ακαδημαϊκών. Ελπίζω να με βλέπουν από εκεί ψηλά και να τους κάνω περήφανους κάθε στιγμή!

**ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ**

[Περίληψη στα Ελληνικά……………………………………………………………………………………………………………………..……….……7](#περιληψη)

[Περίληψη στα Αγγλικά………………………………………………………………………………………………………………………..…….……..8](#abstract)

[Κατάλογος Εικόνων…………………………………………………………………………………………………………………………………………..9](#εικονες)

[Κατάλογος Αποσπασμάτων Κώδικα…………………………………………………………………………………………………………...…..10](#αποσπασματακωδικα)

[Συντομογραφίες………………………………………………………….……………………………………………………………………….………....11](#συντομογραφιες)

**1 ΕΙΣΑΓΩΓΗ………………………………………………………….………………………………………………………………….12**

1.1 Η τεχνολογία στην καθημερινή ζωή του ανθρώπου……………………………………………………………...12

1.2 Αντικείμενο πτυχιακής…………………………………………………………………………………………………………..12

1.2.1 Ορισμός προβλήματος…………………….............................................................................12

1.2.2 Τρόπος επίλυσης προβλήματος…………………………………………………………………………..…….13

1.2.3 Σύντομη περιγραφή λογικής………………………………………………………………………………..…….14

1.3 Οργάνωση κειμένου……………………………………………………………………………………………………..……….15

[**2 ΑΝΑΣΚΟΠΗΣΗ ΤΡΕΧΟΥΣΑΣ ΚΑΤΑΣΤΑΣΗΣ (State of the Art)………………………………………………..…….16**](#ανασκοπηση)

[2.1 Παρόμοιες εφαρμογές στην αγορά…………………………………………………………………………………...16](#ανασκοπηση)

[2.1.1 Εισαγωγή……………………………………………………………………………………..…………………….16](#ανασκοπηση)

[2.1.2 Μια σύντομη γνωριμία με τις εφαρμογές της αγοράς………………………..……………..16](#ανασκοπηση)

[2.1.2.1 Τρόποι επίλυσης του προβλήματος της διαχείρισης χρόνου και γεγονότων από ήδη υπάρχουσες εφαρμογές στην αγορά…………………………………………………………………….……………17](#τροποςεπιλυσηςμεηδηυπαρχουσες)

[2.1.2.1.1 Τεχνικές με τις οποίες το Google Calendar επιτυγχάνει διαχείριση χρόνου και γεγονότων……………………………………………………………………………………………..………17](#τροποςεπιλυσηςμεηδηυπαρχουσες)

[2.1.2.1.2 Τεχνικές με τις οποίες το Outlook Calendar επιτυγχάνει διαχείριση χρόνου και γεγονότων…………………………………………………………………………………………..….……..17](#outlook)

[2.1.2.2 Παρατηρήσεις που προκύπτουν από τον τρόπο προσέγγισης του αυτόματου προγραμματισμού γεγονότων από τις υπάρχουσες εφαρμογές…………………………………...……18](#παρατηρησεις)

[2.1.3 Τρόπος επίλυσης με την προσθήκη επιπλέον χαρακτηριστικών λειτουργικότητας……………………………………………………………………………………………….…18](#επιπλεονχαρακτηριστικαέιτουργιας)

[2.2 Τεχνολογίες που χρησιμοποιήθηκαν για την υλοποίηση της διαδικτυακής εφαρμογής και λογισμικά…………………………………………………………………………………………………………………………………….………20](#τεχνολογιες)

[2.2.1 Node………………………………………………………………………………………………………………….….20](#τεχνολογιες)

[2.2.1.1 Παραδοσιακοί εξυπηρετητές που χρησιμοποιούν μια διεργασία ανά request..….21](#παραδοσιακοιεξυπηρετητες)

[2.2.1.2 Παραδοσιακοί εξυπηρετητές που χρησιμοποιούν Thread Pool……………………….……21](#threadpool)

[2.2.1.3 Γιατί NodeJS ?............................................................................................................22](#giatinode)

[2.2.2 Npm………………………………………………………………………………………………………………………24](#npm)

[2.2.3 Angular……………………………………………………………………………………………………………….…24](#npm)

[2.2.3.1 Γιατί Angular ?...........................................................................................................24](#giatiangular)

[2.2.3.2 Αρχιτεκτονική της Angular……………………………………………………………………………….……25](#archtectureofangular)

[2.2.4 Express…………………………………………………………………………………………………………..……..26](#express)

[2.2.5 MySQL……………………………………………………………………………………………………………..……28](#mysql)

[2.2.6 Xampp…………........................................................................................................…..29](#xampp)

[2.2.7 Visual Studio Code………………………………………………………………………………………….…….29](#xampp)

[2.2.8 Postman……………………………………………………………………………………………………………….30](#postman)

[2.2.9 Google Chrome…………………………………………………………………………………………………….30](#postman)

[**3 ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΣΥΣΤΗΜΑΤΟΣ………………………………………………………………………………………………..31**](#trito)

[3.1 Εισαγωγή…………………………………………………………………………………………………………………………….31](#trito)

[3.2 Συνδεσμολογία……………………………………………………………………………………………………………….…..31](#trito)

[3.2.1 UML Component model Diagram............................................................................31](#componentdiagram)

[3.2.2 Επεξήγηση διαγράμματος…………………………………………………………………………..………..33](#epeksigisidiagrammatos)

[3.2.3 Συσχέτιση συστατικών συνδεσμολογίας με απαιτήσεις συστήματος……………..…..34](#sisxetisistatikon)

[3.2.3.1 Τι είναι ένα use case diagram με λίγα λόγια ?........................................................34](#tieinaitousecase)

[3.2.3.2 Συσχέτιση των δύο παραπάνω διαγραμμάτων (απαιτήσεων-συστατικών)……….…35](#sisxetisidiagrammaton)

[3.2.3.2.1 Λίγα λόγια για τις απαιτήσεις που δεν σχετίζονται με κάποιο συστατικό της εφαρμογής……………………………………………………………………………………………………..36](#ligalogia)

[3.3 Λειτουργικότητα κάθε συστατικού και η επικοινωνία του με τα υπόλοιπα…………………….….38](#litourgikotitakathesistatikou)

[3.3.1 Μια σύντομη γνωριμία με τα Observables και τη βιβλιοθήκη Rxjs……………..………..42](#rxjs)

[**4 ΛΕΠΤΟΜΕΡΕΙΕΣ ΣΥΣΤΗΜΑΤΟΣ………………………………………………………………………………………….……….44**](#leptomereies)

[4.1 Εισαγωγή…………………………………………………………………………………………………………………….……….44](#leptomereies)

[4.2 Εγκατάσταση απαραίτητων τεχνολογιών και εργαλείων……………………………………………………..44](#installtechnologies)

[4.2.1 Εγκατάσταση NodeJS ………………………………………………………………………………………..…44](#installtechnologies)

[4.2.2 Εγκατάσταση Visual Studio……………………………………………………………………………..……45](#installvscode)

[4.2.2.1 Εξοικείωση και γνωριμία με τον editor……………………………………….…………45](#installvscode)

[4.2.3 Εγκατάσταση XAMPP…………………………………………………………………………………………….48](#installxamp)

[4.2.4 Εγκατάσταση Postman……………………………………………………………………………..…………..48](#installpostman)

[4.3 Δημιουργία νέου Project………………………………………………………………………………………………………49](#newproject)

[4.4 Περιγραφή Αλγορίθμων που χρησιμοποιήθηκαν και προγραμματιστικών μοντέλων που ακολουθήθηκαν……………………………………………………………………………………………………….………….52](#algorithms)

[4.4.1 Ο αλγόριθμος της φυσαλίδας……………………………………………………………………….………52](#algorithms)

[4.4.2 Το μοντέλο MVC……………………………………………………………………………………………………54](#mvc)

[4.4.2.1 Μια γνωριμία με το μοντέλο MVC………………………………………………………...54](#mvc)

[4.4.2.2 Η εφαρμογή του μοντέλου MVC στην εφαρμογή που υλοποιήθηκε…………………………………………………………………………………………………………56](#iplementmvc)

[4.4.3 Αρχιτεκτονική 3-Tier……………………………………………………………………………………………..57](#architecturetier)

[4.4.3.1 Η εφαρμογή της αρχιτεκτονικής αυτής στην εφαρμογή ημερολογίου…………………………………………………………………………………………………………58](#tier)

[4.5 Σημεία κώδικα και λογικής που αξίζουν να αναφερθούν………………………………………………….….58](#importantcode)

[4.5.1 Η λογική της ταξινόμησης (με βήματα)…………………………………………………………………59](#sortinglogic)

[4.5.2 Κομμάτια κώδικα ταξινόμησης με εικόνες και σχόλια…………………………………….......60](#codesnippets)

[4.5.3 Σερβίροντας το frontend κάνοντας κλήση στο backend……………………………………..…64](#serversnippets)

[**5 ΑΞΙΟΛΟΓΗΣΗ ΣΥΣΤΗΜΑΤΟΣ………………………………………………………………………………………………………65**](#aksiologisi)

[5.1 Εισαγωγή……………………………………………………………………………………………………………………………..65](#aksiologisi)

[5.2 Σενάρια Εκτέλεσης……………………………………………………………………………………………………………….65](#aksiologisi)

[5.2.1 Αρχικές συνθήκες και στόχοι σεναρίων εκτέλεσης……………………………………………....65](#aksiologisi)

[5.2.2 Ικανοποίηση απαιτήσεων και αντικειμενικών στόχων από το σύστημα……………...68](#ikanopiisiapaitiseon)

[5.2.3 Σύνοψη ικανοποίησης αντικειμενικών στόχων (checklist)……………………………..…….69](#checklist)

[**6 ΑΝΑΠΤΥΞΗ ΕΦΑΡΜΟΓΗΣ ΣΕ ΥΠΟΔΟΜΕΣ ΥΠΟΛΟΓΙΣΤΙΚΟΥ ΝΕΦΟΥΣ..…………………………………………70**](#DEPLOY)

[6.1 Εισαγωγή………………………………………………………………………………………………………………………….….70](#DEPLOY)

[6.2 Τεχνολογίες που θα χρειαστούν και εργαλεία……………………………………………………………….…….70](#DEPLOYTECHNOLOGIES)

[6.2.1 Εξοικείωση με τις έννοιες docker container και docker image……………………….…….74](#whatisdocker)

[6.2.2 Τι είναι το docker hub…………………………………………………………………………………………..74](#whatisdockerhub)

[6.2.3 Τι είναι το azure………………………………………………………………………………………………..….75](#whatisazure)

[6.2.4 Η λογική της αρχιτεκτονικής της φιλοξενίας της εφαρμογής στο cloud……………....76](#cloudlogicarchitecture)

[6.3 Εγκατάσταση επεκτάσεων (azure & docker) στο vs code και δημιουργία repository στο docker hub……………………………………………………………………………………………………..……………………………………78](#installextensions)

[6.4 Δημιουργία dockerFile και docker-Compose.yml και ανέβασμα στο azure web app for containers……………………………………………………..…………………………………………………………………….………………79](#dockerfile)

[**7 ΣΥΜΠΕΡΑΣΜΑΤΑ……………………………………………………………………………………………………………………..85**](#results)

[7.1 Σύνοψη της όλης υλοποίησης και αποτελέσματα επιτυχίας………………………………………….…….85](#results)

[7.2 Μελλοντική αναβάθμιση της εφαρμογής (επιπλέον λειτουργικότητα)………………………………………….86](#upgrade)

[7.3 Συνεισφορά στο ευρύ κοινό ………………………………………………………………………………………………….………86](#sinisfora)

**[ΒΙΒΛΙΟΓΡΑΦΙΑ…………………………………………………………………………………………………………………….……88](#Referencies)**

**[ΠΕΡΙΛΗΨΗ](#ΠΕΡΙΕΧΟΜΕΝΑ)**

Αντικείμενο της παρούσας εργασίας είναι η ανάπτυξη μιας διαδικτυακής εφαρμογής διαδραστικών χαρτών με σκοπό την αναπαράσταση των τροχιών πλοίων καθώς και της δραστηριότητας πλοήγησης την οποία έχουν σε κάθε θέση τους πάνω στον χάρτη ώστε να γίνει εφικτή η μετέπειτα επεξεργασία των δεδομένων θέσεών τους και να παραχθούν οι υπηρεσίες που διατίθενται στον πελάτη μετά την επεξεργασία αυτή. Η υλοποίηση της εφαρμογής είχε δύο στάδια τα οποία έγιναν με χρήση των τεχνολογιών React *(μία javascript βιβλιοθήκη ιδιαίτερα αποδοτική και εύκολη για την ανάπτυξη πολύπλοκων διαδραστικών διεπαφών που συνθέτονται από μικρά και απομονωμένα κομμάτια τα αποκαλούμενα components, μόνο στην έννοια του View, που επιτρέπει τη διαχείριση της κατάστασης των μεταβλητών σε συνεργασία ακόμα και με άλλες βιβλιοθήκες μονοδρομικά.)*, επίσης συνδυάστηκε σε συνεργασία με τη βιβλιοθήκη χαρτών leaflet για το πρώτο στάδιο του frontend και python *(μια γλώσσα κατάλληλη για επεξεργασία ολόκληρων datasets με χρήση ειδικών βιβλιοθηκών για το σκοπό αυτό, όπως η pandas και η numpy)* για το στάδιο της επεξεργασίας του dataset με τις ιστορικές θέσεις των πλοίων και τον υπολογισμό ειδικής στήλης ετικετών πλοήγησης. Επίσης, χρησιμοποιήθηκε και ένα backend το οποίο ανοίγει ένα websocket για την παροχή μιας συνεχούς, αμφίδρομης και ταυτόχρονης επικοινωνίας και τροφοδότησης της εφαρμογής μας με δεδομένα που προέρχονται από την ανάγνωση του αρχείου με τις ιστορικές αυτές θέσεις των πλοίων

Η υλοποίηση μιας τέτοιας εφαρμογής στοχεύει στην απεικόνιση και γραφική αναπαράσταση των παρακολουθούμενων πλοίων για την επεξεργασία και ανάλυση των θέσεων στις οποίες αυτά βρίσκονται με αποτέλεσμα την παραγωγή γνώσης και προβλέψεων μελλοντικών καταστάσεων σχετικά με την κίνηση των πλοίων αυτών προς τον τελικό προορισμό τους, καθώς και πολλή ακόμα γνώση σε περίπτωση επέκτασής της στο προσεχές μέλλον που θα μπορούσε να γίνει διαθέσιμη ως υπηρεσία για το ευρύ κοινό. Πιο αναλυτικά, η εφαρμογή δέχεται ένα dataset μορφής csv από το οποίο λαμβάνει πληροφορίες για κάποιες ιστορικές θέσεις διαφόρων πλοίων και τις στέλνει για γραφική αναπαράσταση στη διεπαφή της εφαρμογής όπου και φορτώνει πάντα την τελευταία πιο πρόσφατη θέση του κάθε πλοίου κάνοντας μη εμφανή την προηγούμενη. Αν ο χρήστης θελήσει να πατήσει πάνω σε κάποια θέση κάποιου πλοίου που υπάρχει ήδη στον χάρτη τότε ενεργοποιείται ολόκληρο το δρομολόγιο που ακολούθησε το πλοίο αυτό και εμφανίζεται με χρώμα κάθε κομμάτι στο οποίο ανήκουν ίδιας κατηγορίας δραστηριότητας θέσεις καθώς και κατά το πέρασμα του δείκτη του ποντικιού από πάνω από αυτά τα κομμάτια. Επίσης με το που ανοίξει το δρομολόγιο στα δεξιά κρατείται σε μια καρτέλα η τελευταία θέση του πλοίου η οποία έχει διαβαστεί μέχρι στιγμής από το dataset και φαινόταν ως κουκκίδα πάνω στον χάρτη πριν την πατήσουμε. Τέλος ο χρήστης μπορεί να διαγράψει τα ενεργοποιημένα δρομολόγια από τον χάρτη ή και να αλλάξει το χρώμα θέματος όλης της εφαρμογής.

**Λέξεις Κλειδιά:** React Interactive Web Maps, Leaflet, Ships’ Latest Positions, View Full Ship’s Trajectory, Historic vessels’ positions

**[ABSTRACT](#ΠΕΡΙΕΧΟΜΕΝΑ)**

The subject of this dissertation is the development of a web application based on interactive maps with the aim of visualizing the trajectories of ships as their navigation activity that they have on each position they are upon the map too for the purpose of analyzing all this positions’ data so the related services can be produced and then be provided to the customer afterwards. The development of the application had two stages that were accomplished by the use of technologies like React *(a javascript library which is extremely effective and flexible for the development of complex interactive user interfaces that are composed from small and isolated pieces of code called components, only under the concept of the View, allowing the management of the state with other libraries working with one-way data binding to achieve a unidirectional data flow)*, furthermore React was combined with another library for the interactive maps called Leaflet making the perfect match together for the first stage of the frontend and python *(an appropriate language for the manipulation of whole datasets with the use of expertised libraries for this purpose, such as pandas and numpy)* for the stage of the historical positions of ships dataset’s manipulation and the calculation of a specific column which stores the navigation labels inside. There was also the need of creating a backend which opens a websocket for the supply of a persistent bidirectional and full-duplex communication with the frontend of the application serving the application with the necessary data that come from the reading of the file with our dataset of all this historical data of the ships.

The development of such an application focuses on the visualization of the tracked ships for the purpose of analyzing of their positions at which they are navigating from with the aim of producing knowledge and other predictions of their future path and movements till their final destination in case this application gets expanded someday in order to be available to customers. In detail, this application takes a dataset of a csv format from which it receives information for some historical positions of several different ships and it sends it to the frontend to be depicted on the map where it loads only the latest received position of each ship making invisible the previous one. If the user wants to click upon a point representing a position of a ship that already exists on the map then it reveals the whole trajectory that this ship followed and it colors every segment in which some of the trajectory points belong and are labeled under the same navigation category not only when the trajectory opens up but also when the user just hovers upon the segment while the trajectory is open. Furthermore, when the trajectory gets opened the latest position that has been read up to the moment is getting stored in the first right panel. Finally, the user can also remove from the map the opened trajectories if he wishes to or even change the whole theme color of the application.

**Keywords:** React Interactive Web Maps, Leaflet, Ships’ Latest Positions, View Full Ship’s Trajectory, Historic vessels’ positions

**[ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ](#περιεχομενα)**

**Εικ.1.** πρώτος παραδοσιακός τρόπος λειτουργίας ενός εξυπηρετητή………………………………………………….……….……………….…….…σ.21

**Εικ.2.** δεύτερος παραδοσιακός τρόπος λειτουργίας ενός εξυπηρετητή……………..………………………………………………………………….….σ.22

**[ΚΑΤΑΛΟΓΟΣ ΑΠΟΣΠΑΣΜΑΤΩΝ ΚΩΔΙΚΑ](#περιεχομενα)**

**Απόσπασμα 1:** Τιμή επιλογής confirmation dialog ……………………………………..σ.41

**Απόσμασμα 2:** Ταξινόμηση μέρος1……………………………………………………………..σ.60

**[ΣΥΝΤΟΜΟΓΡΑΦΙΕΣ](#περιεχομενα)**

|  |  |
| --- | --- |
| SPA | Single Page Application |
| MVC | Model View Controller |
| HTTP | Hypertext Transfer Protocol |
| APP | Application |
| iOS | iPhone Operating System |
| CPU | Central Processing Unit |
| RAM | Random Access Memory |
| OS | Operating System |
| Ι/Ο | Input/Output |
| CLI | Command Line Interface |
| NPM | Node Package Manager |
| DOM | Document Object Model |
| HTML5 | HyperText Markup Language |
| IBM | International Business Machines Corporation |
| UML | Unified Modeling Language |
| ΑPI | Application Programming Interface |
| OBJ | Object |
| CRUD | Create, Read, Update, and Delete operations |
| CSS | Cascading Style Sheets |
| RXJS | Reactive Extensions for JavaScript |
| .ts | typescript file |
| .csv | Comma-separated values Microsoft excel file |
| URL | Uniform Resource Locator |
| JSON | JavaScript Object Notation |
| AJAX | Asynchronous JavaScript And XML |
| IP | Internet Protocol address |
| UI | User interface |
| PaaS | Platform as a Service |
| SSL | Secure Sockets Layer |
| TLS | Transport Layer Security |
| DNS | Domain Name System |