



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ – ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

**Πρόγραμμα Μεταπτυχιακών Σπουδών
«Προηγμένα Συστήματα Πληροφορικής»**

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	Ανάπτυξη διαδικτυακής εφαρμογής για διαχείριση και χρονοπρογραμματισμό εργασιών με εξατομίκευση χρηστών Web application development for task management and time scheduling with user personalization
Όνοματεπώνυμο Φοιτητή	Ιωάννης Κοψιδάς
Πατρώνυμο	Γεράσιμος
Αριθμός Μητρώου	ΜΠΣΠ 16015
Επιβλέπων	Αλέπης Ευθύμιος, Αναπληρωτής Καθηγητής

Ημερομηνία Παράδοσης

Σεπτέμβριος 2022

Τριμελής Εξεταστική Επιτροπή

(υπογραφή)

(υπογραφή)










(υπογραφή)

Μαρία Βίρβου
Καθηγήτρια

Ευθύμιος Αλέπης
Αναπληρωτής Καθηγητής

Σακκόπουλος Ευάγγελος
Αναπληρωτής Καθηγητής

Table of Contents

	ΠΕΡΙΛΗΨΗ	5
	ABSTRACT	5
	ΕΥΧΑΡΙΣΤΙΕΣ	6
	ΕΙΣΑΓΩΓΗ	7
	4.1 Το πλαίσιο ανάπτυξης της εφαρμογής	7
	ΣΧΕΤΙΚΕΣ ΥΛΟΠΟΙΗΣΕΙΣ	8
	5.1 Doodle.com	8
	5.1.1 Εισαγωγή.....	8
	5.1.2 Στοιχεία εφαρμογής.....	8
	5.2 Google calendar	8
	5.2.1 Εισαγωγή.....	8
	5.2.2 Στοιχεία εφαρμογής.....	8
	5.3 Clickup.com	9
	5.3.1 Εισαγωγή.....	9
	5.3.2 Στοιχεία εφαρμογής.....	9
	ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΤΟΥ ΣΥΣΤΗΜΑΤΟΣ	10
	6.1 Εισαγωγή	10
	6.2 Περιβάλλον Ανάπτυξης	10
	6.2.1 Microsoft Visual Studio	10
	6.3 .NET Framework	10
	6.4 ASP.NET MVC	11
	6.5 Η γλώσσα προγραμματισμού C#	13
	6.6 AJAX και JQuery	14
	6.7 CSS	14
	6.8 JavaScript	14
	6.9 Entity Framework	15
	6.10 SQL Server	15
	ΑΝΑΛΥΣΗ ΑΠΑΙΤΗΣΕΩΝ	17
	7.1 Εισαγωγή	17
	7.2 Το πρότυπο Unified Modelling Language	17
	7.3 Διαγράμματα χρήσης	17
	7.4 Δράστες	17
	7.4.1 Δράστης: Διαχειριστής.....	18
	7.4.2 Δράστης: Απλός χρήστης.....	22
	7.4.3 Δράστης: Χρήστης διοργανωτής.....	24
	7.4.4 Δράστης: Χρήστης προώθησης.....	26
	ΣΧΕΔΙΑΣΜΟΣ ΚΑΙ ΥΛΟΠΟΙΗΣΗ	28
	8.1 Εισαγωγή	28
	8.2 Βάση δεδομένων	28
	8.2.1 Εννοιολογικός σχεδιασμός της βάσης.....	28
	8.2.2 Λογικός σχεδιασμός της βάσης δεδομένων	28
	ΣΥΝΟΨΗ	33
	9.1 Συμπεράσματα	33

9.2 Επεκτάσεις	33
📁📁 ΑΝΑΦΟΡΕΣ.....	35

 **ΠΕΡΙΛΗΨΗ**

Η εργασία αυτή επικεντρώνεται στη σχεδίαση ενός συστήματος που ασχολείται με τον χρονοπρογραμματισμό εργασιών και την εξατομίκευση χρηστών για βέλτιστη διαχείριση. Ο χρήστης έχει σκοπό να δεσμεύσει τις αναγκαίες χρονοθυρίδες για τις δραστηριότητες που επιθυμεί και με βάση τη συμπεριφορά του και τα δεδομένα που εισάγει στο σύστημα, αυτό να του προτείνει βελτιώσεις ή/και αλλαγές στα ζητούμενα του για την αποδοτικότερη αξιοποίηση του χρόνου. Ο χρήστης ξεκινάει με την εισαγωγή των δεδομένων, όπως για παράδειγμα την δέσμευση μιας χρονοθυρίδας. Στην συνέχεια η πλατφόρμα αξιολογεί εάν είναι δυνατή η δέσμευση της. Εάν είναι η πρώτη εισαγωγή δεδομένων του χρήστη την καταχωρεί. Σε διαφορετική περίπτωση ελέγχει την δυνατότητα δέσμευσης και την καταλληλότητα της σε σχέση με τις ανάγκες και την συμπεριφορά του χρήστη στο παρόν σύστημα. Το σύστημα βελτιώνει τις επιλογές του χρήστη τόσο ως προς την μετακύλιση των χρόνων θυρίδων για να μην υπάρχει χρονική επικάλυψη όσο και για την παραμετροποίηση της διασποράς των θυρίδων σύμφωνα με τους κανόνες και τις προτεραιότητες που αυτός ορίζει. Τα κριτήρια μπορεί να είναι η διασπορά στα χρονικά όρια που θέτει ο χρήστης, η εύρεση κοινής χρονοθυρίδας δύο ή και παραπάνω χρηστών σε περιπτώσεις που η δραστηριότητα χρειάζεται κοινή συμμετοχή, το είδος της δραστηριότητας, η τοποθεσία κ.α.

 **ABSTRACT**

This thesis focuses on designing a system that deals with task scheduling and user personalization for optimal management. User intends to reserve the necessary time slots for the activities he wishes, and based on his behavior and the data he enters into the system, it will suggest improvements and/or changes to his requests for more efficient use of time. User starts by entering the data, such as the reservation of a time slot. The platform then evaluates whether it is possible to bind it. If it is the user's first data entry, it is inserted. Otherwise, it checks the possibility of binding and its suitability in relation to the needs and the behavior of the user in this system. The system improves the user's options both in terms of moving the time slots so that there is no time overlap and in configuring the dispersion of the slots according to the rules and priorities that he defines. The criteria can be dispersion in the time limits set by the user, finding a common time slot of two or more users in cases where the activity requires joint participation, the type of activity, the location etc.

☰ **ΕΥΧΑΡΙΣΤΙΕΣ**

Η ολοκλήρωση της παρούσας διπλωματικής εργασίας υλοποιήθηκε με την υποστήριξη ενός αριθμού ανθρώπων, στους οποίους θα ήθελα να εκφράσω τις θερμότερες ευχαριστίες μου.

Αρχικά, θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή κύριο Ευθύμιο Αλέπη, ο οποίος μου ανέθεσε αυτή την ενδιαφέρουσα εργασία.

Ιδιαίτερες ευχαριστίες θα ήθελα να απευθύνω στον υποψήφιο διδάκτορα Τμήμα Πληροφορικής του Πανεπιστημίου Πειραιώς, Σπύρο Παπαδημητρίου, για την καθοδήγησή του και την άριστη συνεργασία σε οποιοδήποτε πρόβλημα αντιμετώπισα κατά τη διάρκεια εκπόνησης της εργασίας.

Τέλος, οφείλω ένα μεγάλο ευχαριστώ στην οικογένειά μου και στα αγαπημένα μου πρόσωπα για όλη τη βοήθεια, υποστήριξη, εμπιστοσύνη και αντοχή που έδειξαν κατά τη διάρκεια όλων των σπουδών μου.

ΕΙΣΑΓΩΓΗ

4.1 Το πλαίσιο ανάπτυξης της εφαρμογής

Σήμερα η χρήση του παγκόσμιου ιστού είναι αναπόσπαστο κομμάτι της καθημερινότητας των ανθρώπων. Διάρκως οι χρήστες χρησιμοποιούν το διαδίκτυο με σκοπό τους την οργάνωση της καθημερινότητας τους, την εκπαίδευση, τις συναλλαγές το εμπόριο και πολλές άλλες πτυχές που έχουν ενσωματωθεί στην διαδικτυακή πραγματικότητα κάνοντας την ζωή μας πιο εύκολη αλλά και πιο πολύπλοκη. Σε αυτό συνεισφέρει και το τεράστιο μέγεθος του περιεχομένου και το πλήθος των ροών δεδομένων. Γι' αυτόν τον λόγο είναι σημαντικό να χρησιμοποιούμε εργαλεία που μας βοηθούν να προγραμματίσουμε και να βελτιώσουμε τη διαχείριση του χρόνου μας και την αποτελεσματικότητα των εργασιών μας.

Δεδομένης της ποικιλίας διαδικτυακών εφαρμογών και του μεγάλου εύρους διαδικασιών που υλοποιούνται, πλέον άνθρωποι κάθε ηλικίας είναι αναγκαίο να είναι χρήστες του παγκόσμιου ιστού. Καθώς λοιπόν έχουμε και αρκετούς μη εξοικειωμένους χρήστες, αλλά και χρήστες με πολλές απαιτήσεις, προκύπτει η ανάγκη για περαιτέρω ανάπτυξη εφαρμογών που εξυπηρετούν τις ανάγκες του χρήστη αποδοτικά και με τρόπο που τους απαλλάσσει από ενέργειες που θα καταστήσουν τη εμπειρία του χρήστη δύσκολη. Επομένως καταλήγουμε ότι οι διαδικτυακές πλατφόρμες οφείλουν να έχουν μεθόδους που συντελούν στην εξατομίκευση των χρηστών για μείωση της πολυπλοκότητας στην χρήση της πλατφόρμας και προσαρμοστικότητα στις ανάγκες τους.

Η διαχείριση του χρόνου έχει διαφορετική σημαντικότητα για κάθε άνθρωπο. Ο καθένας θέτει προτεραιότητες και κριτήρια σύμφωνα με τις αξίες και τις συνθήκες διαβίωσής του. Τα κριτήρια μπορεί να είναι η διασπορά στα χρονικά όρια που θέτει ο χρήστης (το πόσο πυκνό χρονοπρόγραμμα θέλει), το είδος και η διάρκεια της δραστηριότητας, τα γεωγραφικά χαρακτηριστικά κ.α. Από την άλλοι υπάρχουν και περιορισμοί που μπορεί να υπάρχουν όπως η χρονική επικάλυψη δύο ή περισσότερων δραστηριοτήτων, ο υπερβολικός όγκος δραστηριοτήτων, προδιαγραφές από θεσμούς κ.α.

Στην παρούσα διπλωματική εργασία σκοπός ήταν η δημιουργία μιας διαδικτυακής πλατφόρμας η οποία θα προσαρμόζεται στις ανάγκες και πολιτικές του εκάστοτε χρήστη για την δέσμευση χρονοθυρίδων. Το αντικείμενο της εφαρμογής είναι η δέσμευση και δυνατόν βελτίωση του χρονοπρογραμματισμού για τον χρήστη σύμφωνα με τα επιλεγμένα κριτήρια και τους υπάρχοντες περιορισμούς. Στο πλαίσιο της εργασίας δεν μπορεί να καλυφθεί το σύνολο των περιορισμών και των κριτηρίων με τα οποία εξατομικεύουμε τον χρονοπρογραμματισμό. Ωστόσο η υλοποίηση της πλατφόρμας έχει γίνει με τρόπο που επιτρέπει την εύκολη εισαγωγή και επέκταση νέων κριτηρίων και περιορισμών. Με την σκέψη αυτή καταλήγουμε σε μία υλοποίηση που αφορά δέσμευση χρόνου για ατομικές ή ομαδικές δραστηριότητες, για παράδειγμα σε ένα τμήμα μιας εταιρείας.

ΣΧΕΤΙΚΕΣ ΥΛΟΠΟΙΗΣΕΙΣ

5.1 Doodle.com

5.1.1 Εισαγωγή

Το doodle είναι μια διαδικτυακή πλατφόρμα για την διαχείριση χρόνου και συντονισμό συναντήσεων που δημιουργήθηκε από την ομώνυμη εταιρεία στην Σουηδία. Βρίσκεται στο διαδίκτυο από το 2007 και έχει χρησιμοποιηθεί σε πολλούς τομείς για την οργάνωση συναντήσεων. Οι χρήστες καταχωρούν την ημερομηνία και την ώρα που θέλουν να γίνει η συνάντηση και ο διοργανωτής της επιλέγει το χρόνο που ταιριάζει καλύτερα στον καθένα για να καταχωρηθεί στα ημερολόγια των εμπλεκομένων. Λαμβάνεται αλληλογραφία για την ενημέρωση της κάθε ενέργειας που αφορά τις δραστηριότητες των χρηστών και ενημερώνονται με τον επιλεγμένο τρόπο. Η πλατφόρμα βρίσκει χαρακτηριστική εφαρμογή στο ακαδημαϊκό περιβάλλον όπου υπάρχει πληθώρα δραστηριοτήτων, με ποικιλία ως προς το περιεχόμενό τους, και πολλοί χρήστες που προσπαθούν να βελτιστοποιήσουν τη διαχείριση του χρόνου τους σε ένα πολύ δυναμικό περιβάλλον.

5.1.2 Στοιχεία εφαρμογής

Η πλατφόρμα doodle.com έχει παραπάνω από 200 εκατομμύρια ενεργούς χρήστες. Διαθέτει πολλά διαφορετικά πακέτα που βοηθούν οργανισμούς, όπως μεγάλες πολυεθνικές εταιρείες, να έχουν ένα σύστημα όπου καταγράφεται το σύνολο των δραστηριοτήτων των εργαζομένων και παρέχονται στατιστικά, δεδομένα που αφορούν τους εργαζόμενους και τους συναδέλφους τους ανάλογα με την διαρθρωτική δομή της εταιρείας, τρόπο διασύνδεσης με άλλες εφαρμογές όπως για παράδειγμα εφαρμογές γεωγραφικών δεδομένων και προσωπικά ημερολόγια.

Χαρακτηριστικά που κάνουν τόσο επιτυχημένη την πλατφόρμα είναι τα εξής:

- Εύκολη χρήση της εφαρμογής ακόμα και για μη εξοικειωμένους χρήστες
- Διασύνδεση με άλλες εφαρμογές όπως GPS, ηλεκτρονική αλληλογραφία
- Πρόσβαση στα δεδομένα και στην χρήση της εφαρμογής πλέον από παντού μέσω διαδικτύου
- Σύγχρονη υλοποίηση με την χρήση υπολογιστικής νέφους για μεγάλη δυνατότητα κλιμάκωσης

5.2 Google calendar

5.2.1 Εισαγωγή

Το google calendar είναι μια διαδικτυακή πλατφόρμα για την διαχείριση χρόνου και την καταγραφή του υποχρεώσεων μέσω ενός ηλεκτρονικού ημερολογίου. Βρίσκεται στο διαδίκτυο από το 2006 και έχει χρησιμοποιηθεί σε πολλούς τομείς για την οργάνωση συναντήσεων. Οι χρήστες καταχωρούν την ημερομηνία και ώρα της δραστηριότητας. Το μεγάλο πλεονέκτημα είναι ότι αν περαστούν δεδομένα στο google calendar αυτόματα αυτά παρέχονται και σε όλο το οικοσύστημα της google που αφορά πολλές εφαρμογές διαδικτύου και κινητών. Επίσης η google έχει τεράστια τεχνογνωσία στην ασφάλεια δεδομένων και επειδή έχει τεράστιο μέγεθος και πόρους μπορεί να προσαρμόζεται άμεσα στις νομοθετικές και άλλες αλλαγές. Η πλατφόρμα βρίσκει εφαρμογή σε κάθε χρήστη με τον τρόπο που πρακτικά ψηφιοποιεί την καθημερινότητα μας και με την ασφάλεια των δεδομένων του, για παράδειγμα έλεγχος ταυτότητας δύο παραγόντων.

5.2.2 Στοιχεία εφαρμογής

Το οικοσύστημα της google έχει παραπάνω από 4,3 δισεκατομμύρια ενεργούς χρήστες. Διαθέτει πολλές διαφορετικές εφαρμογές οι οποίες μπορούν να αλληλοεπιδρούν και να ανταλλάζουν πληροφορίες.

Χαρακτηριστικά που κάνουν τόσο επιτυχημένη την πλατφόρμα είναι τα εξής:

- Εύκολη χρήση της εφαρμογής ακόμα και για μη εξοικειωμένους χρήστες

- Διασύνδεση με ένα γιγαντιαίο οικοσύστημα που παρέχει πάρα πολλή λειτουργικότητα με δικλείδες ασφαλείας, εγγυημένη συμμόρφωση στους ισχύοντες κανονισμούς
- Πρόσβαση στα δεδομένα και στην χρήση της εφαρμογής από πολλών ειδών τερματικά, όπως ηλεκτρονικό υπολογιστή, κινητό τηλέφωνο, έξυπνα ρολόγια και άλλα.
- Σύγχρονη υλοποίηση με την χρήση υπολογιστικής νέφους για μεγάλη δυνατότητα κλιμάκωσης

5.3 Clickup.com

5.3.1 Εισαγωγή

Το clickup.com είναι μια διαδικτυακή πλατφόρμα που δημιουργήθηκε το 2017 και έχει ως γνώμονα την αύξηση της παραγωγικότητας για μια εταιρεία τόσο ατομικά όσο και συλλογικά. Δημιουργήθηκε με την σκέψη ότι υπάρχουν πάρα πολλά εργαλεία για να παρακολουθεί μια εταιρεία πολλά πράγματα σε εντελώς ξεχωριστά οικοσυστήματα. Η υλοποίηση λοιπόν του clickup, στοχεύει στην ενοποίηση των εργαλείων που χρησιμοποιεί μια εταιρεία για την αύξηση την παραγωγικότητας και του χρονοπρογραμματισμού.

5.3.2 Στοιχεία εφαρμογής

Η πλατφόρμα clickup.com έχει παραπάνω από 800 χιλιάδες ενεργές ομάδες χρηστών. Διαθέτει πολλά διαφορετικά πακέτα που βοηθούν οργανισμούς, όπως μεγάλες πολυεθνικές εταιρείες, να έχουν ένα σύστημα όπου καταγράφεται το σύνολο των δραστηριοτήτων των εργαζομένων και παρέχονται στατιστικά, δεδομένα που αφορούν τους εργαζόμενους και τους συναδέλφους τους ανάλογα με την διαρθρωτική δομή της εταιρείας. Εκτός από το πολύ σημαντικό πρόβλημα του χρονοπρογραμματισμού, η πλατφόρμα αυτή εμπλουτίζει το κάθε επίπεδο χρήσης με στοιχεία που βοηθούν στην διοίκηση προσωπικού, την εξωστρέφεια ανάμεσα στα εμπλεκόμενα μέρη της πλατφόρμας και στην ακεραιότητα των διαδικασιών.

Χαρακτηριστικά που κάνουν τόσο επιτυχημένη την πλατφόρμα είναι τα εξής:

- Εύκολη χρήση της εφαρμογής με έμφαση στην παροχή εκπαιδευτικού υλικού ή/και εκπαίδευσης από εξειδικευμένους συμβούλους
- Συγκέντρωση της λειτουργικότητας που χρειάζονται οι εργαζόμενοι σε ένα ενιαίο σύστημα που διευκολύνει τις διαδικασίες και τις καθιστά πιο αποδοτικές
- Πρόσβαση στα δεδομένα και στην χρήση της εφαρμογής από παντού μέσω διαδικτύου
- Σύγχρονη υλοποίηση με την χρήση υπολογιστικής νέφους για μεγάλη δυνατότητα κλιμάκωσης
- Προσαρμοστικότητα στις απαιτήσεις του χρήστη, όπως πρόταση για την καταχώριση μιας νέας χρονοθυρίδας που δεν έχει μικρή χρονική απόσταση από άλλες υποχρεώσεις, δέσμευση χρόνου χωρίς περισπασμούς

🕒 **ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΤΟΥ ΣΥΣΤΗΜΑΤΟΣ**

6.1 Εισαγωγή

Τα τελευταία χρόνια, έχει γίνει μία πραγματική επανάσταση στον χώρο των διαδικτυακών εφαρμογών σε επίπεδα εμπορευματοποίησης, τεχνολογικής προόδου αλλά και δυνατότητα κλιμάκωσης ανεξαρτήτως γεωγραφικών χαρακτηριστικών. Έτσι, έχουν δημιουργηθεί ή/και εξελιχθεί πληθώρα γλωσσών και εργαλείων προσανατολισμένα στον διαδικτυακό προγραμματισμό όπως η ASP.NET, Ruby, JavaScript, Ajax, JQuery, CSS, XHTML, HTML 5 κ.ά. Κάθε μία από αυτές τις τεχνολογίες συμπληρώνει ή/και ανταγωνίζεται ή/και επεκτείνει τις άλλες μα στο σύνολό τους δημιουργούν ένα σύστημα το οποίο σκοπό έχει την επέκταση των δυνατοτήτων του ίδιου του Παγκόσμιου Ιστού.

Στο κεφάλαιο αυτό περιγράφονται συνοπτικά οι τεχνολογίες και τα εργαλεία ανάπτυξης που χρησιμοποιήθηκαν κατά την υλοποίηση της εφαρμογής.

6.2 Περιβάλλον Ανάπτυξης

6.2.1 Microsoft Visual Studio

Το Visual Studio είναι ένα ολοκληρωμένο περιβάλλον ανάπτυξης (Integrated Development Environment - IDE) που χρησιμοποιείται για την ανάπτυξη διαδικτυακών και desktop εφαρμογών καθώς και διαδικτυακών υπηρεσιών (Web Services). Το Visual Studio εισάγει μια νέα διαδικασία ανάπτυξης προγραμμάτων, τον παραστατικό προγραμματισμό, που αλλάζει τον τρόπο εγγραφής και εκτέλεσης των προγραμμάτων, οδηγώντας σε αύξηση της παραγωγικότητας. Παρέχει προχωρημένα εργαλεία για τη διόρθωση λαθών, για τη τεκμηρίωση και εγγραφή κώδικα, για την ανάπτυξη διεπαφών χρήστη, για τη σχεδίαση κλάσεων καθώς και για τη σχεδίαση του σχήματος μίας βάσης δεδομένων (database schema). Επίσης υποστηρίζει πολλά plug-ins που προσφέρουν προηγμένες λειτουργίες όπως unit testing και refactoring. Οι ενσωματωμένες γλώσσες προγραμματισμού του Visual Studio είναι οι Visual C++, Visual C# και Visual Basic. Επίσης, παρέχεται υποστήριξη και για άλλες γλώσσες όπως τις F#, Python, Ruby οι οποίες εγκαθίστανται ξεχωριστά μέσω των language services.

6.3 .NET Framework

Το .NET Framework είναι μια πλατφόρμα ανάπτυξης λογισμικού, για την δημιουργία εφαρμογών που προορίζονται για τα λειτουργικά συστήματα Windows, Windows phone, Windows Server, Windows Azure και unix περιβάλλοντα. Αποτελείται από την Common Language Runtime (CLR), που ουσιαστικά είναι ένα εικονικό περιβάλλον εκτέλεσης των προγραμμάτων .NET, και την βιβλιοθήκη κλάσεων .NET Framework class library. Γενικά, το .NET Framework παρέχει ένα ευέλικτο αλλά ταυτόχρονα ελεγχόμενο περιβάλλον εκτέλεσης εφαρμογών, ευκολία στην ανάπτυξη εφαρμογών και project, και ταυτόχρονη χρήση πολλών γλωσσών προγραμματισμού, οι οποίες είναι οι .NET Languages. Πιο συγκεκριμένα οι βασικές ψηφίδες του .NET Framework είναι οι ακόλουθες:

- **.NET Framework Base Class Library.** Περιλαμβάνει πολλές βιβλιοθήκες κλάσεων με σκοπό να παρέχουν έτοιμο κώδικα στους προγραμματιστές, για λειτουργίες που θεωρείται ότι είναι συχνά χρησιμοποιούμενες, και μπορούν να μοντελοποιηθούν αποτελεσματικά. Παραδείγματα τέτοιων λειτουργιών είναι η εγγραφή και ανάγνωση σε αρχεία ή στην οθόνη, η επικοινωνία με βάσεις δεδομένων, η δημιουργία γραφικών στοιχείων κλπ.
- **.NET Languages.** Είναι ένα σύνολο γλωσσών προγραμματισμού που μπορούν να συνδυαστούν, και στο ίδιο Project, και η επικοινωνία μεταξύ τους. Αυτές οι γλώσσες είναι οι C#, Visual Basic, JS script, F#, J#, C++. Γενικά οι γλώσσες αυτές ακολουθούν τις αρχές του αντικειμενοστραφούς προγραμματισμού.
- **Language Integrated Query (LINQ).** Είναι μία ψηφίδα του .NET Framework που παρέχει την δυνατότητα διεξαγωγής ερωτημάτων σε δεδομένα (querying capabilities)

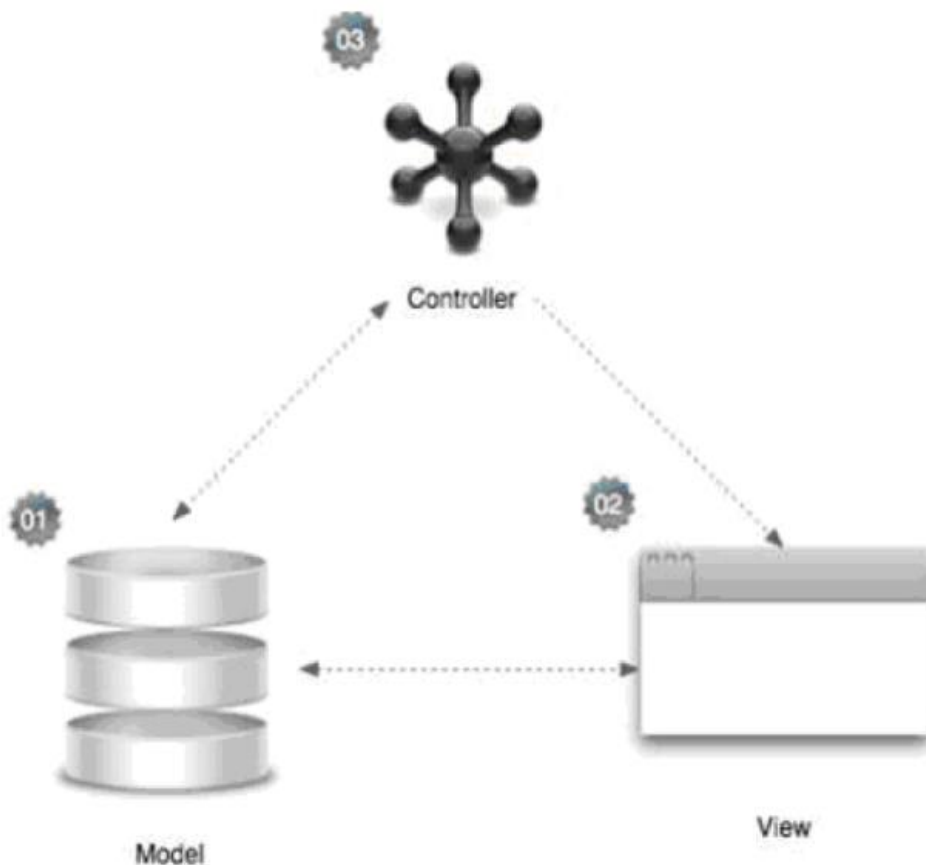
στις γλώσσες του .NET. Αυτή η δυνατότητα είναι πολύ σημαντική, αφού τα περισσότερα πληροφοριακά συστήματα είναι στενά συνδεδεμένα με δεδομένα σε κάποια βάση δεδομένων, και το γεγονός ότι παρέχεται φυσική υποστήριξη από την γλώσσα (δυνατότητα αυτόματης συμπλήρωσης, επισήμανση λαθών, σύνθετες πράξεις με τα δεδομένα) αυξάνει σημαντικά την παραγωγικότητα των προγραμματιστών.

- **ASP.NET.** Το ASP.NET είναι ένα server-side Web application Framework που έχει σχεδιαστεί με σκοπό να παρέχει την ικανότητα στους Web Developers να δημιουργούν δυναμικές ιστοσελίδες. Είναι ένα σχετικά καινούριο Framework, που εμφανίστηκε το 2002. Είναι δομημένο στο Common Language Runtime (CLR), και επιτρέπει στους προγραμματιστές να χρησιμοποιήσουν την .NET γλώσσα προγραμματισμού της αρεσκείας τους. Το ASP.NET υποστηρίζει τρία διαφορετικά μοντέλα ανάπτυξης, που είναι τα Web Pages, Web Forms, και το MVC που είναι και αυτό που χρησιμοποιήθηκε για την ανάπτυξη του παρόντος πληροφοριακού συστήματος.
- **Intermediate Language (IL).** Μία μορφή byte code, που αποτελεί ενδιάμεση μορφή μεταγλώττισης, σε ένα εικονικό περιβάλλον εκτέλεσης, γνωστό ως CLR, και σκοπό έχει ο κώδικας που παράγεται να είναι ανεξάρτητος της αρχιτεκτονικής του εκάστοτε επεξεργαστή.
- **Common Language Runtime (CLR).** Το εικονικό περιβάλλον (virtual machine), που ελέγχει και εκτελεί τις .NET εφαρμογές
- **Garbage Collection.** Το εικονικό περιβάλλον CLR είναι υπεύθυνο για την αποδέσμευση των μη χρησιμοποιούμενων πόρων που δεσμεύουν οι .NET εφαρμογές.
- **Common Type System (CTS),** και Common Language Specification (CLS). Είναι ένα σύνολο από κανόνες οι οποίοι έχουν σκοπό να καταστήσουν δυνατή την επικοινωνία μεταξύ των διαφορετικών .NET Languages. Με αυτόν τον τρόπο είναι δυνατή ή εύκολη μετατροπή κώδικα από μια γλώσσα σε μία άλλη, αφού ουσιαστικά οι τύποι των δεδομένων (τα αντικείμενα) υπακούουν σε κοινούς κανόνες, και έχουν το ίδιο interface προς τον προγραμματιστή.

6.4 ASP.NET MVC

Το MVC (Model – View – Controller) είναι ένα μοντέλο αρχιτεκτονικής, που διαχωρίζει την εφαρμογή σε τρία κύρια συστατικά: το Model, το View, και τον Controller. Το ASP.NET MVC Framework ακολουθεί αυτήν την αρχιτεκτονική ανάπτυξης και αποτελεί ένα ελαφρύ και με μεγάλες δυνατότητες συντήρησης και ελεγχιμότητας Framework, που διατηρεί όλα τα πλεονεκτήματα του ASP.NET όπως τα λεγόμενα Master Pages ή το membership bashed authentication. Ήδη, αρκετοί προγραμματιστές προτιμούν αυτήν την αρχιτεκτονική και έχουν υιοθετήσει το συγκεκριμένο Framework αφού η χρήση του ενδείκνυται για πλήθος εφαρμογών. Οι τρεις βασικές ψηφίδες, όπως αναφέρθηκαν και στην προηγούμενη παράγραφο, είναι οι ακόλουθες:

- **Model.** Είναι τα αντικείμενα εκείνα που υλοποιούν το business logic της εφαρμογής. Αποτελούν δηλαδή τις κύριες οντότητες της εφαρμογής, αφού η κατάσταση των αντικειμένων αυτών προσδιορίζει την συμπεριφορά της εφαρμογής. Τα αντικείμενα του μοντέλου συχνά αντιστοιχούν σε πίνακες κάποιας βάσης δεδομένων, οπότε ανακτούν τις τιμές από την βάση δεδομένων, ενεργούν πάνω στα δεδομένα, και στέλνουν πίσω τις ανανεωμένες πληροφορίες.
- **Views.** Είναι οι ψηφίδες που είναι υπεύθυνες για την παρουσίαση του User Interface (UI) της εφαρμογής. Τυπικά, αυτό το UI παράγεται από το μοντέλο.
- **Controllers.** Είναι οι ψηφίδες που είναι υπεύθυνες για την αλληλεπίδραση με τον χρήστη. Λαμβάνουν δηλαδή την είσοδο του χρήστη, επεξεργάζονται το μοντέλο (model) βάση αυτής, και επιλέγουν ποιο view θα εμφανιστεί τελικά στον χρήστη. Ο controller δηλαδή χειρίζεται την είσοδο από τον χρήστη, την μεταφέρει στο μοντέλο, που με την σειρά του ενεργεί πάνω στα δεδομένα (στην βάση δεδομένων). Τέλος επιλέγεται ποιο View θα εμφανιστεί στον χρήστη.



Το MVC Framework όπως είδαμε κάνει σαφή διαχωρισμό των διαφορετικών τμημάτων κάθε εφαρμογής, δηλαδή των Input logic, business logic και UI Logic, και ταυτόχρονα προσφέρει έναν εύκολο και ευέλικτο τρόπο επικοινωνίας μεταξύ τους. Παρέχει δηλαδή ένα είδος καθοδήγησης στον προγραμματιστή σχετικά με το που πρέπει να βρίσκεται το κάθε τμήμα μέσα στην εφαρμογή, γεγονός που οδηγεί σε ευανάγνωστο και εύκολα συντηρήσιμο κώδικα, και ελαττώνει την πολυπλοκότητα αφού επιτρέπει στον προγραμματιστή να επικεντρώνεται σε ένα κομμάτι της εφαρμογής κάθε φορά (model ή view ή controller). Ακόμη, η προσέγγιση αυτή καθιστά εύκολη την ανάπτυξη μεγάλων project από ομάδες πολλών ατόμων. Γενικά, το ASP.NET MVC προσφέρει:

- Σαφή διαχωρισμό των διαδικασιών της εφαρμογής (Input Logic – Business Logic – UI Logic), Test Driven Development (TDD) και υψηλή συντηρησιμότητα της εφαρμογής.
- Ένα επεκτάσιμο και παραμετροποιήσιμο Framework. Όλες οι ψηφίδες του είναι σχεδιασμένες με τέτοιο τρόπο, ώστε να μπορούν εύκολα να αντικαθίστανται ή να τροποποιούνται. Μπορούμε για παράδειγμα να κατασκευάσουμε και να συνδέσουμε την δικιά μας View engine, Routing Policy και γενικά διάφορες ψηφίδες.
- Έναν ισχυρό μηχανισμό URL – mapping , που ξεφεύγει από την κλασική προσέγγιση της αναζήτησης κάποιου αρχείου σε απομακρυσμένο server. Τα URL's δεν αντιπροσωπεύουν τα αρχεία με τις επεκτάσεις τους (πχ home.php), αλλά είναι ευανάγνωστα και εύκολα ανακτήσιμα, και σχεδιασμένα να υποστηρίζουν διάφορα URL naming patterns που επωφελούνται από το λεγόμενο Search Engine Optimization (SEO) και το Representational State Transfer (REST) addressing.

- Υποστήριξη για την χρήση ισχυρής γλώσσας προγραμματισμού σε κάποια ιστοσελίδα, Master Pages.

Η αλληλεπίδραση μεταξύ των τριών τμημάτων έχει κοινή δομή για όλες τις εφαρμογές που χρησιμοποιούν το πρότυπο MVC. Ο Ελεγκτής βρίσκεται στον πυρήνα της αρχιτεκτονικής και αλληλεπιδρά με τον εκάστοτε χρήστη ενώ στο παρασκήνιο υπάρχει επικοινωνία μεταξύ του στοιχείων του Μοντέλου και της Όψης. Το Μοντέλο έχει πρόσβαση στα δεδομένα έτσι ώστε ο Ελεγκτής να έχει τη δυνατότητα ενημέρωσης σε κάθε είσοδο που δέχεται από το χρήστη κάτι που αυξάνει το βαθμό αλληλεπίδραση ανάμεσα στο χρήστη και την εφαρμογή.

6.5 Η γλώσσα προγραμματισμού C#

Η C# είναι μία αντικειμενοστραφής γλώσσα προγραμματισμού, συνεπώς είναι υψηλού επιπέδου. Αυτό σημαίνει πως είναι φιλική προς τον προγραμματιστή, αφού αποκρύπτει τις λεπτομέρειες του υλικού του υπολογιστικού συστήματος, και παρέχει έναν ενιαίο τρόπο επικοινωνίας προς αυτό, μέσω των έτοιμων κλάσεων και βιβλιοθηκών που προσφέρει. Αποτελεί μια συνεχώς εξελισσόμενη γλώσσα, και σε κάθε νέα της έκδοση προστίθενται νέα χαρακτηριστικά και συντακτικό, με στόχο να κάνει τα πράγματα ευκολότερα για τον προγραμματιστή. Ακόμη, μέσω του ASP.NET μπορεί να ενσωματωθεί κώδικας σε C# ανάμεσα σε τυπικό κώδικα HTML με ξεκάθαρο και αποτελεσματικό τρόπο, βοηθώντας την κατασκευή δυναμικών ιστοσελίδων. Εν συντομία, τα βασικά χαρακτηριστικά της είναι:

- **Απλότητα (Simplicity).** Η C# είναι μια απλή γλώσσα που μπορεί να χρησιμοποιηθεί χωρίς εντατική εκμάθηση, ενώ ταυτόχρονα είναι εναρμονισμένη με σύγχρονες προγραμματιστικές πρακτικές. Οι θεμελιώδεις αρχές της γλώσσας μπορούν να κατανοηθούν γρήγορα κάτι που σημαίνει ότι οι προγραμματιστές θα είναι παραγωγικοί σε σύντομο χρονικό διάστημα. Η C# έχει σχεδιαστεί έτσι ώστε να μειώνεται η πιθανότητα πρόκλησης λαθών από την πολυπλοκότητα του κώδικα, αφού τη μειώνει σε μεγάλο βαθμό με το απλουστευμένο συντακτικό της και την οργάνωση κώδικά της.
- **Αντικειμενοστρέφεια (Object – Orientation).** Η C# από τα θεμέλια της σχεδιάστηκε να είναι αντικειμενοστραφής. Ο αντικειμενοστραφής προγραμματισμός επικράτησε σαν προγραμματιστικό πρότυπο την προηγούμενη δεκαετία και παραμένει στις πρώτες προτιμήσεις των προγραμματιστών. Οι ανάγκες για κατανεμημένα συστήματα πελάτη - εξυπηρετητή συμπίπτουν με την ενθουσία και την ανταλλαγή μηνυμάτων που είναι βασικά χαρακτηριστικά του αντικειμενοστραφούς προγραμματισμού. Κατά πολλούς ειδικούς στις γλώσσες προγραμματισμού, η επιτυχής λειτουργία των προγραμματιστικών συστημάτων σε δικτυακά περιβάλλοντα αυξανόμενης πολυπλοκότητας βασίζεται στην Αντικειμενοστρέφεια. Η C# παρέχει μια ξεκάθαρη και αποδοτική αντικειμενοστραφή πλατφόρμα παρέχοντας στους προγραμματιστές μια συλλογή βιβλιοθηκών δοκιμασμένων αντικειμένων που παρέχουν λειτουργικότητα που ποικίλει από απλούς τύπους δεδομένων, σε διεπαφές εισόδου/εξόδου ή δικτυακές και εργαλεία για τη δημιουργία παραθυρικών εφαρμογών. Αυτές οι βιβλιοθήκες μπορούν να προσαρμοστούν στις ανάγκες του προγραμματιστή. Επιπρόσθετα η C# υποστηρίζει και τον προγραμματισμό βασισμένο σε components (component – based programming) ο οποίος επιτρέπει τον προσδιορισμό αυτόνομων μονάδων λειτουργικότητας (components) που είναι απομονωμένα και τεκμηριωμένα, παρουσιάζοντας ένα μοντέλο με ιδιότητες, μεθόδους, events και μεταδεδομένα για το component. Η C# υποστηρίζει αυτά τα χαρακτηριστικά άμεσα κάνοντας έτσι τη διαδικασία δημιουργίας και χρήσης των components πολύ εύκολη.
- **Οικειότητα (Familiarity).** Στο ξεκίνημα της υλοποίησής της, οι δημιουργοί της τεχνολογίας C# απέρριψαν την ολοκληρωτική χρήση της C++ σαν γλώσσα υλοποίησης. Στη νέα γλώσσα ωστόσο, κράτησαν αρκετά χαρακτηριστικά της C++ αλλά και της Java και αφαίρεσαν την άχρηστη πολυπλοκότητα και των δύο. Έτσι έχοντας κρατήσει αρκετά από τα αντικειμενοστραφή χαρακτηριστικά και τη γενική φιλοσοφία της C++ αλλά και τη γενική ευκολία της Java βελτιώνοντας ορισμένα σημεία της, είναι σχετικά εύκολη τη

«μετακόμιση» στη C# δεδομένου ότι η C η C++ αλλά και η Java είναι ευρέως γνωστές και χρησιμοποιούνται συχνά.

6.6 AJAX και JQuery

Με τον όρο AJAX (Asynchronous JavaScript and XML) αναφερόμαστε σε μια τεχνολογία που επιτρέπει την δυναμική ανανέωση μέρους ή ολόκληρης της ιστοσελίδας, χωρίς να χρειάζεται να την ανανεώσει ο χρήστης, αλλά μέσω των λεγόμενων ασύγχρονων requests. Ο πελάτης (client) ανακτά τα δεδομένα από τον εξυπηρετητή (server) στο παρασκήνιο, χωρίς να είναι εμφανής στον χρήστη αυτή η λειτουργία. Η τεχνολογία αυτή συνετέλεσε τα μέγιστα στην ανάπτυξη δυναμικών, υψηλής ποιότητας ιστοσελίδων. Επειδή συχνά παρεξηγείται ο όρος, να σημειωθεί πως το AJAX είναι απλά μια τεχνολογία, και όχι κάποια γλώσσα προγραμματισμού. Συγκεκριμένα τα λεγόμενα AJAX requests γράφονται σε JavaScript, και συνεπώς εκτελούνται στον περιηγητή του χρήστη. Στην παρούσα Web εφαρμογή η τεχνολογία αυτή χρησιμοποιήθηκε έτσι ώστε να υπάρχει δυναμική ανανέωση των περιεχομένων των πινάκων της βάσης δεδομένων, και συνεπώς να μην είναι υποχρεωμένος ο χρήστης να ανανεώνει συνεχώς την σελίδα. Ακόμη, ανανεώνεται δυναμικά το κεντρικό Master Page της ιστοσελίδας, όπου εμφανίζονται όλες οι ειδοποιήσεις (notifications) προς τον χρήστη. Η jQuery είναι ένα σύνολο από βιβλιοθήκες της γλώσσας JavaScript, οι οποίες ενσωματώνουν λειτουργίες που είναι πολύ πιθανό να θέλει κάποιος προγραμματιστής Web εφαρμογών να συμπεριλάβει στην ιστοσελίδα του. Είναι γραμμένες με αποδοτικό τρόπο, ευανάγνωστες, συντηρούνται και ανανεώνονται συνεχώς από την πολύ ενεργή κοινότητα που την αναπτύσσει. Ο σκοπός της είναι να κάνει την πλοήγηση στις ιστοσελίδες πιο ευχάριστη και διαδραστική γενικά να παρέχει μία ευχάριστη εμπειρία στον χρήστη μέσα από αυτήν. Ακόμη, παρέχει την δυνατότητα τροποποίησης των html κόμβων της ιστοσελίδας με εύκολο τρόπο, ενσωματώνοντας συντομογραφίες, συναρτήσεις και interfaces. Στην παρούσα Web εφαρμογή η jQuery χρησιμοποιήθηκε για την εναλλαγή και παρουσίαση εικόνων στην αρχική σελίδα, καθώς επίσης και για την επαλήθευση (validation) των δεδομένων που εισάγει ο χρήστης. Η επαλήθευση αυτή είναι κομβικής σημασίας, αφού αφ ενός απαγορεύουμε την έλευση λανθασμένων δεδομένων στον Server (ακόμα και όταν κάνει submit ο χρήστης τα δεδομένα δεν αποστέλλονται αν είναι λανθασμένα) , και εμφανίζεται κατάλληλο ενημερωτικό μήνυμα στον χρήστη ώστε να εισάγει σωστά δεδομένα, και αφ ετέρου δεν περιμένει ο χρήστης να αποσταλούν τα δεδομένα στον Server για να γίνει το validation, και μετά να απαντήσει ο Server με κάποιο μήνυμα λάθους. Βλέπουμε δηλαδή ότι η προσέγγιση αυτή είναι πολύ πιο αποδοτική.

6.7 CSS

Η CSS (Cascading Style Sheets) είναι μια γλώσσα που επιτρέπει την εύκολη μορφοποίηση του περιεχομένου (της HTML δηλαδή) των ιστοσελίδων. Η γλώσσα αυτή αναπτύχθηκε από την διεθνή κοινότητα W3C (World Wide Web Consortium), και σε σύντομο χρονικό διάστημα προτυποποιήθηκε και υποστηρίζεται από όλους τους μοντέρνους περιηγητές. Η ιδέα βάσει της οποίας αναπτύχθηκε αυτή η γλώσσα, είναι ο διαχωρισμός του περιεχομένου των ιστοσελίδων από την εμφάνισή τους. Ο διαχωρισμός αυτός επιτρέπει την συγγραφή κώδικα ο οποίος καθορίζει την εμφάνιση των ιστοσελίδων ξεχωριστά από το περιεχόμενό τους, καθιστώντας δυνατή την παράλληλη ανάπτυξη και των δύο από διαφορετικούς προγραμματιστές. Ακόμη είναι δυνατή η συγγραφή των λεγόμενων templates, που είναι κώδικας CSS σε κάποιο αρχείο, και η χρήση τους σε διαφορετικές ιστοσελίδες. Στην παρούσα εφαρμογή χρησιμοποιήθηκε η CSS, μέσω του αρχείου Site.css έτσι ώστε η μορφοποίηση που επιβάλλει να υιοθετείται από όλα τα Views της εφαρμογής, και να κάνει πλοήγηση στον ιστότοπο ευχάριστη.

6.8 JavaScript

Στην αρχή της ιστορίας του Web, υπήρχε μόνο η HTML και το CGI (Common Gateway Interface). Το πρόβλημα με την HTML είναι η στασιμότητά της με την έννοια ότι με το που θα εκτυπωθεί στον browser του χρήστη δε μπορεί να μεταβληθεί από μόνη της. Για να επιτευχθεί

κάτι τέτοιο πρέπει να υπάρξει επικοινωνία με τον server ώστε μέσω μία script γλώσσας προγραμματισμού στην πλευρά του server να αποσταλούν εκ νέου δεδομένα στην πλευρά του πελάτη. Κάτι τέτοιο απαιτούνταν ακόμα και για τις πιο απλές διαδικασίες όπως η απόκρυψη ή η εμφάνιση ενός στοιχείου του εγγράφου κάτι όμως που αποτελεί τροχοπέδη για την ανάπτυξη του διαδικτυακού προγραμματισμού αλλά και για τη λειτουργία των εξυπηρετητών αφού αυξάνεται σημαντικά ο υπολογιστικός φόρτος. Τη λύση σε αυτό το πρόβλημα έδωσε η γλώσσα JavaScript η οποία είναι μία γλώσσα προγραμματισμού που εκτελείται στην πλευρά του πελάτη.

Η JavaScript αναπτύχθηκε από την εταιρεία Netscape σε συνεργασία με την Sun Microsystems και πρωτοδημοσιεύτηκε το 1995 με το όνομα LiveScript. Αργότερα πήρε το τωρινό της όνομα εκμεταλλευόμενη τη δυναμική της πανίσχυρης γλώσσας προγραμματισμού Java αν και δεν έχει σχέση με αυτήν.

Αποτελεί μία διερμηνευμένη (interpreted) γλώσσα προγραμματισμού με ιδιότητες αντικειμενοστραφούς γλώσσας προγραμματισμού, χωρίς όμως να μπορεί να χαρακτηριστεί ως πλήρης αντικειμενοστραφής αν και κάτι τέτοιο λέγεται ότι προβλέπεται στην επόμενη έκδοσή της. Η γλώσσα JavaScript χρησιμοποιείται κυρίως (αλλά όχι μόνο) για την επίτευξη των παρακάτω στόχων:

- Λιγότερος υπολογιστικός φόρτος στην πλευρά του server
- Αμεσότητα αλληλεπίδρασης με τους χρήστες
- Αυξημένη δυνατότητα αλληλεπίδρασης
- Βελτιωμένα γραφικά περιβάλλοντα
- Ελαφρότερα περιβάλλοντα και μικρότεροι χρόνοι αναμονής

6.9 Entity Framework

Το Entity Framework επιτρέπει στους προγραμματιστές να προγραμματίζουν με βάση σχεσιακές βάσεις δεδομένων, χρησιμοποιώντας αντικείμενα .NET και Language Integrated Query (LINQ). Διαθέτει πολλές νέες δυνατότητες, όπως παράβλεψη επιμονής και υποστήριξη POCO, συσχετισμούς εξωτερικών κλειδιών, αργή φόρτωση, υποστήριξη προγραμματισμού βάσει δοκιμής, λειτουργίες στο μοντέλο και νέους τελεστές LINQ. Οι επιπλέον δυνατότητες περιλαμβάνουν καλύτερη υποστήριξη n-tier με οντότητες αυτόματης παρακολούθησης, δημιουργία κώδικα με δυνατότητα προσαρμογής με χρήση προτύπων T4, πρώτη ανάπτυξη μοντέλου, βελτιωμένη εμπειρία σχεδίασης, καλύτερη απόδοση και πλουραλισμό συνόλων οντοτήτων. Οι Υπηρεσίες δεδομένων WCF αποτελούν στοιχείο του .NET Framework, το οποίο σας επιτρέπει να δημιουργήσετε υπηρεσίες και εφαρμογές που βασίζονται σε REST και χρησιμοποιούν το πρωτόκολλο Open Data (OData) για έκθεση και χρήση δεδομένων στο Web. Οι Υπηρεσίες δεδομένων WCF διαθέτουν πολλές νέες δυνατότητες, όπως βελτιωμένη υποστήριξη BLOB, σύνδεση δεδομένων, καταμέτρηση γραμμών, προσαρμογή τροφοδοσίας, προβολές και βελτιώσεις στη διοχέτευση αιτήσεων. Με την ενσωματωμένη ενοποίηση με το Microsoft Office 2010, είναι πλέον δυνατή η έκθεση δεδομένων του Microsoft Office SharePoint Server ως τροφοδοσία OData και η πρόσβαση σε αυτή την τροφοδοσία δεδομένων με χρήση της βιβλιοθήκης - πελάτη των Υπηρεσιών δεδομένων WCF.

6.10 SQL Server

Η Βάση Δεδομένων του πληροφοριακού συστήματος αναπτύχθηκε με κάποια έκδοση του Microsoft SQL Server. Ο Microsoft SQL Server είναι ένα σχεσιακό μοντέλο διαχείρισης βάσεων δεδομένων (RDBMS) του οποίου η υλοποίηση της γλώσσας SQL ονομάζεται T-SQL. Οι λόγοι που χρησιμοποιούμε τα συστήματα διαχείρισης βάσεων δεδομένων είναι επειδή προσφέρουν:

- **Ανεξαρτησία δεδομένων.** Τα δεδομένα είναι ανεξάρτητα από την εφαρμογή που τα χρησιμοποιεί. Μπορούν να δημιουργηθούν εφαρμογές οι οποίες να προσαρμόζονται με σχετική ευκολία σε διαφορετικές πηγές δεδομένων.

- **Ακεραιότητα δεδομένων.** Εξασφαλίζεται ότι δεν υπάρχουν ασυνέπειες μεταξύ εγγραφών που αντιπροσωπεύουν την ίδια πληροφορία, τα δεδομένα 40 συμμορφώνονται με τους περιορισμούς που επιβάλλονται από το σχήμα της βάσης, και οι τιμές των μεταβλητών συμμορφώνονται με το πεδίο τιμών και τον τύπο δεδομένων που έχει οριστεί.
- **Ταυτόχρονη προσπέλαση και κεντρικό έλεγχο.** Εξασφαλίζει την ταυτόχρονη προσπέλαση των δεδομένων και απαλείφει τα λεγόμενα αδιέξοδα (deadlocks).
- **Αποδοτική αποθήκευση και προσπέλαση των δεδομένων.** Χρησιμοποιεί αποδοτικά τον διαθέσιμο χώρο του υπολογιστικού συστήματος, και εισάγει δομές (ευρετήρια, hash tables κλπ) για να επιτυγχάνεται γρήγορη πρόσβαση στα δεδομένα.
- **Αποφυγή πλεονάζουσας πληροφορίας (Non- Redundancy) των δεδομένων.** Το σχεσιακό σύστημα διαχείρισης βάσεων δεδομένων επιχειρεί να εξασφαλίσει ότι δεν αποθηκεύεται πληροφορία η οποία είτε έχει αποθηκευτεί ήδη, είτε μπορεί να εξαχθεί από άλλες, ήδη αποθηκευμένες πληροφορίες. Η δημιουργία και τροποποίηση της βάσης δεδομένων έγινε με την βοήθεια του SQL Server Management Studio. Το SQL Server Management studio είναι μία εφαρμογή με γραφικό περιβάλλον (GUI) και περιλαμβάνει όλα τα απαραίτητα εργαλεία για την επεξεργασία βάσεων δεδομένων, όπως script editors, object explorer και παρέχει αρκετές ευκολίες στον προγραμματιστή, όπως επισήμανση λαθών, Intellisense, αρκετό έτοιμο κώδικα κλπ.

ΑΝΑΛΥΣΗ ΑΠΑΙΤΗΣΕΩΝ

7.1 Εισαγωγή

Ένας ορισμός της παρούσας εφαρμογής είναι ότι αποτελεί μία διαδικτυακή πλατφόρμα η οποία στόχο έχει να βοηθήσει το χρήστη στην καταγραφή των καθημερινών του χρονικών υποχρεώσεων και να την βελτιώνει ανάλογα με το κριτήριο που θα επιλέξει και τους περιορισμούς που υπάρχουν.

Όπως αναφέρθηκε ο σκοπός του χρήστη είναι να καταχωρεί χρόνο θυρίδα και να την δεσμεύει στο προσωπικό του ημερολόγιο. όσο πιο πολλές καταχωρήσεις υπάρχουν τόσο πιο πολλοί περιορισμοί μπορούν να δημιουργηθούν αλλά και τόσο πιο χρήσιμη γίνεται η εξατομίκευση που επιφέρει το σύστημα. Η καταχώρηση μπορεί να έχει τα εξής πιθανά αποτελέσματα:

- Ο χρήστης κάνει την πρώτη του καταχώρηση επιτυχώς.
- Ο χρήστης καταχωρεί το αίτημα του για να το αποδεχτεί ο διοργανωτής της δραστηριότητας.
- Ο χρήστης ενώ καταχωρεί τα στοιχεία για την χρονοθυρίδα που θέλει να δεσμεύσει, το σύστημα επεμβαίνει για να του προτείνει μία καλύτερη λύση δεδομένων των ρυθμίσεων που έχει κάνει, είτε βελτιώνοντας την λύση είτε διαπιστώνοντας ότι αυτή δεν είναι εφικτή.
- Ο χρήστης ενημερώνεται ότι η δέσμευση αυτού του χρόνου δεν είναι εφικτή και δεν μπορεί με κάποιο τρόπο το σύστημα να βρει ένα σενάριο που θα εξυπηρετεί τον χρήστη.

7.2 Το πρότυπο Unified Modelling Language

Το πρώτο βήμα στην ανάπτυξη μιας εφαρμογής είναι να πραγματοποιηθεί ανάλυση για τους στόχους και την εμβέλεια της. Θα πρέπει να διασφαλιστεί ότι η εφαρμογή έχει οριστεί με αρκετή συνέπεια και στη συνέχεια να γίνει ο σχεδιασμός και η υλοποίηση της.

Το πρότυπο Unified Modelling Language (Γλώσσα Μοντελοποίησης UML) δημιουργήθηκε για να περιγράφει και να τεκμηριώνει ανάλυση και σχεδιασμό για συστήματα που χρησιμοποιούν αντικειμενοστραφείς γλώσσες. Στην περίπτωση μας έγινε χρήση της γλώσσας C# και της Html 5. Το κύριο στοιχείο που παρέχει η UML προκειμένου να μοντελοποιήσει τη λειτουργικότητα ενός συστήματος είναι οι περιπτώσεις χρήσης.

7.3 Διαγράμματα χρήσης

Τα διαγράμματα χρήσης αποτελούνται από περιπτώσεις χρήσης, δράστες (actors) και σχέσεις συνάφειας (relationships) μεταξύ των δραστών και των υπηρεσιών του συστήματος. Το πρώτο βήμα στην ανάλυση είναι ο ορισμός του περιβάλλοντος (δραστών) και της συμπεριφοράς που αναμένεται από κάθε δράστη. Επόμενα βήματα είναι ο ορισμός βασικών συμπεριφορών του συστήματος με τη μορφή περιπτώσεων χρήσης, η παραγοντοποίηση κοινών συμπεριφορών σε νέες ενοποιημένες περιπτώσεις χρήσης, η υλοποίηση περιγραφών περιπτώσεων χρήσης και, τέλος, η μοντελοποίηση των περιπτώσεων χρήσης δραστών και σχέσεων με τη μορφή διαγράμματος.

7.4 Δράστες

Οι δράστες της εφαρμογής είναι οι εξής:

- **Διαχειριστής:** Ο διαχειριστής έχει τη δυνατότητα να ρυθμίζει διάφορες παραμέτρους της εφαρμογής ενώ μπορεί να το επεκτείνει προσθέτοντας δεδομένα και ρυθμίσεις που αυξάνουν το εύρος της εφαρμογής.
- **Απλός χρήστης:** Ο απλός χρήστης δημιουργεί λογαριασμό στην ιστοσελίδα και έχει τη δυνατότητα να κάνει χρήση της πλατφόρμας για την καταγραφή δραστηριοτήτων που αφορά τον ίδιο.

- **Χρήστης διοργανωτής:** Ο χρήστης διοργανωτής έχει τη δυνατότητα αφού δημιουργήσει τον αντίστοιχο λογαριασμό να κάνει χρήση της πλατφόρμας για την καταγραφή και διοργάνωση δραστηριοτήτων που αφορά τον ίδιο αλλά και άλλους χρήστες.
- **Χρήστης προώθησης:** Ο χρήστης προώθησης αφού δημιουργήσει τον αντίστοιχο λογαριασμό μπορεί να κάνει χρήση της πλατφόρμας με σκοπό την προώθηση των δραστηριοτήτων που δημιουργεί στους ενδιαφερόμενους χρήστες σύμφωνα με την πλατφόρμα.

7.4.1 Δράστης: Διαχειριστής

Ο διαχειριστής έχει τη δυνατότητα να ρυθμίζει διάφορες παραμέτρους της εφαρμογής και να πραγματοποιεί συνεχώς προσθήκες ώστε να διατηρείται όσο είναι εφικτό, αμείωτο το ενδιαφέρον των χρηστών. Για να προχωρήσει σε τυχόν αλλαγές θα πρέπει να εισέλθει στο σύστημα διαχείρισης της εφαρμογής καταχωρώντας τα στοιχεία εισόδου που έχει.

Περίπτωση χρήσης: Είσοδος διαχειριστή

Κύριος χρήστης: διαχειριστής

Περιγραφή: Ο χρήστης συμπληρώνει το όνομα χρήστη και τον κωδικό που έχει.

Προϋποθέσεις:

- Ο χρήστης θα πρέπει να συμπληρώσει σωστά τα στοιχεία του.

Μετά-συνθήκες:

- Ο χρήστης αποκτά τη δυνατότητα να παραμετροποιήσει την εφαρμογή.

Βασική ροή:

- Ο χρήστης συμπληρώνει τα στοιχεία εισόδου και πατάει το εικονίδιο είσοδος.
- Το σύστημα ελέγχει αν είναι έγκυρα τα στοιχεία του χρήστη και τον εισάγει στο σύστημα διαχείρισης.

Εναλλακτική ροή:

- Ο χρήστης καταχωρεί σε λανθασμένο συνδυασμό ονόματος και κωδικού.
- Το σύστημα ενημερώνει το χρήστη ότι δεν υπάρχει διαχειριστής με τα στοιχεία που έγιναν αποστολή.

Περίπτωση χρήσης: Προβολή τύπων δραστηριοτήτων.

Κύριος χρήστης: διαχειριστής

Περιγραφή: Ο διαχειριστής επιλέγει να δει την λίστα με τους υπάρχοντες τύπους δραστηριοτήτων.

Προϋποθέσεις:

- Ο διαχειριστής θα πρέπει να έχει εισέλθει στο σύστημα διαχείρισης.

Μετά-συνθήκες:

- Ο διαχειριστής θα πρέπει να βλέπει την λίστα με τους υπάρχοντες τύπους δραστηριοτήτων.

Βασική ροή:

- Ο διαχειριστής επιλέγει την ενότητα "Τύποι δραστηριοτήτων".
- Η οθόνη εμφανίζει στην οθόνη τους διαθέσιμους τύπους.

Εναλλακτική ροή:

- Δεν υπάρχει διαθέσιμος τύπος δραστηριότητας.
- Η εφαρμογή εμφανίζει το αντίστοιχο μήνυμα στην οθόνη.

Περίπτωση χρήσης: Δημιουργία τύπου δραστηριότητας.

Κύριος χρήστης: διαχειριστής

Περιγραφή: Ο διαχειριστής δημιουργεί νέο τύπο δραστηριότητας.

Επεκτείνει: Προβολή κατηγοριών

Περιλαμβάνει: Έλεγχο στοιχείων
 Προϋποθέσεις:

- Ο διαχειριστής θα πρέπει να έχει εισέλθει στο σύστημα διαχείρισης.

Μετά-συνθήκες:

- Δημιουργείται νέος τύπος δραστηριότητας.

Βασική ροή:

- Ο διαχειριστής επιλέγει την “Δημιουργία τύπου δραστηριότητας”.
- Πληκτρολογεί τα στοιχεία του νέου τύπου.
- Η εφαρμογή ελέγχει την εγκυρότητα των στοιχείων.
- Η εφαρμογή προσθέτει στην λίστα τύπων δραστηριοτήτων τον νέο τύπο.

Εναλλακτική ροή:

- Τα στοιχεία δεν είναι έγκυρα, οπότε δεν δημιουργείται νέα εγγραφή.

Περίπτωση χρήσης: Επεξεργασία τύπου δραστηριότητας.

Κύριος χρήστης: διαχειριστής
 Περιγραφή: Ο διαχειριστής επεξεργάζεται έναν υπάρχον τύπο δραστηριότητας.
 Επεκτείνει: Προβολή κατηγοριών

Περιλαμβάνει: Έλεγχο στοιχείων
 Προϋποθέσεις:

- Ο διαχειριστής θα πρέπει να έχει εισέλθει στο σύστημα διαχείρισης και βρίσκεται στην οθόνη με την λίστα των τύπων δραστηριοτήτων.

Μετά-συνθήκες:

- Τα στοιχεία του τύπου δραστηριότητας έχουν ενημερωθεί.

Βασική ροή:

- Ο διαχειριστής επιλέγει την επεξεργασία τύπου δραστηριότητας.
- Τροποποιεί τα στοιχεία του τύπου.
- Η εφαρμογή ελέγχει την εγκυρότητα των στοιχείων.
- Η εφαρμογή ενημερώνει τον τύπο δραστηριότητας.

Εναλλακτική ροή:

- Τα στοιχεία δεν είναι έγκυρα, οπότε δεν τροποποιείται η εγγραφή.

Περίπτωση χρήσης: Διαγραφή τύπου δραστηριότητας.

Κύριος χρήστης: διαχειριστής
 Περιγραφή: Ο διαχειριστής διαγράφει έναν τύπο δραστηριότητας.
 Επεκτείνει: Προβολή κατηγοριών

Περιλαμβάνει: Έλεγχο στοιχείων
 Προϋποθέσεις:

- Ο διαχειριστής θα πρέπει να έχει εισέλθει στο σύστημα διαχείρισης και βρίσκεται στην οθόνη με την λίστα των τύπων δραστηριοτήτων.

Μετά-συνθήκες:

- Ο τύπος δραστηριότητας διαγράφεται.

Βασική ροή:

- Ο διαχειριστής επιλέγει την διαγραφή ενός συγκεκριμένου τύπου δραστηριότητας.
- Η εφαρμογή ελέγχει την εγκυρότητα των στοιχείων.
- Η εφαρμογή διαγράφει τον τύπο.

Εναλλακτική ροή:

- Τα στοιχεία δεν είναι έγκυρα, οπότε δεν γίνεται καμία διαγραφή.

Περίπτωση χρήσης: Προβολή τοποθεσιών

Κύριος χρήστης: διαχειριστής
 Περιγραφή: Ο διαχειριστής επιλέγει να δει την λίστα με τις υπάρχουσες τοποθεσίες.
 Προϋποθέσεις:

- Ο διαχειριστής θα πρέπει να έχει εισέλθει στο σύστημα διαχείρισης.

Μετά-συνθήκες:

- Ο διαχειριστής θα πρέπει να βλέπει την λίστα με τις υπάρχουσες τοποθεσίες.

Βασική ροή:

- Ο διαχειριστής επιλέγει την ενότητα “Τοποθεσίες”.
- Η οθόνη εμφανίζει στην οθόνη τις διαθέσιμες τοποθεσίες.

Εναλλακτική ροή:

- Δεν υπάρχει διαθέσιμη τοποθεσία.
- Η εφαρμογή εμφανίζει το αντίστοιχο μήνυμα στην οθόνη.

Περίπτωση χρήσης: Δημιουργία τοποθεσίας.

Κύριος χρήστης: διαχειριστής

Περιγραφή: Ο διαχειριστής δημιουργεί νέα τοποθεσία.

Επεκτείνει: Προβολή τοποθεσιών

Περιλαμβάνει: Έλεγχο στοιχείων

Προϋποθέσεις:

- Ο διαχειριστής θα πρέπει να έχει εισέλθει στο σύστημα διαχείρισης.

Μετά-συνθήκες:

- Δημιουργείται νέα τοποθεσία.

Βασική ροή:

- Ο διαχειριστής επιλέγει την “Δημιουργία τοποθεσίας”.
- Πληκτρολογεί τα στοιχεία τις νέας τοποθεσίας.
- Η εφαρμογή ελέγχει την εγκυρότητα των στοιχείων.
- Η εφαρμογή προσθέτει στην λίστα τοποθεσιών την νέα τοποθεσία.

Εναλλακτική ροή:

- Τα στοιχεία δεν είναι έγκυρα, οπότε δεν δημιουργείται νέα εγγραφή.

Περίπτωση χρήσης: Επεξεργασία τοποθεσίας.

Κύριος χρήστης: διαχειριστής

Περιγραφή: Ο διαχειριστής επεξεργάζεται μία υπάρχουσα τοποθεσία.

Επεκτείνει: Προβολή τοποθεσιών

Περιλαμβάνει: Έλεγχο στοιχείων

Προϋποθέσεις:

- Ο διαχειριστής θα πρέπει να έχει εισέλθει στο σύστημα διαχείρισης και βρίσκεται στην οθόνη με την λίστα των τοποθεσιών.

Μετά-συνθήκες:

- Τα στοιχεία της τοποθεσίας έχουν ενημερωθεί.

Βασική ροή:

- Ο διαχειριστής επιλέγει την επεξεργασία τοποθεσίας.
- Τροποποιεί τα στοιχεία της τοποθεσίας.
- Η εφαρμογή ελέγχει την εγκυρότητα των στοιχείων.
- Η εφαρμογή ενημερώνει την τοποθεσία.

Εναλλακτική ροή:

- Τα στοιχεία δεν είναι έγκυρα, οπότε δεν τροποποιείται η εγγραφή.

Περίπτωση χρήσης: Διαγραφή τοποθεσίας.

Κύριος χρήστης: διαχειριστής

Περιγραφή: Ο διαχειριστής διαγράφει μία τοποθεσία.

Επεκτείνει: Προβολή τοποθεσιών

Περιλαμβάνει: Έλεγχο στοιχείων
 Προϋποθέσεις:

- Ο διαχειριστής θα πρέπει να έχει εισέλθει στο σύστημα διαχείρισης και βρίσκεται στην οθόνη με την λίστα των τοποθεσιών.

Μετά-συνθήκες:

- Η τοποθεσία διαγράφεται.

Βασική ροή:

- Ο διαχειριστής επιλέγει την διαγραφή μιας συγκεκριμένης τοποθεσίας.
- Η εφαρμογή ελέγχει την εγκυρότητα των στοιχείων.
- Η εφαρμογή διαγράφει την τοποθεσία.

Εναλλακτική ροή:

- Τα στοιχεία δεν είναι έγκυρα, οπότε δεν γίνεται καμία διαγραφή.

Περίπτωση χρήσης: Προβολή χρηστών.

Κύριος χρήστης: διαχειριστής

Περιγραφή: Ο διαχειριστής επιλέγει να δει την λίστα με τους υπάρχοντες χρήστες.

Προϋποθέσεις:

- Ο διαχειριστής θα πρέπει να έχει εισέλθει στο σύστημα διαχείρισης.

Μετά-συνθήκες:

- Ο διαχειριστής θα πρέπει να βλέπει την λίστα με τους υπάρχοντες χρήστες.

Βασική ροή:

- Ο διαχειριστής επιλέγει την ενότητα “Χρήστες”.
- Η οθόνη εμφανίζει στην οθόνη τους διαθέσιμους χρήστες.

Εναλλακτική ροή:

- Δεν υπάρχει διαθέσιμος χρήστης.
- Η εφαρμογή εμφανίζει το αντίστοιχο μήνυμα στην οθόνη.

Περίπτωση χρήσης: Δημιουργία χρήστη.

Κύριος χρήστης: διαχειριστής

Περιγραφή: Ο διαχειριστής δημιουργεί νέο χρήστη.

Επεκτείνει: Προβολή χρηστών

Περιλαμβάνει: Έλεγχο στοιχείων

Προϋποθέσεις:

- Ο διαχειριστής θα πρέπει να έχει εισέλθει στο σύστημα διαχείρισης.

Μετά-συνθήκες:

- Δημιουργείται νέος χρήστης.

Βασική ροή:

- Ο διαχειριστής επιλέγει την “Δημιουργία χρήστη”.
- Πληκτρολογεί τα στοιχεία του νέου τύπου.
- Η εφαρμογή ελέγχει την εγκυρότητα των στοιχείων.
- Η εφαρμογή προσθέτει στην λίστα χρηστών τον νέο χρήστη.

Εναλλακτική ροή:

- Τα στοιχεία δεν είναι έγκυρα, οπότε δεν δημιουργείται νέα εγγραφή.

Περίπτωση χρήσης: Επεξεργασία χρήστη.

Κύριος χρήστης: διαχειριστής

Περιγραφή: Ο διαχειριστής επεξεργάζεται έναν υπάρχον χρήστη.

Επεκτείνει: Προβολή χρηστών

Περιλαμβάνει: Έλεγχο στοιχείων

Προϋποθέσεις:

- Ο διαχειριστής θα πρέπει να έχει εισέλθει στο σύστημα διαχείρισης και βρίσκεται στην οθόνη με την λίστα των χρηστών.

Μετά-συνθήκες:

- Τα στοιχεία του χρήστη έχουν ενημερωθεί.

Βασική ροή:

- Ο διαχειριστής επιλέγει την επεξεργασία χρήστη.
- Τροποποιεί τα στοιχεία του χρήστη.
- Η εφαρμογή ελέγχει την εγκυρότητα των στοιχείων.
- Η εφαρμογή ενημερώνει τον χρήστη.

Εναλλακτική ροή:

- Τα στοιχεία δεν είναι έγκυρα, οπότε δεν τροποποιείται η εγγραφή.

Περίπτωση χρήσης: Διαγραφή χρήστη.

Κύριος χρήστης: διαχειριστής

Περιγραφή: Ο διαχειριστής διαγράφει έναν χρήστη.

Επεκτείνει: Προβολή χρήστη

Περιλαμβάνει: Έλεγχο στοιχείων

Προϋποθέσεις:

- Ο διαχειριστής θα πρέπει να έχει εισέλθει στο σύστημα διαχείρισης και βρίσκεται στην οθόνη με την λίστα των χρηστών.

Μετά-συνθήκες:

- Ο χρήστης διαγράφεται.

Βασική ροή:

- Ο διαχειριστής επιλέγει την διαγραφή ενός συγκεκριμένου χρήστη.
- Η εφαρμογή ελέγχει την εγκυρότητα των στοιχείων.
- Η εφαρμογή διαγράφει τον χρήστη.

Εναλλακτική ροή:

- Τα στοιχεία δεν είναι έγκυρα, οπότε δεν γίνεται καμία διαγραφή.

7.4.2 Δράστης: Απλός χρήστης

Περίπτωση χρήσης: Εγγραφή χρήστη

Κύριος χρήστης: Απλός χρήστης

Περιγραφή: Ο χρήστης συμπληρώνει το όνομα χρήστη, κωδικό και τα υπόλοιπα στοιχεία που θέλει, όπως κριτήρια επιλογής χρονοθυρίδας.

Προϋποθέσεις:

- Ο χρήστης θα πρέπει να συμπληρώσει τα απαραίτητα στοιχεία για την εγγραφή όπως όνομα χρήστη, κωδικό και ηλεκτρονική διεύθυνση αλληλογραφίας.

Μετά-συνθήκες:

- Ο χρήστης αποκτά τη δυνατότητα να συνδεθεί στην εφαρμογή.

Βασική ροή:

- Ο χρήστης συμπληρώνει τα στοιχεία εισόδου και πατάει το εικονίδιο εγγραφής.
- Το σύστημα ελέγχει αν είναι έγκυρα τα στοιχεία του χρήστη και τον εισάγει στο σύστημα πλοήγησης.

Εναλλακτική ροή:

- Ο χρήστης καταχωρεί σε λανθασμένο συνδυασμό ονόματος και κωδικού.
- Το σύστημα ενημερώνει το χρήστη ότι δεν υπάρχει χρήστης με τα στοιχεία που έγιναν αποστολή.

Περίπτωση χρήσης: Είσοδος χρήστη

Κύριος χρήστης: Απλός χρήστης

Περιγραφή: Ο χρήστης συμπληρώνει το όνομα χρήστη και τον κωδικό που έχει.

Προϋποθέσεις:

- Ο χρήστης θα πρέπει να συμπληρώσει σωστά τα στοιχεία του.

Μετά-συνθήκες:

- Ο χρήστης αποκτά τη δυνατότητα να παραμετροποιήσει την εφαρμογή.

Βασική ροή:

- Ο χρήστης συμπληρώνει τα στοιχεία εισόδου και πατάει το εικονίδιο είσοδος.
- Το σύστημα ελέγχει αν είναι έγκυρα τα στοιχεία του χρήστη και τον εισάγει στο σύστημα πλοήγησης.

Εναλλακτική ροή:

- Ο χρήστης καταχωρεί σε λανθασμένο συνδυασμό ονόματος και κωδικού.
- Το σύστημα ενημερώνει το χρήστη ότι δεν υπάρχει χρήστης με τα στοιχεία που έγιναν αποστολή.

Περίπτωση χρήσης: Επεξεργασία προφίλ χρήστη.

Κύριος χρήστης: απλός χρήστης

Περιγραφή: Ο απλός χρήστης επεξεργάζεται το δικό του προφίλ χρήστη.

Επεκτείνει: Προβολή προφίλ

Περιλαμβάνει: Έλεγχο στοιχείων

Προϋποθέσεις:

- Ο χρήστης θα πρέπει να εισέλθει στο σύστημα διαχείρισης του προφίλ του από το εικονίδιο στην μενού της εφαρμογή.

Μετά-συνθήκες:

- Τα στοιχεία του χρήστη έχουν ενημερωθεί.

Βασική ροή:

- Τροποποιεί τα στοιχεία του χρήστη.
- Η εφαρμογή ελέγχει την εγκυρότητα των στοιχείων.
- Η εφαρμογή ενημερώνει τον χρήστη.

Εναλλακτική ροή:

- Τα στοιχεία δεν είναι έγκυρα, οπότε δεν τροποποιείται η εγγραφή.

Περίπτωση χρήσης: Προβολή δραστηριοτήτων

Κύριος χρήστης: απλός χρήστης

Περιγραφή: Ο απλός χρήστης επιλέγει να δει την λίστα με τις υπάρχουσες δραστηριότητες.

Προϋποθέσεις:

- Ο απλός χρήστης θα πρέπει να έχει εισέλθει στην εφαρμογή με την χρήση ονόματος χρήστη και κωδικού.

Μετά-συνθήκες:

- Ο απλός χρήστης θα πρέπει να βλέπει την λίστα με τις υπάρχουσες δραστηριότητες.

Βασική ροή:

- Ο απλός χρήστης επιλέγει την ενότητα “Δραστηριότητες”.
- Η οθόνη εμφανίζει στην οθόνη τις διαθέσιμες δραστηριότητες.

Εναλλακτική ροή:

- Δεν υπάρχει διαθέσιμη δραστηριότητα.
- Η εφαρμογή εμφανίζει το αντίστοιχο μήνυμα στην οθόνη.

Περίπτωση χρήσης: Δημιουργία δραστηριότητας

Κύριος χρήστης: απλός χρήστης

Περιγραφή: Ο απλός χρήστης δημιουργεί νέα δραστηριότητα.

Επεκτείνει: Προβολή δραστηριοτήτων

Περιλαμβάνει: Έλεγχο στοιχείων

Προϋποθέσεις:

- Ο χρήστης θα πρέπει να έχει εισέλθει στην εφαρμογή.

Μετά-συνθήκες:

- Δημιουργείται νέα δραστηριότητα.

Βασική ροή:

- Ο χρήστης επιλέγει την “Δημιουργία δραστηριότητας”.
- Πληκτρολογεί τα στοιχεία τις νέας δραστηριότητας.
- Η εφαρμογή ελέγχει την εγκυρότητα των στοιχείων.
- Η εφαρμογή προσθέτει στην λίστα δραστηριοτήτων την νέα δραστηριότητα.

Εναλλακτική ροή:

- Τα στοιχεία δεν είναι έγκυρα, οπότε δεν δημιουργείται νέα εγγραφή.

Περίπτωση χρήσης: Επεξεργασία δραστηριότητας.

Κύριος χρήστης: απλός χρήστης

Περιγραφή: Ο απλός χρήστης επεξεργάζεται μία υπάρχουσα δραστηριότητα.

Επεκτείνει: Προβολή δραστηριοτήτων

Περιλαμβάνει: Έλεγχο στοιχείων

Προϋποθέσεις:

- Ο χρήστης θα πρέπει να έχει εισέλθει στο σύστημα διαχείρισης και βρίσκεται στην οθόνη με την λίστα των δραστηριοτήτων.

Μετά-συνθήκες:

- Τα στοιχεία της δραστηριότητας έχουν ενημερωθεί.

Βασική ροή:

- Ο χρήστης επιλέγει την επεξεργασία δραστηριότητας.
- Τροποποιεί τα στοιχεία της δραστηριότητας.
- Η εφαρμογή ελέγχει την εγκυρότητα των στοιχείων.
- Η εφαρμογή ενημερώνει την δραστηριότητα.

Εναλλακτική ροή:

- Τα στοιχεία δεν είναι έγκυρα, οπότε δεν τροποποιείται η εγγραφή.

Περίπτωση χρήσης: Διαγραφή δραστηριότητας.

Κύριος χρήστης: απλός χρήστης

Περιγραφή: Ο απλός χρήστης διαγράφει μία δραστηριότητα.

Επεκτείνει: Προβολή δραστηριοτήτων

Περιλαμβάνει: Έλεγχο στοιχείων

Προϋποθέσεις:

- Ο χρήστης θα πρέπει να έχει εισέλθει στο σύστημα διαχείρισης και βρίσκεται στην οθόνη με την λίστα των δραστηριοτήτων.

Μετά-συνθήκες:

- Η δραστηριότητα διαγράφεται.

Βασική ροή:

- Ο χρήστης επιλέγει την διαγραφή μιας συγκεκριμένης δραστηριότητας.
- Η εφαρμογή ελέγχει την εγκυρότητα των στοιχείων.
- Η εφαρμογή διαγράφει την δραστηριότητα.

Εναλλακτική ροή:

- Τα στοιχεία δεν είναι έγκυρα, οπότε δεν γίνεται καμία διαγραφή.

7.4.3 Δράστης: Χρήστης διοργανωτής

Ο χρήστης διοργανωτής συμμετέχει με τον ίδιο τρόπο σε όλες τις παραπάνω περιπτώσεις χρήσης όπως ο απλός χρήστης. Επιπρόσθετα μπορεί να συμμετέχει και στις εξής περιπτώσεις χρήσης.

Περίπτωση χρήσης: Προβολή συλλογικών δραστηριοτήτων

Κύριος χρήστης: διοργανωτής

Περιγραφή: Ο χρήστης διοργανωτής επιλέγει να δει την λίστα με τις υπάρχουσες συλλογικές

δραστηριότητες.

Προϋποθέσεις:

- Ο χρήστης διοργανωτής θα πρέπει να έχει εισέλθει στην εφαρμογή με την χρήση ονόματος χρήστη και κωδικού.

Μετά-συνθήκες:

- Ο χρήστης διοργανωτής θα πρέπει να βλέπει την λίστα με τις υπάρχουσες συλλογικές δραστηριότητες.

Βασική ροή:

- Ο χρήστης διοργανωτής επιλέγει την ενότητα “Συλλογικές δραστηριότητες”.
- Η οθόνη εμφανίζει στην οθόνη τις διαθέσιμες συλλογικές δραστηριότητες.

Εναλλακτική ροή:

- Δεν υπάρχει διαθέσιμη συλλογική δραστηριότητα.
- Η εφαρμογή εμφανίζει το αντίστοιχο μήνυμα στην οθόνη.

Περίπτωση χρήσης: Δημιουργία συλλογικής δραστηριότητας

Κύριος χρήστης: χρήστης διοργανωτής

Περιγραφή: Ο χρήστης διοργανωτής δημιουργεί νέα συλλογική δραστηριότητα.

Επεκτείνει: Προβολή συλλογικών δραστηριοτήτων

Περιλαμβάνει: Έλεγχο στοιχείων

Προϋποθέσεις:

- Ο χρήστης θα πρέπει να έχει εισέλθει στην εφαρμογή.

Μετά-συνθήκες:

- Δημιουργείται νέα συλλογική δραστηριότητα.

Βασική ροή:

- Ο χρήστης επιλέγει την “Δημιουργία συλλογικής δραστηριότητας”.
- Πληκτρολογεί τα στοιχεία τις νέας συλλογικής δραστηριότητας.
- Η εφαρμογή ελέγχει την εγκυρότητα των στοιχείων.
- Η εφαρμογή προσθέτει στην λίστα συλλογικών δραστηριοτήτων την νέα συλλογική δραστηριότητα.

Εναλλακτική ροή:

- Τα στοιχεία δεν είναι έγκυρα, οπότε δεν δημιουργείται νέα εγγραφή.

Περίπτωση χρήσης: Επεξεργασία συλλογικής δραστηριότητας.

Κύριος χρήστης: χρήστης διοργανωτής

Περιγραφή: Ο χρήστης διοργανωτής επεξεργάζεται μία υπάρχουσα συλλογική δραστηριότητα.

Επεκτείνει: Προβολή συλλογικών δραστηριοτήτων

Περιλαμβάνει: Έλεγχο στοιχείων

Προϋποθέσεις:

- Ο χρήστης θα πρέπει να έχει εισέλθει στο σύστημα διαχείρισης και βρίσκεται στην οθόνη με την λίστα των συλλογικών δραστηριοτήτων.

Μετά-συνθήκες:

- Τα στοιχεία της συλλογικής δραστηριότητας έχουν ενημερωθεί.

Βασική ροή:

- Ο χρήστης επιλέγει την επεξεργασία συλλογικής δραστηριότητας.
- Τροποποιεί τα στοιχεία της συλλογικής δραστηριότητας.
- Η εφαρμογή ελέγχει την εγκυρότητα των στοιχείων.
- Η εφαρμογή ενημερώνει την συλλογική δραστηριότητα.

Εναλλακτική ροή:

- Τα στοιχεία δεν είναι έγκυρα, οπότε δεν τροποποιείται η εγγραφή.

Περίπτωση χρήσης: Διαγραφή συλλογικής δραστηριότητας.

Κύριος	χρήστης:	χρήστης	διοργανωτής
Περιγραφή: Ο χρήστης διοργανωτής διαγράφει μία συλλογική δραστηριότητα.			
Επεκτείνει: Προβολή συλλογικών δραστηριοτήτων			
Περιλαμβάνει:	Έλεγχο		στοιχείων
Προϋποθέσεις:			
<ul style="list-style-type: none"> Ο χρήστης θα πρέπει να έχει εισέλθει στο σύστημα διαχείρισης και βρίσκεται στην οθόνη με την λίστα των συλλογικών δραστηριοτήτων. 			
Μετά-συνθήκες:			
<ul style="list-style-type: none"> Η συλλογική δραστηριότητα διαγράφεται. 			
Βασική ροή:			
<ul style="list-style-type: none"> Ο χρήστης επιλέγει την διαγραφή μιας συγκεκριμένης συλλογικής δραστηριότητας. Η εφαρμογή ελέγχει την εγκυρότητα των στοιχείων. Η εφαρμογή διαγράφει την συλλογική δραστηριότητα. 			
Εναλλακτική ροή:			
<ul style="list-style-type: none"> Τα στοιχεία δεν είναι έγκυρα, οπότε δεν γίνεται καμία διαγραφή. 			

7.4.4 Δράστης: Χρήστης προώθησης

Ο χρήστης προώθησης συμμετέχει με τον ίδιο τρόπο σε όλες τις παραπάνω περιπτώσεις χρήσης όπως ο χρήστης διοργανωτής. Επιπρόσθετα μπορεί να συμμετέχει και στις εξής περιπτώσεις χρήσης.

Περίπτωση χρήσης: Προβολή προωθητικών δραστηριοτήτων

Κύριος χρήστης: χρήστης προώθησης

Περιγραφή: Ο χρήστης προώθησης επιλέγει να δει την λίστα με τις υπάρχουσες προωθητικές δραστηριότητες.

Προϋποθέσεις:

- Ο χρήστης προώθησης θα πρέπει να έχει εισέλθει στην εφαρμογή με την χρήση ονόματος χρήστη και κωδικού.

Μετά-συνθήκες:

- Ο χρήστης προώθησης θα πρέπει να βλέπει την λίστα με τις υπάρχουσες προωθητικές δραστηριότητες.

Βασική ροή:

- Ο χρήστης προώθησης επιλέγει την ενότητα “Προωθητικές δραστηριότητες”.
- Η οθόνη εμφανίζει στην οθόνη τις διαθέσιμες προωθητικές δραστηριότητες.

Εναλλακτική ροή:

- Δεν υπάρχει διαθέσιμη προωθητική δραστηριότητα.
- Η εφαρμογή εμφανίζει το αντίστοιχο μήνυμα στην οθόνη.

Περίπτωση χρήσης: Δημιουργία προωθητικής δραστηριότητας

Κύριος χρήστης: χρήστης προώθησης

Περιγραφή: Ο χρήστης προώθησης δημιουργεί νέα προωθητική δραστηριότητα.

Επεκτείνει: Προβολή προωθητικών δραστηριοτήτων

Περιλαμβάνει: Έλεγχο

στοιχείων

Προϋποθέσεις:

- Ο χρήστης θα πρέπει να έχει εισέλθει στην εφαρμογή.

Μετά-συνθήκες:

- Δημιουργείται νέα προωθητική δραστηριότητα.

Βασική ροή:

- Ο χρήστης επιλέγει την “Δημιουργία προωθητικής δραστηριότητας”.
- Πληκτρολογεί τα στοιχεία τις νέας προωθητικής δραστηριότητας.
- Η εφαρμογή ελέγχει την εγκυρότητα των στοιχείων.

- Η εφαρμογή προσθέτει στην λίστα προωθητικών δραστηριοτήτων την νέα προωθητική δραστηριότητα.

Εναλλακτική ροή:

- Τα στοιχεία δεν είναι έγκυρα, οπότε δεν δημιουργείται νέα εγγραφή.

Περίπτωση χρήσης: Επεξεργασία προωθητικής δραστηριότητας.

Κύριος χρήστης: χρήστης χρήστης προώθησης

Περιγραφή: Ο χρήστης προώθησης επεξεργάζεται μία υπάρχουσα προωθητική δραστηριότητα.

Επεκτείνει: Προβολή προωθητικών δραστηριοτήτων

Περιλαμβάνει: Έλεγχο στοιχείων

Προϋποθέσεις:

- Ο χρήστης θα πρέπει να έχει εισέλθει στο σύστημα διαχείρισης και βρίσκεται στην οθόνη με την λίστα των προωθητικών δραστηριοτήτων.

Μετά-συνθήκες:

- Τα στοιχεία της προωθητικής δραστηριότητας έχουν ενημερωθεί.

Βασική ροή:

- Ο χρήστης επιλέγει την επεξεργασία προωθητικής δραστηριότητας.
- Τροποποιεί τα στοιχεία της προωθητικής δραστηριότητας.
- Η εφαρμογή ελέγχει την εγκυρότητα των στοιχείων.
- Η εφαρμογή ενημερώνει την προωθητική δραστηριότητα.

Εναλλακτική ροή:

- Τα στοιχεία δεν είναι έγκυρα, οπότε δεν τροποποιείται η εγγραφή.

Περίπτωση χρήσης: Διαγραφή προωθητικής δραστηριότητας.

Κύριος χρήστης: χρήστης χρήστης προώθησης

Περιγραφή: Ο χρήστης προώθησης διαγράφει μία προωθητική δραστηριότητα.

Επεκτείνει: Προβολή προωθητικών δραστηριοτήτων

Περιλαμβάνει: Έλεγχο στοιχείων

Προϋποθέσεις:

- Ο χρήστης θα πρέπει να έχει εισέλθει στο σύστημα διαχείρισης και βρίσκεται στην οθόνη με την λίστα των προωθητικών δραστηριοτήτων.

Μετά-συνθήκες:

- Η προωθητική δραστηριότητα διαγράφεται.

Βασική ροή:

- Ο χρήστης επιλέγει την διαγραφή μιας συγκεκριμένης προωθητικής δραστηριότητας.
- Η εφαρμογή ελέγχει την εγκυρότητα των στοιχείων.
- Η εφαρμογή διαγράφει την προωθητική δραστηριότητα.

Εναλλακτική ροή:

- Τα στοιχεία δεν είναι έγκυρα, οπότε δεν γίνεται καμία διαγραφή.

🔗 ΣΧΕΔΙΑΣΜΟΣ ΚΑΙ ΥΛΟΠΟΙΗΣΗ

8.1 Εισαγωγή

Για να είναι η εφαρμογή προσαρμοστική θα πρέπει με κάποιο τρόπο να αποθηκεύουμε στο σύστημα διάφορα δεδομένα για τον κάθε χρήστη και ανάλογα εκείνο να προσαρμόζεται σε αυτά μεταβάλλοντας κάποια στοιχεία των καταχωρήσεων. Αυτό καθίσταται δυνατό με τη χρήση της βάσης δεδομένων για να αποθηκεύουμε τις επιλογές του χρήστη και με τους αντίστοιχους αλγόριθμους που υλοποιούν την εφαρμογή των κριτηρίων.

8.2 Βάση δεδομένων

8.2.1 Εννοιολογικός σχεδιασμός της βάσης

Οντότητα Χρήστη (User)

Η οντότητα *User* περιέχει τις πληροφορίες για το κάθε χρήστη. Πιο συγκεκριμένα, καταχωρείται το όνομα του χρήστη, ο κωδικός εισόδου του στην εφαρμογή, το είδος του χρήστη, τα στοιχεία επικοινωνίας καθώς και ημερομηνίες εγγραφής και τελευταίας σύνδεσης.

Οντότητα Δραστηριότητα (Activity)

Η οντότητα *Activity* περιέχει τις πληροφορίες της κάθε δραστηριότητας. Πιο συγκεκριμένα, καταχωρείται το όνομα, ο τύπος, η ημερομηνία και ώρα έναρξης και λήξης και ο αριθμός των συμμετεχόντων σε αυτή.

Οντότητα Τύπος Δραστηριότητας (Activity Type)

Η οντότητα *Activity Type* περιέχει τις πληροφορίες όπως το όνομα και η περιγραφή.

Οντότητα Τοποθεσία (Location)

Η οντότητα *Location* περιέχει τις πληροφορίες της κάθε τοποθεσίας στις οποίες θα πραγματοποιηθούν οι δραστηριότητες. Πιο συγκεκριμένα, καταχωρείται το όνομα, η περιγραφή, το γεωγραφικό μήκος και πλάτος του σημείου ενδιαφέροντος και το σημείο επικοινωνίας.

Οντότητα Ρυθμίσεις (Settings)

Η οντότητα *Settings* περιέχει τις πληροφορίες που ο κάθε χρήστης επιλέγει στο προφίλ του σχετικά με την πολιτική εξατομίκευσης που χρησιμοποιεί η εφαρμογή. Πιο συγκεκριμένα, έχει τις εγγραφές που περιέχουν την ιεραρχία των κριτηρίων αλλά και την ενεργοποίησή τους ή όχι.

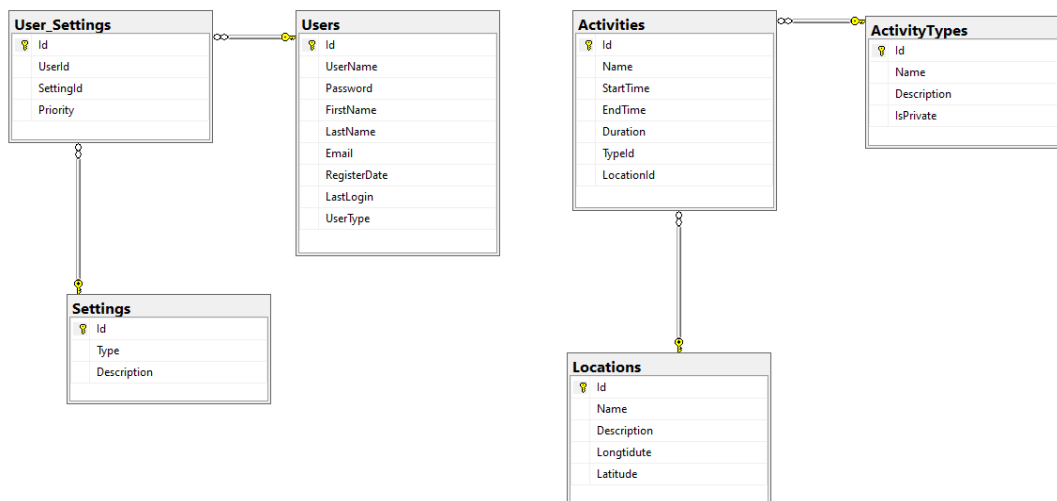
8.2.2 Λογικός σχεδιασμός της βάσης δεδομένων

Στην ενότητα αυτή θα πρέπει να μετασχηματιστεί η σχέση οντοτήτων - συσχετίσεων σε ένα μοντέλο οργάνωσης των δεδομένων που λαμβάνει υπόψη του το σύστημα διαχείρισης στο οποίο θα υλοποιηθεί η βάση δεδομένων. Από τα διάφορα μοντέλα που έχουν εμφανιστεί κατά καιρούς, το σχεσιακό μοντέλο έχει επικρατήσει και χρησιμοποιείται περισσότερο. Το σύστημα διαχείρισης βάσης δεδομένων που επιλέχθηκε για την εφαρμογή είναι η MS SQL η οποία έχει δημιουργηθεί από την Microsoft.

Στη συνέχεια παρουσιάζονται οι πίνακες της βάσης δεδομένων του συστήματος στους οποίους απεικονίζεται το διάγραμμα σχεσιακής βάσης δεδομένων και αποτελούν το λογικό σχήμα της βάσης. Επιπλέον διακρίνονται οι πίνακες με τα χαρακτηριστικά τους. Κατά τη μετατροπή του εννοιολογικού σχήματος σε λογικό, πραγματοποιήθηκε η κανονικοποίηση των πινάκων, ώστε η εκτέλεση των ερωτημάτων στη βάση να είναι αποδοτική και γρήγορη. Το συνολικό σχήμα χωρίστηκε πάλι σε δύο σχήματα το κάθε ένα από τα οποία αντιστοιχεί στο διαγράμματα σχεδιάσαμε.

Στα σχήματα που ακολουθούν τα ονόματα των πινάκων σημειώνονται με γκρι φόντο, τα πρωτεύοντα κλειδιά σημειώνονται στην πρώτη γραμμή με ένα εικονίδιο κίτρινου κλειδιού στα

αριστερά τους ενώ τα ξένα κλειδιά έχουν μία ευθεία γραμμή με το αντίστοιχο κίτρινο κλειδί στο τέλος της. Να σημειωθεί επίσης ότι, όσοι πίνακες έχουν δύο κίτρινα κλειδιά τα συγκεκριμένα παιδιά είναι ξένα κλειδιά αλλά ο συνδυασμός τους αποτελεί το πρωτεύον κλειδί του πίνακα. Τα παιδιά που είναι ξένα κλειδιά σε έναν πίνακα παίρνουν τις τιμές τους από τα παιδιά που είναι πρωτεύοντα κλειδιά στον πίνακα με τον οποίο συνδέονται.



Ακολουθούν οι εντολές για την δημιουργία της βάσης δεδομένων SQL

```

USE [MeetUpDB]
GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
  
```

Για την δημιουργία του πίνακα Activities:

```

CREATE TABLE [dbo].[Activities](
    [Id] [int] IDENTITY(1,1) NOT NULL,
    [Name] [nvarchar](50) NULL,
    [StartTime] [datetime] NULL,
    [EndTime] [datetime] NULL,
    [Duration] [int] NULL,
    [TypeId] [int] NULL,
    [LocationId] [int] NULL,
    CONSTRAINT [PK_MeetUps] PRIMARY KEY CLUSTERED
    (
        [Id] ASC
    )WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
    ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON,
    OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY]
  
```

Για την δημιουργία του πίνακα ActivityTypes:

```

GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[ActivityTypes](
    [Id] [int] IDENTITY(1,1) NOT NULL,
    [Name] [nvarchar](50) NULL,
    [Description] [nvarchar](max) NULL,
    [IsPrivate] [bit] NULL,
    CONSTRAINT [PK_MeetUpTypes] PRIMARY KEY CLUSTERED
(
    [Id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY =
OFF,    ALLOW_ROW_LOCKS    =    ON,    ALLOW_PAGE_LOCKS    =    ON,
OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]

```

Για την δημιουργία του πίνακα Locations:

```

GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Locations](
    [Id] [int] IDENTITY(1,1) NOT NULL,
    [Name] [nvarchar](50) NULL,
    [Description] [nvarchar](max) NULL,
    [Longitude] [decimal](9, 6) NULL,
    [Latitude] [decimal](8, 6) NULL,
    CONSTRAINT [PK_Locations] PRIMARY KEY CLUSTERED
(
    [Id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY =
OFF,    ALLOW_ROW_LOCKS    =    ON,    ALLOW_PAGE_LOCKS    =    ON,
OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]

```

Για την δημιουργία του πίνακα Settings:

```

GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Settings](
    [Id] [int] IDENTITY(1,1) NOT NULL,
    [Type] [nvarchar](50) NULL,
    [Description] [nvarchar](50) NULL,
    CONSTRAINT [PK_Settings] PRIMARY KEY CLUSTERED
(
    [Id] ASC

```

```
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY =
OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON,
OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY]
```

Για την δημιουργία του πίνακα User_Settings:

```
GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[User_Settings](
    [Id] [int] IDENTITY(1,1) NOT NULL,
    [UserId] [int] NULL,
    [SettingId] [int] NULL,
    [Priority] [int] NULL,
    CONSTRAINT [PK_User_Settings] PRIMARY KEY CLUSTERED
(
    [Id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY =
OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON,
OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY]
```

Για την δημιουργία του πίνακα Users:

```
GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Users](
    [Id] [int] IDENTITY(1,1) NOT NULL,
    [UserName] [nvarchar](50) NULL,
    [Password] [nvarchar](50) NULL,
    [FirstName] [nvarchar](50) NULL,
    [LastName] [nvarchar](50) NULL,
    [Email] [nvarchar](100) NULL,
    [RegisterDate] [datetime] NULL,
    [LastLogin] [datetime] NULL,
    [UserType] [nvarchar](50) NULL,
    CONSTRAINT [PK_Users] PRIMARY KEY CLUSTERED
(
    [Id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY =
OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON,
OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY]
```

Για την δημιουργία των δευτερεύοντων κλειδιών:

```
GO
ALTER TABLE [dbo].[Activities] WITH CHECK ADD CONSTRAINT
[FK_Activities_ActivityTypes] FOREIGN KEY([TypeId])
REFERENCES [dbo].[ActivityTypes] ([Id])
```

```
GO
ALTER TABLE [dbo].[Activities] CHECK CONSTRAINT [FK_Activities_ActivityTypes]
GO
ALTER TABLE [dbo].[Activities] WITH CHECK ADD CONSTRAINT
[FK_Activities_Locations] FOREIGN KEY([LocationId])
REFERENCES [dbo].[Locations] ([Id])
GO
ALTER TABLE [dbo].[Activities] CHECK CONSTRAINT [FK_Activities_Locations]
GO
ALTER TABLE [dbo].[User_Settings] WITH CHECK ADD CONSTRAINT
[FK_User_Settings_Settings] FOREIGN KEY([SettingId])
REFERENCES [dbo].[Settings] ([Id])
GO
ALTER TABLE [dbo].[User_Settings] CHECK CONSTRAINT [FK_User_Settings_Settings]
GO
ALTER TABLE [dbo].[User_Settings] WITH CHECK ADD CONSTRAINT
[FK_User_Settings_Users] FOREIGN KEY([UserId])
REFERENCES [dbo].[Users] ([Id])
GO
ALTER TABLE [dbo].[User_Settings] CHECK CONSTRAINT [FK_User_Settings_Users]
```


📌 ΣΥΝΟΨΗ

9.1 Συμπεράσματα

Στην διπλωματική αυτή εργασία περιεγράφηκε η ανάλυση, ο σχεδιασμός, η υλοποίηση και η λειτουργία μιας διαδικτυακής πλατφόρμας η οποία βασίζεται στην απλή ιδέα του συμβατικού ημερολογίου. Απώτερος σκοπός του συστήματος είναι οι χρήστες μέσα από την πλατφόρμα να καταγράφουν και να οργανώνουν με αποδοτικό τρόπο τις δραστηριότητες και τις συναντήσεις τους αξιοποιώντας τους υπολογιστικούς πόρους του συστήματος για να απλοποιηθεί η καθημερινότητα τους και να βελτιωθεί η ποιότητα ζωής.

Για την επίτευξη των παραπάνω δημιουργήθηκε ένα κομμάτι της πλατφόρμας με το οποίο ο διαχειριστής προσθέτει και την παραμετροποιεί για να βελτιωθεί η εμπειρία του χρήστη και να υπάρχει πληρότητα στο περιεχόμενο. Επιπρόσθετα, υπάρχει το κομμάτι στο οποίο γίνονται οι καταχωρίσεις των χρηστών με την βοήθεια πινάκων και φορμών για την ομοιογενή καταγραφή και παρουσίαση των δεδομένων.

Τα τρία μεγάλα πλεονεκτήματα είναι τα εξής:

- Παραμετροποίηση των ρυθμίσεων με γνώμονα τις ανάγκες και επιθυμίες των χρηστών βελτιώνοντας την εμπειρία τους και καταφέρνει να επιφέρει εξατομίκευση σε σχέση με άλλες πλατφόρμες βάση των κριτηρίων που έχουν θεσπιστεί.
- Χρήση αποδοτικών αλγορίθμων μικρής πολυπλοκότητας για την άμεση εξαγωγή αποτελεσμάτων και προτάσεων.
- Εύκολη προσθήκη επιπλέον κριτηρίων και περιορισμών στο αλγοριθμικό κομμάτι αφού οι υπολογισμοί γίνεται με ιεραρχία ως προς τα επιλεγμένα κριτήρια.

9.2 Επεκτάσεις

Η εφαρμογή είναι υλοποιημένη κατά τέτοιο τρόπο ώστε να μπορεί να επεκταθεί και να βελτιωθεί με πολλούς τρόπους, μερικοί από τους οποίους αναφέρονται παρακάτω:

- *Προσθήκη επιπλέον κριτηρίων και περιορισμών:* Στην πραγματικότητα το περιβάλλον μεταβάλλεται πολύ γρήγορα και με απρόβλεπτο τρόπο και έτσι δημιουργούνται συνέχεια άλλοι περιορισμοί αλλά και κριτήρια που αξιολογούμε για να λάβουμε μια απόφαση. Μια περίπτωση επέκτασης, λοιπόν, είναι η συγγραφή ακόμα παραπάνω περιορισμών και κριτηρίων οι οποίοι θα ενεργοποιούνται αντίστοιχα ώστε να υπάρχει πιο πλήρες και δυναμικό σύνολο κανόνων και πολιτικών χρονοπρογραμματισμού. Η δομή και η υλοποίηση των περιορισμών είναι άκρως αυτόνομη και αποδοτική και έτσι ενθαρρύνονται τέτοιες αλλαγές.
- *Ενσωμάτωση των δεδομένων και σε άλλα συστήματα που λειτουργούν στο διαδίκτυο:* Για παράδειγμα θα μπορούσαμε να περνάμε τα απαραίτητα δεδομένα που έχει η εφαρμογή μας στην εφαρμογή Google Calendar που αναφέραμε στην εισαγωγή μέσω της διαθέσιμης δημόσιας διεπαφής προγραμματισμού εφαρμογών (public web API). Με τον τρόπο αυτό θα μπορούσαμε να διαχειριστούμε παράλληλα τα δεδομένα μας ίσως και σε διαφορετικά περιβάλλοντα με διαφορετικές ανάγκες και δυνατότητες. Επιπρόσθετα, άλλη εφαρμογή που θα μπορούσε να χρησιμοποιηθεί για να επεκτείνει την λειτουργικότητα της πλατφόρμας θα μπορούσε να είναι η διασύνδεση με το Google Maps από το οποίο θα μπορούσαμε να φέρουμε δεδομένα για την τοποθεσία που θα πραγματοποιηθεί η εκάστοτε δραστηριότητα, ακόμα και να πλοηγηθούμε με τις παρεχόμενες οδηγίες για να διευκολυνθεί η μετάβαση στην φυσική τοποθεσία.
- Η επόμενη πιθανή προέκταση είναι ίσως πιο θεωρητική και είναι μια προσπάθεια μετατροπής του μοντέλου ανάθεσης μιας εταιρείας σε ένα Partially Observable Markov Decision Process (POMDP) όπου σε κάθε βάση (αφετηρία) της εταιρείας, με την

βοήθεια ενός συντονιστή θα γίνεται η ανάθεση χρονοθυρίδων λαμβάνοντας υπ' όψιν την διαμοιραζόμενη πληροφορία με την μέθοδο της κοινής πληροφόρισης βελτιστοποιώντας έτσι την πολιτική ανάθεσης. Γενικά ο κλάδος αποκεντρωμένης λήψης αποφάσεων είναι υπό ανάπτυξη και αρκετά απαιτητικός και γι αυτό μια τέτοια εργασία θα είχε πολύ μεγάλη δυσκολία αλλά και ακαδημαϊκό ενδιαφέρον.

 **ΑΝΑΦΟΡΕΣ**

1. C. Bussler. Workflow Instance Scheduling with Project Management Tools. In 9th Workshop on Database and Expert Systems Applications DEXA'98, Vienna, Austria, 1998. IEEE Computer Society Press.
2. CSESystems. Benutzerhandbuch V 4.1 Workflow. CSE Systems, Computer & Software Engineering GmbH, Klagenfurt, Austria, 1996.
3. CSE Systems Homepage. <http://www.csesys.co.at/>, February 1998.
4. J. Eder, H. Groiss, and W. Liebhart. The Workflow Management System Panta Rhei. In A. Dogac et al., editor, Advances in Workflow Management Systems and Interoperability. Springer, Istanbul, Turkey, August 1997.
5. J. Eder and W. Liebhart. Workflow Transactions. In P. Lawrence, editor, Workflow Handbook 1997. John Wiley, 1997.
6. Johann Eder, Heinz Pozewaunig, and Walter Liebhart. Timing issues in workflow management systems. Technical report, Institut f"ur Informatik-Systeme, Universit"at Klagenfurt, 1997.
7. TeamWare Flow. Collaborative workflow system for the way people work. P.O. Box 780, FIN-00101, Helsinki, Finland.
8. InConcert. Technical product overview. XSoft, a division of xerox. 3400 Hillview Avenue, Palo Alto, CA 94304. <http://www.xsoft.com>.
9. Heinrich Jasper and Olaf Zukunft. Zeitaspekte bei der Modellierung und Ausf"uhrung von Workflows. In S. Jablonski, H. Groiss, R. Kaschek, and W. Liebhart, editors, Gesch"äftsprozessmodellierung und Workflowsysteme, volume 2 of Proceedings Reihe der Informatik' 96, pages 109–119, Escherweg 2, 26121 Oldenburg, 1996.
10. F. Leymann and D. R"oller. Business process management with flowmark. In Proceedings of the 39th IEEE Computer Society International Conference, pages 230–233, San Francisco, California, February 1994. <http://www.software.ibm.com/workgroup>.
11. E. Panagos and M. Rabinovich. Escalations in workflow management systems. In DART Workshop, Rockville, Maryland, November 1996.
12. E. Panagos and M. Rabinovich. Predictive workflow management. In Proceedings of the 3rd International Workshop on Next Generation Information Technologies and Systems, Neve Ilan, ISRAEL, June 1997.
13. E. Panagos and M. Rabinovich. Reducing escalation-related costs in WFMSs. In A. Dogac et al., editor, NATO Advanced Study Institute on Workflow Management Systems and Interoperability. Springer, Istanbul, Turkey, August 1997.
14. Web application that implements part of functionality, 2007: <https://www.doodle.com>
15. Web application that implements similar functionality, 2006: <https://calendar.google.com/>
16. Web application that expands functionality, 2017: <https://clickup.com/>