



**ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ**

**ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ  
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ**

**ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ**

**ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ**

**Σχεδιασμός και Ανάπτυξη Εφαρμογής με Δυνατότητες  
Ασύγχρονης Εκτέλεσης και Χρήση Σχεσιακής Βάσης  
Δεδομένων Ανεξάρτητης Εξυπηρετητών**

**Γεώργιος Α. Τσιλιπάκος**

**Επιβλέπουσα**

**Αφροδίτη Τσαλαγατίδου, Αναπληρώτρια Καθηγήτρια**

**ΑΘΗΝΑ**

**ΣΕΠΤΕΜΒΡΙΟΣ 2019**

## **ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ**

Σχεδιασμός και Ανάπτυξη Εφαρμογής με Δυνατότητες Ασύγχρονης Εκτέλεσης και  
Χρήση Σχεσιακής Βάσης Δεδομένων Ανεξάρτητης Εξυπηρετητών

**Γεώργιος Α. Τσιλιπάκος**

**A.M.: 0684**

**ΕΠΙΒΛΕΠΟΥΣΑ** **Αφροδίτη Τσαλαγιάδου**, Αναπληρώτρια Καθηγήτρια

Σεπτέμβριος 2019

## ΠΕΡΙΛΗΨΗ

Σ' αυτήν την εργασία παρουσιάζεται η ανάπτυξη εφαρμογής ανεξάρτητης εξυπηρετητή με δυνατότητα εκτέλεσης χωρίς σύνδεση στο διαδίκτυο και διαχείριση των πληροφοριών της σε σχεσιακή βάση δεδομένων. Η προσπάθεια ξεκίνησε με τη συλλογή και κωδικοποίηση των απαιτήσεων και συνεχίστηκε με την εύρεση μίας σχεσιακής βάσης δεδομένων ανεξάρτητης εξυπηρετητή. Επιλέχθηκε στη συνέχεια να χρησιμοποιηθεί η ανάπτυξη αρχικής αποτύπωσης της διεπαφής για την αποτελεσματικότερη επικοινωνία της με τους χρήστες και ερευνήθηκαν πιθανά εργαλεία για την ανάπτυξη της. Μετά πραγματοποιήθηκε εκτενής έρευνα για την εύρεση κατάλληλου περιβάλλοντος, εργαλείων και τεχνολογιών για την ανάπτυξη εφαρμογής ανεξάρτητης πλατφόρμας και τέλος εντοπίστηκαν τρόποι για την ασύγχρονη εκτέλεση της εφαρμογής.

Το αποτέλεσμα των παραπάνω ερευνών ήταν να χρησιμοποιηθεί η SQLite ως σχεσιακή βάση δεδομένων, το Cacao ως εργαλείο για την ανάπτυξη wireframe, το Microsoft Visual Studio (MVS) ως ολοκληρωμένο περιβάλλον ανάπτυξης, το Apache Cordova και το jQuery ως ολοκληρωμένο πλαίσιο ανάπτυξης, η HTML και η JavaScript ως γλώσσες προγραμματισμού και το Dropbox ως ψηφιακό αποθετήριο της εφαρμογής για την ασύγχρονη εκτέλεσή της. Αφού συλλέχθηκαν οι απαιτήσεις, σχεδιάστηκε αρχικά η σχεσιακή βάση δεδομένων δημιουργώντας ένα Διάγραμμα Οντοτήτων Συσχετίσεων και στη συνέχεια μετατρέποντάς το στο σχεσιακό μοντέλο. Αναπτύχθηκε μετά η αρχική αποτύπωση της διεπαφής και ξεκίνησε η υλοποίηση της εφαρμογής στο MVS. Η διεπαφή της εφαρμογής σχεδιάστηκε για τη χρήση της εφαρμογής σε ποικίλα περιβάλλοντα και συσκευές, ώστε να διευκολύνεται η μεταφορά της εφαρμογής σε περιβάλλοντα διαφορετικά από Microsoft Windows. Η ανάπτυξη των λειτουργιών της εφαρμογής υλοποιήθηκε με συναρτήσεις σε JavaScript, όπου αναπτύχθηκαν επίσης αντικείμενα και λίστες. Τέλος αναπτύχθηκε και η δυνατότητα της εφαρμογής για ασύγχρονη εκτέλεση.

Ο έλεγχος του λογισμικού πραγματοποιούνταν κατά την υλοποίηση αλλά και στη συνέχεια με την εκτέλεση σεναρίων αποσφαλμάτωσης, τα οποία εκτελέστηκαν και από τους χρήστες της εφαρμογής. Με τον τρόπο αυτό συμμετείχαν και οι χρήστες στην επικύρωση και την επαλήθευση της εφαρμογής. Εντοπίστηκαν πολλά σημεία για τα οποία απαιτήθηκε εν συνεχεία η τροποποίηση του σχεδιασμού και της υλοποίησής της, ένδειξη ότι ο έλεγχος λειτούργησε προσθετικά για το τελικό αποτέλεσμα.

Με το πέρας της υλοποίησης της εφαρμογής εντοπίστηκαν σημαντικές τάσεις στην ανάπτυξη εφαρμογών, στη διαχείριση των δεδομένων των εφαρμογών, στην αναγκαιότητα της ανάπτυξης wireframe και στις δυνατότητες ασύγχρονης εκτέλεσης εφαρμογών σ' ένα περιβάλλον όπου η διασύνδεση με το διαδίκτυο συνεχώς αυξάνεται.

**ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ:** Σχεδιασμός και Ανάπτυξη Εφαρμογών

**ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ:** εφαρμογή ανεξάρτητη πλατφόρμας, σχεσιακή βάση δεδομένων, βάση δεδομένων ανεξάρτητη εξυπηρετητών, αρχική αποτύπωση διεπαφής, ασύγχρονη εκτέλεση

## **ABSTRACT**

This project presents the development of a server-less application, which has the ability to be executed asynchronously and uses a server-less, relational database to store information. At the start of the effort, user requirements were gathered and a server-less, relational database was investigated. After that, there was a research on tools for application wireframes development, which could help on meeting user requirements for the interface at the early stages of the project. Afterwards, a thorough investigation was contacted to specify the proper Integrated Development Environment, the proper framework and technologies. Finally, there was an effort on pinpointing possible ways for asynchronous execution.

The outcome of the above investigation was the selection of SQLite as the relational database engine, Cacao as the wireframe development tool, Microsoft Visual Studio (MVS) as the IDE, Apache Cordova and jQuery as the development frameworks, HTML and JavaScript as the scripting languages and Dropbox as the tool to help asynchronous execution. An Entity-Relational Diagram was designed at the start based on user requirements, which afterwards was converted into a relational database schema. Then, the interface wireframe was created and the development started using the MVS. User interface was designed having in mind the cross-platform execution of the application. The functionality of the application was created using functions developed in JavaScript with the help of user-created objects and lists. At the final stages of the development, the possibility of asynchronous execution was created using the Dropbox Application Program Interface (API).

The software control was contacted throughout the development and with the use of user-specified testing scenarios, which were executed also from the users. Thus users were involved in the validation and the verification of the application. There were many changes in the design and implementation of the application, which were decided through the process of software testing, highlighting the value of this process.

The whole effort assisted in pinpointing application development trends, information storage trends for application, wireframe trends and asynchronous execution possibilities, in an application environment, where internet connectivity is considered more and more mandatory.

**SUBJECT AREA:** Application Design and Development

**KEYWORDS:** cross-platform application, relational database, server-less database, wireframes, asynchronous execution

Στην Ελένη για την τεράστια υπομονή της,  
στην μικρή Ολυμπία για το χρόνο που της στέρησα και  
στη μητέρα Ολυμπία για τη χαρά που δεν της πρόσφερα.

## **ΕΥΧΑΡΙΣΤΙΕΣ**

Θερμές ευχαριστίες στην Αναπληρώτρια Καθηγήτρια Αφροδίτη Τσαλγατίδου για την ευκαιρία που μου έδωσε να δω έναν αγώνα να ολοκληρώνεται που έμοιαζε χαμένος.

# ΠΕΡΙΕΧΟΜΕΝΑ

<b>ΠΡΟΛΟΓΟΣ .....</b>	<b>13</b>
<b>1. ΕΙΣΑΓΩΓΗ.....</b>	<b>14</b>
1.1 Στόχος Ανάπτυξης.....	14
1.2 Μεθοδολογία Προσέγγισης.....	14
1.3 Συνοπτική Παρουσίαση .....	15
<b>2. ΜΕΘΟΔΟΛΟΓΙΑ .....</b>	<b>17</b>
2.1 Απαιτήσεις Εφαρμογής.....	17
2.2 Επιλογή Σχεσιακής Βάσης Δεδομένων Ανεξάρτητης Εξυπηρετητή.....	18
2.3 Αρχική Αποτύπωση Διεπαφής .....	19
2.4 Επιλογή Τεχνολογιών .....	20
2.4.1 Netbeans, Java .....	20
2.4.2 Eclipse, HTML, JavaScript.....	21
2.4.3 Visual Studio, Apache Cordova, HTML, JavaScript .....	21
2.5 Ασύγχρονη Εκτέλεση .....	22
2.6 Αναλυτική Περιγραφή Απαιτήσεων Εφαρμογής.....	22
2.6.1 Εισαγωγή Θεραπείας .....	22
2.6.2 Προβολή Θεραπείας .....	24
2.6.3 Επεξεργασία Θεραπείας .....	25
2.6.4 Λίστα Θεραπειών .....	25
2.6.5 Αναζήτηση Θεραπείας .....	25
2.6.6 Διαχείριση Περιφερειακών Δεδομένων .....	25
2.6.7 Περιβάλλον Εκτέλεσης Εφαρμογής .....	25
2.6.8 Πλήθος Δεδομένων .....	26
2.6.9 Ασύγχρονη Εκτέλεση .....	26
<b>3. ΥΛΟΠΟΙΗΣΗ ΣΧΕΣΙΑΚΗΣ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ ΑΝΕΞΑΡΤΗΤΗΣ ΕΞΥΠΗΡΕΤΗΤΗ.....</b>	<b>27</b>
3.1 Σχεδιασμός Σχεσιακής Βάσης Δεδομένων .....	27
3.1.1 Πρώτη Κανονική Μορφή (First Normal Form – 1NF).....	30

3.1.2	Δεύτερη Κανονική Μορφή (Second Normal Form – 2NF)	30
3.1.3	Τρίτη Κανονική Μορφή (Third Normal Form – 3NF)	30
3.2	Επιλογή Εφαρμογής Διαχείρισης	31
3.3	Υλοποίηση Βάσης Δεδομένων	31
4.	ΑΡΧΙΚΗ ΑΠΟΤΥΠΩΣΗ ΔΙΕΠΑΦΗΣ	37
4.1	Επιλογή Εφαρμογής Αρχικής Αποτύπωσης Διεπαφής	37
4.2	Σχεδιασμός Αρχικής Αποτύπωσης Διεπαφής	37
4.3	Παρουσίαση Αρχικής Αποτύπωσης Διεπαφής	39
5.	ΥΛΟΠΟΙΗΣΗ ΕΦΑΡΜΟΓΗΣ	43
5.1	Εγκατάσταση Εργαλείων και Πακέτων	43
5.2	Ανάπτυξη Αντικειμένων	45
5.3	Ανάπτυξη Λιστών	47
5.4	Ανάπτυξη Ανάκλησης και Αποθήκευσης Δεδομένων	47
5.5	Ανάπτυξη Διεπαφής	48
5.6	Ανάπτυξη Λειτουργικότητας	49
5.7	Ανάπτυξη Ασύγχρονης Εκτέλεσης	50
5.8	Εκτέλεση σε Πολλαπλές Πλατφόρμες	50
5.9	Εγκατάσταση Εφαρμογής	51
5.10	Παρουσίαση Εφαρμογής	51
6.	ΕΛΕΓΧΟΣ ΛΟΓΙΣΜΙΚΟΥ	56
6.1	Επικύρωση Εφαρμογής	56
6.2	Επαλήθευση Εφαρμογής	57
6.3	Δημιουργία Σεναρίων Αποσφαλμάτωσης	57
6.4	Εκτέλεση Σεναρίων Αποσφαλμάτωσης και Διορθωτικές Ενέργειες	59



<b>7. ΣΥΜΠΕΡΑΣΜΑΤΑ .....</b>	<b>60</b>
7.1 Γενικά .....	60
7.2 Σχεσιακή Βάση Δεδομένων Ανεξάρτητη Εξυπηρετητών .....	60
7.3 Αρχική Αποτύπωση Διεπαφής .....	61
7.4 Τεχνολογίες Ανάπτυξης Εφαρμογών .....	61
7.5 Ασύγχρονη Εκτέλεση Εφαρμογών .....	62
<b>ΠΙΝΑΚΑΣ ΟΡΟΛΟΓΙΑΣ .....</b>	<b>64</b>
<b>ΣΥΝΤΜΗΣΕΙΣ – ΑΡΚΤΙΚΟΛΕΞΑ – ΑΚΡΩΝΥΜΙΑ .....</b>	<b>65</b>
<b>ΠΑΡΑΡΤΗΜΑ Ι - ΟΔΗΓΙΕΣ ΕΓΚΑΤΑΣΤΑΣΗΣ.....</b>	<b>66</b>
<b>ΠΑΡΑΡΤΗΜΑ ΙΙ - ΟΔΗΓΙΕΣ ΧΡΗΣΗΣ.....</b>	<b>72</b>
<b>ΑΝΑΦΟΡΕΣ .....</b>	<b>91</b>

## ΚΑΤΑΛΟΓΟΣ ΣΧΗΜΑΤΩΝ

Σχήμα 1: Πρώτο Διάγραμμα Οντοτήτων Συσχετίσεων .....	27!
Σχήμα 2: Δεύτερο Διάγραμμα Οντοτήτων Συσχετίσεων .....	28!
Σχήμα 3: Σχεσιακό Μοντέλο Βάσης Δεδομένων .....	29!

## ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ

Εικόνα 1: Γραφικό περιβάλλον εφαρμογής DB Browser for SQLite.....	34!
Εικόνα 2: Φόρμα δημιουργία και μετατροπής πινάκων στην εφαρμογή DB Browser for SQLite .....	35!
Εικόνα 3: Φόρμα εισαγωγής δεδομένων στην εφαρμογή DB Browser for SQLite .....	35!
Εικόνα 4: Αρχικό περιβάλλον της εφαρμογής Cacao.....	38!
Εικόνα 5: Περιβάλλον επεξεργασίας της εφαρμογής Cacao.....	38!
Εικόνα 6: Αρχική σελίδα εφαρμογής θεραπειών.....	39!
Εικόνα 7: Σελίδα αναζήτησης εφαρμογής θεραπειών.....	40!
Εικόνα 8: Σελίδα ενημέρωσης θεραπείας εφαρμογής θεραπειών.....	41!
Εικόνα 9: Σελίδα εισαγωγής θεραπείας εφαρμογής θεραπειών.....	41!
Εικόνα 10: Σελίδα λίστας εφαρμογής θεραπειών (πρώτη εκδοχή) .....	42!
Εικόνα 11: Σελίδα λίστας εφαρμογής θεραπειών (δεύτερη εκδοχή) .....	42!
Εικόνα 12: Αρχικό περιβάλλον στο MSV Community 2015 .....	43!
Εικόνα 13: Διαχείριση πακέτων στο MVS Community 2015.....	44!
Εικόνα 14: Διαχείριση επεκτάσεων στο MVS Community 2015.....	44!
Εικόνα 15: Δέντρο αρχείων του project για την εφαρμογή Tai Yi.....	45!
Εικόνα 16: Περιβάλλον ανάπτυξης κώδικα HTML στο MVS .....	46!
Εικόνα 17: Περιβάλλον ανάπτυξης κώδικα JavaScript στο MVS .....	46!
Εικόνα 18: Αρχική σελίδα εφαρμογής Tai Yi.....	52!
Εικόνα 19: Αναζήτηση θεραπειών στην εφαρμογή Tai Yi.....	52!
Εικόνα 20: Εισαγωγή θεραπείας στην εφαρμογή Tai Yi .....	53!
Εικόνα 21: Επεξεργασία θεραπείας στην εφαρμογή Tai Yi .....	53!
Εικόνα 22: Λίστα του συνόλου των θεραπειών στην εφαρμογή Tai Yi .....	54!
Εικόνα 23: Προβολή μιας θεραπείας στην εφαρμογή Tai Yi .....	54!
Εικόνα 24: Επεξεργασία περιφερειακών δεδομένων στην εφαρμογή Tai Yi (Special Categories).....	55!

## **ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ**

Πίνακας 1: Σύγκριση Συστημάτων Βάσεων Δεδομένων ..... 19!

Πίνακας 2: Σύντμηση ονοματολογίας σχέσεων και γνωρισμάτων ..... 31!

## ΠΡΟΛΟΓΟΣ

Η εργασία περιγράφει το σχεδιασμό και την ανάπτυξη εφαρμογής λογισμικού με δυνατότητες ασύγχρονης εκτέλεσης και χρήση σχεσιακής βάσης δεδομένων ανεξάρτητης από εξυπηρετητές. Πραγματοποιήθηκε εκτενής διερεύνηση εργαλείων και τεχνολογιών κατάλληλων για την υλοποίησή της. Η συγκέντρωση των απαιτήσεων των χρηστών, η κωδικοποίησή τους σε τεχνικές απαιτήσεις, ο καθορισμός κριτηρίων για την διερεύνηση των κατάλληλων εργαλείων, η αναζήτησή τους καθώς η τελική επιλογή τους, πραγματοποιήθηκε στα αρχικά στάδια σχεδιασμού, σε ποικίλα περιβάλλοντα και υπολογιστές. Στη συνέχεια, η εγκατάσταση των εργαλείων, και η υλοποίηση του λογισμικού, πραγματοποιήθηκαν στο γραφείο Α11 του Τμήματος, και με τη βοήθεια εκπαιδευτικής άδειας που μου χορηγήθηκε για να μπορέσω ανεμπόδιστος να συγκεντρωθώ και να υλοποιήσω το λογισμικό με τις δοθείσες απαιτήσεις. Τέλος, η συγγραφή της διπλωματικής εργασίας ξεκίνησε στον ίδιο χώρο που υλοποιήθηκε η εφαρμογή και συνεχίστηκε σε ποικίλα περιβάλλοντα, αφού πλέον είχε καταγραφεί λεπτομερώς η όλη διαδικασία σχεδιασμού και υλοποίησης. Μετά την ολοκλήρωση της ανάπτυξης λογισμικού και την αποσφαλμάτωση, έγιναν πολλαπλές δοκιμές από χρήστες της εφαρμογής. Βάσει των αποτελεσμάτων των δοκιμών έγιναν οι απαραίτητες διορθώσεις στο λογισμικό, ώστε το τελικό αποτέλεσμα να ικανοποιεί τους χρήστες της εφαρμογής.

## 1. ΕΙΣΑΓΩΓΗ

### 1.1 Στόχος Ανάπτυξης

Ο κύριος στόχος της διπλωματικής εργασίας είναι η ανάπτυξη εφαρμογής (application), η οποία θα διαχειρίζεται τα δεδομένα της σε μία σχεσιακή βάση δεδομένων (relational database) ανεξάρτητης εξυπηρετητή (serverless). Η εφαρμογή αναπτύχθηκε για να λειτουργεί σε λειτουργικό σύστημα (operating system) Microsoft Windows.

Ο κύριος αυτός στόχος αναλύεται περαιτέρω, αρχικά στην αναζήτηση και επιλογή μίας σχεσιακής βάσης δεδομένων ανεξάρτητης εξυπηρετητή, δηλαδή μίας σχεσιακής βάσης δεδομένων που μπορεί να λειτουργήσει χωρίς την ύπαρξη εξυπηρετητή (server) και χωρίς μόνιμη σύνδεση στο διαδίκτυο (internet access), και στη συνέχεια στην αναζήτηση και επιλογή ολοκληρωμένου περιβάλλοντος ανάπτυξης (Integrated Development Environment - IDE) για την υλοποίηση της εφαρμογής.

Για τις παραπάνω αναζητήσεις, καταβλήθηκε προσπάθεια να εντοπιστούν σχεσιακές βάσεις δεδομένων και IDE που χρησιμοποιούνται ευρέως, δηλαδή είναι δημοφιλής, δοκιμασμένα και έχουν σημαντική τεκμηρίωση. Η συνεργασία του IDE με τη σχεσιακή βάση δεδομένων που επιλέχθηκε ήταν ένα από τα σημαντικότερα κριτήρια για την επιλογή τους.

Η όλη προσπάθεια δεν ήταν εύκολη, διότι η παγκόσμια χρήση του διαδικτύου αυξάνεται συνεχώς, σύμφωνα με τα στατιστικά δεδομένα του International Telecommunication Union - ITU [1], οπότε και τα αναπτυσσόμενα IDE πλέον είναι προσανατολισμένα στην ανάπτυξη εφαρμογών που έχουν μόνιμη σύνδεση στο διαδίκτυο.

### 1.2 Μεθοδολογία Προσέγγισης

Η εργασία που έγινε στο πλαίσιο της διπλωματικής, οδηγήθηκε από τις ανάγκες ύπαρξης μιας εφαρμογής για την καταγραφή και αξιοποίηση δεδομένων συγκεκριμένων πληροφοριών μιας θεραπευτικής ενεργειακής μεθόδου που λέγεται Tai Yi ή Ροή Ζωτικής Ενέργειας (Life Energy Flow) [2] που σημαίνει Υπέρτατη Κίνηση, Υπέρτατη Υγεία. Η ύπαρξη πολλών θεραπειών Tai Yi, με πληθώρα σημαντικών πληροφοριών για την κάθε μία, αποτέλεσε το κατάλληλο case study για τους σκοπούς της διπλωματικής εργασίας. Συγκεκριμένα, οδήγησε στη δημιουργία εφαρμογής που θα δίνει τη δυνατότητα της διατήρησης, διαχείρισης και αναζήτησης όλων των δεδομένων για κάθε θεραπεία με ασύγχρονο τρόπο.

Για την υλοποίηση της συγκεκριμένης εφαρμογής, συγκεντρώθηκαν οι αρχικές απαιτήσεις των χρηστών, μέσα από τις οποίες περιγράφηκαν σε γενικές γραμμές οι λειτουργίες που καλείται να εκτελεί η εφαρμογή, καθώς και το είδος των πληροφοριών που πρέπει τηρεί.

Έτσι λοιπόν, η εργασία ξεκίνησε με τον τρόπο που θα γίνονταν η διαχείριση των δεδομένων. Η διαχείριση των δεδομένων με το σχεσιακό μοντέλο κάλυπτε την ανάγκη για δομημένα δεδομένα και επεκτασιμότητα (scalability). Επομένως, αναζητήθηκαν σχεσιακές βάσεις δεδομένων που μπορούν να λειτουργήσουν χωρίς την ύπαρξη εξυπηρετητή, έτσι ώστε τα δεδομένα να είναι διαθέσιμα ανεξάρτητα από την ύπαρξη σύνδεσης στο διαδίκτυο. Μετά την εύρεση της κατάλληλης βάσης δεδομένων έγινε ο σχεσιακός σχεδιασμός της.

Το επόμενο στάδιο ήταν η σχεδίαση της διεπαφής της εφαρμογής χρησιμοποιώντας μία αρχική αποτύπωση διεπαφής (wireframe). Για το σκοπό αυτό πραγματοποιήθηκε συνοπτική έρευνα στα διαθέσιμα εργαλεία λογισμικού για wireframes και επιλέχθηκε ένα

απλό εργαλείο με μικρή καμπύλη εκμάθησης έτσι ώστε η όλη προσπάθεια να ολοκληρωθεί σε σύντομο χρονικό διάστημα.

Η επόμενη έρευνα που πραγματοποιήθηκε αφορούσε την εύρεση ενός κατάλληλου ολοκληρωμένου περιβάλλοντος ανάπτυξης για την υλοποίηση της λειτουργικότητας καθώς και της διεπαφής της εφαρμογής, χρησιμοποιώντας τεχνολογίες ευρέως διαδεδομένες, ώστε να υπάρχει αρκετή διαθέσιμη τεκμηρίωση. Στο σημείο αυτό πραγματοποιήθηκαν δοκιμές, οι οποίες περιγράφονται στο επόμενο κεφάλαιο, μετά τις οποίες επιλέχθηκε ο κατάλληλος συνδυασμός τεχνολογιών για την υλοποίηση της εφαρμογής.

Ένα ακόμη βήμα για το οποίο χρειάστηκε κάποια έρευνα, ήταν ο τρόπος με τον οποίο τα δεδομένα θα συγχρονίζονταν, έτσι ώστε να μπορούν να είναι διαθέσιμα από πολλούς χρήστες σε διαφορετικές συσκευές, ακόμη και αν η ενημέρωσή τους γίνεται χωρίς την ύπαρξη σύνδεσης στο διαδίκτυο. Εδώ εξετάστηκαν οι δυνατότητες για ασύγχρονη εκτέλεση της εφαρμογής και συγχρονισμό των δεδομένων τη χρονική στιγμή που επιλέγει ο χρήστης και υπάρχει σύνδεση στο διαδίκτυο.

Για τον έλεγχο του λογισμικού, την επικύρωση και την επαλήθευσή του, πέρα από τη συνεχή εξέταση της συμμόρφωσης στις απαιτήσεις κατά την υλοποίηση της εφαρμογής, δημιουργήθηκαν πολλά σενάρια αποσφαλμάτωσης, τα οποία εκτελέστηκαν εκτενώς.

### 1.3 Συνοπτική Παρουσίαση

Στο επόμενο κεφάλαιο περιγράφεται αναλυτικά η μεθοδολογία που ακολουθήθηκε για την υλοποίηση της εφαρμογής. Αρχικά παρουσιάζονται με συνοπτικό τρόπο οι απαιτήσεις της εφαρμογής, η διαδικασία επιλογής σχεσιακής βάσης δεδομένων ανεξάρτητης εξυπηρετητή και η διαδικασία επιλογής της εφαρμογής για την ανάπτυξη της αρχικής αποτύπωσης της διεπαφής. Στη συνέχεια περιγράφεται η έρευνα που έγινε για τον εντοπισμό των κατάλληλων τεχνολογιών ανάπτυξης, παρουσιάζοντας τις προσπάθειες που έγιναν και τα αποτελέσματά τους. Επίσης, παρουσιάζεται η διαδικασία που επιλέχθηκε για την κάλυψη της ανάγκης για ασύγχρονη εκτέλεση της εφαρμογής. Ακολουθώντας, περιγράφεται η προσπάθεια για την εκτέλεση της εφαρμογής και σε άλλα περιβάλλοντα και πλατφόρμες, οι δυσκολίες που αντιμετωπίστηκαν και τελικά η εγκατάλειψη της απαίτησης εκτέλεσης και σε άλλα περιβάλλοντα. Στο τέλος του κεφαλαίου παρουσιάζονται αναλυτικά οι τελικές απαιτήσεις της εφαρμογής.

Στο τρίτο κεφάλαιο παρουσιάζεται αναλυτικά ο τρόπος υλοποίησης της σχεσιακής βάσης δεδομένων. Συγκεκριμένα παρουσιάζονται τα βήματα για το σχεδιασμό της, η επιλογή της εφαρμογής διαχείρισης και ο τρόπος που δημιουργήθηκε καθώς και ο τρόπος εισαγωγής δοκιμαστικών δεδομένων.

Στη συνέχεια, στο επόμενο κεφάλαιο παρουσιάζεται η έρευνα για την εύρεση εργαλείου ανάπτυξης της αρχικής αποτύπωσης της διεπαφής, την επιλογή του εργαλείου, τον τρόπο που έγινε ο σχεδιασμός της, καθώς και η παρουσίαση της αρχικής αποτύπωσης της διεπαφής που αναπτύχθηκε.

Στο πέμπτο κεφάλαιο παρουσιάζεται αναλυτικά η υλοποίηση της εφαρμογής χρησιμοποιώντας τις τεχνολογίες και το περιβάλλον ανάπτυξης που επιλέχτηκε. Παρουσιάζονται τα αντικείμενα, οι λίστες, ο τρόπος ανάκλησης των δεδομένων, ο τρόπος ανάπτυξης της διεπαφής, ο τρόπος υλοποίησης της λειτουργικότητας, ο τρόπος κάλυψης της ανάγκης ασύγχρονης εκτέλεσης της εφαρμογής καθώς και το τρόπος εγκατάστασής της. Τέλος γίνεται παρουσίαση της ολοκληρωμένης εφαρμογής.

Το επόμενο κεφάλαιο αφορά στον έλεγχο του λογισμικού και παρουσιάζονται αναλυτικά η επικύρωση της εφαρμογής, η επαλήθευσή της, η δημιουργία σεναρίων αποσφαλμάτωσης, η εκτέλεση των σεναρίων αυτών καθώς και όλες οι διορθωτικές ενέργειες που προέκυψαν από τις παραπάνω ενέργειες.

Στο τελευταίο κεφάλαιο παρουσιάζονται τα συμπεράσματα της όλης προσπάθειας ομαδοποιημένα σε θεματικές ενότητες. Συγκεκριμένα χωρίζονται σε ότι αφορά στην σχεσιακή βάση δεδομένων ανεξάρτητη εξυπηρετητή, στην αρχική αποτύπωση της διεπαφής, στις τεχνολογίες ανάπτυξης εφαρμογών και στις δυνατότητες ασύγχρονης εκτέλεσης εφαρμογών.



## 2. ΜΕΘΟΔΟΛΟΓΙΑ

### 2.1 Απαιτήσεις Εφαρμογής

Οι αρχικές απαιτήσεις για την εφαρμογή συζητήθηκαν και χορηγήθηκαν από τους χρήστες της εφαρμογής, ασκούμενους της Σχολής Θεραπειών Tai Yi - Ροής Ζωτικής Ενέργειας [2]. Στις πρώτες συναντήσεις έγινε μία λεκτική περιγραφή των λειτουργικών απαιτήσεων της εφαρμογής και παραδόθηκε σε κείμενο η αποτύπωση μερικών θεραπειών.

Οι αρχικές λειτουργικές απαιτήσεις που δόθηκαν παρουσιάζονται συνοπτικά παρακάτω:

1. Εισαγωγή θεραπείας: Να υπάρχει η δυνατότητα να γίνει εισαγωγή μίας νέας θεραπείας στην εφαρμογή.
2. Επεξεργασία θεραπείας: Να δίνει τη δυνατότητα η εφαρμογή να γίνεται επεξεργασία των δεδομένων μίας θεραπείας.
3. Λίστα θεραπειών: Να εμφανίζεται μία συνολική λίστα με το σύνολο των θεραπειών που έχουν εισαχθεί στην εφαρμογή.
4. Αναζήτηση θεραπείας: Να δίνεται η δυνατότητα αναζήτησης στις ήδη καταχωρημένες θεραπείες, με βάσει συγκεκριμένες πληροφορίες, που θα παρέχει ο χρήστης και να εμφανίζονται σε λίστα τα αποτελέσματα. Αυτή αποτελεί μία από τις σημαντικότερες λειτουργικές απαιτήσεις της εφαρμογής.
5. Περιβάλλον εκτέλεσης της εφαρμογής: Να υπάρχει η δυνατότητα η εφαρμογή να εκτελείται σε πολλαπλά περιβάλλοντα, δηλαδή είτε σε προσωπικούς υπολογιστές με λειτουργικό σύστημα Microsoft Windows ή Apple, είτε σε φορητές συσκευές με λειτουργικό σύστημα Android ή Apple.
6. Πλήθος δεδομένων: Ο μέγιστος αριθμός θεραπειών που θα κληθεί να φιλοξενήσει η εφαρμογή είναι 5.000, δηλαδή στην τάξη των χιλιάδων.
7. Δυνατότητα σύνδεσης στο διαδίκτυο: Λόγω της πιθανότητας εκτέλεσης και εκμάθησης των θεραπειών σε μέρη χωρίς πρόσβαση στο διαδίκτυο, ζητήθηκε η δυνατότητα εκτέλεσης της εφαρμογής χωρίς σύνδεση στο διαδίκτυο, με το σύνολο ωστόσο των δεδομένων και τη δυνατότητα εμπλουτισμού τους. Αυτή η απαίτηση δεν δόθηκε σαν απαραίτητη στην αρχή.

Παραδόθηκε επίσης κείμενο με την αναλυτική περιγραφή δέκα συγκεκριμένων θεραπειών από τους χρήστες.

Οι απαιτήσεις αυτές κωδικοποιήθηκαν αναλυτικά σε τεχνικές προδιαγραφές σταδιακά όπως προχωρούσε ο σχεδιασμός και η υλοποίηση της εφαρμογής, ωστόσο κάποιες σημαντικές τεχνικές λεπτομέρειες αποφασίστηκαν αρχικά.

Τα δεδομένα της εφαρμογής επιλέχθηκε αρχικά να δομηθούν βάσει του σχεσιακού μοντέλου το οποίο στηρίζεται στη σχεσιακή άλγεβρα και για το οποίο υπήρχε πολύ καλή τεχνογνωσία. Επιπλέον, σύμφωνα με έρευνα που έχει γίνει [3], οι μη σχεσιακές βάσεις δεδομένων ή όπως αναφέρονται NoSQL databases, εμφανίζουν ανωριμότητα, δεν έχουν συγκεκριμένη γλώσσα για τη δημιουργία ερωτημάτων, δεν έχουν συγκεκριμένη διεπαφή, η διαχείρισή τους είναι δύσκολη και δεν εγγυούνται τις ιδιότητες της ατομικότητας (atomicity), συνοχής (consistency), απομόνωσης (isolation) και αντοχής (durability) ή αλλιώς ACID (Atomicity, Consistency, Isolation, Durability) ιδιότητες.

Περαιτέρω, από σύγκριση που έγινε στη χρήση ανάμεσα σε μία μη σχεσιακή βάση δεδομένων και μία σχεσιακή για μία εφαρμογή διαδικτύου [4], διαπιστώθηκε ότι η χρονική απόκριση ανάμεσα στις λειτουργίες μίας βάσης δεδομένων, όπως η εισαγωγή,

η ενημέρωση, η διαγραφή και η αναζήτηση δεν έχουν μεγάλη διαφορά για αριθμό δεδομένων της τάξης των 1.000. Μάλιστα στη σύγκριση αυτή φάνηκε ότι η σχεσιακή βάση δεδομένων ήταν ανώτερη στην αναζήτηση όταν ο αριθμός των δεδομένων αυξάνονταν πέρα των 1.000, κάτι που ενίσχυε τη χρησιμοποίησή της αφού το πλήθος δεδομένων που καλείται να φιλοξενήσει η εφαρμογή είναι στην τάξη των χιλιάδων. Ένας επιπλέον λόγος να επιλεγεί το σχεσιακό μοντέλο για τη βάση δεδομένων, είναι το γεγονός ότι η κυριότερη απαίτηση της εφαρμογής είναι η αναζήτηση των δεδομένων και η επεξεργασία των δεδομένων θα γίνεται μεμονωμένα και όχι με μεγάλη συχνότητα.

Η απαίτηση της εφαρμογής να εκτελείται χωρίς σύνδεση στο διαδίκτυο οδήγησε στην επιλογή σχεσιακής βάσης δεδομένων που να μπορεί να λειτουργεί ανεξάρτητα από την ύπαρξη εξυπηρετητή και μάλιστα να υπάρχει δυνατότητα ασύγχρονης εκτέλεσης και ενημέρωσής της.

Και τέλος, η απαίτηση η εφαρμογή να εκτελείται σε ποικίλα περιβάλλοντα οδήγησε στην εύρεση περιβάλλοντος ανάπτυξης και στη χρησιμοποίηση τεχνολογιών που υποστηρίζουν την ανάπτυξη εφαρμογών ανεξαρτήτων πλατφόρμας.

Αναλυτική περιγραφή των απαιτήσεων για την εφαρμογή θα παρουσιαστούν στην τελευταία παράγραφο του κεφαλαίου.

## **2.2 Επιλογή Σχεσιακής Βάσης Δεδομένων Ανεξάρτητης Εξυπηρετητή**

Λόγω της απαίτησης η εφαρμογή να εκτελείται σε ποικίλα περιβάλλοντα και χωρίς σύνδεση στο διαδίκτυο, πραγματοποιήθηκε έρευνα στις διαθέσιμες επιλογές σε συστήματα βάσεων δεδομένων που μπορούν να εκτελεστούν σε φορητές συσκευές χωρίς τη χρήση εξυπηρετητή. Από τις διαθέσιμες επιλογές ουσιαστικά αποκλείστηκαν όσες δεν ήταν βασισμένες στο σχεσιακό μοντέλο, αφού έχει γίνει ήδη η επιλογή για χρησιμοποίηση σχεσιακής βάσης δεδομένων. Τα κριτήρια βάσει των οποίων έγινε η σύγκριση είναι:

- Υποστήριξη πολλαπλών συσκευών
- Το πόσο δημοφιλές είναι
- Μοντέλο άδειας χρήσης

Για την έρευνα χρησιμοποιήθηκε μία παρουσίαση των διαφορετικών συστημάτων βάσεων δεδομένων που εκτελούνται σε φορητές συσκευές [5] σε συνδυασμό με τα αποτελέσματα της σειράς κατάταξης όλων των συστημάτων βάσεων δεδομένων που υπάρχουν στην αγορά [6]. Η σειρά κατάταξης αυτή μετρά το πόσο δημοφιλής είναι μία βάση δεδομένων λαμβάνοντας υπόψη τον αριθμό αναφοράς του κάθε συστήματος σε ιστοσελίδες (λήφθηκε από τις μηχανές αναζήτησης (search engines) Google, Bing και Yandex), το γενικό ενδιαφέρον για το κάθε σύστημα ως συχνότητα αναζήτησης (λήφθηκε από το Google Trends), τη συχνότητα εμφάνισης τεχνικών ερωτήσεων για το κάθε σύστημα και τον αριθμό των χρηστών που ενδιαφέρονται (λήφθηκε από τις ιστοσελίδες ερωτήσεων και απαντήσεων για τεχνικά θέματα Stack Overflow και DBA Stack Exchange), τον αριθμό των προσκλήσεων εύρεσης εργασίας στις οποίες αναφέρεται το κάθε σύστημα (λήφθηκε από τις ιστοσελίδες εύρεσης εργασίας Indeed και Simply Hired), τον αριθμό των επαγγελματικών προφίλ - λογαριασμών (profile) στα οποία αναφέρεται το κάθε σύστημα (λήφθηκε από τα επαγγελματικά δίκτυα LinkedIn και Urwork) καθώς και τον αριθμών αναφορών στα κοινωνικά δίκτυα (λήφθηκε ο αριθμός των ανακοινώσεων στο Twitter).

**Πίνακας 1: Σύγκριση Συστημάτων Βάσεων Δεδομένων**

<b>Σύστημα Βάσης Δεδομένων</b>	<b>Υποστήριξη Πολλαπλών Συσκευών</b>	<b>Πόσο Δημοφιλές είναι (rank Sep. 2019)</b>	<b>Μοντέλο Άδειας Χρήσης</b>
InterBase ToGo / IBLite	NAI	50	Ιδιοκτησία
Oracle Database Lite	NAI	1	Ιδιοκτησία
SQL Anywhere; Ultralite is the mobile version	NAI	58	Ιδιοκτησία
SQLite	NAI	11	Ελεύθερο

Τα αποτελέσματα τις έρευνας δίνονται στον πίνακα 2 και είναι εμφανές ότι από τα δωρεάν συστήματα βάσεων δεδομένων το μόνο που καλύπτει τις απαιτήσεις μας είναι η SQLite. Μάλιστα για ελεύθερο μοντέλο αδειών χρήσης είναι πολύ δημοφιλές και αυτό θα βοηθήσει να βρεθεί κατάλληλη τεκμηρίωση για τη χρήση του, καθώς και για να λυθούν τυχόν προβλήματα που μπορεί να εμφανιστούν κατά την υλοποίηση.

Ο τρόπος σχεδιασμού, η επιλογή της εφαρμογής και ο τρόπος υλοποίησης της σχεσιακής βάσης δεδομένων θα παρουσιαστούν στο τρίτο κεφάλαιο.

## **2.3 Αρχική Αποτύπωση Διεπαφής**

Η αρχική αποτύπωση της διεπαφής μίας ιστοσελίδας ή μίας εφαρμογής είναι μία τεχνική που εμπλέκει τους χρήστες με τους δημιουργούς και είναι μία αρχική σύμβαση ανάμεσά τους για τη λειτουργικότητα και την αρχιτεκτονική της διεπαφής. Εξετάζοντας τα πλεονεκτήματα και τα μειονεκτήματα [7] της χρησιμοποίησης αυτής της τεχνικής διαπιστώθηκε ότι για την εφαρμογή που υλοποιείται δεν έχουν αντίκρισμα τα μειονεκτήματα ενώ βοηθούν τα πλεονεκτήματα.

Συγκεκριμένα, η εμπλοκή των χρηστών στο σχεδιασμό της λειτουργικότητας και της αρχιτεκτονικής της διεπαφής, εξασφαλίζει ότι θα είναι ικανοποιημένοι με το τελικό αποτέλεσμα και δεν θα απαιτηθούν αλλαγές σχεδιασμού την τελευταία στιγμή. Επιπλέον, ο εντοπισμός των απαιτήσεων όσον αφορά το σχεδιασμό της διεπαφής γίνεται σε αρχικό στάδιο και εξασφαλίζει ένα καλό αποτέλεσμα. Έτσι οι αλλαγές σε αυτό το στάδιο μπορούν να πραγματοποιηθούν πολύ πιο εύκολα, εξοικονομώντας χρόνο. Επίσης, η χρήση αυτής της τεχνικής επιτρέπει να σχεδιαστεί ο τρόπος παρουσίασης των περιεχομένων της κάθε σελίδας της εφαρμογής από την αρχή.

Λόγω της απλότητας των λειτουργικών απαιτήσεων της εφαρμογής, δεν υπάρχει κίνδυνος να εστιάσουν οι χρήστες στις λεπτομέρειες και να χρειαστεί επιπλέον χρόνος για την ολοκλήρωση του σταδίου αυτού, ούτε είναι δύσκολες οι σχεδιαστικές αλλαγές που τυχόν απαιτηθούν. Τέλος, ο περιορισμός που θέτει η χρήση αυτής της τεχνικής στην δημιουργικότητα που μπορεί να εκφραστεί από το σχεδιαστή της διεπαφής όταν υλοποιεί την εφαρμογή, δεν αποτελεί μειονέκτημα διότι δεν υπάρχουν συγκεκριμένες απαιτήσεις των χρηστών για την καλαισθησία της διεπαφής.

Η έρευνα για την επιλογή της εφαρμογής, ο τρόπος σχεδιασμού της αρχικής αποτύπωσης της διεπαφής και το αποτέλεσμα θα παρουσιαστούν στο τέταρτο κεφάλαιο.

## 2.4 Επιλογή Τεχνολογιών

Έχοντας καταλήξει στο σύστημα της σχεσιακής βάσης δεδομένων που θα χρησιμοποιηθεί και έχοντας μία αρχική αποτύπωση της διεπαφής, ξεκίνησε η έρευνα για το απαραίτητο ολοκληρωμένο περιβάλλον ανάπτυξης (IDE), το οποίο θα έδινε την δυνατότητα να υλοποιηθεί μία εφαρμογή ανεξάρτητη πλατφόρμας και ανεξάρτητη εξυπηρετητή. Χρησιμοποιήθηκαν τα παρακάτω κριτήρια για τον εντοπισμό του κατάλληλου IDE:

- 1.Υποστήριξη πολλαπλών συσκευών (PC, mobile, tablet κτλ)
- 2.Υποστήριξη πολλαπλών γλωσσών
- 3.Καμπύλη εκμάθησης
- 4.Πλήθος επεκτάσεων

Πραγματοποιήθηκε έρευνα για τυχόν οδηγούς που θα βοηθούσαν στην έναρξη της προσπάθειας και οι οποίοι θα προσέφεραν πληροφορίες για τον τρόπο χρήσης του εκάστοτε IDE, αφού βέβαια επιβεβαιώνονταν ότι καλύπτονται τα παραπάνω κριτήρια. Παρακάτω παρουσιάζεται το αποτέλεσμα της έρευνας αυτής, η προσπάθεια που έγινε για το κάθε IDE, οι τεχνολογίες που χρησιμοποιήθηκαν και που κατέληξε η κάθε προσπάθεια.

### 2.4.1 Netbeans, Java

Μια αρχική προσπάθεια για την ανάπτυξη της εφαρμογής πραγματοποιήθηκε θεωρώντας ότι η εφαρμογή θα λειτουργεί με σύνδεση στο διαδίκτυο. Εξετάστηκε το NetBeans ως ένα πολύ δημοφιλές περιβάλλον με εκτεταμένη τεκμηρίωση, αφού είναι ανοικτού κώδικα (open source), και υποστήριξη στην ανάπτυξη εφαρμογών σε πολλές γλώσσες και για πολλαπλές συσκευές. Μάλιστα από παρουσίαση των διαφορών IDE [8] έχει βαθμολογηθεί από τους χρήστες με 4,1 στα 5 και έχει αρκετές (στην τάξη των εκατοντάδων) κρητικές.

Μάλιστα αρχικά αναπτύχθηκε η σχεσιακή βάση δεδομένων σε MySQL, ένα πολύ δημοφιλές σύστημα (δεύτερο στην σειρά κατάταξης που εξετάστηκε σε προηγούμενη παράγραφο [6]), το οποίο όμως λειτουργεί πάνω σε εξυπηρετητή, αλλά στην προσπάθεια αυτή θεωρήθηκε ότι μπορεί να λειτουργήσει η εφαρμογή με μόνιμη σύνδεση στο διαδίκτυο. Πραγματοποιήθηκε ο σχεδιασμός της βάσης που παρουσιάζεται στο επόμενο κεφάλαιο και η υλοποίηση της βάσης δεδομένων στην MySQL. Χρησιμοποιήθηκε ως σύστημα διαχείρισης της βάσης δεδομένων το phpMyAdmin το οποίο εγκαταστάθηκε όπως και η MySQL σε έναν Apache Tomcat εξυπηρετητή.

Αποφασίστηκε να σχεδιαστεί η εφαρμογή χρησιμοποιώντας αρχιτεκτονική Model-View-Controller (MVC) [9] και χρησιμοποιήθηκε ένας αναλυτικός οδηγός για υλοποίηση εφαρμογής δικτύου στο NetBeans [10]. Η συγκεκριμένη αρχιτεκτονική προτείνει τη δημιουργία της εφαρμογής σε τρία επίπεδα. Το επίπεδο του μοντέλου (model) που διαχειρίζεται τα δεδομένα, το επίπεδο της παρουσίασης (view) που διαχειρίζεται τον τρόπο παρουσίασης των δεδομένων και το επίπεδο του ελεγκτή (controller) που διαχειρίζεται τις λειτουργίες και είναι υπεύθυνο για τις αλλαγές στα δύο άλλα επίπεδα. Τα διάφορα συστατικά (components) που αναπτύχθηκαν υλοποιήθηκαν σε java.

Η προσέγγιση αυτή εγκαταλείφθηκε όταν έγινε σαφές ότι η εφαρμογή έπρεπε να λειτουργεί χωρίς σύνδεση στο διαδίκτυο. Επιπλέον, πραγματοποιήθηκε δοκιμή για την εκτέλεση στο κινητό του αρχικού σχεδίου της εφαρμογής, που είχε ήδη αναπτυχθεί, αλλά διαπιστώθηκε ότι τα αρχεία .java δεν ήταν δυνατόν να εκτελεστούν με τον ίδιο απλό τρόπο που μπορούσε να γίνει σε λειτουργικό σύστημα Microsoft Windows. Για την εκτέλεση της εφαρμογής σε κινητές συσκευές είναι απαραίτητο να γίνει εκ νέου σύνταξη της εφαρμογής σε διαφορετικό πηγαίο κώδικα, κάτι για το οποίο δεν έγινε περαιτέρω έρευνα, αφού η προσέγγιση τελικά αυτή δεν εξυπηρετούσε τις απαιτήσεις.

#### **2.4.2 Eclipse, HTML, JavaScript**

Στη συνέχεια πραγματοποιήθηκε μία σύντομη προσπάθεια χρησιμοποιώντας την Eclipse, ένα επίσης δημοφιλές IDE με μεγάλη γκάμα από επεκτάσεις. Στην παρουσίαση των IDE [8] που χρησιμοποιήθηκε και πριν, έχει βαθμολογηθεί από δύο χιλιάδες χρήστες περίπου με 4,2 στα 5. Εξετάστηκε η προσέγγιση να δημιουργηθεί μία εφαρμογή διαδικτύου χρησιμοποιώντας ως γλώσσες προγραμματισμού τις HTML και JavaScript, η οποία θα εκτελείται με μεγαλύτερη ευκολία σε φορητές συσκευές και θα μπορεί να εκτελεστεί και χωρίς την ύπαρξη σύνδεσης στο διαδίκτυο. Στην προσπάθεια αυτή, είχε ληφθεί η απόφαση να χρησιμοποιηθεί ως σύστημα βάσης δεδομένων η SQLite, το οποίο επιτρέπει η βάση δεδομένων να είναι ένα αρχείο.

Η προσπάθεια δεν προχώρησε πολύ διότι διαπιστώθηκε ότι η σύγχρονη HTML δεν επιτρέπει την αλληλεπίδραση με το σύστημα αρχείων (file system) της συσκευής που εκτελείται για λόγους ασφαλείας. Χωρίς τη δυνατότητα πρόσβασης σε συγκεκριμένα αρχεία, και συγκεκριμένα σε ένα αρχείο βάσης δεδομένων που θα βρίσκεται πάντα στο ίδιο σημείο και δεν θα μπορεί να διαγραφεί από το σύστημα ανάλογα με τις τυχόν ρυθμίσεις, δεν υπήρχε δυνατότητα να μπορεί να εκτελεστεί χωρίς σύνδεση στο διαδίκτυο με παρόντα όλα τα δεδομένα.

#### **2.4.3 Visual Studio, Apache Cordova, HTML, JavaScript**

Στην τελευταία αυτή προσπάθεια πραγματοποιήθηκε έρευνα για την εύρεση ενός ολοκληρωμένου πλαισίου ανάπτυξης (framework) εφαρμογών ανεξάρτητων πλατφόρμας. Από τα συστήματα αυτά ερευνήθηκαν περισσότερο αυτά που προσφέρονταν δωρεάν [11] και πάρθηκε η απόφαση να χρησιμοποιηθεί η Apache Cordova, διότι χρησιμοποιεί γλώσσες προγραμματισμού ευρέως γνωστές και τεκμηριωμένες όπως η HTML και η JavaScript. Μάλιστα το γεγονός ότι είναι ένα ολοκληρωμένο πλαίσιο ανάπτυξης ανοικτού κώδικα προσφέρει τη δυνατότητα να ερευνηθεί σε βάθος ο τρόπος λειτουργίας του και να αναζητηθεί βοήθεια στη μεγάλη κοινότητα ανάπτυξης που το υποστηρίζει. Επίσης, ένα από τα πλεονεκτήματα που παρέχει είναι η συνεχής υποστήριξη από την ομάδα ανάπτυξης της Apache, η οποία ενημερώνει τακτικά τη Cordova.

Η ύπαρξη τεχνογνωσίας στις γλώσσες προγραμματισμού HTML και JavaScript βοήθησε περαιτέρω στην επιλογή αυτή. Επιπλέον, επιλέχθηκε να γίνει χρήση και του jQuery mobile που είναι μία βιβλιοθήκη που χρησιμοποιείται με τη JavaScript και βοηθάει σε αποτελεσματικότερη συγγραφή κώδικα, αφού απλοποιεί τον τρόπο που γράφεται ο κώδικας, προσφέροντας μεθόδους που μπορεί να καλέσει ο προγραμματιστής στον κώδικά του. Η συγκεκριμένη βιβλιοθήκη είναι προσανατολισμένη για την ανάπτυξη εφαρμογών σε ποικίλα περιβάλλοντα και

συσκευές, είναι ευρέως χρησιμοποιούμενη και τεκμηριωμένη, οπότε η εκμάθηση και η χρήση της θεωρήθηκε σκόπιμη.

Το ολοκληρωμένο πλαίσιο ανάπτυξης Apache Cordova μπορεί να χρησιμοποιηθεί σε ποικίλα IDE, αλλά επιλέχθηκε το Visual Studio της Microsoft το οποίο στην παρουσίαση των IDE που εξετάστηκαν [8] βαθμολογήθηκε με 4,4 στα 5 από σχεδόν δύο χιλιάδες χρήστες. Είναι δημοφιλές περιβάλλον ανάπτυξης με σημαντική τεκμηρίωση. Μάλιστα στους διαθέσιμους οδηγούς του Visual Studio εντοπίστηκε ένας οδηγός για την ανάπτυξη εφαρμογής για φορητές συσκευές με χρήση της Apache Cordova [12], ο οποίος αποτέλεσε ουσιαστικά τον οδηγό για να ξεκινήσει η προσπάθεια αυτή.

Με το σύνολο των παραπάνω εργαλείων και τη χρήση μίας βιβλιοθήκης υλοποίησης της SQLite σε JavaScript [13], η οποία δημιουργήθηκε μεταγλωττίζοντας την SQLite σε JavaScript χρησιμοποιώντας το emscripten [14], προχώρησε η υλοποίηση της διεπαφής της εφαρμογής, η σύνδεσή της με τη βάση δεδομένων και η ανάπτυξη της λειτουργικότητάς της. Η εφαρμογή αναπτύχθηκε σε λειτουργικό περιβάλλον Microsoft Windows, στο οποίο έγιναν και οι αρχικές δοκιμές λειτουργίας.

Ο τρόπος με τον οποίον σχεδιάστηκε και υλοποιήθηκε η εφαρμογή παρουσιάζονται αναλυτικά στο πέμπτο κεφάλαιο.

## **2.5 Ασύγχρονη Εκτέλεση**

Μία από τις κυριότερες απαιτήσεις της εφαρμογής, όπως φάνηκε τελικά, είναι η εκτέλεσή της χωρίς τη σύνδεση στο διαδίκτυο και τον συγχρονισμό των δεδομένων όταν υπάρχει σύνδεση στο διαδίκτυο. Η κάλυψη αυτής της ανάγκης ήταν το επόμενο βήμα.

Αρχικά έγινε προσπάθεια να βρεθεί ένας τρόπος να αποθηκεύεται το αρχείο της βάσης δεδομένων σε ένα απομακρυσμένο σημείο, από το οποίο θα μπορεί η εφαρμογή όταν εκτελείτε με σύνδεση στο διαδίκτυο να το πάρει και να χρησιμοποιήσει τα δεδομένα του. Στη συνέχεια η εφαρμογή όταν εκτελείται χωρίς σύνδεση στο διαδίκτυο συνδέεται και χρησιμοποιεί το αρχείο της βάσης δεδομένων που υπάρχει τοπικά στη συσκευή που εκτελείται. Πραγματοποιήθηκε έρευνα η οποία ανέδειξε ότι το Dropbox προσφέρει πακέτα ανάπτυξης λογισμικού (Software Development Kit – SDK) για τη χρησιμοποίηση του Dropbox σε εφαρμογές. Μάλιστα προσφέρει και ένα πακέτο σε JavaScript [15], κάτι που εξυπηρετεί πλήρως την ενσωμάτωσή του στην εφαρμογή που αναπτύσσονταν, και το οποίο τελικά χρησιμοποιήθηκε.

## **2.6 Αναλυτική Περιγραφή Απαιτήσεων Εφαρμογής**

Στο σημείο αυτό παρουσιάζονται οι τελικές αναλυτικές απαιτήσεις από την εφαρμογή, οι οποίες διαμορφώθηκαν από τις αρχικές και περιορίστηκαν ή εμπλουτίστηκαν σταδιακά όπως περιγράφηκε στα προηγούμενα υποκεφάλαια του κεφαλαίου και στις οποίες έχουν συμπεριληφθεί και οι ανάγκες που εντοπίστηκαν μετά τον έλεγχο του λογισμικού που παρουσιάζεται στο έκτο κεφάλαιο.

### **2.6.1 Εισαγωγή Θεραπείας**

Να υπάρχει δυνατότητα εισαγωγής μίας θεραπείας στη βάση δεδομένων με τα παρακάτω δεδομένα, τα οποία δόθηκαν στα Αγγλικά. Το σύνολο των πληροφοριών - γνωρισμάτων αποθηκεύονται στα Αγγλικά εκτός από τις περιπτώσεις που σημειώνεται κάτι διαφορετικό :

- **Treatment Name** : Το όνομα της θεραπείας το οποίο είναι κείμενο. Δεν πρέπει να υπάρχει η δυνατότητα να υπάρχουν δύο θεραπείες με το ίδιο όνομα.
- **Date Taught** : Η ημερομηνία που παρουσιάστηκε αυτή η θεραπεία. Μπορεί να είναι και δύο οπότε και αυτή η πληροφορία είναι κείμενο.
- **Issues Addressed** : Να σημειώνεται ποια θέματα (Issues) αντιμετωπίζει. Το κάθε θέμα που αντιμετωπίζεται αφορά συγκεκριμένο επίπεδο θεραπειών (Disease Aspect). Επειδή μία θεραπεία μπορεί να αντιμετωπίζει πολλά θέματα, μπαίνει και μία σειρά ως προς την αποτελεσματικότητα με όνομα (Level Name) και αριθμητική τιμή (Level Degree). Επίσης, τα θέματα εκτός από το όνομά τους έχουν μία περιγραφή (Issue Description), μία κατηγορία (Issue Category) και μία υποκατηγορία (Issue Sub Category) το κάθε ένα. Σημειώνεται εδώ ότι μπορεί να υπάρξει μία θεραπεία που αντιμετωπίζει διαφορετικά θέματα με την ίδια αποτελεσματικότητα, και ότι υπάρχει περίπτωση να συνδέεται μία θεραπεία με ένα επίπεδο θεραπειών χωρίς σύνδεση στα θέματα και την αποτελεσματικότητα.
- **Treatment Type** : Σημειώνεται ο τύπος της θεραπείας σύμφωνα με μία λίστα που δόθηκε. Μία θεραπεία μπορεί να συνδέεται με έναν ή περισσότερους τύπους.
- **Treatment Description** : Η αναλυτική περιγραφή της θεραπείας το οποίο είναι κείμενο.
- **Treatment Short Description Greek** : Σύντομη περιγραφή της θεραπείας στα Ελληνικά, δηλαδή κείμενο.
- **Treatment Short Description Greek in Unicode Normal Form D** : Η παραπάνω σύντομη περιγραφή της θεραπείας στα Ελληνικά αλλά σε μία κανονικοποιημένη μορφή που εξυπηρετεί στην αναζήτηση. Δεν είναι ακριβώς η Unicode Normal Form D (NFD) που προβλέπει το πρότυπο Unicode [16] αλλά είναι το ελληνικό κείμενο χωρίς τόνους και κεφαλαία. Δεν εμφανίζεται στη διεπαφή της εφαρμογής.
- **Treatment Short Description English** : Σύντομη περιγραφή της θεραπείας που είναι κείμενο.
- **Treatment Special Category** : Κάθε θεραπεία μπορεί να συνδέεται με μία ή περισσότερες ειδικές κατηγορίες που δόθηκαν σαν λίστα στα Ελληνικά και έχουν και μία περιγραφή η κάθε μία (Special Category Description). Όπως και παραπάνω το με τη σύντομη περιγραφή στα Ελληνικά της θεραπείας, αποθηκεύεται το όνομα της ειδικής κατηγορίας χωρίς τόνους και κεφαλαία (Special Category Name in Unicode Normal Form D).
- **Treatment Parts** : Εδώ σημειώνεται αν η συγκεκριμένη θεραπεία γίνεται σε πολλά μέρη και συγκεκριμένα σε πόσα (Parts Number) που ζητήθηκε να είναι κείμενο. Επιπλέον σημειώνονται σε κείμενο το μικρότερο και μεγαλύτερο επιτρεπτό χρονικό διάστημα ανάμεσα στις θεραπείες (Minimum Interval between Parts, Maximum Interval between Parts), το πώς μετριέται το διάστημα (Interval Metric) για παράδειγμα εβδομάδες, μήνες κτλ και το βέλτιστο χρονικό διάστημα (Best Interval).
- **Body Area** : Η περιοχή του σώματος που πραγματοποιείται η θεραπεία και είναι κείμενο.
- **Start with** : Η περιοχή του σώματος από την οποία ξεκινάει η θεραπεία. Ζητήθηκε να είναι κείμενο.
- **Treatment Limitations** : Περιγραφή των περιορισμών που τυχόν υπάρχουν για μία θεραπεία, δηλαδή κείμενο.

- **Treatment Specific Timing** : Περιγραφή αν υπάρχουν συγκεκριμένος χρόνος στον οποίον πρέπει να γίνει η θεραπεία, που είναι κείμενο.
- **Treatment Series Type** : Σημειώνεται αν η θεραπεία είναι ανεξάρτητη ή αποτελεί μέρος μίας σειράς θεραπειών.
- **Gauging Resistance** : Εδώ σημειώνεται ο τρόπος με τον οποίο μπορεί να μετρήσει κανείς την αντίσταση του αποδέκτη της θεραπείας, άρα είναι κείμενο.
- **Possible Negative Effects** : Περιγράφονται τυχόν αντιδράσεις που μπορεί να εμφανιστούν σε μορφή κειμένου.
- **Treatment Integration Period** : Σημειώνεται ο τρόπος ενσωμάτωσης της θεραπείας το οποίο είναι κείμενο.
- **Treatment Links** : Στο σημείο αυτό σημειώνεται η σύνδεση κάθε θεραπείας με άλλες. Υπάρχουν συγκεκριμένοι τρόποι σύνδεσης (Link Type Name) και για κάθε σύνδεση μπορεί να υπάρχει μία περιγραφή (Link Description) που είναι κείμενο.
- **Recommended Actions for Recipient** : Περιγράφονται εδώ προτεινόμενες ενέργειες για τον αποδέκτη της θεραπείας και είναι κείμενο.
- **Treatment Timing Post** : Εδώ σημειώνονται τυχόν χρονικοί περιορισμοί μετά την θεραπεία και ζητήθηκε να είναι κείμενο.
- **Treatment Requirements** : Περιγραφή τυχόν απαιτήσεων για τη συγκεκριμένη θεραπεία που είναι κείμενο.
- **Energy Type** : Ο τύπος της ενέργειας που απαιτεί μία θεραπεία ο οποίος πρέπει να έχει και μία περιγραφή (Energy Type Description) ο κάθε ένας. Μπορεί μία θεραπεία να έχει περισσότερους από ένα τύπους ενέργειας.
- **Minimum Energy Requirements** : Περιγράφονται εδώ τυχόν απαιτήσεις στην ενέργεια και είναι κείμενο.
- **Treatment Duration** : Σημειώνεται η διάρκεια της θεραπείας και ζητήθηκε να είναι κείμενο.
- **Treatment Special Requirements** : Εδώ σημειώνονται τυχόν περαιτέρω απαιτήσεις της θεραπείας που είναι ποιο εξειδικευμένες και είναι κείμενο.

Ζητήθηκε να υπάρχει δυνατότητα να μπει μία θεραπεία μόνο με το όνομά της και να ενημερωθεί στη συνέχεια με τα υπόλοιπα δεδομένα.

Κάποια από τα παραπάνω γνωρίσματα ομαδοποιήθηκαν ως βασικά και τα οποία απαιτήθηκε να είναι άμεσα διαθέσιμα όπου εμφανίζεται μία θεραπεία (μετά από μία αναζήτηση για παράδειγμα). Τα βασικά γνωρίσματα είναι : Treatment Name, Treatment Duration, Date Taught, Treatment Short Description Greek και το Treatment Parts.

## 2.6.2 Προβολή Θεραπείας

Να παρέχεται η δυνατότητα να προβάλλονται όλες οι παραπάνω πληροφορίες για μία θεραπεία και να δίνεται η δυνατότητα στο χρήστη να διαγράψει τη θεραπεία αν το επιθυμεί, καθώς και να επεξεργαστεί τα δεδομένα της. Για την προβολή όλων των πληροφοριών εκφράστηκε η απαίτηση να είναι ομαδοποιημένες για ευκολότερη πλοήγηση σε αυτές. Σημειώνεται ότι απαιτήθηκε να μην εμφανίζονται οι τίτλοι των γνωρισμάτων που δεν έχουν δεδομένα, έτσι ώστε η παρουσίαση της θεραπείας να περιέχει μόνο τα γνωρίσματα με αποθηκευμένη πληροφορία.



### 2.6.3 Επεξεργασία Θεραπείας

Η εφαρμογή να παρέχει τη δυνατότητα να γίνεται η ενημέρωση και επεξεργασία των δεδομένων μίας θεραπείας που παρουσιάστηκαν παραπάνω. Ο χρήστης να μπορεί να αλλάξει όλα τα στοιχεία της θεραπείας και να επιλέγει το πότε θα ενημερώσει τη βάση δεδομένων.

### 2.6.4 Λίστα Θεραπειών

Να υπάρχει η δυνατότητα να εμφανίζεται το σύνολο των θεραπειών που υπάρχει καταχωρημένο στη βάση δεδομένων. Απαιτήθηκε να υπάρχει λίστα με το όνομα της θεραπείας το αν έχει ή όχι μέρη και να υπάρχει η δυνατότητα να εμφανίζονται άμεσα για κάθε θεραπεία κάποια βασικά γνωρίσματα. Τα βασικά αυτά γνωρίσματα είναι αυτά που προαναφέρθηκαν, δηλαδή Treatment Name, Treatment Duration, Date Taught, Treatment Short Description Greek και Treatment Parts. Επίσης, να υπάρχει δυνατότητα να μπορεί να δει κανείς το σύνολο των πληροφοριών μίας θεραπείας, καθώς και η δυνατότητα να ξεκινήσει την επεξεργασία των δεδομένων της.

### 2.6.5 Αναζήτηση Θεραπείας

Αυτή είναι και η σημαντικότερη απαίτηση από την εφαρμογή. Η αναζήτηση οφείλει να γίνεται με συγκεκριμένα κριτήρια. Να υπάρχει η δυνατότητα να αναζητηθεί ένα κείμενο στο σύνολο των πληροφοριών μίας θεραπείας ή να αναζητηθεί ένα κείμενο σε συγκεκριμένο σημείο των γνωρισμάτων (για παράδειγμα στο γνώρισμα Treatment Requirements). Επίσης, να παρέχεται η δυνατότητα να ορίζονται περισσότερα του ενός κριτήρια τα οποία είτε να λειτουργούν αθροιστικά (να ισχύουν όλα δηλαδή) είτε να λειτουργούν συμπληρωματικά (να ισχύει τουλάχιστον ένα δηλαδή). Αν μάλιστα δοθούν στο γενικό κείμενο (το κείμενο που αναζητείται παντού) περισσότερες από μία λέξεις, να μπορεί να λειτουργήσουν όπως και πριν, είτε αθροιστικά είτε συμπληρωματικά.

Απαιτήθηκε επίσης στη λίστα με τις θεραπείες που ικανοποιούν τα κριτήρια να εμφανίζονται όλες οι θεραπείες με το σύνολο των βασικών γνωρισμάτων που περιγράφηκαν παραπάνω και αν θέλει ο χρήστης να τα αποκρύπτει.

### 2.6.6 Διαχείριση Περιφερειακών Δεδομένων

Κατά τη διάρκεια ανάπτυξης της εφαρμογής εντοπίστηκε η απαίτηση για τη διαχείρισης κάποιων περιφερειακών δεδομένων για το χαρακτηρισμό των θεραπειών. Με τον όρο διαχείριση εννοείται η εισαγωγή, ενημέρωση και διαγραφή των δεδομένων αυτών όπως είναι για παράδειγμα το «Energy Type» ή το «Treatment Type». Αρχικά δεν υπήρξε πρόβλεψη για τον τρόπο που θα ενημερώνονταν οι τιμές αυτών των δεδομένων, αλλά στον έλεγχο του λογισμικού βρέθηκε το κενό.

### 2.6.7 Περιβάλλον Εκτέλεσης Εφαρμογής

Όπως περιγράφηκε σε προηγούμενες υποπαραγράφους του κεφαλαίου, μετά τις αποτυχημένες προσπάθειες να εκτελεστεί η εφαρμογή σε ποικίλα περιβάλλοντα,

κατέληξε η απαίτηση αυτή να αρθεί και η εφαρμογή να εκτελείται σε περιβάλλον Microsoft Windows. Η εφαρμογή δοκιμάστηκε και επιβεβαιώθηκε η εκτέλεσή της στο λειτουργικό σύστημα Microsoft Windows 10.

### **2.6.8 Πλήθος Δεδομένων**

Απαιτήθηκε αρχικά να μπορεί η εφαρμογή να λειτουργεί με περίπου πέντε χιλιάδες θεραπείες, αλλά ήταν αδύνατο να δοκιμαστεί η εφαρμογή σε αυτές τις συνθήκες. Οι θεραπείες που καταχωρήθηκαν στην εφαρμογή ήταν στη τάξη των δεκάδων, οι οποίες ωστόσο ήταν αρκετές για να γίνει ο έλεγχος του λογισμικού σε κανονικές συνθήκες αλλά όχι σε συνθήκες κορεσμού.

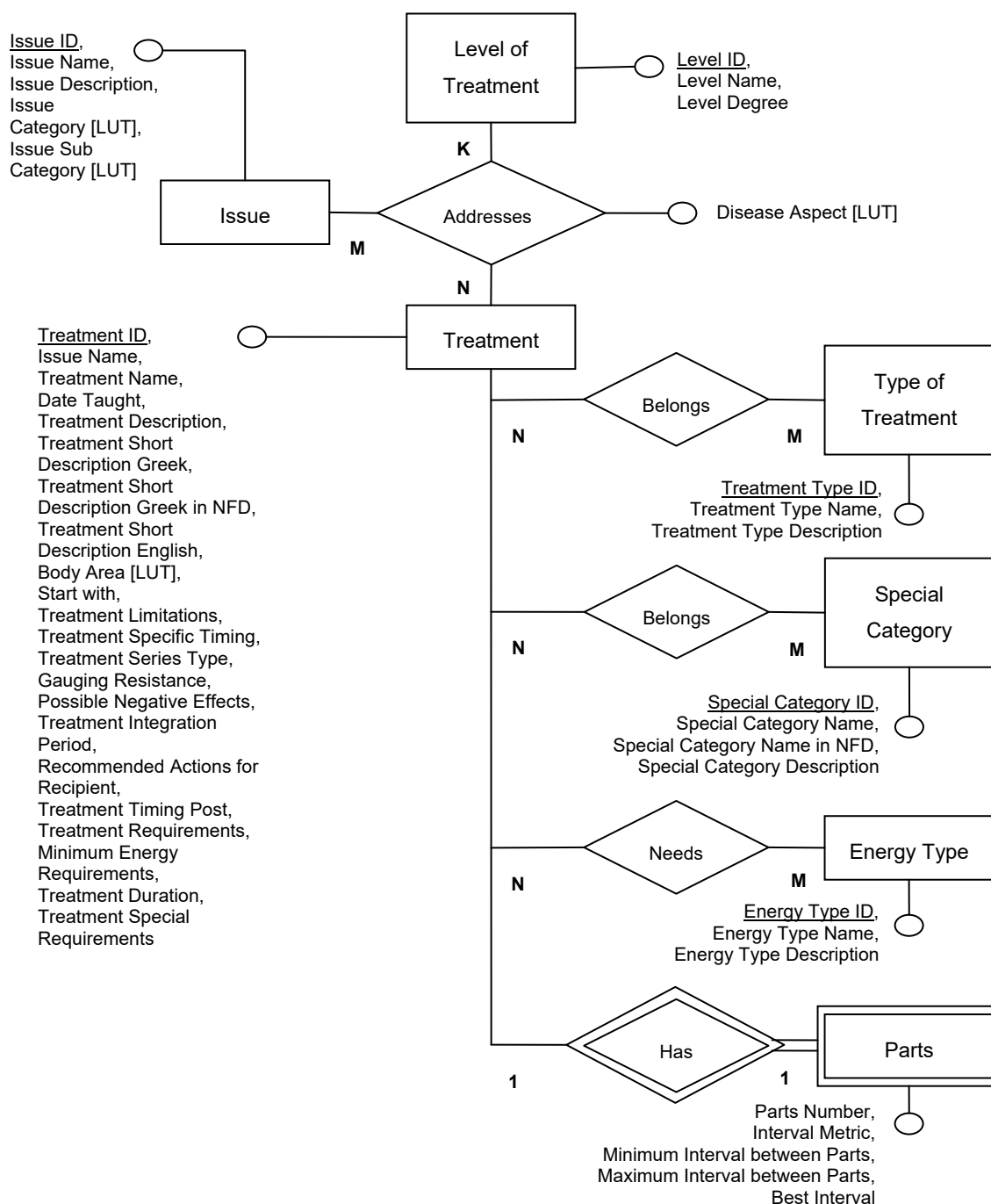
### **2.6.9 Ασύγχρονη Εκτέλεση**

Η απαίτηση να μπορεί η εφαρμογή να εκτελεστεί ασύγχρονα και ανεξάρτητα από την ύπαρξη σύνδεσης στο διαδίκτυο περιορίστηκε. Και αυτό διότι μόνο ένας χρήστης μπορεί να κάνει αλλαγές κάθε φορά και άρα πρέπει να υπάρχει επικοινωνία μεταξύ των χρηστών της εφαρμογής. Η απαίτηση λοιπόν διαμορφώθηκε στο να μπορεί να εκτελείται η εφαρμογή ανεξάρτητα από τη σύνδεση στο διαδίκτυο.

### 3. ΥΛΟΠΟΙΗΣΗ ΣΧΕΣΙΑΚΗΣ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ ΑΝΕΞΑΡΤΗΤΗΣ ΕΞΥΠΗΡΕΤΗΤΗ

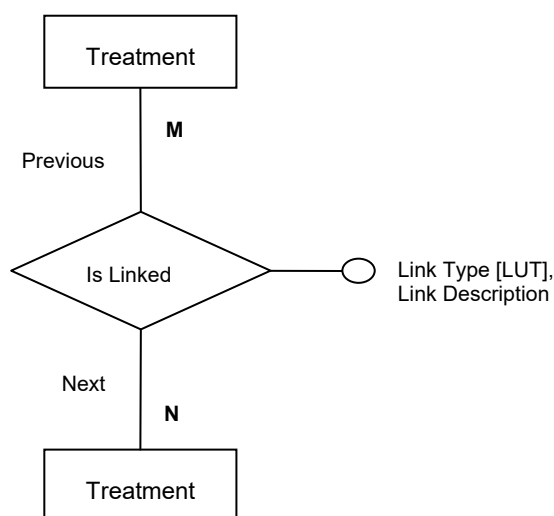
#### 3.1 Σχεδιασμός Σχεσιακής Βάσης Δεδομένων

Για το σχεδιασμό της βάσης δεδομένων χρησιμοποιήθηκε αρχικά το μοντέλο οντοτήτων συσχετίσεων και δημιουργήθηκαν δύο Διαγράμματα Οντοτήτων Συσχετίσεων (ΔΟΣ). Στη συνέχεια, πραγματοποιήθηκε η μετατροπή των ΔΟΣ στο σχεσιακό μοντέλο και αξιολογήθηκε αν το αποτέλεσμα είναι στις κανονικές μορφές του μοντέλου. Η όλη διαδικασία στηρίχθηκε σε βιβλιογραφία [17] και σε προηγούμενη εμπειρία στο σχεδιασμό και ανάπτυξη βάσεων δεδομένων. Έχοντας την ονομασία των γνωρισμάτων στα Αγγλικά, η ονοματολογία που χρησιμοποιήθηκε είναι στα Αγγλικά.



Σχήμα 1: Πρώτο Διάγραμμα Οντοτήτων Συσχετίσεων

Για την καλύτερη αποτύπωση των συσχετίσεων δημιουργήθηκαν δύο διαγράμματα, στο δεύτερο εκ των οποίων δεν σημειώθηκαν εκ νέου τα γνωρίσματα της θεραπείας (treatment), αφού είχαν σημειωθεί στο πρώτο. Βάσει λοιπόν των αναλυτικών απαιτήσεων που δόθηκαν για τα δεδομένα δημιουργήθηκαν το πρώτο ΔΟΣ που δίνεται στο σχήμα 1 και το δεύτερο ΔΟΣ που δίνεται στο σχήμα 2. Στα διαγράμματα δίνονται οι οντότητες με τα γνωρίσματά τους και τις συσχετίσεις τους. Για κάποια γνωρίσματα των οντοτήτων που ήταν τυποποιημένα χρησιμοποιήθηκε η ορολογία του πίνακα αναφοράς (look-up table – LUT). Η βασική κυρίαρχη οντότητα στα διαγράμματα όπως είναι φυσικό είναι η θεραπεία (Treatment).

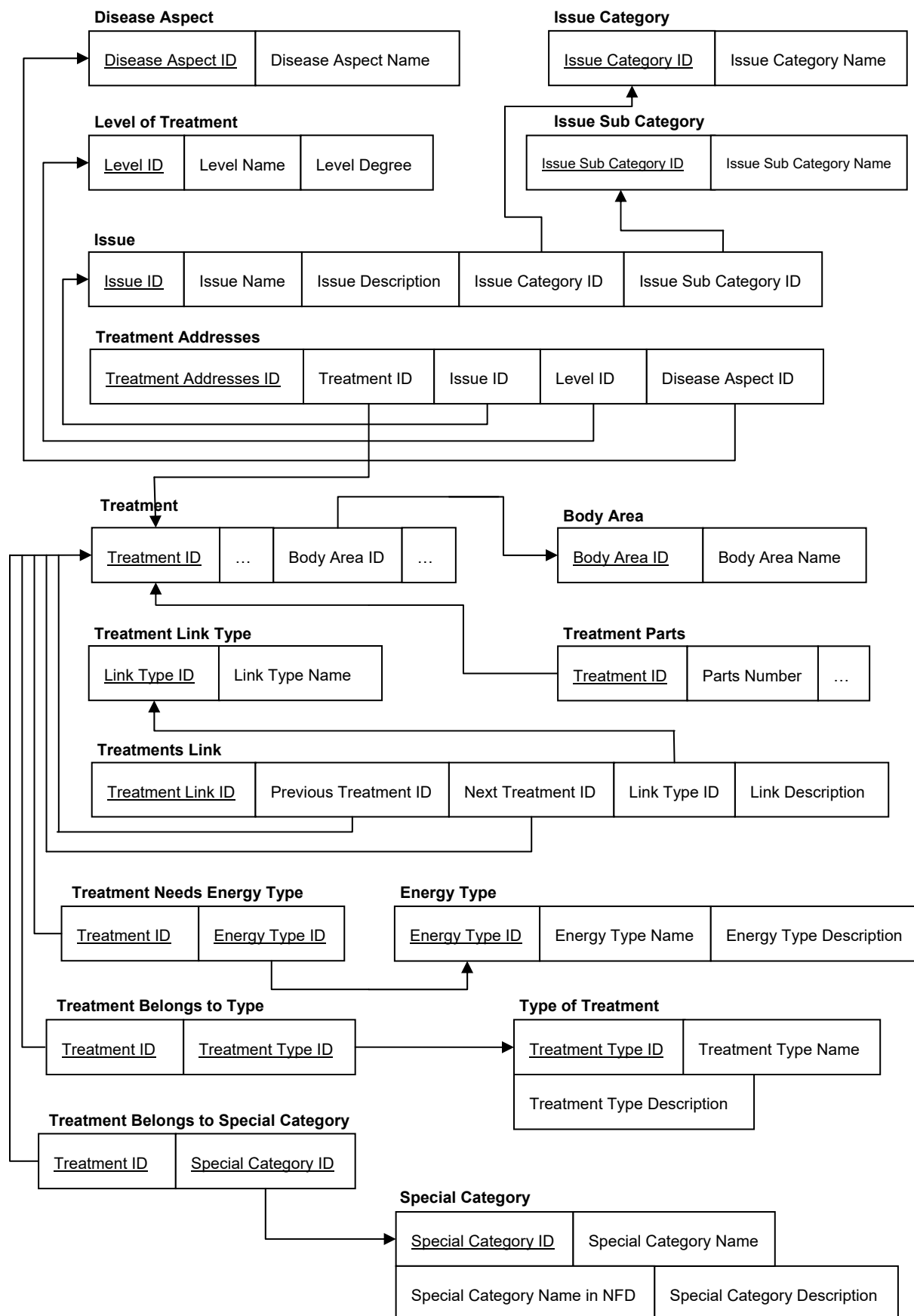


**Σχήμα 2: Δεύτερο Διάγραμμα Οντοτήτων Συσχετίσεων**

Σημειώνεται στο σημείο αυτό ότι η ασθενής οντότητα που δημιουργήθηκε για τα μέρη (parts) μίας θεραπείας αν και δεν έχει μοναδικό κλειδί απεικονίστηκε σαν ξεχωριστή οντότητα διότι τα συγκεκριμένα δεδομένα απαιτούν ενημέρωση μόνο αν μία θεραπεία έχει μέρη και αυτές οι θεραπείες είναι συγκριτικά λίγες σε σχέση με το σύνολο. Αποφασίστηκε λοιπόν να τις αποθηκεύουμε σαν ξεχωριστή οντότητα για να αποφύγουμε στη συνέχεια έναν πίνακα με κάποια πεδία γεμάτα κενά.

Η μετατροπή των ΔΟΣ στο σχεσιακό μοντέλο μας έδωσε τις σχέσεις που δίνονται στο σχήμα 3, στο οποίο φαίνονται τα πρωτεύοντα κλειδιά των σχέσεων υπογραμμισμένα και τα ξένα κλειδιά με κατευθυντήρια βέλη. Λόγω του πλήθους και του μήκους των ονομάτων των γνωρισμάτων της θεραπείας και των μερών της, εμφανίζονται μόνο το πρωτεύον κλειδί και τα ξένα κλειδιά στις σχέσεις αυτές.

Ο έλεγχος για το αν η σχεσιακή βάση δεδομένων που σχεδιάσαμε καλύπτει τις κανονικές μορφές του μοντέλου έδωσε τα παρακάτω αποτελέσματα για κάθε κανονική μορφή.



Σχήμα 3: Σχεσιακό Μοντέλο Βάσης Δεδομένων

### 3.1.1 Πρώτη Κανονική Μορφή (First Normal Form – 1NF)

Η πρώτη κανονική μορφή αποτελεί σήμερα μέρος του τυπικού ορισμού μίας σχέσης, οπότε καλύπτεται από όλες τις σχέσεις οι οποίες εμφανίζονται. Οι μόνες τιμές γνωρισμάτων που επιτρέπονται από την 1NF είναι απλές ατομικές τιμές, το οποίο ισχύει για όλα τα γνωρίσματα.

### 3.1.2 Δεύτερη Κανονική Μορφή (Second Normal Form – 2NF)

Η μορφή αυτή στηρίζεται στην πλήρη συναρτησιακή εξάρτηση μεταξύ των γνωρισμάτων των σχέσεων. Με απλά λόγια η δεύτερη κανονική μορφή απαιτεί όλα τα γνωρίσματα μίας σχέσης να εξαρτώνται πλήρως από το πρωτεύον κλειδί μίας σχέσης. Για τη βάση που δημιουργήθηκε οι περισσότερες σχέσεις έχουν απλά κλειδιά και ικανοποιούν την 2NF. Οι σχέσεις που έχουν διπλά πρωτεύοντα κλειδιά είναι σχέσεις που δημιουργήθηκαν για να συνδέσουν την κεντρική σχέση «Treatment» με άλλες οντότητες και δεν έχουν από μόνες τους γνωρίσματα. Η μόνη σχέση που εμφανίζει πολυπλοκότητα είναι η σχέση «Treatment Addresses» αλλά για τη συγκεκριμένη σχέση δημιουργήθηκε ξεχωριστό πρωτεύον κλειδί διότι υπάρχει περίπτωση να μην είναι διαθέσιμες όλες οι πληροφορίες για μία συσχέτιση, δηλαδή η συσχέτιση «Addresses» να μην ακουμπάει σε όλες τις οντότητες που συνδέονται. Επειδή λοιπόν για κάθε πλειάδα της σχέσης αυτής υπάρχει ξεχωριστό πρωτεύον κλειδί από το οποίο εξαρτώνται όλα τα γνωρίσματα της σχέσης, άρα ικανοποιείται η δεύτερη κανονική μορφή.

### 3.1.3 Τρίτη Κανονική Μορφή (Third Normal Form – 3NF)

Εδώ εξετάζεται η μεταβατική εξάρτηση ανάμεσα στα γνωρίσματα κάθε σχέσης και δεν πρέπει να υπάρχει τέτοια εξάρτηση από το πρωτεύον κλειδί μίας σχέσης σε ένα γνώρισμα που δεν ανήκει στο κλειδί. Λόγω των απαιτήσεων της εφαρμογής δημιουργήθηκαν κάποια γνωρίσματα όπως για παράδειγμα το «Treatment Short Description Greek in NFD», το οποίο εξαρτάται μεταβατικά από το «Treatment Short Description Greek» και μετά από το πρωτεύον κλειδί. Το ίδιο ισχύει και για το γνώρισμα «Special Category Name in NFD». Επομένως η βάση που δημιουργήθηκε δεν είναι στην 3NF. Δεν έχει νόημα να εξεταστεί περαιτέρω αν ικανοποιεί άλλες κανονικές μορφές, αφού κάθε επόμενη κανονική μορφή προϋποθέτει ο σχεδιασμός της βάσης να ικανοποιεί τις προηγούμενες.

Το γεγονός ότι δεν καλύπτει η σχεσιακή βάση δεδομένων που σχεδιάστηκε την τρίτη κανονική μορφή δεν αποτελεί πρόβλημα στη λειτουργία της βάσης. Θεωρητικά η λύση στο πρόβλημα αυτό θα ήταν να δημιουργηθεί μία νέα σχέση στην οποία το πρωτεύον κλειδί θα ήταν το γνώρισμα από το οποίο εξαρτάται μεταβατικά το γνώρισμα που εντοπίστηκε με πρόβλημα, στη δική μας περίπτωση το «Treatment Short Description Greek in NFD», και στη συνέχεια να συνδεθεί με την αρχική σχέση. Ωστόσο το να δημιουργήσει κανείς ένα πρωτεύον κλειδί με το γνώρισμα «Treatment Short Description Greek» είναι τεχνικά άστοχο από τη στιγμή που το γνώρισμα αυτό είναι κείμενο και μάλιστα εκτενές.

### 3.2 Επιλογή Εφαρμογής Διαχείρισης

Στη συνέχεια πραγματοποιήθηκε έρευνα για τον εντοπισμό κατάλληλων εργαλείων δημιουργίας και διαχείρισης μίας σχεσιακής βάσης δεδομένων SQLite και εντοπίστηκαν κάποια πιθανά εργαλεία. Από μία παρουσίαση διαφόρων εργαλείων διαχείρισης βάσεων δεδομένων [18] επιλέχθηκαν και ερευνήθηκαν τα συστήματα αυτά που υποστήριζαν SQLite. Ο στόχος ήταν να επιλεγεί ένα πρόγραμμα διαχείρισης με απλότητα, χωρίς ιδιαίτερες απαιτήσεις εγκατάστασης και αν ήταν εφικτό ανοικτού κώδικα, δηλαδή χρησιμοποιήσιμο δωρεάν.

Αρχικά εξετάστηκε το Adminer [19], ένα πρόγραμμα διαχείρισης πολλών διαφορετικών βάσεων δεδομένων το οποίο είναι γραμμένο σε php, οπότε εκτελείτε πάνω σε εξυπηρετητή (server). Η απαίτηση να εκτελείται η εφαρμογή διαχείρισης της βάσης σε εξυπηρετητή απέκλεισε τη χρήση του συγκεκριμένου προγράμματος.

Στη συνέχεια εντοπίστηκε το DB Browser for SQLite [20], το οποίο είναι ένα πρόγραμμα διαχείρισης βάσεων δεδομένων SQLite και μόνο. Είναι ελαφρύ πρόγραμμα ανοικτού κώδικα χωρίς ιδιαίτερες απαιτήσεις εγκατάστασης και λειτουργεί ανεξάρτητα από την ύπαρξη εξυπηρετητή. Τα χαρακτηριστικά αυτά το καθιστούν εξαιρετική επιλογή.

Εξετάστηκε στη συνέχεια το DBBeaver [21], ένα εργαλείο ανοικτού κώδικα για τη διαχείριση πολλών διαφορετικών συστημάτων βάσεων δεδομένων. Δεν απαιτεί εξυπηρετητή για να εγκατασταθεί αλλά χρειάζεται την εγκατάσταση ενός Java Runtime Environment (JRE) για να λειτουργήσει. Αυτό καθιστά τις απαιτήσεις εγκατάστασης σχετικά υψηλές και το πρόγραμμα αρκετά πολύπλοκο τη στιγμή που έχει σχεδιαστεί για να διαχειρίζεται ποικίλες βάσεις δεδομένων.

Τέλος ερευνήθηκε το DbVisualizer Free [22], το οποίο είναι και αυτό ένα πρόγραμμα διαχείρισης ποικίλων συστημάτων βάσεων δεδομένων. Το συγκεκριμένο πρόγραμμα είναι ιδιοκτησία εταιρείας αλλά παρέχεται και δωρεάν έκδοση, μάλιστα τα αρχεία της εφαρμογής είναι κοινά και για τις δύο περιπτώσεις. Η εμπειρία σε παρόμοιες περιπτώσεις δείχνει ότι η δωρεάν έκδοση παρέχει περιορισμένη λειτουργικότητα, οπότε και αυτή η περίπτωση αποκλείστηκε.

Από τα παραπάνω εργαλεία επιλέχθηκε το DB Browser for SQLite, εγκαταστάθηκε και χρησιμοποιήθηκε για τη δημιουργία της σχεσιακής βάσης δεδομένων.

### 3.3 Υλοποίηση Βάσης Δεδομένων

Μετά την εγκατάσταση του προγράμματος διαχείρισης σχεσιακών βάσεων δεδομένων SQLite, το γραφικό περιβάλλον της εφαρμογής είναι όπως φαίνεται στην εικόνα 1. Δημιουργήθηκε ένα νέο αρχείο βάσεως δεδομένων και στη συνέχεια περιγράφηκαν οι πίνακες στους οποίους κατέληξε ο σχεδιασμός της βάσης. Λόγω των μεγάλων ονομάτων των σχέσεων των γνωρισμάτων, έγιναν συντμήσεις αυτών όπως φαίνονται στον πίνακα 3. Χρησιμοποιήθηκε μία αρχική σύντμηση πριν κάθε όνομα που ήταν βασισμένη στο Tai Yi User Table (tyut\_), διότι αρχικά η βάση δημιουργήθηκε στη MySQL και βοήθησε στη διαφοροποίηση από του πίνακες που προϋπάρχουν σε αυτές τις βάσεις δεδομένων και είναι πίνακες του συστήματος. Με έντονη γραφή εμφανίζονται τα ονόματα των σχέσεων.

Πίνακας 2: Σύντμηση ονοματολογίας σχέσεων και γνωρισμάτων

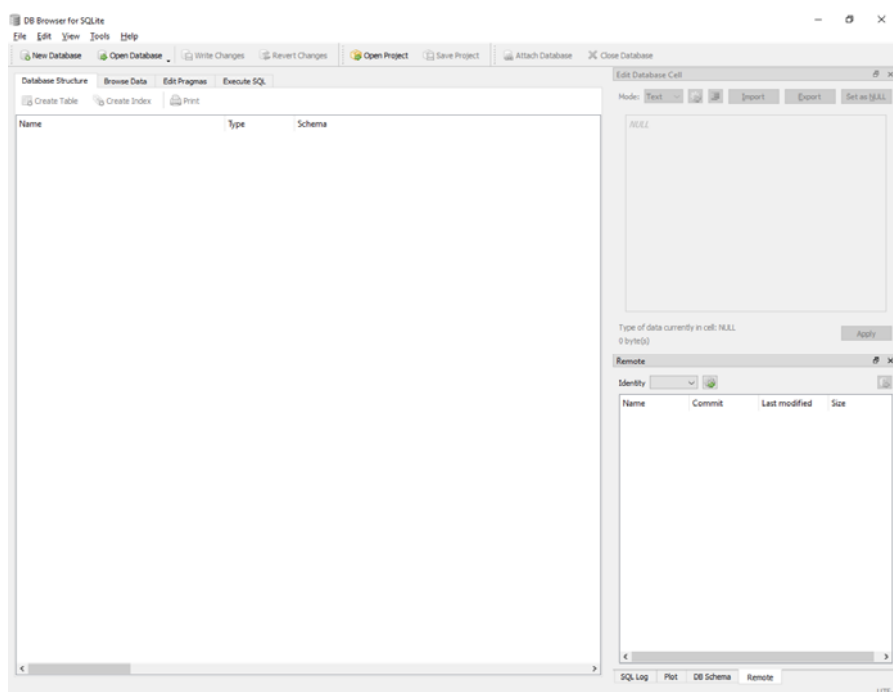
Όνομα Σχέσης ή Γνωρίσματος	Σύντμηση ονόματος
Body Area	tyut_bd_area

Όνομα Σχέσης ή Γνωρίσματος	Σύντμηση ονόματος
Body Area ID	tyut_bd_area_id
Body Area Name	tyut_bd_area_name
<b>Disease Aspect</b>	<b>tyut_dis_asp</b>
Disease Aspect ID	tyut_dis_asp_id
Disease Aspect Name	tyut_dis_asp_name
<b>Energy Type</b>	<b>tyut_enrg_type</b>
Energy Type ID	tyut_enrg_type_id
Energy Type Name	tyut_enrg_type_name
Energy Type Description	tyut_enrg_type_descr
<b>Issue</b>	<b>tyut_iss</b>
Issue ID	tyut_iss_id
Issue Name	tyut_iss_name
Issue Description	tyut_iss_descr
Issue Category ID	tyut_iss_cat_id
Issue Sub Category ID	tyut_iss_subcat_id
<b>Issue Category</b>	<b>tyut_iss_cat</b>
Issue Category ID	tyut_iss_cat_id
Issue Category Name	tyut_iss_cat_name
<b>Issue Sub Category</b>	<b>tyut_iss_subcat</b>
Issue Sub Category ID	tyut_iss_subcat_id
Issue Sub Category Name	tyut_iss_subcat_name
<b>Treatment Link Type</b>	<b>tyut_lnk_type</b>
Link Type ID	tyut_lnk_type_id
Link Type Name	tyut_lnk_type_name
<b>Level of Treatment</b>	<b>tyut_lvl_of_treat</b>
Level ID	tyut_lvl_id
Level Name	tyut_lvl_name
Level Degree	tyut_lvl_degr
<b>Special Category</b>	<b>tyut_sp_cat</b>
Special Category ID	tyut_sp_cat_id
Special Category Name	tyut_sp_cat_name
Special Category Name in NFD	tyut_sp_cat_name_NFD
Special Category Description	tyut_sp_cat_descr
<b>Treatment</b>	<b>tyut_tr</b>
Treatment ID	tyut_tr_id
Treatment Name	tyut_tr_name
Date Taught	tyut_tr_date_taught
Treatment Description	tyut_tr_descr
Treatment Short Description Greek	tyut_tr_shrt_descr_el



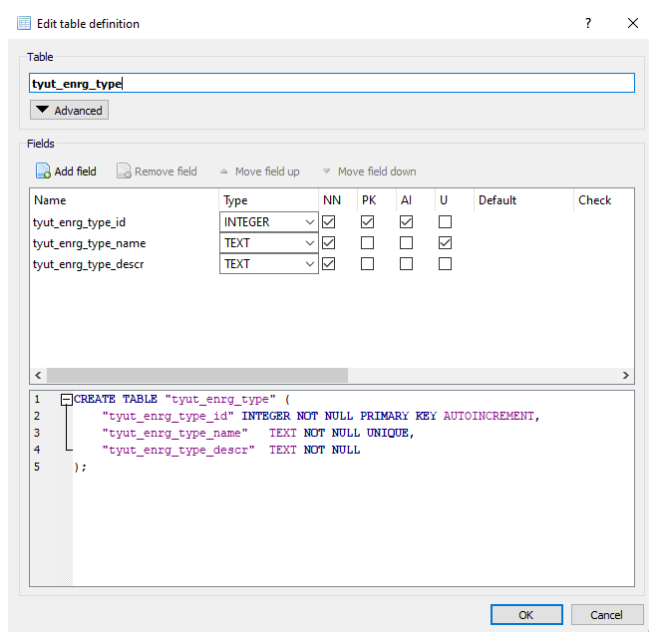
Όνομα Σχέσης ή Γνωρίσματος	Σύντμηση ονόματος
Treatment Short Description Greek in NFD	tyut_tr_shrt_descr_el_NFD
Treatment Short Description English	tyut_tr_shrt_descr_en
Body Area ID	tyut_bd_area_id
Start with	tyut_strt_with
Treatment Limitations	tyut_tr_limits
Treatment Specific Timing	tyut_tr_sp_time
Treatment Series Type	tyut_tr_ser_type
Gauging Resistance	tyut_gaug_res
Possible Negative Effects	tyut_neg_eff
Treatment Integration Period	tyut_integr_per
Recommended Actions for Recipient	tyut_recom_act
Treatment Timing Post	tyut_tr_tp
Treatment Requirements	tyut_tr_req
Minimum Energy Requirements	tyut_min_enrg_req
Treatment Duration	tyut_tr_duration
Treatment Special Requirements	tyut_tr_sp_req
<b>Treatment Addresses</b>	<b>tyut_tr_addr</b>
Treatment Addresses ID	tyut_tr_addr_id
Treatment ID	tyut_tr_id
Issue ID	tyut_iss_id
Level ID	tyut_lvl_id
Disease Aspect ID	tyut_dis_asp_id
<b>Treatment Needs Energy Type</b>	<b>tyut_tr_enrg_type</b>
Treatment ID	tyut_tr_id
Energy Type ID	tyut_enrg_type_id
<b>Treatments Link</b>	<b>tyut_tr_lnk</b>
Treatment Link ID	tyut_tr_lnk_id
Previous Treatment ID	tyut_tr_pr_id
Next Treatment ID	tyut_tr_next_id
Link Type ID	tyut_lnk_type_id
Link Description	tyut_lnk_descr
<b>Treatment Parts</b>	<b>tyut_tr_parts</b>
Treatment ID	tyut_tr_id
Parts Number	tyut_parts_num
Interval Metric	tyut_interval
Minimum Interval between Parts	tyut_min_inter
Maximum Interval between Parts	tyut_max_inter
Best Interval	tyut_best_inter
<b>Treatment Belongs to Special Category</b>	<b>tyut_tr_sp_cat</b>

Όνομα Σχέσης ή Γνωρίσματος	Σύντμηση ονόματος
Treatment ID	tyut_tr_id
Special Category ID	tyut_sp_cat_id
<b>Treatment Belongs to Type</b>	<b>tyut_tr_tr_type</b>
Treatment ID	tyut_tr_id
Treatment Type ID	tyut_tr_type_id
<b>Type of Treatment</b>	<b>tyut_tr_type</b>
Treatment Type ID	tyut_tr_type_id
Treatment Type Name	tyut_tr_type_name
Treatment Type Description	tyut_tr_type_descr



Εικόνα 1: Γραφικό περιβάλλον εφαρμογής DB Browser for SQLite

Η δημιουργία των πινάκων είχε γίνει ήδη σε MySQL και είχαν μπει κάποια αρχικά δεδομένα στους πίνακες από την αρχική προσπάθεια δημιουργίας της εφαρμογής διαχείρισης των θεραπειών. Η εφαρμογή DB Browser for SQLite υποστηρίζει την εκτέλεση κώδικα SQL οποιαδήποτε στιγμή, οπότε έγινε εξαγωγή του κώδικα δημιουργίας της βάσης δεδομένων από την MySQL και εκτελέστηκε στη νέα εφαρμογή διαχείρισης. Ο κώδικας αυτός παραδόθηκε στην επιβλέπουσα σε αρχείο «.txt» μαζί με τον κώδικα για τον εμπλουτισμό των πινάκων με τα αρχικά δεδομένα.



Εικόνα 2: Φόρμα δημιουργία και μετατροπής πινάκων στην εφαρμογή DB Browser for SQLite

Οι τροποποιήσεις που χρειάστηκαν μετά τον έλεγχο του λογισμικού πραγματοποιήθηκαν στο ίδιο πρόγραμμα διαχείρισης. Στην εικόνα 2 παρουσιάζεται η εικόνα στην οποία δημιουργούνται και τροποποιούνται οι πίνακες. Η τροποποίηση των δεδομένων των πινάκων γίνεται στο περιβάλλον που εμφανίζεται στην εικόνα 3. Ο χρήστης μπορεί να επιλέξει να κάνει εισαγωγή εγγραφής, το σύστημα προσθέτει μία γραμμή βάζοντας τα προεπιλεγμένα δεδομένα στα πεδία ανάλογα με τον ορισμό και ο χρήστης μπορεί να την επεξεργαστεί.

tyut_tr_id	tyut_parts_num	tyut_interval	tyut_min_inter	tyut_max_inter	tyut_best_inter
53	2	Hours	1	24	2
58	3	Days	2	30	2
65	3	Week	1	4	1
67	2	Days	1	3	
78	3	Days	1	14	
79	2	Hours	2	48	
85	2	Hours	3	72	
86	3	Days	1	30	
87	2	Hours	24	48	
88	2	Days	2	7	
89	2	Hours	2	72	
91	2	Days	1	14	1
92	3	Hours	2	48	2
108	5	Months	1	3	1
115	5	Week	1	2	
126	2	Hours	3	8	
128	3	Hours	24	48	
133	3	Days	3	7	
152	2	Days	2	7	2
165	4	Days	1	7	
173	2	Hours	24	36	
185	4	Days	4	90	
193	2	Days	1	7	
211	4	Days	7	30	
212	5	Days	3	14	7
217	2	Days	1	7	
218		Day			NULL

Εικόνα 3: Φόρμα εισαγωγής δεδομένων στην εφαρμογή DB Browser for SQLite

Η σχεσιακή βάση δεδομένων πλέον ήταν έτοιμη για να χρησιμοποιηθεί και αποτελούνταν από ένα αρχείο της μορφής «.sqlite».

## 4. ΑΡΧΙΚΗ ΑΠΟΤΥΠΩΣΗ ΔΙΕΠΑΦΗΣ

### 4.1 Επιλογή Εφαρμογής Αρχικής Αποτύπωσης Διεπαφής

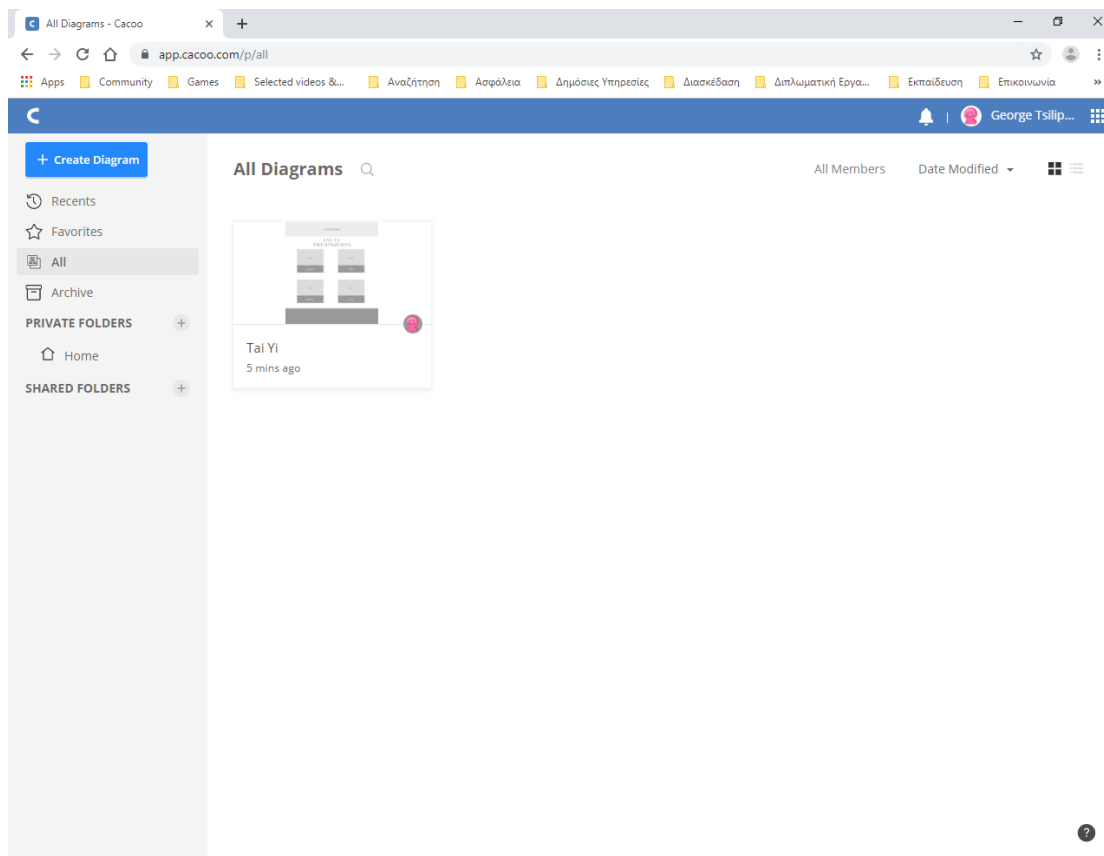
Για τη δημιουργία της αρχικής αποτύπωσης της διεπαφής πραγματοποιήθηκε έρευνα για οδηγούς, πληροφορίες και τελικά εφαρμογές που μπορούν να χρησιμοποιηθούν για το σκοπό αυτό. Ένας συνοπτικός και κατατοπιστικός οδηγός [23] παρείχε μία συνοπτική λίστα με εργαλεία ανάπτυξης αρχικών αποτυπώσεων διεπαφής. Από τη πληθώρα των εργαλείων εξετάστηκαν τα περισσότερα από αυτά, αλλά δυστυχώς τα περισσότερα αποτελούσαν ιδιοκτησία και παρέχονταν μετά από αγορά. Η δημιουργία αρχικών αποτυπώσεων και προτύπων διεπαφών είναι μία δυνατή τακτική στην ανάπτυξη ιστοσελίδων πλέον και η χρήση από επαγγελματίες του χώρου είναι πλέον ένα σημαντικό εργαλείο. Για το λόγο αυτό οι εφαρμογές δεν είναι δωρεάν. Ο στόχος, όπως και για το εργαλείο διαχείρισης της βάσης δεδομένων, ήταν να επιλεγεί ένα πρόγραμμα διαχείρισης με απλότητα στη χρήση, χωρίς ιδιαίτερες απαιτήσεις εγκατάστασης και αν ήταν εφικτό χρησιμοποιήσιμο δωρεάν.

Εξετάσαμε πιθανή χρήση των εφαρμογών Balsamiq [24], Axure [25], Flairbuilder [26], Adobe Fireworks [27] και Adobe Indesign [28] αλλά όλες αυτές είναι επαγγελματικές εφαρμογές και δεν παρέχονται δωρεάν. Οι εφαρμογές της Adobe μάλιστα είναι μεγάλα πακέτα λογισμικού που προσφέρουν πάρα πολλές δυνατότητες σχεδίασης, μη απαραίτητες για τη δημιουργία απλών αρχικών αποτυπώσεων διεπαφών, αλλά περισσότερο προσανατολισμένες για την υλοποίηση του δημιουργικού μίας εφαρμογής ή ιστοσελίδας. Οι τρεις πρώτες εφαρμογές, αν και είναι προσανατολισμένες στη δημιουργία wireframes, είναι και αυτές εμπορικές και αποκλείστηκαν ως πιθανά εργαλεία.

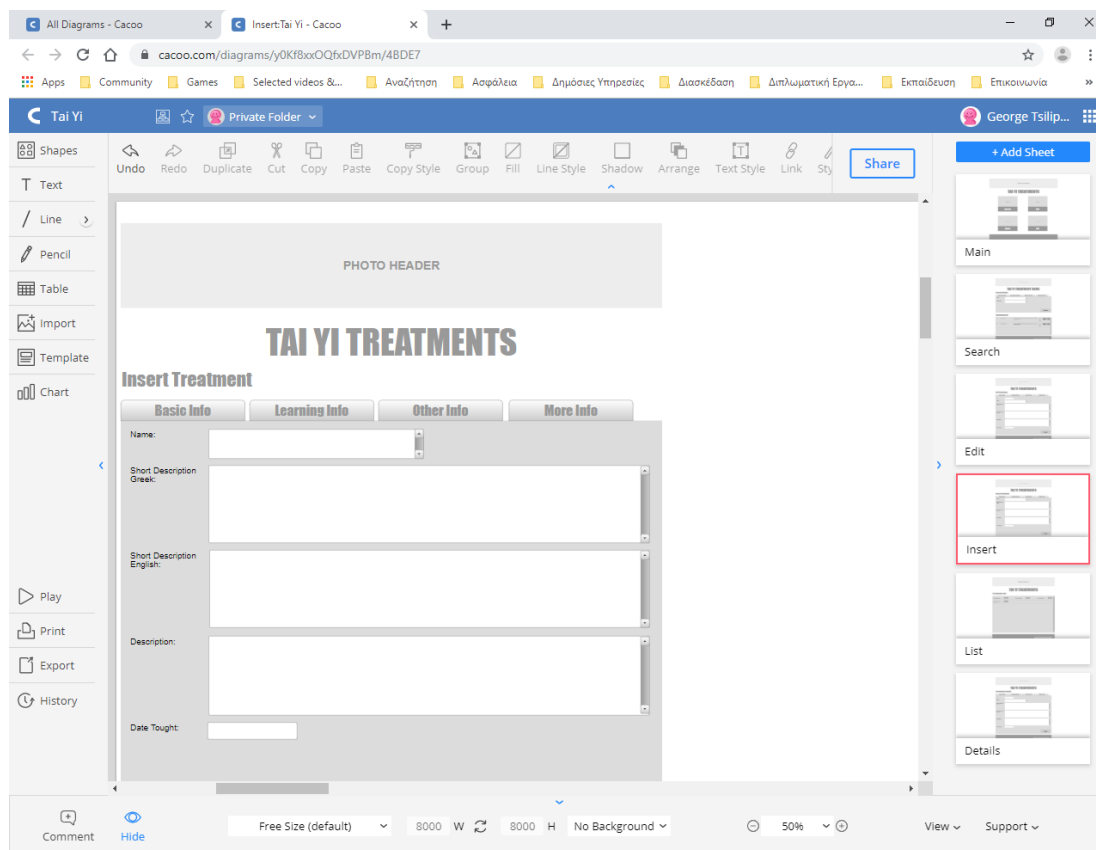
Στη συνέχεια εντοπίστηκε το Keynotopia [29] ένα πακέτο που προσφέρει μία πολύ μεγάλη συλλογή από έτοιμα στοιχεία για δημιουργία αρχικών αποτυπώσεων διεπαφών με χρήση άλλων εργαλείων και συγκεκριμένα με το Microsoft PowerPoint και το Apple Keynote. Κάτι τέτοιο ήταν εκτός των στόχων της έρευνας. Καταβλήθηκε προσπάθεια να εντοπιστούν δωρεάν εργαλεία και εξετάστηκε το Pencil Project [30], το οποίο είναι ανοικτού κώδικα, αλλά φάνηκε να έχει εγκαταλειφτεί και να μην ανανεώνεται, οπότε και απορρίφθηκε και αυτό. Εξετάστηκαν επίσης το MockFlow [31], το Mockingbird [32] και το Cacao [33]. Και τα τρία εργαλεία αυτά παρέχονται κατόπιν αγοράς αλλά προσφέρουν δωρεάν δοκιμαστικά πακέτα για την αξιολόγησή τους και την εξέταση της καταλληλότητάς τους από πιθανούς αγοραστές. Δοκιμάστηκαν και τα τρία και επιλέχθηκε το Cacao, διότι παρείχε με τη δωρεάν έκδοση περισσότερη λειτουργικότητα.

### 4.2 Σχεδιασμός Αρχικής Αποτύπωσης Διεπαφής

Η εφαρμογή που επιλέχθηκε (Cacao) λειτουργεί σε οποιοδήποτε πρόγραμμα πλοήγησης στο διαδίκτυο. Δημιουργήθηκε ένας λογαριασμός και ένα διάγραμμα, στο οποίο αποτυπώθηκε η αρχική αποτύπωση της διεπαφής. Στην εικόνα 4 εμφανίζεται το αρχικό περιβάλλον της εφαρμογής στο οποίο δημιουργούνται τα διαγράμματα. Λόγω περιορισμών στη δωρεάν χρήση δεν παρέχεται δυνατότητα δημιουργίας περισσότερων του ενός διαγραμμάτων για κάθε χρήστη. Στην εικόνα 5 εμφανίζεται το περιβάλλον στο οποίο τροποποιείται το διάγραμμα και στο οποίο υπάρχει δυνατότητα δημιουργίας 6 φύλλων (λόγω της δωρεάν χρήσης και πάλι, διαφορετικά μπορεί ο χρήστης να δημιουργήσει περισσότερα). Για τις ανάγκες μας αυτό ήταν αρκετό.



Εικόνα 4: Αρχικό περιβάλλον της εφαρμογής Cacoo

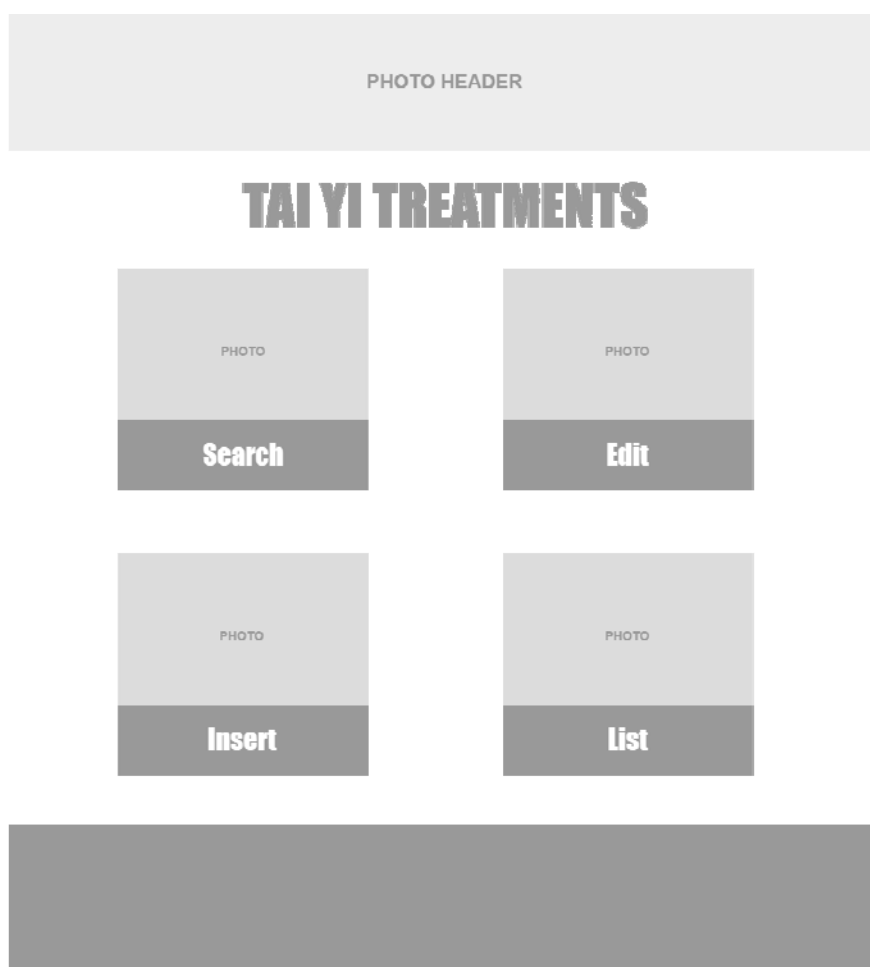


Εικόνα 5: Περιβάλλον επεξεργασίας της εφαρμογής Cacoo

Η αρχική αποτύπωση της διεπαφής σχεδιάστηκε για να καλύψει τις λειτουργικές απαιτήσεις και να δώσει μία αρχική εικόνα της εφαρμογής στους χρήστες. Κάθε εικόνα του wireframe συνοδευόταν με μία συνοπτική περιγραφή για τον τρόπο λειτουργίας. Στη συνέχεια παρουσιάζουμε τις εικόνες που αναπτύχθηκαν μαζί με την περιγραφή λειτουργίας.

#### 4.3 Παρουσίαση Αρχικής Αποτύπωσης Διεπαφής

Η αρχική εικόνα της εφαρμογής δίνεται στην εικόνα 6 και εμφανίζεται μετά την έναρξη της εφαρμογής. Η γραμματοσειρές που έχουν χρησιμοποιηθεί είναι ενδεικτικές και το footer, μπορεί να περιέχει πληροφορίες ή απλά κάποιο χρώμα.



Εικόνα 6: Αρχική σελίδα εφαρμογής θεραπειών

Η σελίδα για την αναζήτηση δίνεται στην εικόνα 7, όπου ενδεικτικά παρουσιάζονται δύο κριτήρια αναζήτησης, αλλά θα εμπλουτιστούν. Επιπλέον κριτήρια θα εμφανίζονται σε διαφορετικά tabs σύμφωνα με την κατηγοριοποίηση των στοιχείων που θα γίνει. Τα αποτελέσματα της αναζήτησης θα εμφανίζονται από κάτω ως λίστα, και για κάθε θεραπεία ο χρήστης θα μπορεί να επιλέξει να δει λεπτομέρειες, ή να επεξεργαστεί τα στοιχεία της.

PHOTO HEADER

## TAI YI TREATMENTS

### Search Criteria

**Basic Criteria** | Learning Criteria | Other Criteria | More Criteria

Name:

Search text:

**Search**

### Treatments List

ID	Brain Passion	A short description of what this treatment is about, and some important notices.	<b>Details</b> <b>Edit</b>
ID	Brain Passion 2	A short description of what this treatment is about, and some important notices.	<b>Details</b> <b>Edit</b>

**Back to main**

Εικόνα 7: Σελίδα αναζήτησης εφαρμογής θεραπειών

Η σελίδα στην οποία ο χρήστης θα τροποποιεί τα στοιχεία μίας θεραπείας δίνεται στην εικόνα 8. Θα εμφανίζονται οι διάφορες πληροφορίες σε διαφορετικά tabs σύμφωνα με την ομαδοποίηση των γνωρισμάτων. Ενδεικτικά παρουσιάζονται κάποια βασικά γνωρίσματα στην εικόνα αυτή.

Όπως και στην προηγούμενη σελίδα, εδώ ο χρήστης θα μπορεί να εισάγει τα στοιχεία μίας θεραπείας. Η εικόνα 9 παρουσιάζει τη σελίδα αυτή του wireframe. Θα εμφανίζονται οι διάφορες πληροφορίες σε διαφορετικά tabs σύμφωνα με την ομαδοποίηση των γνωρισμάτων και παρουσιάζονται κάποια ενδεικτικά βασικά γνωρίσματα.

Για τη σελίδα στην οποία ο χρήστης θα μπορεί να δει το σύνολο των θεραπειών που έχουν καταχωρηθεί και να μπορεί να επιλέξει να δει λεπτομέρειες, ή να επεξεργαστεί τα στοιχεία της, δημιουργήθηκαν δύο εκδοχές, μία όπως εμφανίζεται στην εικόνα 10 και μία όπως εμφανίζεται στην εικόνα 11. Η τελευταία είναι μία εκδοχή ώστε να φαίνονται πολλές περισσότερες μαζί και να είναι εποπτική η εικόνα. Τελικά ωστόσο οι χρήστες επέλεξαν την πρώτη.



The screenshot shows a web application titled "TAI YI TREATMENTS". At the top is a "PHOTO HEADER" placeholder. Below the title is the "Edit Treatment" section. It features four tabs: "Basic Info", "Learning Info", "Other Info", and "More Info". The "Basic Info" tab is active, displaying a form with the following fields: "ID:" (a small text input), "Name:" (a text input with a dropdown arrow), "Short Description Greek:" (a large text area with a scrollbar), "Short Description English:" (a large text area with a scrollbar), "Description:" (a large text area with a scrollbar), and "Date Taught:" (a date input). An "Update" button is located at the bottom right of the form. Below the form is a dark gray footer bar containing a "Back to main" button.

Εικόνα 8: Σελίδα ενημέρωσης θεραπείας εφαρμογής θεραπειών

The screenshot shows a web application titled "TAI YI TREATMENTS". At the top is a "PHOTO HEADER" placeholder. Below the title is the "Insert Treatment" section. It features four tabs: "Basic Info", "Learning Info", "Other Info", and "More Info". The "Basic Info" tab is active, displaying a form with the following fields: "Name:" (a text input with a dropdown arrow), "Short Description Greek:" (a large text area with a scrollbar), "Short Description English:" (a large text area with a scrollbar), "Description:" (a large text area with a scrollbar), and "Date Taught:" (a date input). An "Insert" button is located at the bottom right of the form. Below the form is a dark gray footer bar containing a "Back to main" button.

Εικόνα 9: Σελίδα εισαγωγής θεραπείας εφαρμογής θεραπειών



Εικόνα 10: Σελίδα λίστας εφαρμογής θεραπειών (πρώτη εκδοχή)



Εικόνα 11: Σελίδα λίστας εφαρμογής θεραπειών (δεύτερη εκδοχή)

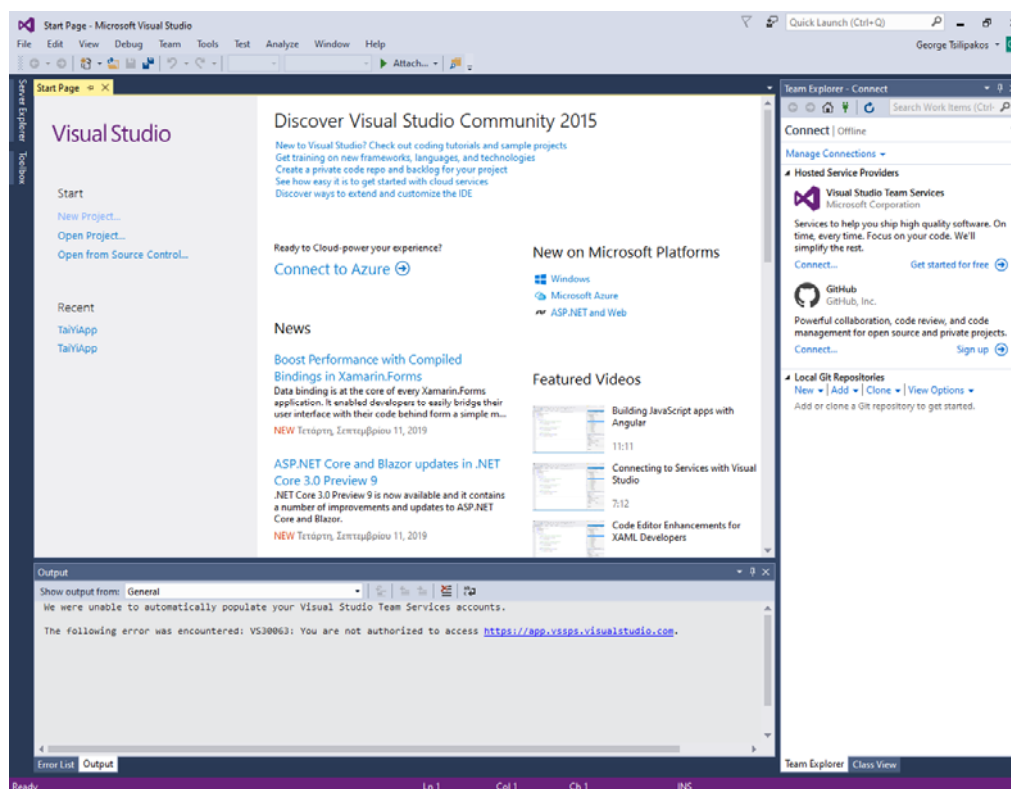
Πάνω στην αρχική αποτύπωση της διεπαφής η οποία συμφωνήθηκε με τους χρήστες, αναπτύχθηκε η διεπαφή, σε συνδυασμό πάντα με τις προσφερόμενες δυνατότητες των εργαλείων που επιλέχθηκαν για την ανάπτυξη.

## 5. ΥΛΟΠΟΙΗΣΗ ΕΦΑΡΜΟΓΗΣ

### 5.1 Εγκατάσταση Εργαλείων και Πακέτων

Για την υλοποίηση της διεπαφής και της λειτουργικότητας της εφαρμογής αποφασίστηκε να χρησιμοποιηθεί το Microsoft Visual Studio (MSV). Εγκαταστάθηκε λοιπόν το Microsoft Visual Studio Community 2015. Το αρχικό περιβάλλον του IDE που εγκαταστάθηκε δίνεται στην εικόνα 12. Σημειώνεται στο σημείο αυτό ότι η εφαρμογή ονομάστηκε Tai Yi, και με το όνομα αυτό θα γίνεται αναφορά σε αυτή στη συνέχεια.

Στη συνέχεια ακολουθώντας οδηγίες [34] εγκαταστάθηκαν τα απαραίτητα εργαλεία για τη δημιουργία έργων (project) Apache Cordova. Πέρα από τα εργαλεία εγκαταστάθηκαν και επιπλέον πακέτα για την υποστήριξη των λειτουργιών της εφαρμογής. Τα πακέτα αυτά ήταν το jQuery και το jQuery mobile και εμφανίζονται στην εικόνα 13. Επίσης, ενεργοποιήθηκαν και κάποιες επεκτάσεις που ήταν τα Compact, Notification, File, File Transfer, InAppBrowser και Whitelist όπως εμφανίζεται στην εικόνα 14. Κάποιες επεκτάσεις απαιτούσαν προϋπόθεση για τη λειτουργία του Apache Cordova στο MSV, και κάποιες άλλες εγκαταστάθηκαν για να καλύψουν λειτουργικότητες που απαιτούσε η εφαρμογή Tai Yi.

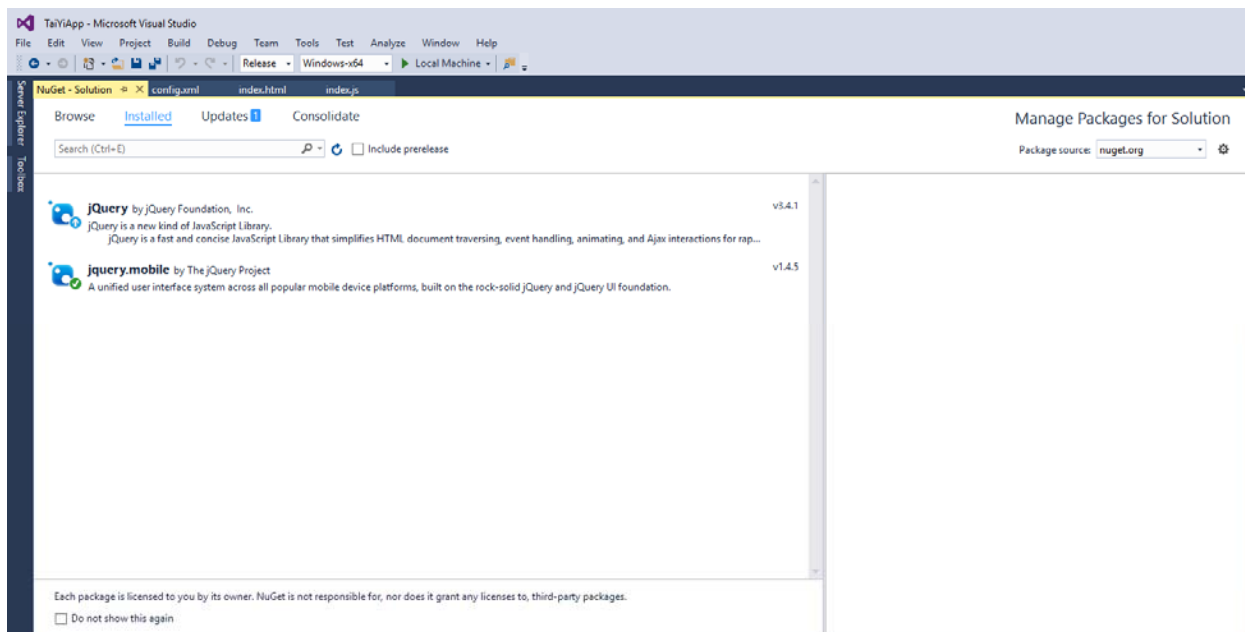


Εικόνα 12: Αρχικό περιβάλλον στο MSV Community 2015

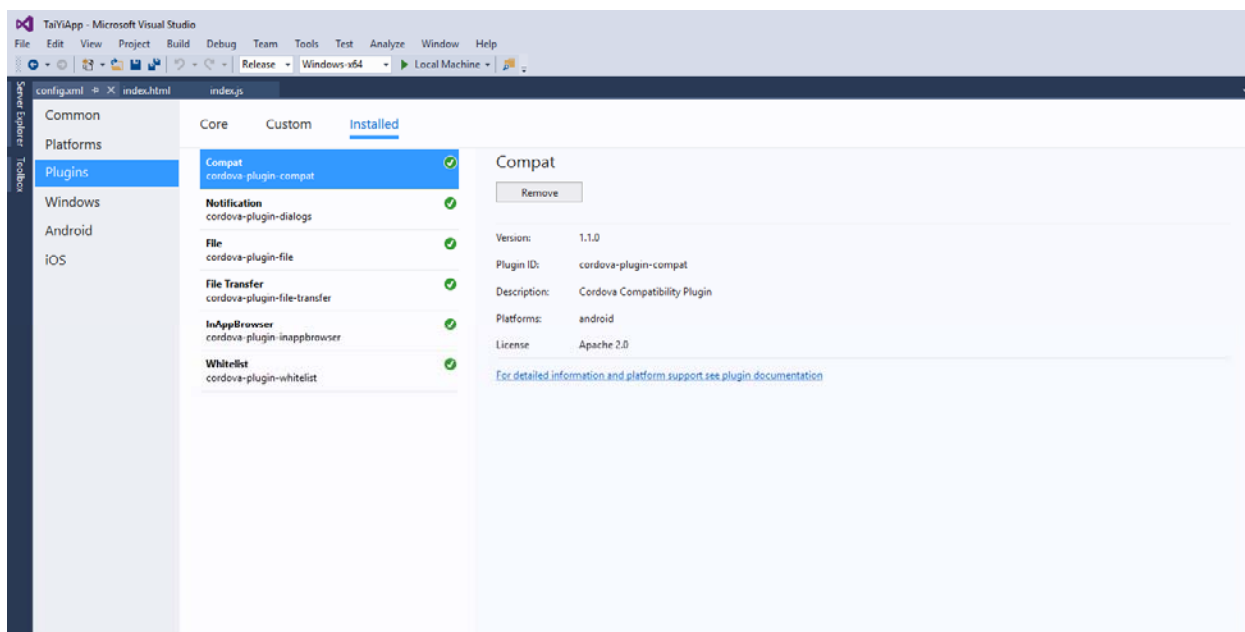
Σε συνέχεια και με τη βοήθεια του οδηγού που αναφέραμε αρχικά [12] προχωρήσαμε στην τροποποίηση του κώδικα της προτεινόμενης απλής εφαρμογής και την προσαρμογή του στις ανάγκες του υπό ανάπτυξη λογισμικού. Ο κύριος κώδικας όσον αφορά στη διεπαφή αναπτύχθηκε σε ένα αρχείο «.html» και είναι ουσιαστικά σε HTML και ο κύριος κώδικας όσον αφορά τη λειτουργικότητα, τη διαχείριση των δεδομένων και την ενημέρωση της διεπαφής με τα απαιτούμενα κάθε φορά δεδομένα αναπτύχθηκε σε ένα αρχείο «.js» και είναι σε JavaScript. Το σύνολο του κώδικα που αναπτύχθηκε και βρίσκεται στα δύο παραπάνω αρχεία παραδόθηκε στην επιβλέπουσα με το κείμενο της

Σχεδιασμός και Ανάπτυξη Εφαρμογής με Δυνατότητες Ασύγχρονης Εκτέλεσης και Χρήση Σχεσιακής Βάσης Δεδομένων Ανεξάρτητης Εξυπηρητητών

εργασίας. Το δέντρο με τα αρχεία του project που δημιουργήθηκε δίνεται στην εικόνα 14.

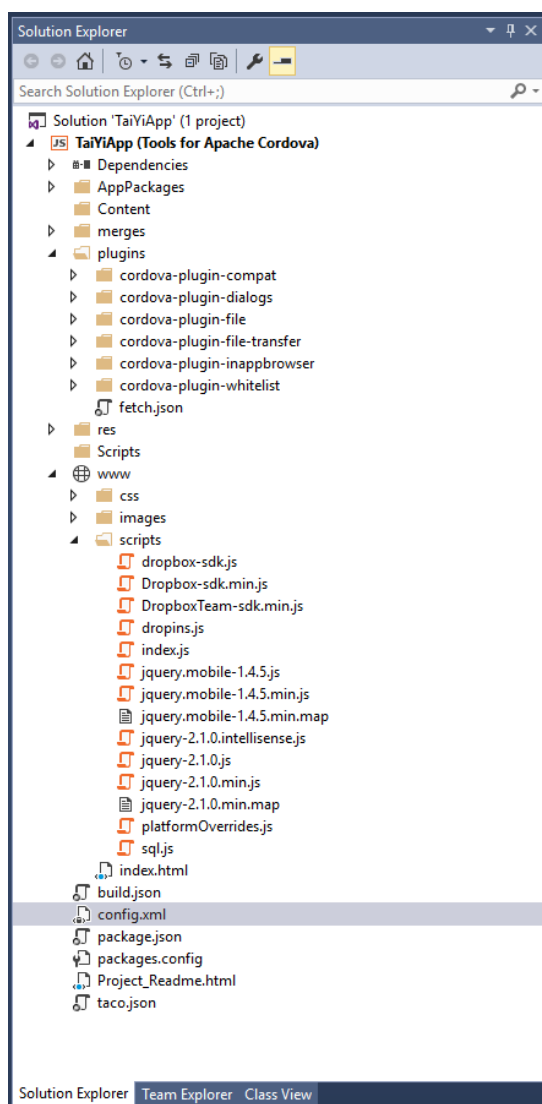


Εικόνα 13: Διαχείριση πακέτων στο MVS Community 2015



Εικόνα 14: Διαχείριση επεκτάσεων στο MVS Community 2015

Παρακάτω παρουσιάζεται ο τρόπος που αναπτύχθηκε ο κώδικας της εφαρμογής παρουσιάζοντας τα αντικείμενα, τις λίστες, τον τρόπο αλληλεπίδρασης με τη βάση δεδομένων, τη διεπαφή, τον τρόπο ανάπτυξης της λειτουργικότητας, τον τρόπο ασύγχρονης λειτουργίας καθώς και την προσπάθεια εκτέλεσης σε πολλαπλές πλατφόρμες.

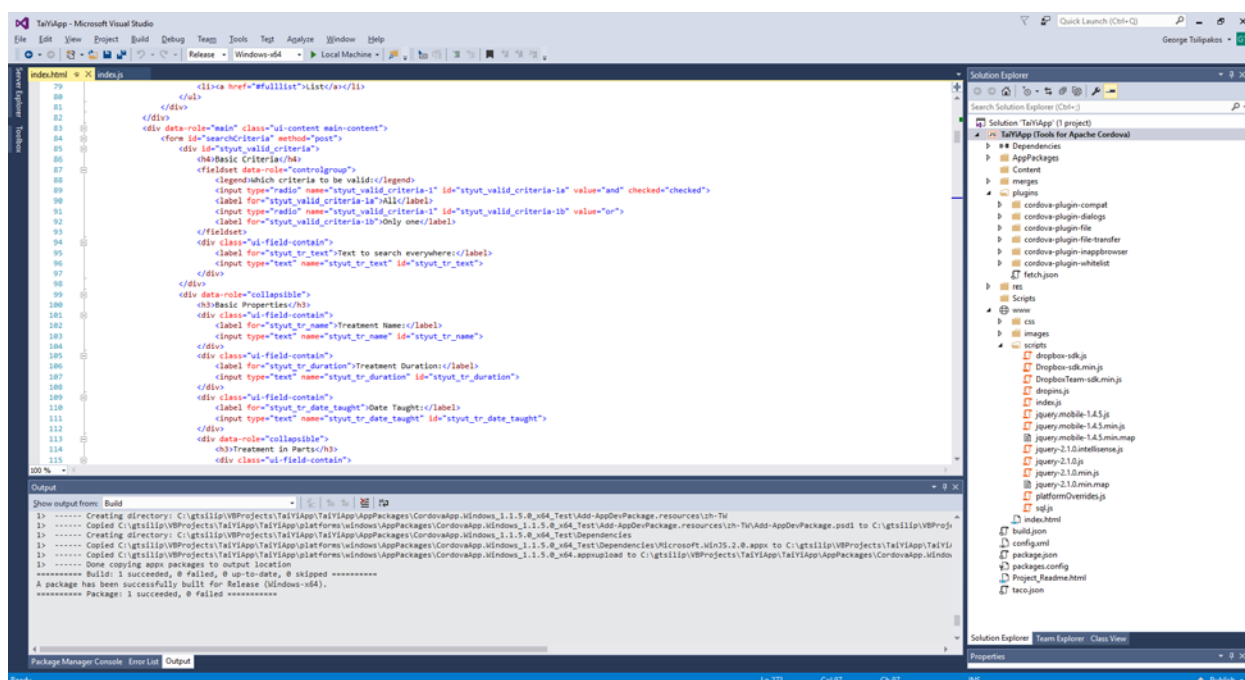


Εικόνα 15: Δέντρο αρχείων του project για την εφαρμογή Tai Yi

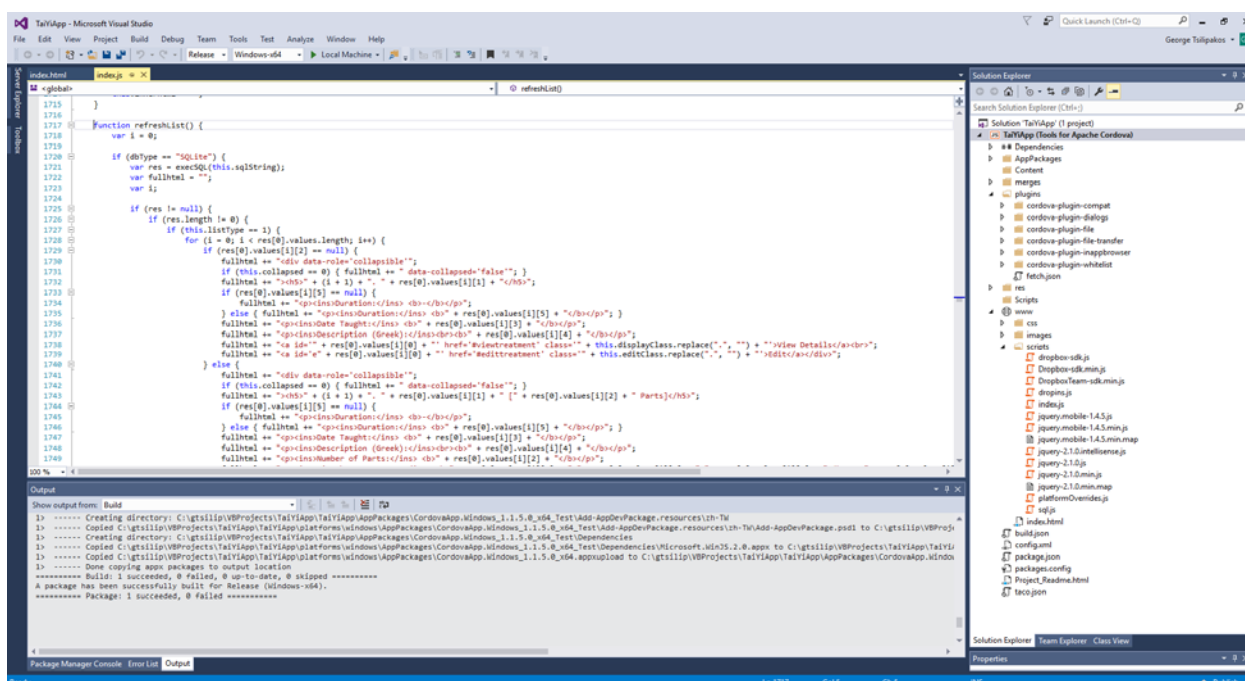
Στις εικόνες 16 και 17 παρουσιάζεται το περιβάλλον στο οποίο πραγματοποιήθηκε η συγγραφή του κώδικα, στο οποίο παρέχεται συντακτική βοήθεια για την κάθε γλώσσα που χρησιμοποιείται. Ο κώδικας σε HTML εμφανίζεται στην εικόνα 16 και ο κώδικας σε JavaScript εμφανίζεται στην εικόνα 17.

## 5.2 Ανάπτυξη Αντικειμένων

Για τη διαχείριση των δεδομένων και την επαναχρησιμοποίηση τους κατά τη λειτουργία της εφαρμογής Tai Yi, αναπτύχθηκαν κάποια αντικείμενα. Το κεντρικό βεβαίως αντικείμενο είναι για τη θεραπεία και ονομάστηκε «selectedTreatment». Τα υπόλοιπα αντικείμενα αφορούν είτε τις λίστες (λίστες θεραπειών για παράδειγμα) που θα παρουσιαστούν παρακάτω είτε τα δεδομένα που συνδέονται με την εκάστοτε θεραπεία αλλά παίρνουν πολλαπλές τιμές διότι βρίσκονται σε άλλους πίνακες της σχεσιακής βάσης δεδομένων.



Εικόνα 16: Περιβάλλον ανάπτυξης κώδικα HTML στο MVS



Εικόνα 17: Περιβάλλον ανάπτυξης κώδικα JavaScript στο MVS

Δημιουργήθηκαν επίσης νέοι μέθοδοι για τα αντικείμενα αυτά, με τις οποίες ουσιαστικά υλοποιούνταν τα στιγμιότυπα των αντικειμένων. Για το κεντρικό αντικείμενο αναπτύχθηκε η μέθοδος «loadTreat», η οποία λαμβάνει ως όρισμα ένα νούμερο που αναμένεται να είναι ο κωδικός της θεραπείας και ουσιαστικά φορτώνει τα δεδομένα της θεραπείας με τον κωδικό αυτό στο αντικείμενο. Για τα υπόλοιπα αντικείμενα αναπτύχθηκαν δύο μέθοδοι για το κάθε ένα, η μία ήταν η «initListorTable» για την αρχικοποίηση του κάθε αντικειμένου και η άλλη ήταν είτε η «refreshTable» είτε η «refreshList» για την ανανέωση των δεδομένων ενός αντικειμένου, ανάλογα με το τι είναι το κάθε αντικείμενο. Υπάρχουν δηλαδή αντικείμενα που θεωρούνται λίστες και

αντικείμενα που θεωρούνται πίνακες. Τα αντικείμενα που θεωρούνται πίνακες είναι ουσιαστικά για την διαχείριση των περιφερειακών δεδομένων της βάσης δεδομένων. Οι μέθοδοι αυτές είναι κοινές για όλα τα αντικείμενα, αλλά η λειτουργία τους διαφοροποιείται ανάλογα με τις τιμές που έχουν συγκεκριμένες ιδιότητες των αντικειμένων, οι οποίες καθορίστηκαν στην ανάπτυξη.

Εδώ είναι σημαντικό να σημειωθεί ότι για τα περιφερειακά δεδομένα δημιουργήθηκαν παρόμοια αντικείμενα με αντιγραφή, διότι κάθε σελίδα της εφαρμογής, ανάλογα με την λειτουργικότητα που επιτελεί, απαιτούσε στοιχεία με διαφορετική ονοματολογία.

### 5.3 Ανάπτυξη Λιστών

Για την αναπαράσταση των λιστών αναπτύχθηκε η συνάρτηση «refreshList» η οποία ουσιαστικά δημιουργεί τον κώδικα σε HTML μαζί με τις πληροφορίες από τη βάση δεδομένων και ενημερώνεται στη συνέχεια η διεπαφή. Η δυσκολία στη σημείο αυτό ήταν να μπορεί κατά τη διάρκεια εκτέλεσης της εφαρμογής, όταν γίνονται αλλαγές στις πληροφορίες που τηρούνται στη βάση δεδομένων, να ενημερώνονται και οι τυχόν λίστες που είναι διαθέσιμες. Αυτό οδηγεί στον να εκτελούνται κομμάτια του κώδικα επανειλημμένως όταν γίνεται επεξεργασία των πληροφοριών.

Οι λίστες δεδομένων εμφανίζονται σε πολλά σημεία της εφαρμογής, αφού υπάρχουν πολλές πληροφορίες για την κάθε θεραπεία που διατηρούνται σε περιφερειακούς πίνακες και χρήζουν διαχείρισης. Καταβλήθηκε προσπάθεια για την παραμετροποίηση των απαιτούμενων δεδομένων σε ιδιότητες για τη κάθε λίστα έτσι ώστε ο κώδικας να μπορεί να είναι όσο το δυνατόν πιο συνοπτικός.

Ένας τύπος λιστών είναι οι λίστες με θεραπείες. Για τις λίστες αυτές καθορίστηκαν κάποια βασικά γνωρίσματα, όπως αναφέρθηκε και στις απαιτήσεις, τα οποία ζητήθηκε να είναι διαθέσιμα για κάθε θεραπεία.

### 5.4 Ανάπτυξη Ανάκλησης και Αποθήκευσης Δεδομένων

Βάσει του οδηγού [13] για τη χρήση του «sql.js», δημιουργήθηκαν κάποιες συναρτήσεις για τον άνοιγμα και την ενημέρωση της βάσης δεδομένων. Η διαδικασία που ακολουθείται, είναι να διαβάζεται και να φορτώνεται σε ένα αντικείμενο όλη η βάση δεδομένων με την έναρξη της εφαρμογής. Μάλιστα στο σημείο αυτό προστέθηκε και παραμετροποίηση έτσι ώστε να μπορεί να αλλάξει ο σχεδιασμός της εφαρμογής σε περίπτωση που επιλεγεί να χρησιμοποιηθεί διαφορετική βάση δεδομένων, να δίνεται δηλαδή η δυνατότητα εκτέλεσης διαφορετικών εντολών ανάλογα με το τύπο της βάσης δεδομένων που χρησιμοποιείται. Αν η εφαρμογή δεν εντοπίσει βάση δεδομένων, ενημερώνει το χρήστη της εφαρμογής ότι δεν υπάρχουν διαθέσιμα δεδομένα. Παρακάτω θα παρουσιάσουμε τις δυνατότητες που αναπτύχθηκαν για την λήψη των δεδομένων.

Όταν λοιπόν είναι διαθέσιμες οι απαιτούμενες πληροφορίες, η εφαρμογή μπορεί να λειτουργήσει κανονικά. Στην πορεία εκτέλεσης της εφαρμογής, μετά από κάθε ενέργεια που διαφοροποιεί τα δεδομένα, ενημερώνεται η βάση δεδομένων στο αντικείμενο που τη φιλοξενεί και γίνεται και η ενημέρωση του τοπικού αρχείου «.sqlite». Η ενημέρωση της βάσης κατά τη διάρκεια χρήσης της εφαρμογής παρέχεται από τη δυνατότητα εκτέλεσης SQL κώδικα στη βάση με τη χρήση συνάρτησης που αναπτύχθηκε. Μέσω της συνάρτησης αυτής επιστρέφονται και τυχόν προβλήματα στην εκτέλεση του SQL κώδικα.

## 5.5 Ανάπτυξη Διεπαφής

Στην ανάπτυξη της διεπαφής, όπως προαναφέρθηκε, χρησιμοποιήθηκε η γλώσσα προγραμματισμού HTML και η βιβλιοθήκη jQuery mobile. Για τη συγκεκριμένη βιβλιοθήκη εντοπίστηκε ένας οδηγός επίδειξης τρόπου λειτουργίας [35] και εκτεταμένος βοηθός [36], από τους οποίους χρησιμοποιήθηκαν πολλά κομμάτια. Η βιβλιοθήκη αυτή προσφέρει εργαλεία για την ανάπτυξη εφαρμογών που εκτελούνται σε πολλά περιβάλλοντα και συσκευές και χρησιμοποιεί την τεχνική της σελιδοποίησης. Αναπτύσσονται δηλαδή σε ένα αρχείο «.html» διαφορετικές περιοχές που καθορίζονται ως σελίδες (pages) και ενεργοποιείται κάθε φορά κάποια συγκεκριμένη. Το ποια σελίδα είναι ενεργή κάθε φορά καθορίζεται από τον κώδικα JavaScript που εκτελείται παράλληλα με το αρχείο «.html».

Η διεπαφή αναπτύχθηκε λαμβάνοντας βέβαια υπόψη την αρχική αποτύπωσή της που παρουσιάστηκε στο τέταρτο κεφάλαιο, καθώς και τους περιορισμούς των εργαλείων που χρησιμοποιήθηκαν. Δεν υπήρχε γνώση για παράδειγμα για τον τρόπο που διαχειρίζεται τη σελιδοποίηση το jQuery, και σχεδιάστηκε η αρχική αποτύπωση στηριζόμενη σε tabs, ενώ δεν δημιουργήθηκε κάτι τέτοιο. Βέβαια η λειτουργικότητα είναι κοινή ανεξάρτητα από το στοιχείο που χρησιμοποιείται.

Για τις ανάγκες της εφαρμογής δημιουργήθηκαν οι παρακάτω σελίδες στο αρχείο, κάθε μία εκ των οποίων εξυπηρετεί διαφορετική λειτουργικότητα. Λόγω της ύπαρξης όλων των σελίδων στο ίδιο αρχείο, ήταν αναπόφευκτο να αναπτυχθούν διαφορετικά στοιχεία HTML για κάθε σελίδα και μάλιστα να ονομαστούν με ξεχωριστό και μοναδικό όνομα. Αυτό ήταν που συνετέλεσε και στη δημιουργία ξεχωριστών αντικειμένων για κάθε περίπτωση, όπως αναφέρθηκε παραπάνω.

Επιπλέον, σε κάθε σελίδα που παρουσιάζονταν τα γνωρίσματα μίας θεραπείας, είτε για αναζήτηση, είτε για εισαγωγή, είτε για επεξεργασία, είτε για προβολή, ομαδοποιούνταν χρησιμοποιώντας άλλο ένα εργαλείο του jQuery mobile, το «collapsible». Το στοιχείο αυτό δημιουργεί μία περιοχή στη διεπαφή που μπορεί είτε να φαίνεται είτε όχι, ανάλογα με μία αρχική ρύθμιση και την αλληλεπίδραση με το χρήστη. Τα γνωρίσματα της κάθε θεραπείας ομαδοποιήθηκαν σε τέτοιες περιοχές με την εξής τρόπο και ονομασία:

- Basic Properties : Treatment Name, Treatment Duration, Date Taught, Treatment Parts (Number of Parts, Min Interval, Max Interval, Best Interval), Treatment Types, Treatment Description, Treatment Short Description (Greek), Treatment Short Description (English)
- Issues Addressed : Issue, Aspect of Disease, Effectiveness
- Details : Treatment Body Area, Start with, Treatment Limitations, Treatment Specific Timing, Treatment Series Type, Gauging Resistance, Special Categories
- Treatment Relations : Relation Type, Relation Description
- Post Details : Possible Negative Effects, Treatment Integration Period, Recommended Actions for Recipient, Treatment Timing Post
- Requirements : Treatment Requirements, Energy Types, Minimum Energy Requirements

Η ομαδοποίηση αυτή των γνωρισμάτων βοηθάει τον χρήστη αν επικεντρώνεται στα σημεία που βρίσκεται και να μην χάνεται στο πλήθος των πληροφοριών που είναι διαθέσιμες για κάθε θεραπεία.



Οι σελίδες τώρα που αναπτύχθηκαν για να καλύψουν την λειτουργικότητα της εφαρμογής ήταν:

1. **mainpage** : Αυτή είναι η αρχική σελίδα της εφαρμογής και κεντρική για την προήγηση στις υπόλοιπες σελίδες. Δεν αποτελείται από πολλά στοιχεία, κυρίως από κουμπιά που σε παραπέμπουν σε άλλες σελίδες της εφαρμογής.
2. **searchtreatment** : Η σελίδα στην οποία ο χρήστης μπορεί να κάνει αναζήτηση μίας θεραπείας. Εδώ υπάρχει γενικό κριτήριο κειμένου και όλα τα στοιχεία για μία θεραπεία, έτσι ώστε ο χρήστης να μπορεί να αναζητήσει σε συγκεκριμένο γνώρισμα. Τα γνωρίσματα είναι ομαδοποιημένα και σε αυτή τη σελίδα.
3. **edittreatment** : Αυτή είναι η σελίδα για την επεξεργασία των πληροφοριών μίας θεραπείας, οι οποίες είναι ομαδοποιημένες όπως αναφέρθηκε παραπάνω.
4. **inserttreatment** : Η συγκεκριμένη σελίδα είναι για την εισαγωγή μίας θεραπείας, και όλα τα στοιχεία της σελίδας εμφανίζονται κενά, εκτός από τις περιπτώσεις που έχουμε προεπιλεγμένη τιμή γνωρίσματος. Και στην περίπτωση αυτή τα δεδομένα είναι ομαδοποιημένα.
5. **fulllist** : Αποτελεί ουσιαστικά τη σελίδα που εμφανίζονται όλες οι θεραπείες που έχουν καταχωρηθεί στη βάση δεδομένων.
6. **viewtreatment** : Η συγκεκριμένη σελίδα είναι μόνο για προβολή των πληροφοριών, οι οποίες και αυτές εμφανίζονται ομαδοποιημένες. Από τη συγκεκριμένη σελίδα μπορεί ο χρήστης να επιλέξει να διαγράψει μία θεραπεία.
7. **update\_enrg\_type\_data** : Αυτή η σελίδα και οι επόμενες, αναπτύχθηκαν μετά τον εντοπισμό της αναγκαιότητας να διαχειρίζεται η εφαρμογή τα περιφερειακά δεδομένα, έτσι ώστε οι χρήστες να μπορούν να τα τροποποιούν. Εξυπηρετεί δηλαδή την παρουσίαση των δεδομένων για το γνώρισμα «Energy Type» και επιτρέπει την εισαγωγή νέων δεδομένων και την τροποποίηση και τη διαγραφή των υπαρχόντων.
8. **update\_tr\_type\_data** : Η σελίδα αυτή είναι για τη διαχείριση του γνωρίσματος «Treatment Type».
9. **update\_sp\_cat\_data**: Αυτή η σελίδα είναι για τη διαχείριση του γνωρίσματος «Special Category».
10. **update\_issue\_data** : Η συγκεκριμένη σελίδα είναι για τη διαχείριση του γνωρίσματος «Issue», όπως αναφέρθηκε και παραπάνω.

Να σημειωθεί στο σημείο αυτό ότι δεν αναπτύχθηκε ιδιαίτερα το αισθητικό κομμάτι της εφαρμογής. Είναι ένα πολύ σημαντικό μέρος για κάθε εφαρμογή, αλλά έγινε η επιλογή σε επικοινωνία με τους χρήστες, να ολοκληρωθεί με το σωστό τρόπο η λειτουργικότητα και να μείνει ανοικτό το θέμα της αισθητικής βελτίωσης της για τη συνέχεια.

Τέλος στην επικεφαλίδα της εφαρμογής τοποθετήθηκαν κουμπιά για την πλοήγηση στην εφαρμογή. Υπάρχει η δυνατότητα να επιστρέψει κανείς στην αρχική σελίδα ή να επιστρέψει στην προηγούμενη σελίδα. Επίσης, υπάρχουν κουμπιά για να κάνει εκ νέου έναρξη των σελίδων της αναζήτησης, της εισαγωγής μίας θεραπείας ή της λίστας όλων των θεραπειών.

## 5.6 Ανάπτυξη Λειτουργικότητας

Ο τρόπος που υλοποιήθηκε η λειτουργικότητα της διεπαφής είναι βασισμένος σε συναρτήσεις που εκτελούνται από τις επιλογές του χρήστη. Σε επιλεγμένα σημεία της

διεπαφής συνδέθηκαν συγκεκριμένες συναρτήσεις. Υπάρχουν συναρτήσεις αρχικοποίησης κάποιων αντικειμένων, συναρτήσεις για την ενημέρωση της βάσης δεδομένων και συναρτήσεις για την ανανέωση κάποιων στοιχείων της διεπαφής. Δεν θεωρείται σκόπιμο να παρουσιαστούν όλες οι συναρτήσεις που αναπτύχθηκαν και ο σκοπός τους, από τη στιγμή που η λειτουργικότητα που απαιτήθηκε δεν απαιτεί την αλληλεπίδραση μεταξύ ενεργειών. Υπάρχουν μόνο άμεσες συνέπειες μετά την ολοκλήρωση των διαφόρων ενεργειών, όπως για παράδειγμα αν προστεθεί μία νέα τιμή στο γνώρισμα «Treatment Types», τότε θα πρέπει να ανανεωθεί η λίστα που παρουσιάζει τις τιμές αυτού του γνωρίσματος.

Ένα σημαντικό κομμάτι λειτουργικότητας που αναπτύχθηκε αφορά την διαχείριση των περιφερειακών πληροφοριών και ουσιαστικά περιέχει για κάθε μία από αυτές το σύνολο των ενεργειών που απαιτούνται για την κεντρική πληροφορία, δηλαδή τη θεραπεία. Δυστυχώς η ανάγκη αυτή ήρθε στο προσκήνιο αφού είχε ξεκινήσει η υλοποίηση, και δεν υπήρχαν πολλά περιθώρια να αναπτυχθεί ένα μοντέλο παραμετροποίησης του κώδικα, έτσι ώστε να μην επαναλαμβάνεται άσκοπα ο ίδιος κώδικας. Αυτό σε συνδυασμό με την ανάγκη ξεχωριστής ονομασίας των στοιχείων της διεπαφής, ακόμη και αυτών που διαχειρίζονταν το ίδιο γνώρισμα, συνετέλεσε στο γεγονός ότι ο κώδικας είναι εκτενής και πολλές φορές επαναλαμβανόμενος.

## 5.7 Ανάπτυξη Ασύγχρονης Εκτέλεσης

Για το συγχρονισμό των δεδομένων και την υλοποίηση της ασύγχρονης λειτουργίας, σχεδιάστηκε να δημιουργηθεί ένα αποθετήριο για την αποθήκευση των εντολών επεξεργασίας των δεδομένων της βάσης, έτσι ώστε όταν η εφαρμογή συνδέονταν στο δίκτυο, να εκτελούσε στην απομακρυσμένη βάση δεδομένων τις εντολές και να γίνονταν ο συγχρονισμός. Με τον τρόπο αυτό θα μπορούσαν περισσότεροι του ενός χρήστες να χρησιμοποιούν την εφαρμογή και να ενημερώνουν ανεξάρτητα την κεντρική βάση δεδομένων (όχι βέβαια ταυτόχρονα).

Λόγω πίεσης χρόνου για την ολοκλήρωση της εφαρμογής όμως, επιλέχθηκε να γίνονται αλλαγές από έναν χρήστη μόνο, έτσι ώστε να μην παρουσιάζονται προβλήματα με επικαλύψεις δεδομένων. Ο χρήστης που υλοποιεί τις αλλαγές στη βάση, μπορεί μετά το πέρας αυτών, και αφού συνδεθεί στο διαδίκτυο, να ενημερώσει το αρχείο της βάσης δεδομένων στο Dropbox, το οποίο να είναι πλέον διαθέσιμο για όλους του χρήστες.

Το Dropbox προσφέρει τη δυνατότητα δημιουργίας φακέλου για συγκεκριμένη εφαρμογή, και μέσω του API που προσφέρει [15] μπορεί μία εφαρμογή να συνδεθεί στο φάκελο αυτό και να κατεβάσει ή να κάνει προσθήκη ή ενημέρωση των αρχείων στο χώρο αυτό. Αναπτύχθηκαν λοιπόν δύο συναρτήσεις, μία για το κατέβασμα της βάσης δεδομένων από το Dropbox και μία για την ενημέρωση. Στο σημείο αυτό θα ήταν χρήσιμο να αναπτυχθεί και ένας μηχανισμός ελέγχου της ημερομηνίας τροποποίησης του τοπικού αρχείου της βάσης δεδομένων και του αρχείου που βρίσκεται στο Dropbox, έτσι ώστε να αποφευχθεί η περίπτωση να είναι παλαιότερο το τοπικό αρχείο από το αρχείο που βρίσκεται στο δίκτυο και να χαθούν πληροφορίες που μπήκαν στη βάση δεδομένων από άλλο χρήστη. Δυστυχώς η ανάπτυξη ενός τέτοιου μηχανισμού δεν κατέστη δυνατή εντός εύλογου χρονικού διαστήματος.

## 5.8 Εκτέλεση σε Πολλαπλές Πλατφόρμες

Αφού ολοκληρώθηκε η αρχική υλοποίηση της εφαρμογής έγινε προσπάθεια να δοκιμαστεί η εκτέλεσή της σε φορητές συσκευές Apple [37], αλλά δυστυχώς δεν ολοκληρώθηκε η προσπάθεια με επιτυχία. Απαιτούνταν η κατοχή ενός υπολογιστή της

Apple στον οποίο απαιτούνταν να εγκατασταθούν πολλές εφαρμογές. Βρέθηκε ένας παλιός φορητός υπολογιστής Apple, στον οποίο δυστυχώς δεν μπορούσαν να εγκατασταθούν οι απαιτούμενες εφαρμογές.

Στη συνέχεια καταβλήθηκε προσπάθεια να τρέξει η εφαρμογή σε Android φορητές συσκευές σύμφωνα με οδηγίες που εντοπίστηκαν [38]. Απαιτούνταν η εγκατάσταση και η χρήση καινούργιων εργαλείων αλλά ο διαθέσιμος χρόνος είχε ήδη εξαντληθεί. Αποφασίστηκε να μην καταβληθεί επιπλέον προσπάθεια και να καταγραφεί ότι είχε επιτευχθεί.

## 5.9 Εγκατάσταση Εφαρμογής

Η εγκατάσταση της εφαρμογής Tai Yi προϋποθέτει συγκεκριμένες ενέργειες για τη δημιουργία του πακέτου της εφαρμογής μέσα από το MSV, και στη συνέχεια κάποιες ενέργειες στον υπολογιστή που θα εγκατασταθεί. Εξαιτίας του γεγονότος ότι το Apache Cordova είναι λογισμικό για την ανάπτυξη εφαρμογών ανεξάρτητων πλατφόρμας, είναι σχεδιασμένο να ανεβάζει τις εφαρμογές που αναπτύσσονται με αυτό σε συγκεκριμένα αποθετήρια για τη διακίνηση ασφαλών εφαρμογών όπως είναι το Google Play ή το iTunes. Υπάρχουν οδηγίες [39] για τη δημιουργία των απαιτούμενων πακέτων για την ανάρτηση της εφαρμογής σε αποθετήρια για περιβάλλον Microsoft Windows. Ακολουθώντας τις οδηγίες αυτές δημιουργήθηκε ένα πακέτο εγκατάστασης της εφαρμογής για υποστήριξη περιβάλλοντος Microsoft Windows x64 (λειτουργία σε 64 bit).

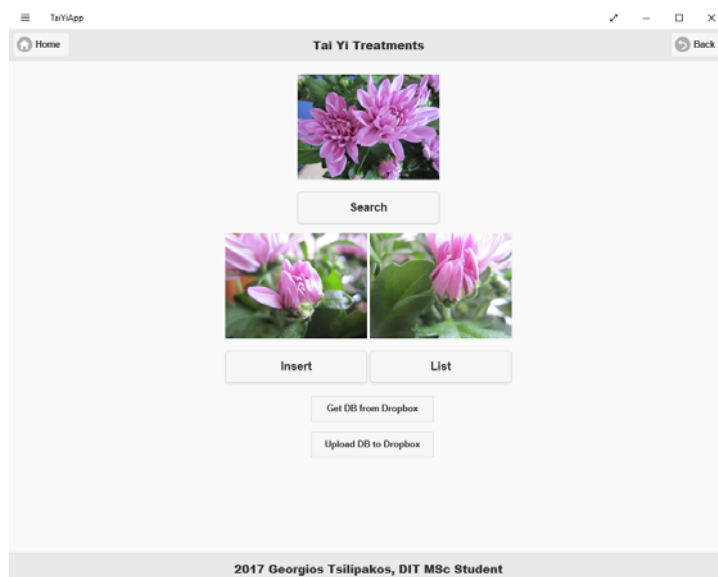
Στη συνέχεια χρησιμοποιώντας το πακέτο αυτό δίνεται η δυνατότητα να εγκατασταθεί σε περιβάλλον Microsoft Windows σύμφωνα και πάλι με τις παραπάνω οδηγίες. Παρέχεται η δυνατότητα η εφαρμογή να εγκατασταθεί για δοκιμή αλλά αυτό προϋποθέτει ο υπολογιστής στον οποίο θα εγκατασταθεί να είναι ρυθμισμένος για να υποστηρίξει την ανάπτυξη [40]. Αν είναι ενεργή αυτή η ρύθμιση η εφαρμογή Tai Yi εγκαθίσταται επιλέγοντας να εκτελεστεί με το PowerShell ένα από τα αρχεία του πακέτου εγκατάστασης που δημιουργήθηκε. Με τον τρόπο αυτό η εφαρμογή πλέον είναι διαθέσιμη για εκτέλεση.

Η εφαρμογή δοκιμάστηκε και λειτουργεί σε περιβάλλον Microsoft Windows 10. Αναλυτικές οδηγίες εγκατάστασης εμπλουτισμένες με εικόνες από τη διαδικασία δίνονται στο παράρτημα I.

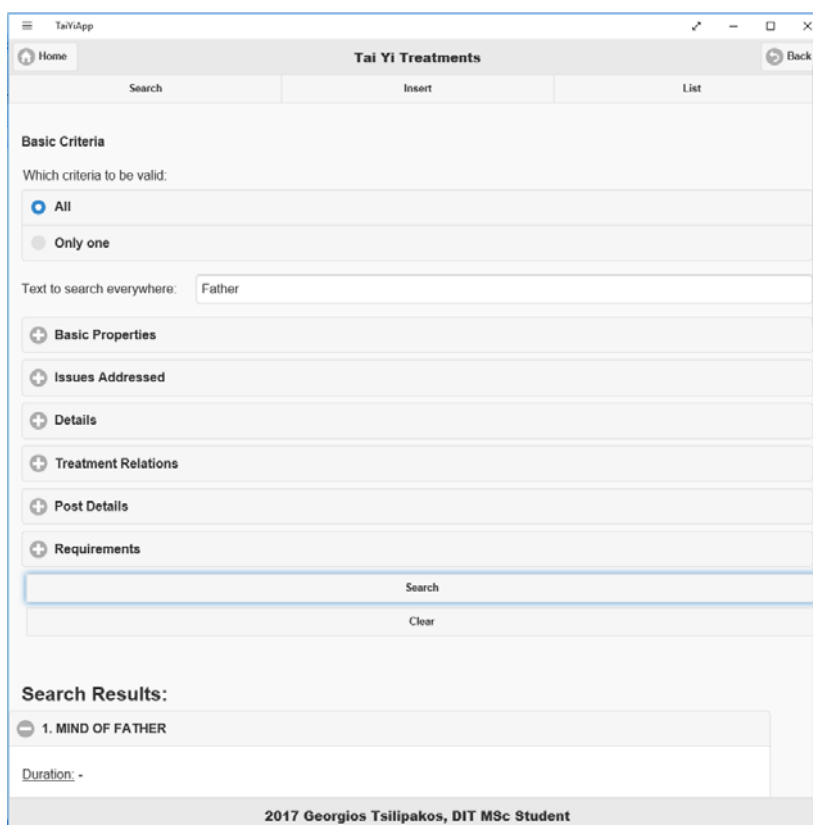
## 5.10 Παρουσίαση Εφαρμογής

Στο σημείο αυτό δίνεται μία συνοπτική παρουσίαση της εφαρμογής Tai Yi με εικόνες κατά τη διάρκεια της εκτέλεσής της. Για τις βασικές λειτουργίες της παρουσιάζεται και από μία εικόνα. Η αρχική λοιπόν εκτέλεση της εφαρμογής Tai Yi εμφανίζει το παράθυρο που δίνεται στην εικόνα 16, που είναι η αρχική σελίδα από την οποία ο χρήστης πλέον μπορεί να πλοηγηθεί στις διάφορες επιλογές που του δίνονται.

Στην εικόνα 17 εμφανίζεται η σελίδα αναζήτησης θεραπειών με κάποια από τα κριτήρια που φαίνονται και τυχαία αποτελέσματα αναζήτησης. Η σελίδα για την εισαγωγή θεραπειών δίνεται στην εικόνα 18 και είναι παρόμοια με τη σελίδα για την επεξεργασία μίας θεραπείας, η οποία δίνεται στην εικόνα 19.



Εικόνα 18: Αρχική σελίδα εφαρμογής Tai Yi



Εικόνα 19: Αναζήτηση θεραπειών στην εφαρμογή Tai Yi

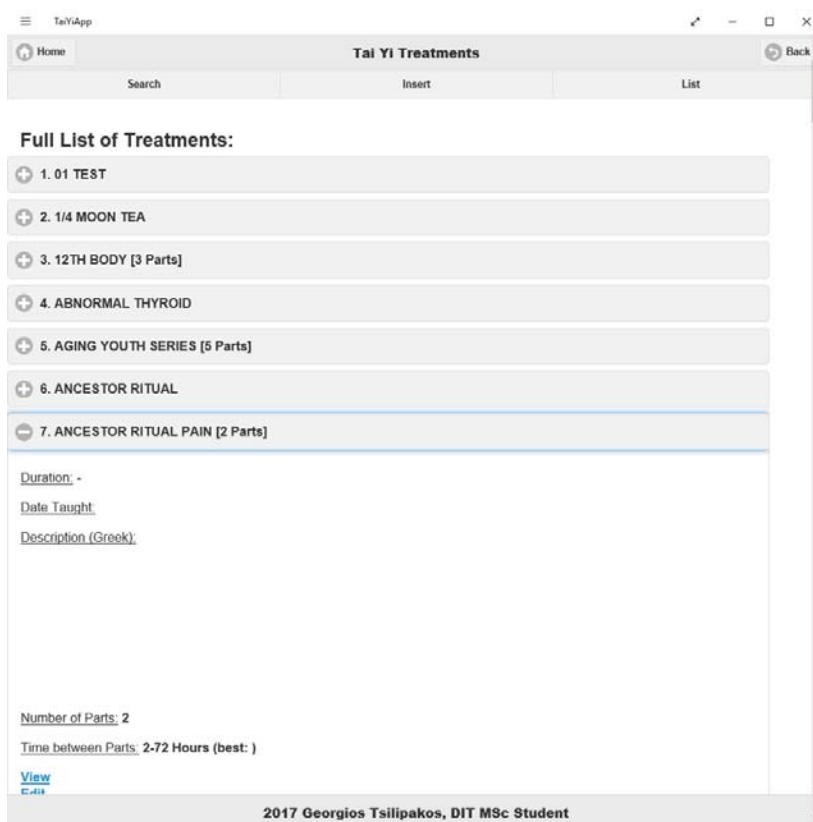
Στη συνέχεια στην εικόνα 20 δίνεται η λίστα του συνόλου των θεραπειών, όσες τουλάχιστον είναι ορατές, και η σελίδα για την προβολή των θεραπειών εμφανίζεται στην εικόνα 21. Τέλος, η σελίδα για την ενημέρωση των περιφερειακών δεδομένων εμφανίζεται στην εικόνα 22 και είναι για την ενημέρωση των πληροφοριών για το γνώρισμα «Special Category».

The screenshot shows the 'Tai Yi Treatments' app interface. At the top, there's a header with 'Tai YiApp' and a 'Home' button. Below the header, there's a navigation bar with 'Search', 'Insert', and 'List' options. The main content area is titled 'Tai Yi Treatments' and features a 'Back' button. The form is divided into several sections: 'Basic Properties' (with fields for Treatment Name, Duration, and Date Taught), 'Treatment in Parts', 'Type of Treatment' (with an 'Update Types of Treatments' button), 'Treatment Description' (with fields for Short Description in Greek and English), 'Issues Addressed', 'Details', 'Treatment Relations', and 'Post Details'. The footer displays '2017 Georgios Tsilipakos, DIT MSc Student'.

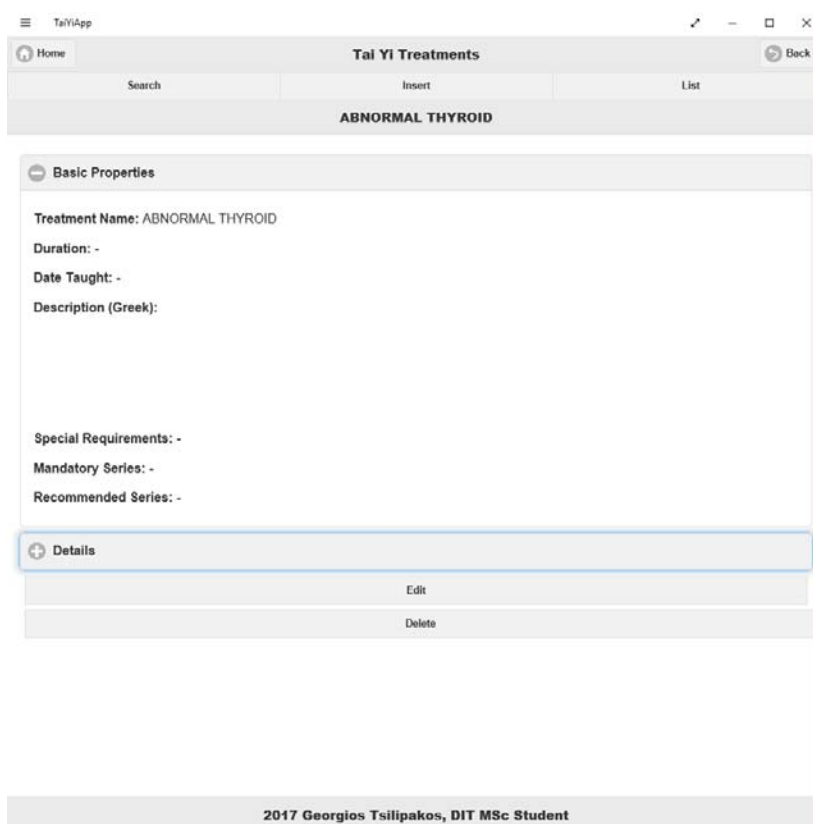
Εικόνα 20: Εισαγωγή θεραπείας στην εφαρμογή Tai Yi

This screenshot shows the same app interface but with the 'ANXIETY OF LOVE' treatment selected. The 'Basic Properties' section now includes a 'Treatment ID' field with the value '210'. The 'Treatment Name' field contains 'ANXIETY OF LOVE'. The other fields (Duration, Date Taught, Short Description in Greek and English) are empty. The 'Issues Addressed' section is also visible. The footer remains the same, showing '2017 Georgios Tsilipakos, DIT MSc Student'.

Εικόνα 21: Επεξεργασία θεραπείας στην εφαρμογή Tai Yi



Εικόνα 22: Λίστα του συνόλου των θεραπειών στην εφαρμογή Tai Yi



Εικόνα 23: Προβολή μιας θεραπείας στην εφαρμογή Tai Yi

The screenshot displays the 'Tai Yi Treatments' application interface. At the top, there is a navigation bar with a 'Home' button and a 'Back' button. Below the navigation bar, the 'Special Categories' section is visible. It contains three input fields: 'Special Category ID' with the value '3', 'Special Category Name' with the value 'Ανώτερα Ενεργειακά Σώματα', and 'Special Category Description' which is empty. Below these fields are four buttons: 'Insert', 'Update', 'Delete', and 'Clear'. Underneath the buttons is a 'Special Categories List' section. This list contains 12 items, each with a number and a description, and a right-pointing arrow icon. The items are: 1. Αναζωογόνηση, 2. Ανοσοποιητικό Σύστημα, 3. Ανώτερα Ενεργειακά Σώματα, 4. Αποτοξίνωση, 5. Αυτο-ενδυνάμωση, 6. Αφαίρεση Ενεργειακών Επιρροών, 7. Δημιουργικότητα, 8. Διαύγεια και εσωτερική επίγνωση, 9. Εθισμοί, 10. Νους, 11. Ορμονικά θέματα, and 12. Στρες. At the bottom of the application, there is a footer that reads '2017 Georgios Tsilipakos, DIT MSc Student'.

**Εικόνα 24: Επεξεργασία περιφερειακών δεδομένων στην εφαρμογή Tai Yi (Special Categories)**

Αναλυτικές οδηγίες χρήσης της εφαρμογής δίνονται στο παράρτημα II, το οποίο είναι εμπλουτισμένο με εικόνες.

## 6. ΕΛΕΓΧΟΣ ΛΟΓΙΣΜΙΚΟΥ

### 6.1 Επικύρωση Εφαρμογής

Ο έλεγχος του λογισμικού είναι ένα μεγάλο κεφάλαιο στο σχεδιασμό και την ανάπτυξη εφαρμογών και έχει μελετηθεί εκτενώς. Βάσει της βιβλιογραφίας [41], των μαθημάτων για την τεχνολογία λογισμικού [42], μελετών που έχουν γίνει [43], καθώς και των σύγχρονων μεθόδων προσέγγισης της ανάπτυξης λογισμικού από την αγορά [44], η επικύρωση (validation) και η επαλήθευση (verification) των εφαρμογών είναι ένα βασικό τμήμα στην ανάπτυξή τους. Μάλιστα είναι σημαντικό να υπάρχει εξωτερική ομάδα ελέγχου για την επικύρωση και την επαλήθευση του υπό ανάπτυξη λογισμικού, έτσι ώστε οι ελεγκτές να μην είναι επηρεασμένοι από την όλοια διαδικασία ανάπτυξης. Χωρίς βέβαια να σημαίνει αυτό ότι ο προγραμματιστής δεν πρέπει να έχει κατά τη διάρκεια της ανάπτυξης στραμμένη την προσοχή του και στην επικύρωση και επαλήθευση του λογισμικού. Στην συγκεκριμένη περίπτωση, απουσία ανεξάρτητες ομάδας, την επικύρωση και την επαλήθευση ανέλαβαν οι χρήστες του.

Αρχικά παρουσιάζεται ο τρόπος επικύρωσης της εφαρμογής και συγκεκριμένα η διαδικασία που προσεγγίστηκε η ερώτηση: φτιάχνεται το σωστό λογισμικό; Δηλαδή, η εφαρμογή που αναπτύσσεται καλύπτει τις απαιτήσεις των χρηστών; Ένα από τα σημαντικότερα σημεία για την ορθή επικύρωση της εφαρμογής ήταν η αρχική αποτύπωση της διεπαφής που συμφωνήθηκε με τους χρήστες. Με τον τρόπο αυτό δεν ήρθαν οι χρήστες σε επαφή με τη διεπαφή της εφαρμογής αφού είχε ολοκληρωθεί η ανάπτυξή της, αλλά πολύ νωρίτερα, δίνοντας μάλιστα και κατεύθυνση στον τρόπο παρουσίασης των δεδομένων. Επίσης, ήταν από την αρχή καθορισμένη η λειτουργικότητα που θα κάλυπτε η εφαρμογή.

Εκτός όμως από την αρχική αποτύπωση της διεπαφής, η εφαρμογή παρουσιάστηκε στους χρήστες περισσότερες από μία φορές μόλις ολοκληρώθηκε το βασικό κομμάτι της λειτουργικότητας της. Με τον τρόπο αυτό οι χρήστες κλήθηκαν να εξετάσουν τη λειτουργία της επί της πράξης και βλέποντας οι ίδιοι αν καλύπτει τις απαιτήσεις τους, εκτελώντας τα σενάρια αποσφαλμάτωσης και γενικότερα χρησιμοποιώντας την. Εδώ να σημειωθεί ότι ενώ οι τεχνικές απαιτήσεις της εφαρμογής να ήταν πολύπλοκες, οι απαιτήσεις όσον αφορά τη λειτουργικότητα δεν εμφάνιζαν αυξημένη πολυπλοκότητα. Σε περισσότερο πολύπλοκες εφαρμογές, δεν είναι πάντα προς τη σωστή κατεύθυνση να εμπλέκονται οι χρήστες εκτενώς στην επικύρωση του λογισμικού, διότι μπορεί να εμφανιστούν φαινόμενα συνεχούς αλλαγής των απαιτήσεων που μπορεί να επιφέρουν αλλαγές και στο σχεδιασμό. Αυτό έχει ως αποτέλεσμα να χάνεται πολύτιμος χρόνος και να επανασχεδιάζονται πολλά τμήματα του λογισμικού καθιστώντας την ανάπτυξης αργή και επίπονη. Στην περίπτωση που εξετάζεται εδώ η αλληλεπίδραση με τους χρήστες επέφερε εμπλουτισμό των απαιτήσεων και μεγαλύτερη ανάλυση στο τι τελικά ήθελαν από το λογισμικό οι χρήστες, δηλαδή βοήθησε πραγματικά στην επικύρωση τελικά της εφαρμογής.

Σημειώνονται μερικά σημεία των απαιτήσεων των χρηστών που εντοπίστηκαν στη συγκεκριμένη διαδικασία. Η απαίτηση να μην φαίνονται στην προβολή μίας θεραπείας οι τίτλοι των γνωρισμάτων που δεν έχουν αποθηκευμένα δεδομένα είναι ένα παράδειγμα, καθώς και η απαίτηση να εμφανίζονται στις λίστες των αποτελεσμάτων μετά την αναζήτηση όλα τα βασικά γνωρίσματα όπως καθορίστηκαν στη λειτουργικότητα των λιστών. Σημαντικότερο επίσης που δεν είχε εντοπιστεί από την αρχή ήταν η απαίτηση να υπάρχει περιγραφή της θεραπείας στα Ελληνικά, καθώς και διαφορές όσον αφορά στην πληθικότητα της συσχέτισης των οντοτήτων και των πληροφοριών που απαιτούνταν να τηρούνται για τις περιφερειακές πληροφορίες, όπως



για παράδειγμα η σχέση μεταξύ των «Treatments» και του «Energy Type» που αρχικά ήταν ένα γνώρισμα με LUT και τελικά έγινε σχέση N προς M.

## 6.2 Επαλήθευση Εφαρμογής

Ένα άλλο κομμάτι του ελέγχου είναι όπως προαναφέρθηκε η επαλήθευση του λογισμικού, δηλαδή ο έλεγχος της απάντησης στο ερώτημα: φτιάχεται το λογισμικό σωστά; Ικανοποιούνται δηλαδή οι απαιτήσεις του σχεδιασμού; Ο έλεγχος αυτός διενεργείται ουσιαστικά μετά την ολοκλήρωση του σχεδιασμού και καθ' όλη τη διάρκεια ανάπτυξης και επικύρωσης του λογισμικού. Λαμβάνοντας υπόψη ότι η επικύρωση της εφαρμογής οδήγησε πολλές φορές σε τροποποίηση του σχεδιασμού, είναι εμφανές ότι η διαδικασία της επαλήθευσης διενεργήθηκε πολλές φορές. Εδώ ο έλεγχος έγινε κυρίως κατά τη διάρκεια υλοποίησης της εφαρμογής και οι χρήστες βοήθησαν στο έλεγχο που έγινε με την εκτέλεση των σεναρίων αποσφαλμάτωσης.

Στην επαλήθευση της εφαρμογής ελέγχεται κυρίως η τεχνική λειτουργία της εφαρμογής όπως το αν ενημερώνονται σωστά τα δεδομένα στη βάση δεδομένων, αν επιστρέφονται τα σωστά αποτελέσματα κατά την αναζήτηση και αν η λήψη του αρχείου της βάσης από το απομακρυσμένο σημείο γίνεται σωστά. Υπάρχουν πολλές λειτουργίες της εφαρμογής που απαιτούσαν επαλήθευση η οποία πραγματοποιήθηκε σταδιακά κατά την υλοποίηση του κώδικα για τις τεχνικές λειτουργίες της και από την εκτέλεση των σεναρίων αποσφαλμάτωσης για τις λειτουργικές απαιτήσεις της.

Κατά τη δημιουργία του κώδικα εκτελούνταν συνεχώς η εφαρμογή για να δοκιμάζεται το αν η λειτουργία του κάθε τμήματος ανταποκρίνονταν στο σχεδιασμό. Για παράδειγμα, οι συναρτήσεις που αναπτύχθηκαν σε JavaScript και αναφέρονται στο πέμπτο κεφάλαιο δοκιμάζονταν κατά τη δημιουργία τους για να αναδειχθούν τυχόν προβλήματα. Η ανάγκη για παράδειγμα για την ενημέρωση όλων των λιστών συγκεκριμένων γνωρισμάτων μετά την ενημέρωση των δεδομένων αυτού του γνωρίσματος εμφανίστηκε μέσω αυτής της διαδικασίας. Η παραπάνω διαδικασία ακολουθούνταν και κατά τη διάρκεια ανάπτυξης της διεπαφής με HTML, δηλαδή η εφαρμογή εκτελούνταν συνεχώς και ελέγχονταν για την επαλήθευσή της.

## 6.3 Δημιουργία Σεναρίων Αποσφαλμάτωσης

Μετά την ολοκλήρωση της πρώτης έκδοσης της εφαρμογής Tai Yi, δημιουργήθηκαν κάποια σενάρια για να εκτελεστούν σε συνεργασία με τους χρήστες. Τα σενάρια αυτά αποτελούνταν από παραδείγματα εκτέλεσης συγκεκριμένων λειτουργιών στην εφαρμογή και είχαν προετοιμαστεί πριν το καθορισμό συνάντησης με τους χρήστες, στην οποία θα εκτελούνταν. Σχεδιάστηκαν λοιπόν τα παρακάτω σενάρια:

- Εγκατάσταση εφαρμογής : Το πρώτο σενάριο ήταν η εγκατάσταση της εφαρμογής σε έναν νέο υπολογιστή που δεν είχε εγκατασταθεί ξανά. Όπως έχει προαναφερθεί, η εγκατάσταση λειτουργεί για υπολογιστές με Microsoft Windows. Αυτό θα εξυπηρετούσε έτσι ώστε να δοκιμαστεί ότι πραγματοποιείται επιτυχώς η εγκατάσταση της εφαρμογής ακολουθώντας τις οδηγίες εγκατάστασης που παρουσιάζονται στο παράρτημα Ι.
- Κατέβασμα (downloading) της βάσης δεδομένων από το αποθετήριο : Εδώ ο στόχος είναι να δοκιμαστεί η λειτουργία της διασύνδεσης της εφαρμογής με τον χώρο της εφαρμογής που τηρείται στο Dropbox.

- **Λίστα θεραπειών :** Ουσιαστικά με το κατέβασμα της βάσης δεδομένων, το πρώτο πράγμα που μπορεί εύκολα να δοκιμαστεί είναι η λίστα με το σύνολο των θεραπειών που έχουν καταχωρηθεί.
- **Προβολή θεραπείας :** Εδώ δοκιμάζεται ουσιαστικά ο τρόπος που προβάλλεται μία θεραπεία, και είναι ένα σημείο που μπορεί να δοκιμαστεί πολλές φορές και για πολλές θεραπείες. Μάλιστα μπορεί να εκτελείται πάντα σε συνδυασμό με το επόμενο σενάριο.
- **Επεξεργασία θεραπείας :** Στη σενάριο αυτό ελέγχεται ο τρόπος επεξεργασίας μίας θεραπείας. Καταβλήθηκε προσπάθεια να δοκιμαστούν επιμέρους τα παρακάτω σημεία, δηλαδή το σενάριο αυτό να εκτελεστεί όσο περισσότερες φορές είναι δυνατόν:
  - Να γίνει ενημέρωση είτε μεμονωμένα είτε συνδυαστικά στα περισσότερα γνωρίσματα μίας θεραπείας μέσω επαναλαμβανόμενης εκτέλεσης.
  - Να γίνει τροποποίηση στα μέρη μίας θεραπείας (να αφαιρεθούν σε μία περίπτωση, να προστεθούν σε θεραπεία χωρίς μέρη και να τροποποιηθούν σε άλλη περίπτωση).
  - Να γίνει τροποποίηση στις συνδέσεις μίας θεραπείας (όπως και πριν, να αφαιρεθούν σε μία περίπτωση, να προστεθούν σε θεραπεία χωρίς συνδέσεις και να τροποποιηθούν σε κάποια άλλη περίπτωση).
  - Για τα γνωρίσματα που παίρνουν πολλαπλές τιμές να γίνουν όλες οι πιθανές τροποποιήσεις (είτε να αφαιρεθούν πλήρως, είτε να προστεθούν χωρίς να υπάρχουν, είτε να τροποποιηθεί το σύνολο των τιμών τους).
  - Να δοκιμαστεί να τροποποιηθεί το όνομα μίας θεραπείας και να δοθεί όνομα που υπάρχει ήδη.
- **Εισαγωγή θεραπείας :** Το σενάριο αυτό δοκιμάζει τον τρόπο εισαγωγής μίας θεραπείας στη βάση δεδομένων. Αποφασίστηκε για το συγκεκριμένο σενάριο να ακολουθήσουμε τις παρακάτω οδηγίες :
  - Να αποθηκευτεί μία δοκιμαστική θεραπεία που θα έχει δεδομένα για το σύνολο των γνωρισμάτων που υπάρχουν, έτσι ώστε να δοκιμαστούν όλα τα στοιχεία της διεπαφής της εφαρμογής.
  - Να δημιουργηθούν μέρη για τη συγκεκριμένη θεραπεία.
  - Να συνδεθεί με υπάρχουσες θεραπείες.
  - Για τις περιφερειακές πληροφορίες να συνδεθεί με περισσότερες από μία τιμές (συγκεκριμένα για τα γνωρίσματα «Treatment Type», «Treatment Special Category» και «Energy Type»).
  - Να δοκιμαστεί να εισαχθεί μία θεραπεία με ίδιο όνομα με κάποια άλλη.
  - Να δοκιμαστεί να εισαχθεί μία θεραπεία μόνο με όνομα.
- **Διαγραφή θεραπείας :** Να γίνει δοκιμή να διαγραφεί μία θεραπεία. Σημαντική πρόβλεψη για το σενάριο αυτό είναι να δοκιμαστεί η διαγραφή μίας θεραπείας η οποία έχει συνδεθεί ήδη με μία άλλη.
- **Αναζήτηση θεραπείας :** Στο σενάριο αυτό καταβλήθηκε προσπάθεια να εκτελεστεί αναζήτηση είτε μεμονωμένα είτε συνδυαστικά στα περισσότερα γνωρίσματα μίας θεραπείας μέσω πολλαπλής εκτέλεσης της αναζήτησης. Επίσης, να εκτελεστούν γενικές αναζητήσεις με μία ή περισσότερες λέξεις, οι οποίες να συνδέονται είτε αθροιστικά είτε συμπληρωματικά. Λόγω της μεγάλης

σημασίας που έδιναν οι χρήστες στη συγκεκριμένη λειτουργία της εφαρμογής, η εκτέλεση των συγκεκριμένων σεναρίων αποσφαλμάτωσης έγινε πολλές φορές, έτσι ώστε να αναδειχθεί οποιοδήποτε πρόβλημα υπάρχει.

- Αποστολή βάσης δεδομένων στο αποθετήριο : Να δοκιμαστεί, μετά το πέρας των αλλαγών και των τροποποιήσεων των αποθηκευμένων πληροφοριών στη βάση δεδομένων, να γίνει αποστολή της στο Drobox, έτσι ώστε να είναι διαθέσιμη σε οποιοδήποτε άλλο σύστημα που είχε εγκαταστημένο την εφαρμογή.

#### 6.4 Εκτέλεση Σεναρίων Αποσφαλμάτωσης και Διορθωτικές Ενέργειες

Η εγκατάσταση της εφαρμογής έγινε σε τρεις διαφορετικούς υπολογιστές με λειτουργικό σύστημα Microsoft Windows 10. Δυστυχώς δεν επιτευχθεί να δοκιμαστεί η εγκατάσταση της εφαρμογής σε παλαιότερη έκδοση των Windows. Δεν εμφανίστηκαν προβλήματα κατά την εγκατάσταση της εφαρμογής, αλλά έγινε πραγματικά απροβλεπτό και σύμφωνα με όσα έχουν περιγραφεί στο παράρτημα Ι.

Στη συνέχεια, η εκτέλεση των σεναρίων αποσφαλμάτωσης, έγινε από τρεις χρήστες της εφαρμογής. Η πρώτη εκτέλεση των σεναρίων έγινε σε προσωπικό υπολογιστή όπου έγινε και η πρώτη εγκατάσταση. Στη συνέχεια, από του χρήστες σε άλλους υπολογιστές. Οι εκτέλεση των σεναρίων από τους χρήστες έγινε σε συναντήσεις που πραγματοποιήθηκαν για την παρουσίαση της εφαρμογής και την δοκιμαστική της λειτουργία παρουσία όλων των εμπλεκόμενων. Έγιναν τέσσερις συναντήσεις και μάλιστα μετά την πρώτη συνάντηση και τη δεύτερη εκτέλεση των σεναρίων αποσφαλμάτωσης, δόθηκαν πολλές θεραπείες για εισαγωγή των πληροφοριών τους στη βάση δεδομένων, κάτι που υλοποιήθηκε σε περιβάλλον εκτέλεσης της εφαρμογής και όχι σε περιβάλλον ανάπτυξης για να δοκιμαστεί η εφαρμογή.

Η εκτέλεση των σεναρίων έφερε στο προσκήνιο προβλήματα που δεν είχαν προβλεφτεί κατά το σχεδιασμό και την υλοποίηση βοηθώντας στην αποσφαλμάτωση της εφαρμογής. Με τον τρόπο αυτό έγινε καλύτερα η επικύρωση και η επαλήθευση του λογισμικού. Ουσιαστικά δηλαδή η εκτέλεση των σεναρίων αποσφαλμάτωσης οδήγησε στην επιτυχή επικύρωση και επαλήθευση της εφαρμογής.

Δεν θεωρείται σκόπιμο να παρουσιαστούν αναλυτικά τα αποτελέσματα της κάθε εκτέλεσης των σεναρίων αποσφαλμάτωσης, το κυριότερο είναι να αναφερθούν τα προβλήματα που αναδείχθηκαν. Ένα από τα σημαντικότερα προβλήματα που εντοπίστηκε ήταν η ανικανότητα της βιβλιοθήκης της SQLite που χρησιμοποιήθηκε να εκτελέσει αναζητήσεις ανεξάρτητες μεγέθους και τονισμού (case and accent insensitive) στα ελληνικά. Για το λόγο αυτό απαιτήθηκε να δημιουργηθούν άλλα πεδία στη βάση δεδομένων έτσι να μπορεί να γίνει η αναζήτηση με ταχύτητα και όχι να παράγεται κάθε φορά το κάθε κείμενο χωρίς τόνους και μεγάλα γράμματα. Μάλιστα αναπτύχθηκαν και συναρτήσεις για τις μετατροπές των κειμένων.

Επίσης, οι αλλαγές που παρουσιάστηκαν στις προηγούμενες παραγράφους, αποτελούν ουσιαστικά αποτέλεσμα της εκτέλεσης των σεναρίων αποσφαλμάτωσης κατά τις συναντήσεις με τους χρήστες της εφαρμογής, μέσω της οποίας έγινε και η επικύρωση και η επαλήθευση της από αυτούς.

## 7. ΣΥΜΠΕΡΑΣΜΑΤΑ

### 7.1 Γενικά

Ο στόχος της ανάπτυξης εφαρμογής για τη διαχείριση των θεραπειών του Ταί Υι προσεγγίστηκε διενεργώντας πολλαπλές έρευνες για κατάλληλες τεχνολογίες και εργαλεία. Αναζητήθηκαν σχεσιακές βάσεις δεδομένων ανεξάρτητες εξυπηρετητή, εργαλεία για τη δημιουργία αρχικών αποτυπώσεων της διεπαφής εφαρμογών, ολοκληρωμένα περιβάλλοντα ανάπτυξης λογισμικού και τεχνολογίες ευρέως χρησιμοποιούμενες, καθώς και τρόποι για την ασύγχρονη εκτέλεση εφαρμογών χωρίς σύνδεση στο διαδίκτυο.

Επίσης, υλοποιήθηκε μία λειτουργική σχεσιακή βάση δεδομένων και αναπτύχθηκε η εφαρμογή διαχείρισης των θεραπειών με τεχνολογίες που επιτρέπουν την εύκολη εγκατάσταση και εκτέλεση της εφαρμογής σε ποικίλα περιβάλλοντα. Μπορεί να μην επιτεύχθηκε η εγκατάσταση σε φορητές συσκευές ή άλλα λειτουργικά συστήματα εκτός από Microsoft Windows, αλλά πραγματοποιήθηκε προσπάθεια προς αυτήν την κατεύθυνση.

Παρακάτω συνοψίζονται τα συμπεράσματα από την όλη προσπάθεια εντοπισμού των κατάλληλων εργαλείων και από τη χρήση τους.

### 7.2 Σχεσιακή Βάση Δεδομένων Ανεξάρτητη Εξυπηρετητών

Η έρευνα στις βάσεις δεδομένων αποκάλυψε ότι η χρησιμοποίηση των σχεσιακών βάσεων δεδομένων, αν και είναι σχεδιαστικά από τις παλαιότερες, συνεχίζει να βρίσκεται σε πολύ υψηλά ποσοστά. Με τον καιρό αναδεικνύεται μία τάση για αύξηση των NoSQL βάσεων δεδομένων, η οποία φαίνεται να έχει σχέση με τη βελτίωση της αποτελεσματικότητάς αυτών των βάσεων. Μπορεί η αποτελεσματικότητα να αυξάνει λόγω και της αύξησης της υπολογιστικής δύναμης των επεξεργαστών, αφού σύμφωνα με τον G. Moore η πυκνότητα των τρανζίστορ στους επεξεργαστές διπλασιάζεται κάθε δύο χρόνια περίπου.

Ωστόσο, οι σχεσιακές βάσεις δεδομένων που είναι διαδεδομένες δεν είναι αυτές που είναι ανεξάρτητες εξυπηρετητή. Δηλαδή, για χρήση βάσεων ανεξάρτητων εξυπηρετητή προτιμούνται NoSQL βάσεις. Εντοπίστηκε ωστόσο η SQLite που υποστηρίζεται από μεγάλη και ενεργή κοινότητα. Μάλιστα υπάρχουν αρκετά εργαλεία για τη διαχείριση σε γραφικό περιβάλλον βάσεων δεδομένων SQLite, τα οποία προσφέρουν προσεγγμένες διεπαφές για την εύκολη διαχείριση αυτών των βάσεων.

Όσον αφορά στην υλοποίηση μίας σχεσιακής βάσης δεδομένων, υπάρχει τόσο μεγάλη τεκμηρίωση στον τρόπο σχεδιασμού, που είναι δύσκολο να μην λύσει κανείς τυχόν απορίες ή προβληματισμούς που θα προκύψουν. Επιπλέον, για το σχεδιασμό σχεσιακών βάσεων δεδομένων υπάρχει εκτενής βιβλιογραφία. Η δυσκολία έγκειται στην μεταφορά του σχεδιασμού στην πραγματική υλοποίηση της εφαρμογής και σε πιθανά προβλήματα που μπορεί να εμφανιστούν με τη διαχείριση των δεδομένων. Στο σημείο αυτό υπάρχουν ακόμη θέματα που έχουν να κάνουν με την αντιπαράβολή χαρακτήρων (collation) σε μία σχεσιακή βάση δεδομένων, κυρίως όταν χρησιμοποιείται γλώσσα εκτός από τα Αγγλικά. Βέβαια σε ευρέως διαδεδομένα και ιδιόκτητα συστήματα σχεσιακών βάσεων δεδομένων όπως την Oracle Database ή την Microsoft SQL Server αυτά τα θέματα έχουν λυθεί εδώ και πολλά χρόνια. Είναι όμως λογικό όταν χρησιμοποιεί κανείς βάσεις δεδομένων που εκτελούνται χωρίς την ύπαρξη εξυπηρετητή, να μην μπορούν να καλύψουν όλες τις απαιτήσεις των χρηστών, αφού πρέπει να εκτελούνται χωρίς υψηλές υπολογιστικές απαιτήσεις.

Τέλος, να σημειωθεί ότι είναι πλέον πολύ διαδεδομένη η χρήση βάσεων δεδομένων για την ανάπτυξη εφαρμογών. Παλαιότερα υπήρχε δυνατότητα να αναπτυχθεί εφαρμογή αποθηκεύοντας τα δεδομένα σε αρχεία, αλλά πλέον είναι τόσο εύκολη η χρήση βάσεων δεδομένων και είναι τόσο αποτελεσματικές που δεν εννοείται να αποθηκεύονται πλέον δεδομένα στο σύστημα αρχείων.

### 7.3 Αρχική Αποτύπωση Διεπαφής

Η έρευνα για τα εργαλεία αρχικής αποτύπωσης της διεπαφής μίας εφαρμογής, δηλαδή το λεγόμενο wireframe, ανέδειξε το πόσο διαδεδομένη πλέον διαδικασία αποτελεί στην ανάπτυξη εφαρμογών. Όλα τα εργαλεία που εντοπίστηκαν ήταν ιδιόκτητα και απαιτούσαν ουσιαστικά αγορά για τη χρήση τους, αποδεικνύοντας τη δύναμή τους στην εξοικονόμηση χρόνου και πόρων. Έχει προχωρήσει μάλιστα τόσο η τεχνολογία σε αυτόν τον τομέα, που υπάρχουν εργαλεία που μπορούν μετά το σχεδιασμό της αρχικής αποτύπωσης της διεπαφής να δημιουργήσουν και τον κώδικα για την ανάπτυξη της λειτουργικότητας. Κυρίως βέβαια το wireframing χρησιμοποιείται στη δημιουργία ιστοσελίδων, αλλά με τον ίδιο τρόπο μπορεί να χρησιμοποιηθεί και στη δημιουργία λογισμικού ανεξάρτητου πλατφόρμας.

Εδώ ωστόσο τα σημαντικότερα συμπεράσματα προκύπτουν από το πόσο εύκολα υλοποιήθηκε η διεπαφή της εφαρμογής σε σχέση με τις απαιτήσεις των χρηστών. Δηλαδή, η δημιουργία του wireframe, οδήγησε στην ουσιαστική αποδοχή της διεπαφής που αναπτύχθηκε χωρίς προβλήματα ή σημαντικές τροποποιήσεις. Έμπρακτα λοιπόν διαπιστώθηκε η εξοικονόμηση χρόνου σε ένα θέμα τόσο ευαίσθητο όσο η διεπαφή της εφαρμογής, που είναι ουσιαστικά ο τρόπος επικοινωνίας του χρήστη με την εφαρμογή και καθορίζει τον τρόπο μεταφοράς των εντολών του στον κώδικα της εφαρμογής. Ήταν από την αρχή καθορισμένα τα παραπάνω και αυτό βοήθησε στην ταχύτατη υλοποίησή της.

### 7.4 Τεχνολογίες Ανάπτυξης Εφαρμογών

Η ποικιλία που εντοπίστηκε στα ολοκληρωμένα περιβάλλοντα ανάπτυξης, δημιουργεί σύνθετες απορίες στο κατά πόσο καλύπτουν τις απαιτήσεις ενός προγραμματιστή. Διαπιστώθηκε από την έρευνα που έγινε ότι το περιβάλλον ανάπτυξης έχει να κάνει περισσότερο με την εξοικείωση του καθενός με αυτό, παρά με προτερήματα ή μειονεκτήματα που έχει το κάθε IDE. Η πλειοψηφία των IDE μπορεί να φιλοξενήσει ανάπτυξη με όλα τα σύγχρονα εργαλεία και γλώσσες προγραμματισμού, παρέχοντας τεράστιες δυνατότητες στον τρόπο ανάπτυξης εφαρμογών.

Έχουν αναπτυχθεί τόσα πολλά ολοκληρωμένα πλαίσια ανάπτυξης για την δημιουργία εφαρμογών που είναι πολύ δύσκολο να εντοπίσει κανείς αυτό που εξυπηρετεί καλύτερα τους στόχους του. Και εδώ μάλλον η εξοικείωση του καθενός με τις τεχνολογίες ανάπτυξης παίζει καθοριστικό ρόλο στην επιλογή ολοκληρωμένου πλαισίου ανάπτυξης. Η δυσκολία στο σημείο αυτό έγκειται στην επιλογή τεχνολογίας πάνω στην οποία να επενδύσει κανείς. Ανεξάρτητα από τη γλώσσα προγραμματισμού που μπορεί να χρησιμοποιεί κάποιος, τα ολοκληρωμένα πλαίσια ανάπτυξης πλέον είναι τόσο πολύπλοκα που πρέπει να επενδύσει κανείς πάρα πολύ χρόνο για να εμβαθύνει και να κατακτήσει τον τρόπο λειτουργίας του. Η καμπύλη εκμάθησης ενός ολοκληρωμένου πλαισίου ανάπτυξης είναι πολλές φορές πολύ δύσκολη. Υπάρχουν βέβαια εργαλεία όπως το jQuery που είναι σχετικά απλά και δεν απαιτούν ιδιαίτερες απαιτήσεις για την εγκατάσταση και χρησιμοποίησή τους, τα οποία όμως έχουν διανύσει πολλά χρόνια ζωής και αρχίζουν να παροπλίζονται.

Οι τεχνολογίες που επιλέχθηκαν στη συγκεκριμένη περίπτωση, δηλαδή το Apache Cordova και το jQuery, παρουσίασαν ευκολία στο σχεδιασμό και την υλοποίηση, ωστόσο λόγω των ιδιαίτερων απαιτήσεων της εφαρμογής, δημιουργήθηκαν προβλήματα όταν έγινε προσπάθεια να εκτελούνται σε ποικίλα περιβάλλοντα. Οι ιδιαίτερες αυτές απαιτήσεις ήταν η χρήση σχεσιακής βάσης δεδομένων ανεξάρτητης εξυπηρετητή και η ασύγχρονη εκτέλεσή της. Παρ' όλες τις δυσκολίες όμως επιτεύχθηκε ο στόχος και υλοποιήθηκε η απαιτούμενη εφαρμογή, η οποία εγκαθίσταται και εκτελείται σε περιβάλλον Microsoft Windows 10.

Η ανάπτυξη της λειτουργικότητας της εφαρμογής, ενώ καταβλήθηκε προσπάθεια να παραμετροποιηθεί έτσι ώστε ο κώδικας να μην επαναλαμβάνεται, δεν ήταν στο τέλος εφικτό στο βαθμό που θα ήταν δυνατόν. Το πρόβλημα αυτό εντοπίζεται κυρίως στο γεγονός ότι η παραμετροποίηση του κώδικα οφείλει να σχεδιαστεί από την αρχή και τυχόν αλλαγές που προκύπτουν από τον έλεγχο του λογισμικού, μπορούν να επιφέρουν τέτοιες αλλαγές στον κώδικα που είναι πρακτικά πολύ δύσκολο να γίνει περαιτέρω παραμετροποίηση. Εδώ, μπορεί τα εργαλεία και η γλώσσα να παρέχουν δυνατότητες ανάπτυξης αντικειμένων, αλλά ο τρόπος διασύνδεσης των αντικειμένων με τα στοιχεία της διεπαφής και η ανταλλαγή δεδομένων μεταξύ τους, αν έχει τις παραμικρές αποκλίσεις από το κανονικό, δημιουργεί τέτοιες ανάγκες που αναγκάζει τον προγραμματιστή να αναπτύξει μόνος του το μεγαλύτερο μέρος του κώδικα.

Η ανάπτυξη της εφαρμογής σε ένα MVC μοντέλο, ενώ φαίνεται δελεαστική, είναι πολύ δύσκολο να επιτευχθεί όταν υπάρχουν μη κανονικά δεδομένα που πρέπει να διαχειριστεί ο προγραμματιστής, όπως για παράδειγμα ανάπτυξη επιμέρους κομματιών που εμφωλεύουν διαχείριση δεδομένων, και η οποία απαιτείται να γίνεται οποιαδήποτε στιγμή επιθυμεί ο χρήστης να ρυθμίσει αυτά τα δεδομένα.

Στην περίπτωση αυτή έγινε προσπάθεια αποδόμησης του τρόπου λειτουργίας των συναρτήσεων που αναπτύχθηκαν, έτσι ώστε να μπορούν να ομαδοποιούνται ανά αντικείμενο χωρίς να εκτελούν σειρά ενεργειών η κάθε μία, αλλά να εκτελούν μεμονωμένες εργασίες. Ο τρόπος ανάπτυξης λοιπόν του κώδικα μίας εφαρμογής που υλοποιεί την λειτουργικότητά της, αν δεν είναι η λειτουργικότητα απλή και συνήθης, δεν είναι εύκολο να μπει σε συγκεκριμένα αρχιτεκτονικά μοντέλα που περιγράφουν σε επίπεδα τις διάφορες λειτουργίες. Η σπουδαιότητα να υπάρχει διαχωρισμός των λειτουργιών σε επίπεδα είναι παραπάνω από προφανής, αφού επιτρέπει τις αλλαγές σε επιμέρους μόνο σημεία, αλλά δεν είναι πάντα εφικτός.

## 7.5 Ασύγχρονη Εκτέλεση Εφαρμογών

Στο σημείο αυτό η έρευνα που πραγματοποιήθηκε ήταν συνοπτική, αφού δεν υπήρχαν απαιτήσεις για πολύπλοκες λειτουργίες ασύγχρονης εκτέλεσης. Η απαίτηση ήταν να υπάρχει το αρχείο της βάσης σε ένα σημείο προσβάσιμο από πολλούς χρήστες. Αυτό καλύπτεται εύκολα με την εύρεση κάποιου αποθετηρίου και υπάρχουν πολλά που προσφέρουν τις υπηρεσίες αυτές δωρεάν.

Ο σχεδιασμός που εγκαταλείφθηκε και αφορούσε την εκτέλεση του κώδικα τροποποίησης της σχεσιακής βάσης δεδομένων στην κεντρική βάση δεδομένων, δηλαδή την αποθήκευση του κώδικα SQL κατά τη χρήση της εφαρμογής, παρουσίαζε μεγάλη πολυπλοκότητα. Τα προβλήματα στην υλοποίηση ενός τέτοιου σεναρίου και ο τρόπος αντιμετώπισής τους θα απαιτούσε τεράστια προσπάθεια και δεν θεωρήθηκε δόκιμο να γίνει. Ένα μικρό παράδειγμα είναι όταν ο κώδικας αυτός θα εκτελούνταν στην κεντρική βάση, θα μπορούσε να εμφανίζει λάθη λόγω μεταβολών των δεδομένων από άλλους χρήστες. Πως θα διαχειρίζονταν η εφαρμογή τέτοιες περιπτώσεις και ποιος θα τις αξιολογούσε;

Καταλήξαμε ότι η αλληλεπίδραση των εφαρμογών με αποθετήρια αρχείων στο διαδίκτυο είναι κάτι πολύ διαδεδομένο και εύκολα υλοποιήσιμο, αλλά η χρήση σχεσιακών βάσεων δεδομένων χωρίς την ύπαρξης εξυπηρετητή δεν συνηθίζεται.

## ΠΙΝΑΚΑΣ ΟΡΟΛΟΓΙΑΣ

Ξενόγλωσσος όρος	Ελληνικός Όρος
Atomicity	Ατομικότητα
Application	Εφαρμογή
Component	Συστατικό
Consistency	Συνοχή
Controller	Ελεγκτής
Durability	Αντοχή
File System	Σύστημα Αρχείων
Framework	Ολοκληρωμένο Πλαίσιο Ανάπτυξης
Integrated Development Environment (IDE)	Ολοκληρωμένο Περιβάλλον Ανάπτυξης
Internet Access	Σύνδεση στο Διαδίκτυο
Isolation	Απομόνωση
Look-up Table	Πίνακας Αναφοράς
Model	Μοντέλο
Open Source	Ανοικτού Κώδικα
Operating System	Λειτουργικό Σύστημα
Profile	Προφίλ - λογαριασμός
Relational Database	Σχεσιακή Βάση Δεδομένων
Scalability	Επεκτασιμότητα
Search Engine	Μηχανή Αναζήτησης
Serverless	Ανεξάρτητη Εξυπηρετητή
Software Development Kit (SDK)	Πακέτο Ανάπτυξης Λογισμικού
Validation	Επικύρωση
Verification	Επαλήθευση
View	Παρουσίαση
Wireframe	Αρχική Αποτύπωση Διεπαφής



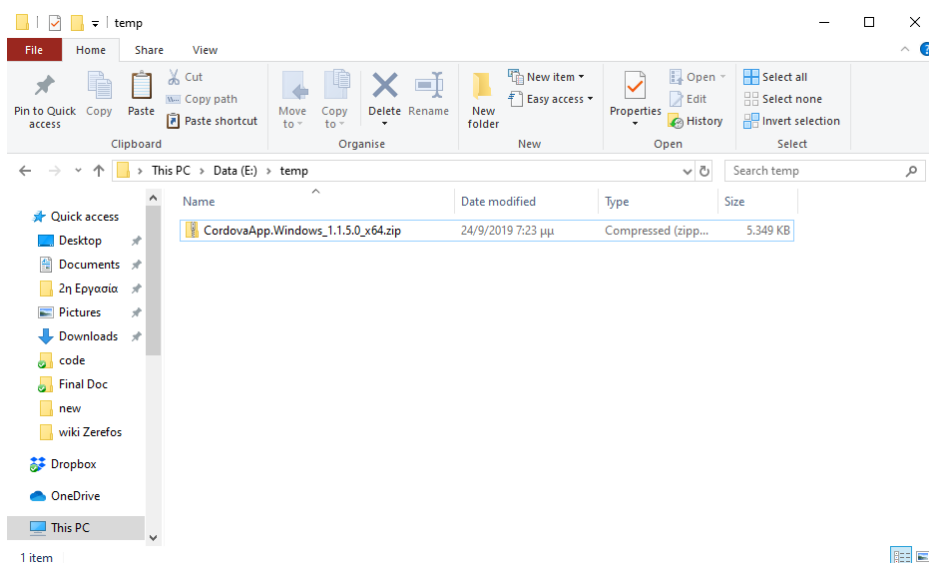
## ΣΥΝΤΜΗΣΕΙΣ – ΑΡΚΤΙΚΟΛΕΞΑ – ΑΚΡΩΝΥΜΙΑ

ACID	Atomicity, Consistency, Isolation, Durability
JRE	Java Runtime Environment
IDE	Integrated Development Environment
ITU	International Telecommunication Union
LUT	Look-up Table
MSV	Microsoft Visual Studio
MVC	Model-View-Controller
SDK	Software Development Kit
ΔΟΣ	Διάγραμμα Οντοτήτων Συσχετίσεων

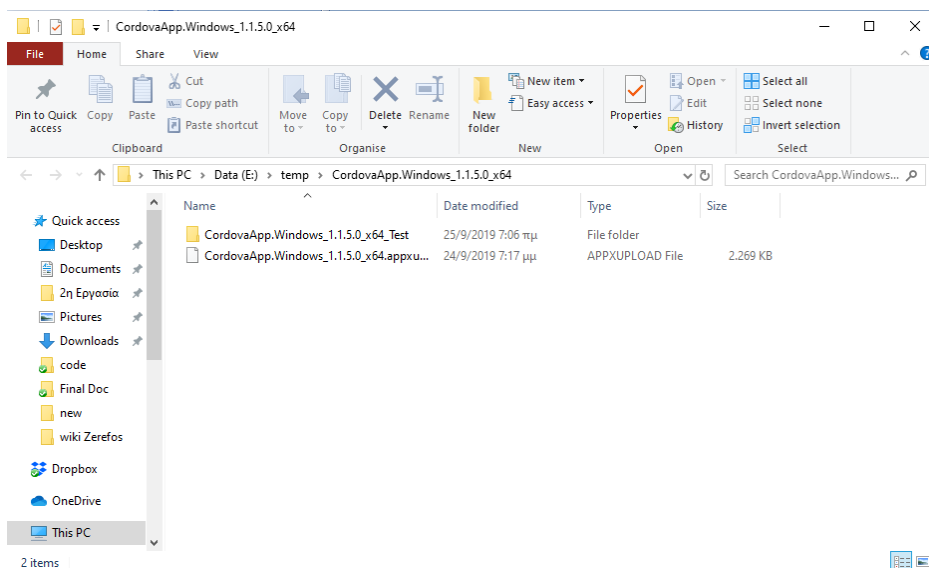
## ΠΑΡΑΡΤΗΜΑ Ι - Οδηγίες Εγκατάστασης

Στο παρόν παράρτημα παρουσιάζεται ο τρόπος εγκατάστασης της εφαρμογής Tai Yi σε έναν υπολογιστή που λειτουργεί με Microsoft Windows 10 (Version 1903, OS Build 18362.295), εμπλουτισμένος με εικόνες από τη διαδικασία. Ο υπολογιστής από τον οποίον πάρθηκαν οι εικόνες λειτουργούσε στα Αγγλικά, οπότε και οι εικόνες εμφανίζουν όλα τα μενού σε αυτή τη γλώσσα. Παρακάτω ωστόσο παρουσιάζονται οι επιλογές και στα Ελληνικά.

Τα αρχεία εγκατάστασης της εφαρμογής δίνονται συμπιεσμένα σε ένα αρχείο όπως φαίνεται στην παρακάτω εικόνα.

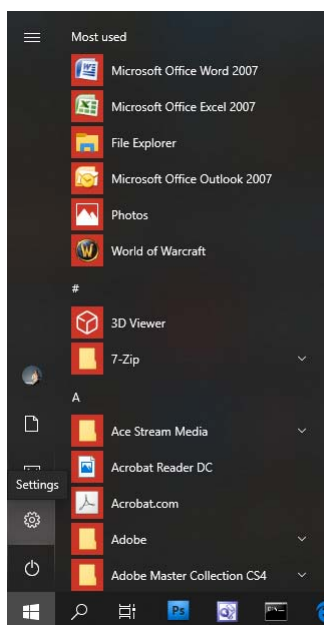


Ο χρήστης αρχικά πρέπει να τα αποσυμπιέσει όπως φαίνεται στην παρακάτω εικόνα και να ακολουθήσει τα βήματα που παρουσιάζονται στη συνέχεια για να εγκαταστήσει την εφαρμογή Tai Yi.



1. Το πρώτο στάδιο αφορά την ενεργοποίηση της δυνατότητας χρήσης του υπολογιστή για ανάπτυξη (development). Τα επόμενα βήματα καθορίζουν τη διαδικασία ενεργοποίησης αυτής της δυνατότητας σε περιβάλλον Microsoft Windows 10.

- 1.1. Αρχικά επιλέγονται οι ρυθμίσεις (settings) του υπολογιστή όπως φαίνεται στην παρακάτω εικόνα. Στην επόμενη εικόνα εμφανίζεται το μενού των ρυθμίσεων.

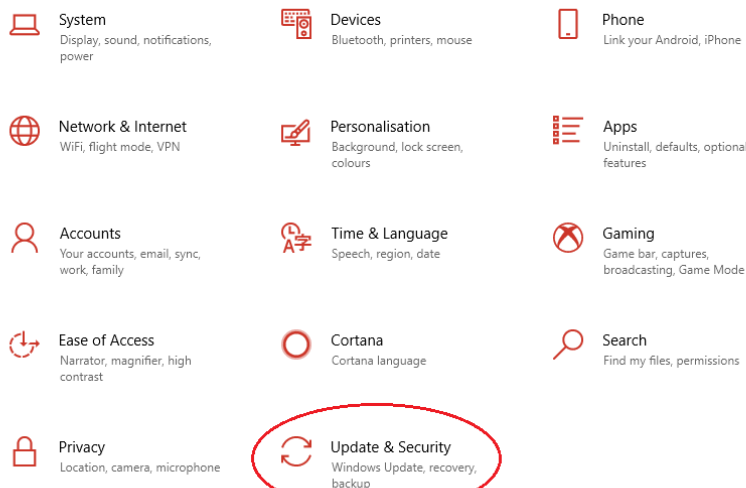


Settings

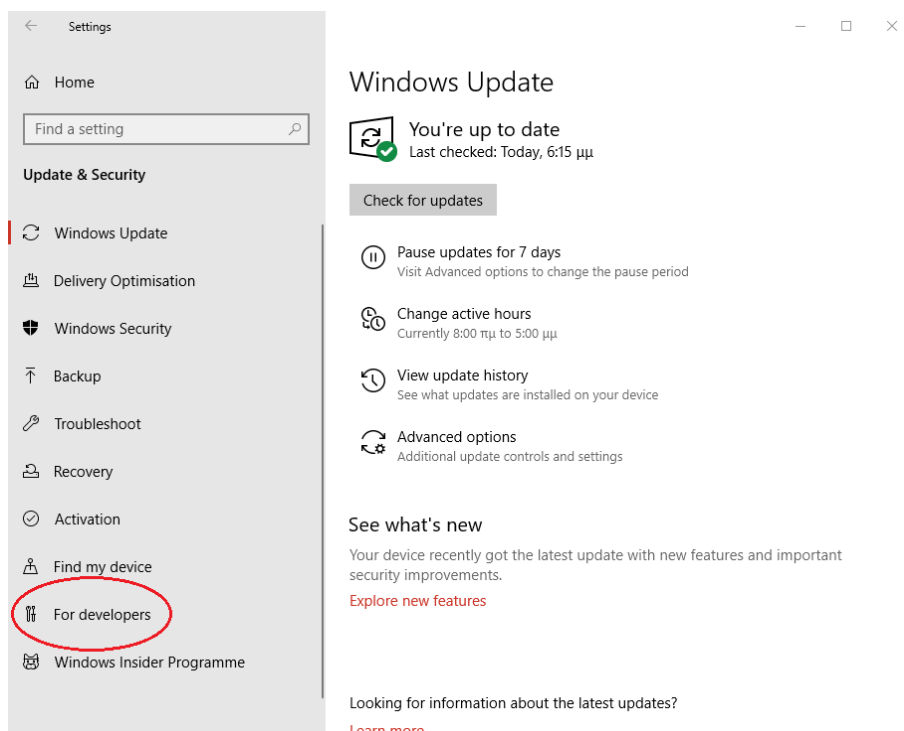
— □ ×

## Windows Settings

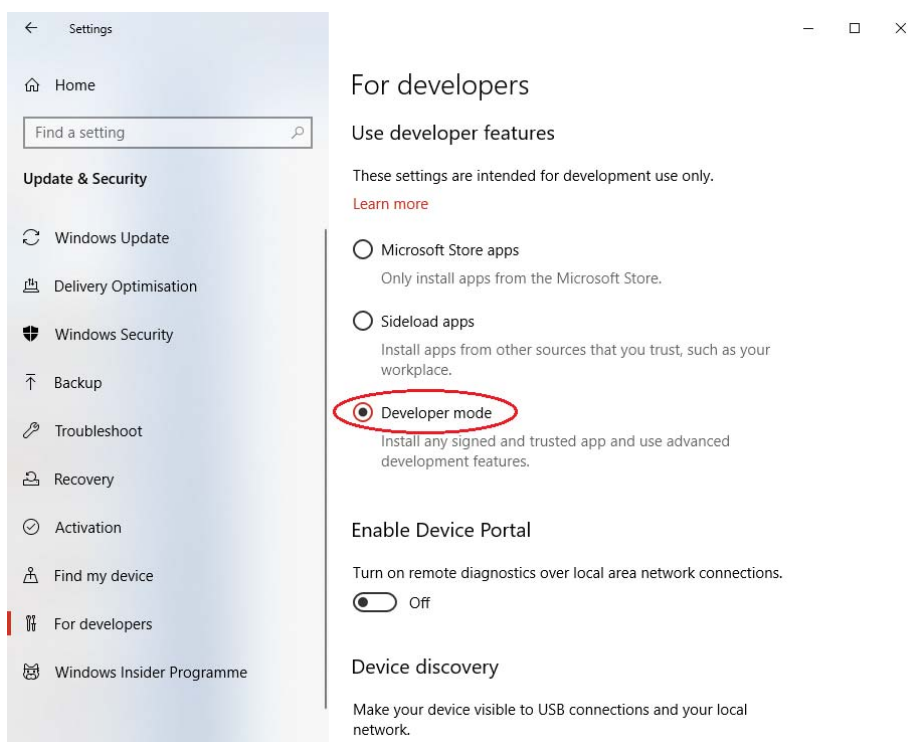
Find a setting



- 1.2. Στη συνέχεια από τις ρυθμίσεις επιλέγεται το «Ενημερώσεις και Ασφάλεια» (Update & Security) που βρίσκεται στο σημείο που έχει μαρκαριστεί στην προηγούμενη εικόνα και εμφανίζεται το μενού της επόμενης εικόνας.



1.3. Στο μενού αυτό επιλέγεται το «Για προγραμματιστές» (For developers) στο σημείο που έχει μαρκιαριστεί και εμφανίζονται οι επιλογές που παρουσιάζονται στην επόμενη εικόνα.

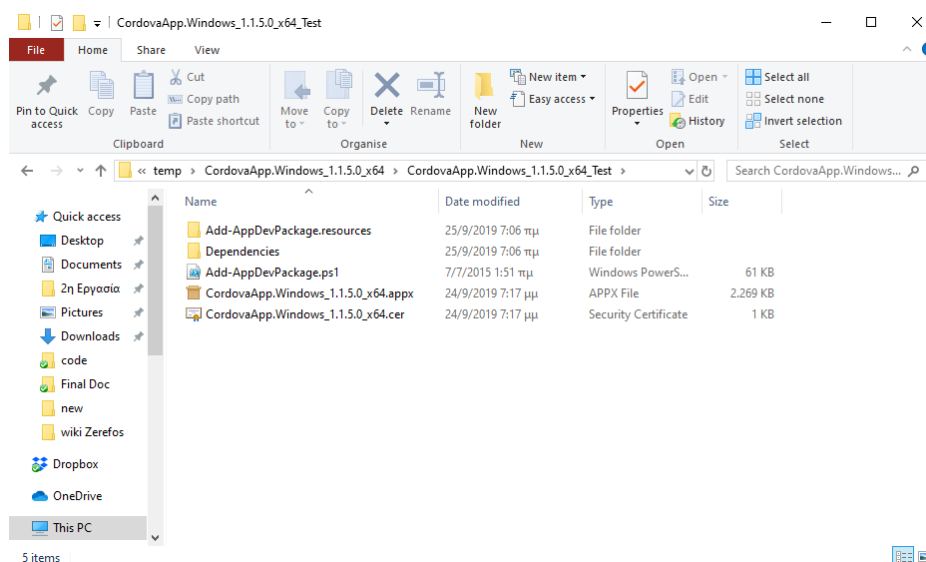


1.4. Επιλέγεται το «Λειτουργία Προγραμματιστή» (Developer mode), για την ενεργοποίηση της οποίας απαιτείται κάποια αναμονή. Μόλις ενεργοποιηθεί η

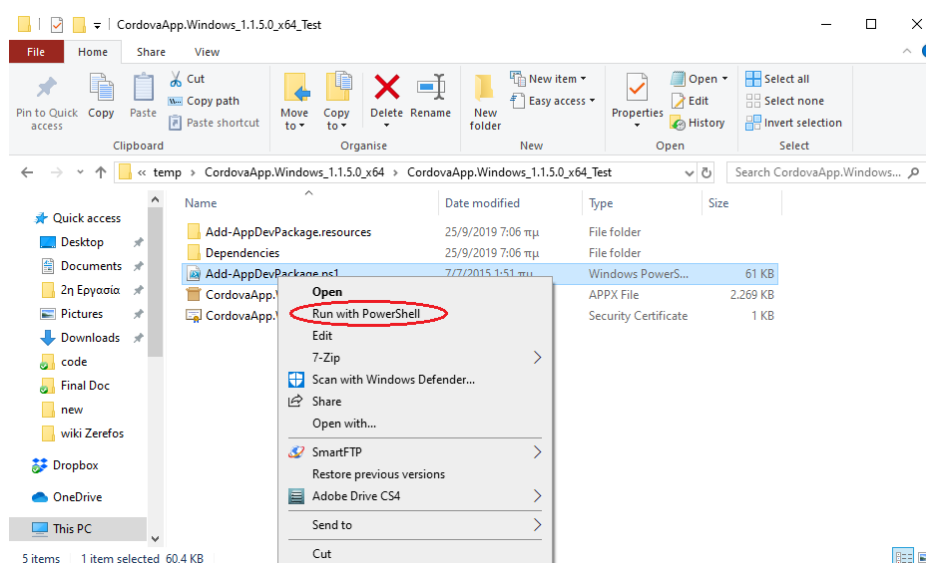
συγκεκριμένη λειτουργία, είναι πλέον έτοιμος ο υπολογιστής για να εγκατασταθεί η εφαρμογή Tai Yi.

2. Το επόμενο στάδιο είναι η εγκατάσταση της εφαρμογής και ακολουθούνται τα παρακάτω βήματα.

2.1. Αρχικά στο χώρο που έγινε η αποσυμπίεση και εμφανίζεται ένας φάκελος και ένα αρχείο «.appxupload», ο χρήστης ανοίγει το φάκελο και βλέπει τα αρχεία που εμφανίζονται στην παρακάτω εικόνα.

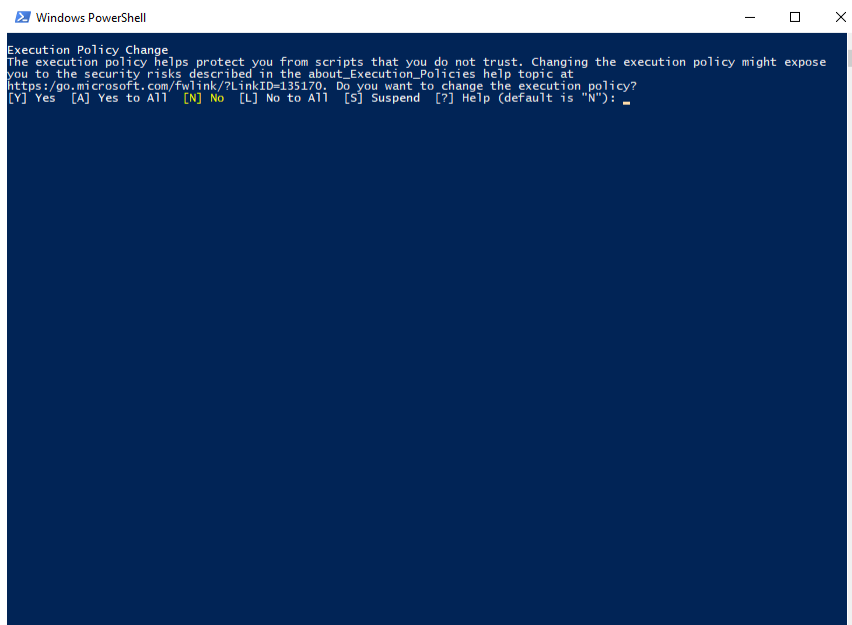


2.2. Στη συνέχεια επιλέγει το αρχείο με κατάληξη «.ps1» και κάνοντας δεξί κλικ πάνω του εμφανίζεται ένα μενού. Από το μενού επιλέγεται το «Εκτέλεση με PowerShell» (Run with PowerShell), όπως φαίνεται στην παρακάτω εικόνα στην οποία έχει μαρκαριστεί η επιλογή.

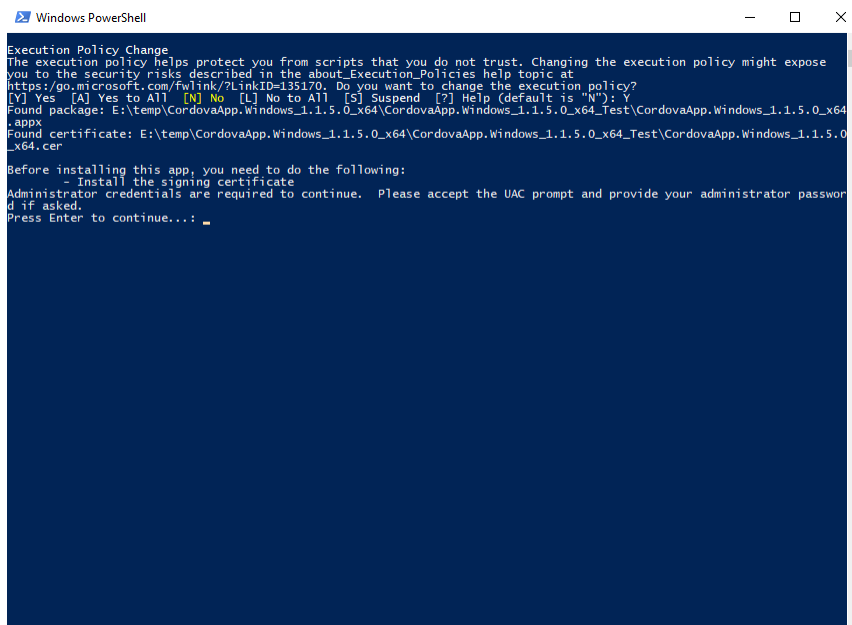


2.3. Στο σημείο αυτό εκτελείται το PowerShell όπως φαίνεται στην παρακάτω εικόνα και εμφανίζεται το μήνυμα: «Do you want to change the execution policy?», στο

οποίο επιλέγεται το «Yes». Το συγκεκριμένο μήνυμα ρωτάει αν θέλουμε να αλλάξουμε την πολιτική του λειτουργικού συστήματος για τα εκτελέσιμα αρχεία και είναι απαραίτητο να δεχθούμε για να συνεχίσει η εγκατάσταση, απλά δεν πρέπει να επιλέξουμε «Yes to All».

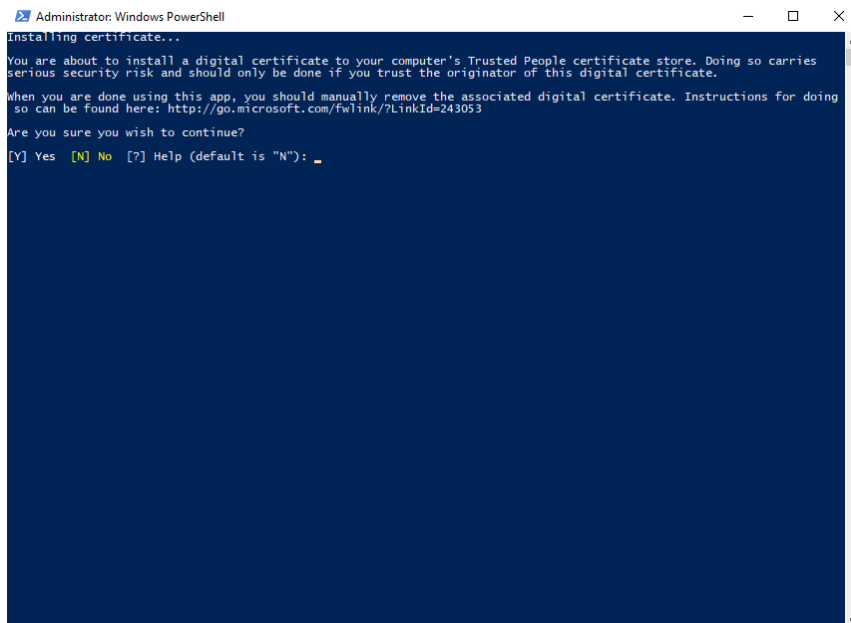


2.4. Στην συνέχεια ενημερώνετε ο χρήστης ότι μπορεί να ζητηθούν τα διαπιστευτήρια του διαχειριστή (administrator password) για την ολοκλήρωση της εγκατάστασης και περιμένει είσοδο από τον χρήστη όπως φαίνεται στην επόμενη εικόνα.

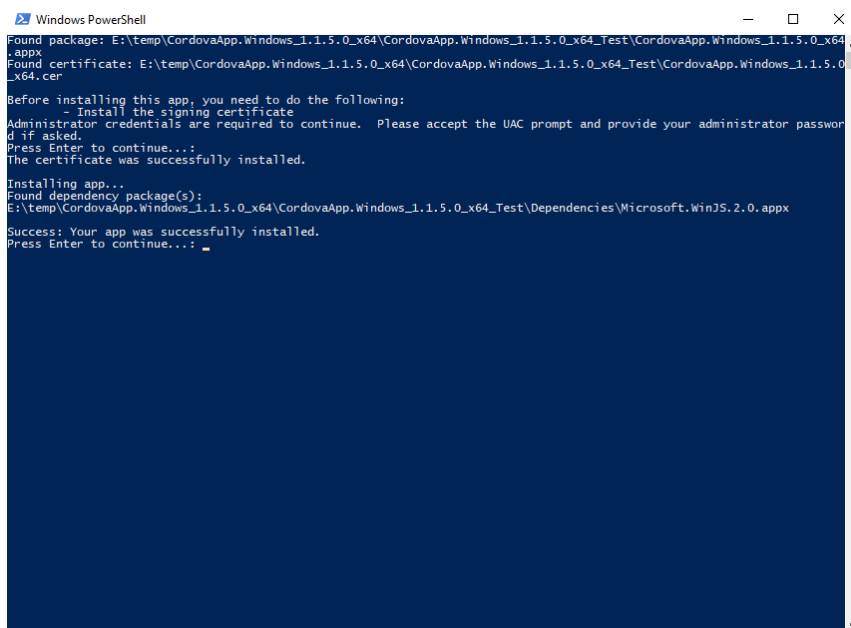


2.5. Στο σημείο αυτό γίνεται η εγκατάσταση του πιστοποιητικού της εφαρμογής και αν είναι ενεργοποιημένο το «User Account Control» (που συνήθως είναι),

εμφανίζεται η ερώτηση «Do you want to allow this app to make changes to your device?» για την εφαρμογή «Microsoft PowerShell». Επιλέγει η χρήστης το «Yes» και ανοίγει άλλο παράθυρο του «PowerShell» για την εγκατάσταση του πιστοποιητικού που ζητά επιβεβαίωση για συνέχεια όπως φαίνεται στην παρακάτω εικόνα.



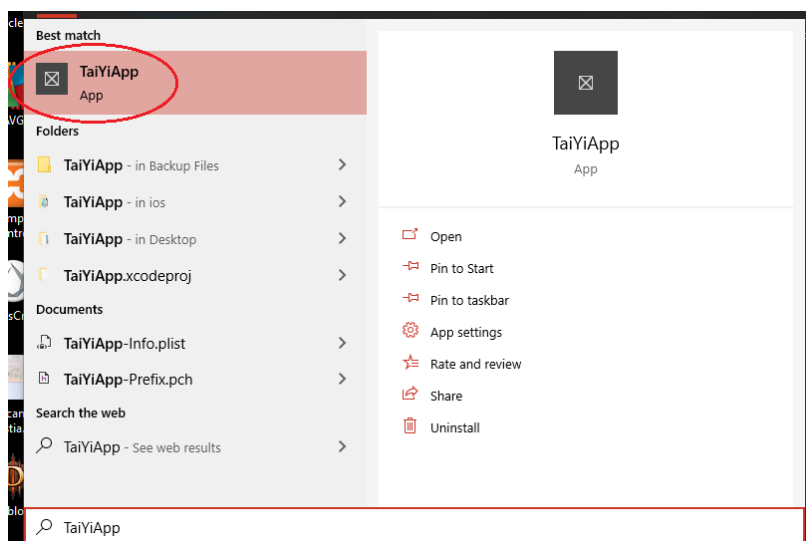
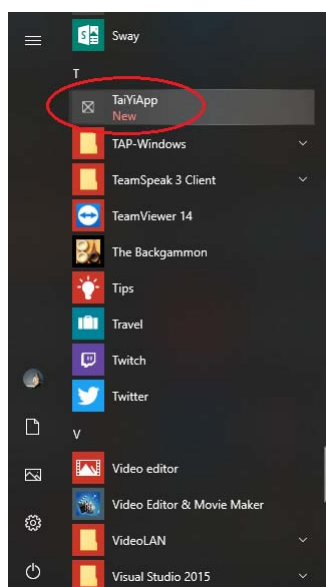
2.6. Με την ολοκλήρωση της εγκατάστασης εμφανίζεται το μήνυμα «Your app was successfully installed.» όπως δίνεται στην παρακάτω εικόνα και η εφαρμογή Tai Yi είναι πλέον εγκατεστημένη στον υπολογιστή.



Ακολουθούν στο επόμενο παράρτημα αναλυτικές οδηγίες χρήσης της εφαρμογής. Το όνομα της εφαρμογής στο μενού είναι «TaiYiApp».

## ΠΑΡΑΡΤΗΜΑ II - Οδηγίες Χρήσης

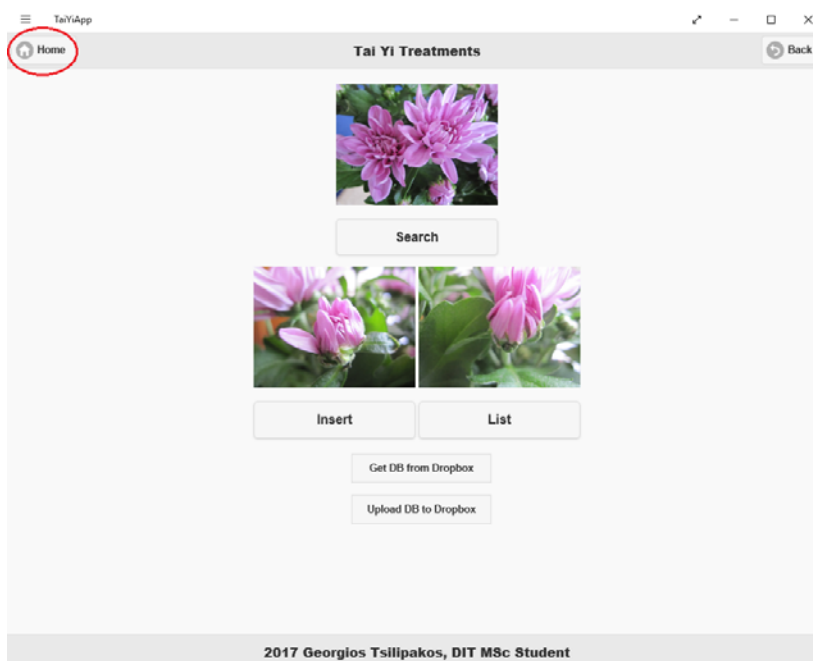
Στο παράρτημα αυτό παρουσιάζεται αναλυτικά ο τρόπος χρήσης της εφαρμογής Tai Yi, αφού έχει εγκατασταθεί σε περιβάλλον Microsoft Windows 10 (Version 1903, OS Build 18362.295). Το όνομα της εφαρμογής στο μενού του λειτουργικού είναι «TaiYiApp». Η εκτέλεση της εφαρμογής ξεκινάει είτε επιλέγοντάς την από το μενού των εφαρμογών είτε αναζητώντας την εφαρμογή στην αναζήτηση. Οι δύο αυτές περιπτώσεις εμφανίζονται στις παρακάτω εικόνες.



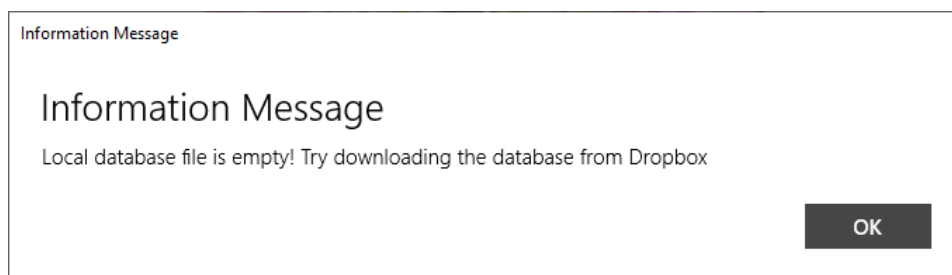
Στην συνέχεια παρουσιάζονται οι λειτουργίες της εφαρμογής για κάθε μία από τις επιλογές που δίνονται στους χρήστες.

Η αρχική σελίδα της εφαρμογής δίνεται παρακάτω. Στη σελίδα αυτή επιστρέφουν οι χρήστες και όποτε επιλέξουν το «Home» στο πάνω αριστερό μέρος της διεπαφής της εφαρμογής (έχει μαρκαριστεί στην επόμενη εικόνα).

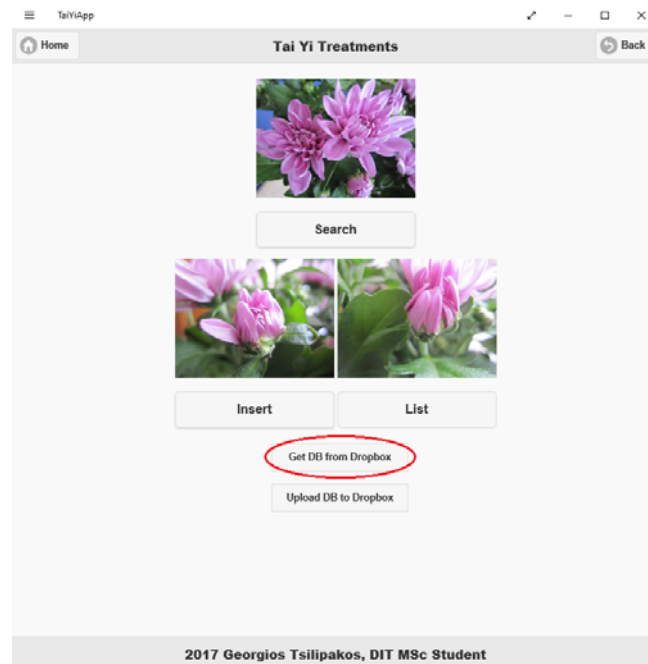




Όταν ολοκληρώνεται η αρχική εκτέλεση της εφαρμογής, εμφανίζεται ένα μήνυμα. Το μήνυμα αυτό είτε είναι το «Local database file loaded successfully!», που σημαίνει ότι η εφαρμογή εντόπισε το τοπικό αρχείο της βάσης δεδομένων και είναι έτοιμη να λειτουργήσει, είτε είναι το «Local database file is empty! Try downloading the database from Dropbox», το οποίο σημαίνει ότι η εφαρμογή δεν εντόπισε τοπικό αρχείο και πρέπει να πραγματοποιηθεί λήψη του αρχείου. Σημειώνεται εδώ ότι δεν μπορεί να γίνει λήψη του αρχείου χωρίς σύνδεση στο διαδίκτυο. Ένα παράδειγμα μηνύματος εμφανίζεται στην επόμενη εικόνα.

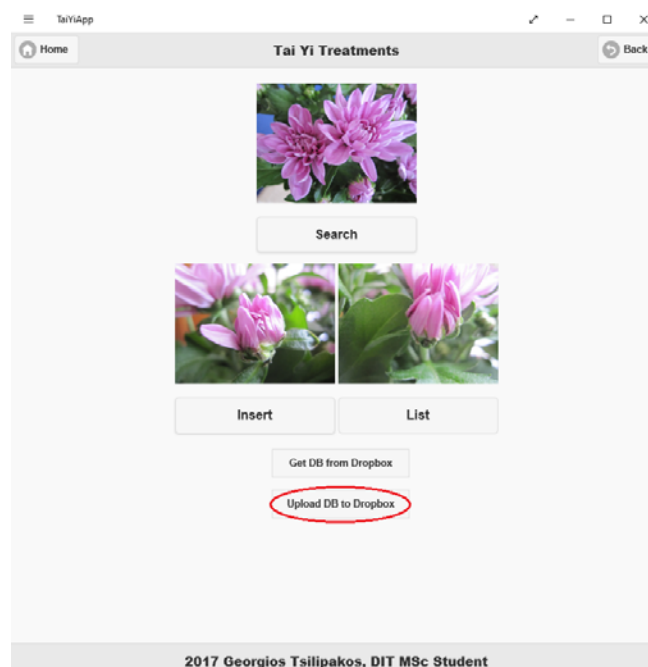


1. Η πρώτη επιλογή που οφείλουν μάλιστα να κάνουν οι χρήστες αν δεν έχει εντοπιστεί τοπικό αρχείο της βάσης δεδομένων, είναι να πραγματοποιήσουν λήψη του αρχείου. Αυτό επιτυγχάνεται επιλέγοντας «Get DB from Dropbox», το οποίο φαίνεται μαρκαρισμένο στην επόμενη σελίδα.



Με την ολοκλήρωση της λήψης του αρχείου εμφανίζεται το μήνυμα «Database successfully downloaded at enjoyinglife.sqlite». Αν κάτι δεν πάει καλά στη διαδικασία λήψης ενημερώνονται οι χρήστες.

2. Μία επίσης επιλογή σε σχέση με το αρχείο της βάσης δεδομένων, είναι η αποστολή του τοπικού αρχείου της βάσης στο Dropbox και πραγματοποιείται με την επιλογή «Upload DB to Dropbox», το οποίο έχει μαρκαριστεί στην επόμενη σελίδα.

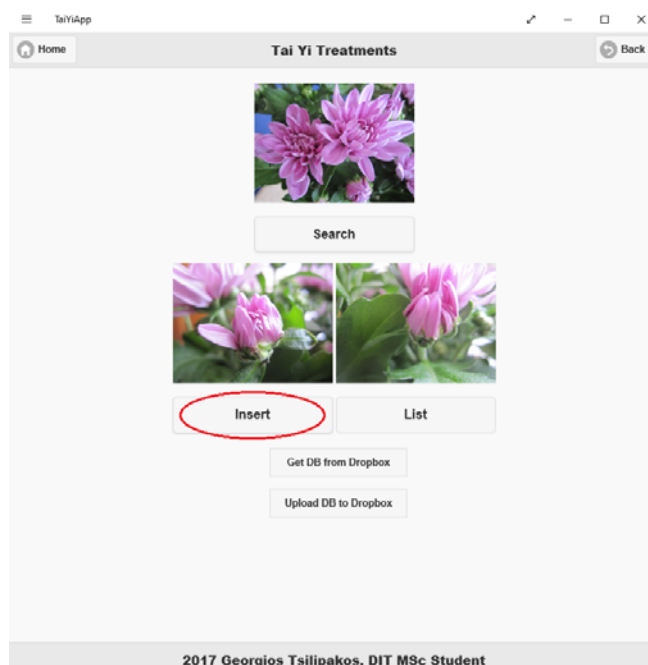


Το μήνυμα «Database successfully uploaded!» εμφανίζεται με την επιτυχή ολοκλήρωση της αποστολής του αρχείου, διαφορετικά ενημερώνονται οι χρήστες για την αποτυχία.

3. Στο σημείο αυτό θεωρείται σκόπιμο να παρουσιαστεί το μενού της εφαρμογής όταν οι χρήστες βρίσκονται σε οποιαδήποτε σελίδα της (πέραν της αρχικής). Το μενού αυτό εμφανίζεται στην επόμενη εικόνα και έχει τις παρακάτω επιλογές, εκ των οποίων το «Home» έχει εξηγηθεί ήδη.



- 3.1. «Search»: Κάνοντας αυτή την επιλογή οι χρήστες μεταφέρονται στη σελίδα αναζήτησης θεραπειών που παρουσιάζεται παρακάτω στην παράγραφο 8.
- 3.2. «Insert»: Επιλέγοντας αυτό, οι χρήστες ανοίγουν τη σελίδα για την εισαγωγή μίας νέας θεραπείας στην εφαρμογή. Ο τρόπος εισαγωγής παρουσιάζεται στην επόμενη παράγραφο (4).
- 3.3. «List»: Αυτή η επιλογή οδηγεί τους χρήστες στην λίστα με το σύνολο των θεραπειών που έχουν καταχωρηθεί στο αρχείο της βάσης που έχει ανοίξει η εφαρμογή και παρουσιάζεται παρακάτω στη παράγραφο 5.
- 3.4. «Back»: Σε οποιοδήποτε σημείο κάνουν οι χρήστες αυτήν την επιλογή μεταφέρονται στην προηγούμενη σελίδα που βρίσκονταν.
4. Η εισαγωγή θεραπείας είναι μία δυνατότητα που δίνεται για την εισαγωγή των στοιχείων μίας νέας θεραπείας στη βάση δεδομένων και γίνεται με την επιλογή «Insert» που είναι μαρκαρισμένη στην επόμενη εικόνα.



Εδώ υπάρχει η δυνατότητα εισαγωγής όλων των πληροφοριών μίας θεραπείας, οι οποίες είναι ομαδοποιημένες για να μπορούν οι χρήστες να εμφανίζουν ή να αποκρύπτουν τις ομάδες γνωρισμάτων που τους ενδιαφέρουν. Αρχικά η ομάδα των

βασικών γνωρισμάτων (Basic Properties) εμφανίζεται αναλυτικά και οι επόμενες ομάδες όχι. Παρουσιάζεται λοιπόν παρακάτω ο τρόπος εισαγωγής των γνωρισμάτων ανά ομάδα.

- 4.1.Βασικές πληροφορίες («Basic Properties»). Εδώ εισάγονται τα γνωρίσματα «Treatment Name», «Treatment Duration», «Date Taught», «Treatment Description», «Treatment Short Description (Greek)» και «Treatment Short Description (English)» σαν κείμενο όπως φαίνεται στην επόμενη εικόνα.

The screenshot shows a web application titled "Tai Yi Treatments". At the top, there is a navigation bar with "Home", "Search", "Insert", and "List" buttons. Below this, the "Basic Properties" section is active, showing input fields for "Treatment Name", "Treatment Duration", and "Date Taught". Below these are two expandable sections: "Treatment in Parts" and "Type of Treatment". The "Type of Treatment" section has a button labeled "Update Types of Treatments". Below that are input fields for "Treatment Description", "Treatment Short Description (Greek)", and "Treatment Short Description (English)". At the bottom, there is an "Issues Addressed" section. The footer of the application reads "2017 Georgios Tsilipakos, DIT MSc Student".

Το γνώρισμα «Treatment in Parts» έχει διαφορετικό τρόπο εισαγωγής που δίνεται στην επόμενη εικόνα και οι πληροφορίες «Number of Parts», «Min Interval», «Max Interval» και «Best Interval» εισάγονται σαν κείμενο.

The screenshot shows a form titled "Treatment in Parts". It contains input fields for "Number of Parts", "Parts Interval Metric" (which has a dropdown menu currently showing "null"), "Min Interval", "Max Interval", and "Best Interval".

Το γνώρισμα «Parts Interval Metric» εισάγεται από λίστα τιμών όπως φαίνεται στην παρακάτω εικόνα. Αυτή είναι η μορφή όλων των λιστών που συναντούν οι χρήστες της εφαρμογής.

Parts Interval Metric:	null
Min Interval:	Days
Max Interval:	Hours
Best Interval:	Months
	Weeks
	Years

Η επιλογή του γνωρίσματος «Type of Treatment», που μπορεί να έχει πολλαπλές τιμές, γίνεται σε λίστα πολλαπλών τιμών, όπου σημειώνονται οι επιλογές των χρηστών. Η λίστα δίνεται στην επόμενη εικόνα με ένα παράδειγμα επιλεγμένων τιμών.

Type of Treatment

☒ Counterbalancing

☐ Harmonizing

☒ Purging

☐ Reinforcing

4.2.Θέματα που αντιμετωπίζονται («Issues Addressed»). Εδώ εισάγονται οι πληροφορίες για τα θέματα που αντιμετωπίζει μία θεραπεία και πραγματοποιείται όπως φαίνεται στην παρακάτω εικόνα και περιγράφεται παρακάτω.

TaiYiApp

Tai Yi Treatments

Home Search Insert List Back

Issues Addressed

Rank	Issue	Body	Effectiveness	Control
-	-	-	-	-

Issue: null

Update Issues

Aspects of Disease:

☐ Spiritual

☐ Mental

☐ Emotional

☐ Physical

☐ All

Effectiveness: null

Add

Remove

Details

2017 Georgios Tsilipakos, DIT MSc Student

Επιλέγεται ουσιαστικά από λίστα τιμών οι τιμές των «Issue», «Effectiveness» και μία ή πολλές εκ των τιμών «Aspects of Disease». Στη συνέχεια οι χρήστες μπορούν να προσθέσουν αυτή την καταχώρηση επιλέγοντας «Add». Όταν έχει γίνει η καταχώρηση εμφανίζεται στον πίνακα στην αρχή της ομάδας «Issues

Addressed» όπως δίνεται στην παρακάτω εικόνα, και μπορεί να κάνουν επιλογή συγκεκριμένης καταχώρησης (επιλέγοντας το «Control» που έχει μαρκαριστεί στην επόμενη εικόνα) και να επιλέξουν να τη διαγράψουν με την επιλογή «Remove».

Rank	Issue	Body	Effectiveness	Control
1	Lack of Passion	Mental	Secondary 3	<input checked="" type="checkbox"/>
2	Lack of Passion	Emotional	Secondary 3	<input type="checkbox"/>

Issue:

Aspects of Disease:

- ☐ Spiritual
- ☒ Mental
- ☒ Emotional
- ☐ Physical
- ☐ All

Effectiveness:

2017 Georgios Tsilipakos, DIT MSc Student

Σημειώνεται εδώ ότι παρέχεται η επιλογή για τη διαχείριση του γνωρίσματος «Issue» στο σημείο αυτό. Αν οι χρήστες επιλέξουν «Update Issue» μεταφέρονται στην παρακάτω σελίδα που μπορεί να κάνει διαχείριση. Η διαχείριση αυτή περιγράφεται στη συνέχεια στη παράγραφο 9.

Issue ID:

Issue Name:

Issue Description:

Issues List

- 1. Compression >
- 2. Compression 3 >
- 3. Denial of Expression >
- 4. Empowering the Lower Bodies vs. the Higher Bodies >
- 5. Lack of Passion >
- 6. Non-Acceptance of Path >

2017 Georgios Tsilipakos, DIT MSc Student

4.3.Λεπτομέρειες («Details»). Εδώ εισάγονται τα γνωρίσματα «Treatment Limitations», «Treatment Specific Timing» και «Gauging Resistance» σαν κείμενο και τα γνωρίσματα «Treatment Body Area», «Start with» και «Treatment Series Type» από λίστα τιμών.

The screenshot displays the 'Tai Yi Treatments' application interface. At the top, there's a navigation bar with 'Home', 'Tai Yi Treatments', and 'Back' buttons. Below this is a search bar and three tabs: 'Search', 'Insert', and 'List'. The main content area is titled 'Issues Addressed' and contains a 'Details' section. This section has several input fields: 'Treatment Body Area' (a dropdown menu showing 'null'), 'Start with' (a dropdown menu showing 'null'), 'Treatment Limitations' (a text input field), 'Treatment Specific Timing' (a text input field), 'Treatment Series Type' (a dropdown menu showing 'null'), and 'Gauging Resistance' (a text input field). Below these fields are two more sections: 'Treatment Special Categories' with an 'Update Special Categories' button, and 'Treatment Relations'. The footer of the application shows '2017 Georgios Tsilipakos, DIT MSc Student'.

Επίσης επιλέγονται τιμές και για το γνώρισμα «Treatment Special Categories» που μπορεί να είναι περισσότερες από μία. Η προηγούμενη εικόνα εμφανίζει τις λεπτομέρειες. Στο σημείο αυτό με την επιλογή «Update Special Categories» ο χρήστης μπορεί να διαχειριστεί τις περιφερειακές πληροφορίες για το γνώρισμα αυτό, όπως παρουσιάζεται στην παράγραφο 9 και φαίνεται στην επόμενη εικόνα.

The screenshot displays the 'Tai Yi Treatments' application interface for managing 'Special Categories'. At the top, there's a navigation bar with 'Home', 'Tai Yi Treatments', and 'Back' buttons. Below this is a section titled 'Special Categories' with three input fields: 'Special Category ID', 'Special Category Name', and 'Special Category Description'. Below these fields are four buttons: 'Insert', 'Update', 'Delete', and 'Clear'. Below the buttons is a section titled 'Special Categories List' which contains a list of 8 items, each with a right-pointing arrow icon. The footer of the application shows '2017 Georgios Tsilipakos, DIT MSc Student'.

4.4. Σύνδεση θεραπειών («Treatment Relations»). Εδώ καταχωρούνται οι συνδέσεις ανάμεσα στις θεραπείες στο τμήμα της διεπαφής που δίνεται στην επόμενη εικόνα. Είναι σημαντικό να κατανοήσουν οι χρήστες ότι υπάρχουν δύο τύποι συνδέσεων. Μπορούν να συνδέσουν μία θεραπεία που προηγείται από αυτήν που προσπαθούν να εισάγουν (οι σχέσεις αυτές φαίνονται στον δεύτερο πίνακα που εμφανίζεται) και μπορούν να συνδέσουν και μία θεραπεία που έπεται από αυτήν που προσπαθούν να εισάγουν (οι σχέσεις αυτές φαίνονται στον πρώτο πίνακα που εμφανίζεται). Αυτό καθορίζεται από το τι θα επιλέξουν από τη λίστα τιμών «Position of Linked Treatment». Αν επιλέξουν «Next» εισάγουν μία θεραπεία (που είναι ήδη καταχωρημένη στη βάση δεδομένων) που έπεται αυτής που προσπαθούν να καταχωρήσουν. Αν επιλέξουν «Previous» εισάγουν μία θεραπεία που προηγείται αυτής που προσπαθούν να καταχωρήσουν.

Tai Yi Treatments

Search Insert List

Treatment Relations

Treatments to follow this one:

Rank	Treatment	Relation	Relation Description	Control
-	-	-	-	-

Treatments to be done before this one:

Rank	Treatment	Relation	Relation Description	Control
-	-	-	-	-

Position of linked Treatment: Next

Treatment: null

Relation: Mandatory

Relation Description:

Add Remove

2017 Georgios Tsilipakos, DIT MSc Student

Τα υπόλοιπα γνωρίσματα που επιλέγονται εδώ είναι το «Relation Description» που είναι κείμενο και «Treatment» καθώς και «Relation» που επιλέγονται από λίστα τιμών. Στην παρακάτω εικόνα δίνεται ένα παράδειγμα καταχωρημένων σχέσεων θεραπειών. Όπως και πριν οι χρήστες όταν συμπληρώσουν τις απαιτούμενες πληροφορίες μπορούν να επιλέξουν «Add» για να προσθέσουν τη σχέση και να μαρκάρουν το «Control» για να επιλέξουν «Remove» και να αφαιρέσουν μία σχέση.



4.5.Λεπτομέρειες μετά τη θεραπεία («Post Details»). Στην ομάδα αυτή εισάγονται τα γνωρίσματα «Possible Negative Effects», «Treatment Integration Period», «Recommended Actions for Recipient» και «Treatment Timing Post» σαν κείμενο. Εμφανίζονται στην επόμενη εικόνα.

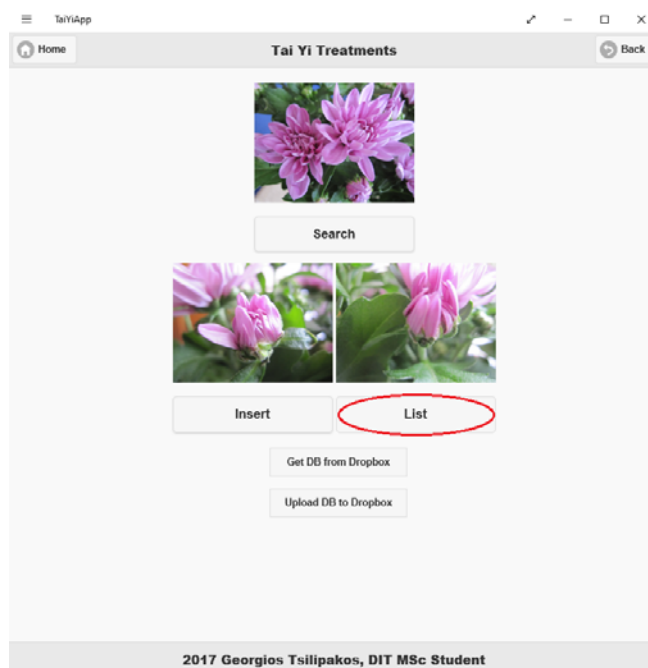
4.6.Απαιτήσεις («Requirements»). Εδώ εισάγονται τα γνωρίσματα «Treatment Requirements» και «Minimum Energy Requirements» σαν κείμενο και το γνώρισμα «Energy Types» σαν πολλαπλή επιλογή. Επιλέγοντας «Update Energy Types» μπορούν οι χρήστες να διαχειριστούν τις περιφερειακές πληροφορίες για το γνώρισμα αυτό σύμφωνα με τις οδηγίες που παρέχονται στην παράγραφο 9 όπως φαίνεται στην επόμενη εικόνα.

The screenshot shows the 'Tai Yi Treatments' app interface. At the top, there's a header with 'Home' and 'Back' buttons. Below the header, the title 'Tai Yi Treatments' is centered. The main section is titled 'Energy Types' and contains three input fields: 'Energy Type ID:', 'Energy Type Name:', and 'Energy Type Description:'. Below these fields are four buttons: 'Insert', 'Update', 'Delete', and 'Clear'. Underneath the buttons is a section titled 'Energy Types List' which displays a list of five items: '1. Energy Type 1', '2. Energy Type 3', '3. Energy Type 4', and '4. Energy Type 5'. Each item has a right-pointing arrow. At the bottom of the screen, there is a footer that reads '2017 Georgios Tsilipakos, DIT MSc Student'.

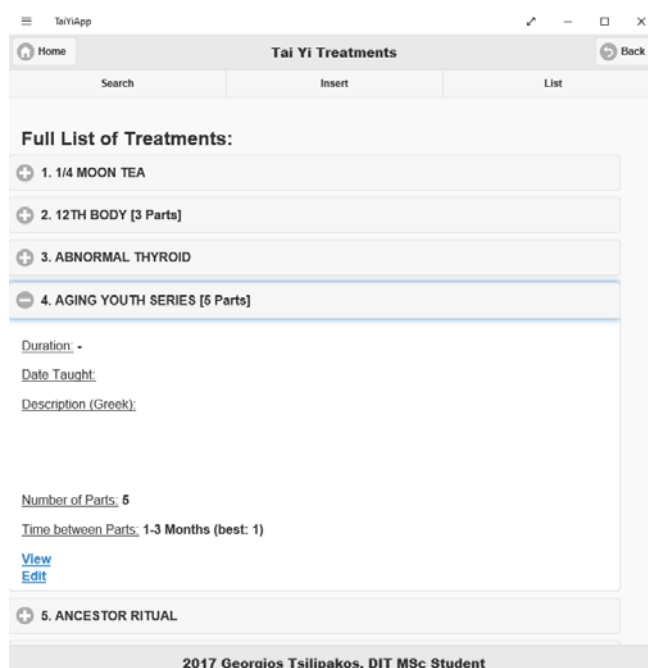
The screenshot shows the 'Tai Yi Treatments' app interface with the 'Requirements' section expanded. The header is the same as the previous screenshot. Below the header, there are three tabs: 'Search', 'Insert', and 'List'. The 'Insert' tab is selected. The main section is titled 'Requirements' and contains several input fields: 'Treatment Requirements:', 'Energy Types' (with a plus icon), 'Update Energy Types' (a button), 'Minimum Energy Requirements:', and 'Insert' (a button circled in red). At the bottom of the screen, there is a footer that reads '2017 Georgios Tsilipakos, DIT MSc Student'.

Η ομάδα αυτή γνωρισμάτων (Απαιτήσεις) εμφανίζεται στην παραπάνω εικόνα.

Με την ολοκλήρωση της εισαγωγής των πληροφοριών που επιθυμούν οι χρήστες, επιλέγουν «Insert» και αναμένουν να λάβουν μήνυμα επιτυχούς καταχώρησης της νέας θεραπείας (η επιλογή φαίνεται μαρκαρισμένη στην παραπάνω εικόνα). Αν υπάρξει οποιοδήποτε πρόβλημα στη διαδικασία οι χρήστες ενημερώνονται με μήνυμα από την εφαρμογή και επιστρέφουν στην αρχική οθόνη της εφαρμογής. Να σημειωθεί εδώ ότι υπάρχει δυνατότητα να καταχωρηθεί θεραπεία μόνο με το όνομά της και να επεξεργαστούν οι χρήστες τα δεδομένα της αργότερα.



5. Επιλέγοντας «List» (εμφανίζεται μαρκαρισμένη επιλογή στην παραπάνω εικόνα) παρουσιάζεται μία λίστα με το σύνολο των θεραπειών που υπάρχουν στη βάση δεδομένων. Η εικόνα της λίστας δίνεται στην παρακάτω εικόνα. Κάθε μία θεραπεία περιέχει κάποια γνωρίσματα που μπορούν να δουν άμεσα οι χρήστες και είναι τα «Treatment Duration», «Date Taught», «Treatment Short Description Greek» και «Treatment Parts» και λειτουργούν όπως οι ομάδες, δηλαδή μπορεί να εμφανίζονται ή να αποκρύπτονται. Μαζί με τα γνωρίσματα αυτά υπάρχει και η δυνατότητα επιλογής περαιτέρω ενεργειών. Επιλέγοντας «View» μπορούν να δουν το σύνολο των πληροφοριών που υπάρχει στη βάση δεδομένων για τη συγκεκριμένη θεραπεία όπως περιγράφεται στην παράγραφο 6, ενώ επιλέγοντας «Edit» μπορούν οι χρήστες να επεξεργαστούν τα δεδομένα της θεραπείας όπως περιγράφεται στην παράγραφο 7.



6. Στην προηγούμενη παράγραφο παρουσιάστηκε η δυνατότητα στη λίστα των θεραπειών να επιλέξουν οι χρήστες «View» και να μεταβούν στη σελίδα που εμφανίζεται στην παρακάτω εικόνα και χρησιμοποιείται για την προβολή των πληροφοριών μίας θεραπείας. Οι πληροφορίες εδώ είναι ομαδοποιημένες διαφορετικά και εμφανίζονται ως βασικά γνωρίσματα (Basic Properties) τα «Treatment Name», «Treatment Duration», «Date Taught», «Treatment Short Description (Greek)», «Treatment Parts», «Special Requirements», «Mandatory Series» και «Recommended Series». Να σημειωθεί εδώ ότι τα γνωρίσματα «Mandatory Series» και «Recommended Series» είναι αποτύπωση των σχέσεων ανάμεσα στις θεραπείες. Οι σχέσεις μεταξύ των θεραπειών εμφανίζονται επίσης μαζί με τις υπόλοιπες πληροφορίες μίας θεραπείας.

The screenshot shows a mobile application interface titled 'Tai Yi Treatments'. At the top, there is a navigation bar with 'Home', 'Search', 'Insert', and 'List' buttons. Below this, the title 'MISSING FATHER' is displayed. The main content area is divided into sections: 'Basic Properties' (which includes fields for Treatment Name, Duration, Date Taught, and Description (Greek)), 'Special Requirements', 'Mandatory Series', and 'Recommended Series'. A 'Details' button is visible at the bottom of the 'Basic Properties' section. Below the 'Details' button are 'Edit' and 'Delete' buttons. At the very bottom, a footer displays '2017 Georgios Tsilipakos, DIT MSc Student'.

Οι υπόλοιπες πληροφορίες λοιπόν εμφανίζονται ομαδοποιημένες όλες μαζί στην ομάδα λεπτομέρειες (Details). Μία εικόνα με τις λεπτομέρειες μίας θεραπείας δίνεται στην επόμενη εικόνα.

TaiYiApp

Treatment Short Description (English):

Treatment Limitations:

Treatment Specific Timing:

Treatment Series Type:

Gauging Resistance:

Possible Negative Effects:

Treatment Integration Period:

Recommended Actions for Recipient:

Treatment Timing Post:

Treatment Requirements:

Οι χρήστες στο σημείο αυτό έχουν τη δυνατότητα να επιλέξουν είτε να επεξεργαστούν τη συγκεκριμένη θεραπεία που βλέπουν πατώντας «Edit» και να προχωρήσουν στην επεξεργασία της όπως περιγράφεται στην παράγραφο 7, είτε να διαγράψουν τη θεραπεία πατώντας «Delete». Στην τελευταία περίπτωση η εφαρμογή ζητάει επιβεβαίωση από το χρήστη όπως φαίνεται στην παρακάτω εικόνα.

Confirmation

Confirmation

Do you really want to delete Treatment 5 - MISSING FATHER

OK

Cancel

7. Για να πραγματοποιήσουν οι χρήστες επεξεργασία στα δεδομένα μίας θεραπείας μπορούν να επιλέξουν «Edit» είτε από τη λίστα των θεραπειών, είτε από την προβολή της θεραπείας. Ανεξάρτητα με ποια μεθοδολογία θα οδηγηθούν στην επεξεργασία, ο τρόπος ενημέρωσης των δεδομένων των γνωρισμάτων είναι ακριβώς ίδιος με τον τρόπο που γίνεται η εισαγωγή των δεδομένων που παρουσιάστηκε στην παράγραφο 4. Μία ενδεικτική εικόνα επεξεργασίας μίας θεραπείας δίνεται στην επόμενη εικόνα.

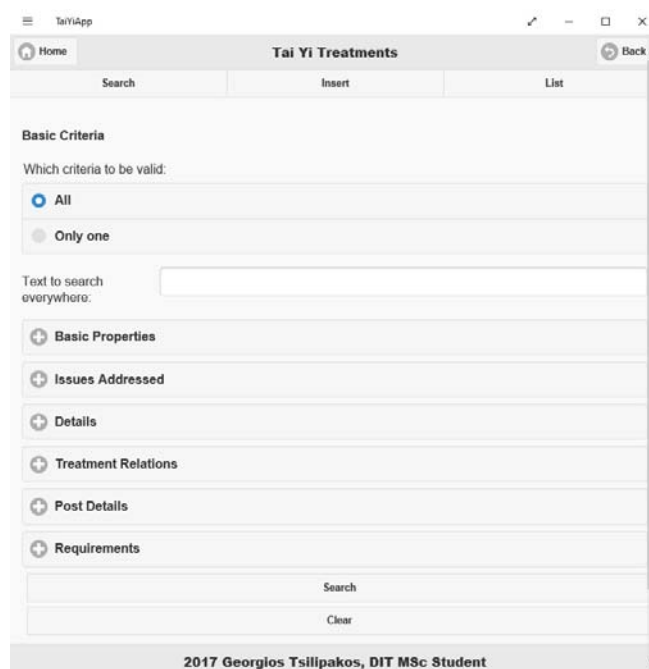
The screenshot shows a web application window titled 'Tai Yi Treatments'. At the top, there are three buttons: 'Home', 'Search', and 'List'. Below these, the title 'MISSING FATHER' is displayed. The main section is titled 'Basic Properties' and contains several input fields: 'Treatment ID' with the value '5', 'Treatment Name' with the value 'MISSING FATHER', 'Treatment Duration' (empty), and 'Date Taught' with the value 'July 2016, 11/2006'. Below these fields are two expandable sections: 'Treatment in Parts' and 'Type of Treatment'. There is also a button labeled 'Update Types of Treatments'. At the bottom, there is a 'Treatment Description' label and a large text area. The footer of the application reads '2017 Georgios Tsilipakos, DIT MSc Student'.

Η διαφοροποίηση έγκειται στο ότι υπάρχει και το γνώρισμα «Treatment ID», το οποίο όμως δεν μπορούν να αλλάξουν οι χρήστες της εφαρμογής. Μετά την ολοκλήρωση των απαιτούμενων αλλαγών, στο τέλος της σελίδας υπάρχει η επιλογή «Edit» που διενεργείται πλέον η αλλαγή των δεδομένων της θεραπείας, ενημερώνονται οι χρήστες για την ολοκλήρωση της διαδικασίας και επανέρχεται η εφαρμογή στην αρχική οθόνη.

8. Για την αναζήτηση μίας θεραπείας επιλέγουν οι χρήστες το «Search» στην αρχική οθόνη (φαίνεται στην επόμενη εικόνα μαρκαρισμένο) ή στο μενού της επικεφαλίδας.

The screenshot shows the main menu of the 'Tai Yi Treatments' application. At the top, there are three buttons: 'Home', 'Search', and 'List'. Below these, there is a large image of a pink flower. Below the image, the 'Search' button is highlighted with a red circle. Below the 'Search' button, there are two smaller images of pink flowers. Below these images are two buttons: 'Insert' and 'List'. At the bottom, there are two buttons: 'Get DB from Dropbox' and 'Upload DB to Dropbox'. The footer of the application reads '2017 Georgios Tsilipakos, DIT MSc Student'.

Η σελίδα της αναζήτησης παρουσιάζεται στην παρακάτω σελίδα. Αρχικά υπάρχει η επιλογή αν θέλουν οι χρήστες να ισχύουν όλα τα κριτήρια που θα βάλουν ή θέλουν να ισχύει μόνο ένα. Αν λοιπόν επιλέξουν το «All» τότε κάθε κριτήριο που θα συμπληρωθεί θα πρέπει να ισχύει, ενώ αν επιλέξουν το «Only one» τότε αρκεί να ισχύει ένα από όσα κριτήρια δοθούν. Τα κριτήρια είναι σε δύο μεγάλες ομάδες όπως εξηγείται παρακάτω. Να σημειωθεί εδώ ότι το γενικό κριτήριο θεωρείται ένα κριτήριο. Δηλαδή ισχύει ή όχι είτε αθροιστικά είτε συμπληρωματικά με τα κριτήρια για τα υπόλοιπα γνωρίσματα.



- 8.1. Γενικό κριτήριο κειμένου. Αν προσθέσουν οι χρήστες κάποιο κείμενο στο σημείο «Text to search everywhere» γίνεται αναζήτηση σε όλα τα γνωρίσματα μίας θεραπείας. Μάλιστα μπορούν να δοθούν μία ή περισσότερες λέξεις σε συνδυασμό. Αν δηλαδή δοθεί το κείμενο για παράδειγμα «father+mother,pain», τότε θα εμφανιστούν όλες οι θεραπείες που έχουν κάπου την λέξη «father» και έχουν κάπου είτε τη λέξη «mother» είτε τη λέξη «pain». Δηλαδή το «+» θεωρείται για τα κριτήρια «και», ενώ το «,» θεωρείται για τα κριτήρια «ή». Αν δοθεί για παράδειγμα η έκφραση «Missing father» θα αναζητηθεί να βρεθεί το σύνολο της έκφρασης.
- 8.2. Κριτήρια γνωρισμάτων. Εδώ μπορούν οι χρήστες να προσθέσουν κάποιο κριτήριο σε όλα τα γνωρίσματα μίας θεραπείας όπως έχουν παρουσιαστεί παραπάνω στην παράγραφο 4. Η μόνη διαφορά είναι ότι στα περιφερειακά γνωρίσματα δεν μπορούν να δώσουν περισσότερες από μία τιμές για αναζήτηση αλλά μόνο μία. Για παράδειγμα δεν μπορούν να αναζητήσουν μία θεραπεία που θα έχει δύο τιμές στο γνώρισμα «Treatment Special Category». Στην παρακάτω εικόνα δίνεται ένα παράδειγμα για την ομάδα γνωρισμάτων «Post Details». Σημειώνεται επίσης ότι τα επιμέρους κριτήρια δεν λειτουργούν όπως το γενικό κριτήριο, δηλαδή δεν αντιλαμβάνονται τους χαρακτήρες «+» και «,» σαν χαρακτήρες ελέγχου.

TaiYiApp

Basic Properties

Issues Addressed

Details

Treatment Relations

Post Details

Possible Negative Effects:

Treatment Integration Period:

Recommended Actions for Recipient: something

Treatment Timing Post Value:

Requirements

Search

Clear

Search Results:

Οι χρήστες έχουν τη δυνατότητα αφού ολοκληρώσουν την καταγραφή των κριτηρίων να επιλέξουν είτε το «Search» για να προχωρήσει η εφαρμογή στην αναζήτηση των θεραπειών που πληρούν τα κριτήρια που δόθηκαν, είτε το «Clear» για να διαγραφούν όλα τα κριτήρια από το σύνολο των γνωρισμάτων και να ξεκινήσει μία νέα αναζήτηση.

Τα αποτελέσματα της αναζήτησης εμφανίζονται σε λίστα όπου όλες οι θεραπείες που βρέθηκαν να πληρούν τα κριτήρια εμφανίζουν το σύνολο των βασικών γνωρισμάτων τους όπως περιγράφηκε στη παράγραφο 5. Ένα παράδειγμα αποτελεσμάτων εμφανίζεται στην παρακάτω εικόνα.

TaiYiApp

Search Results:

1. MIND OF FATHER

Duration: -

Date Taught: August 2015

Description (Greek):

[View Details](#)

[Edit](#)

2. MIND POWER

Duration: -

Date Taught: August 2015

Description (Greek):



9. Τελευταία παρουσιάζεται ο τρόπος διαχείρισης των περιφερειακών δεδομένων των γνωρισμάτων μίας θεραπείας. Ο τρόπος διαχείρισης είναι κοινός για όλες αυτά τα γνωρίσματα, και επιλέχθηκε να παρουσιαστεί η διαχείριση του γνωρίσματος «Special Categories». Η σελίδα για τη διαχείριση παρουσιάζεται στην επόμενη εικόνα.

The screenshot shows a mobile application interface for 'Tai Yi Treatments'. It features a 'Home' button and a 'Back' button. The main content area is titled 'Special Categories' and contains three input fields for 'Special Category ID', 'Special Category Name', and 'Special Category Description'. Below these fields are four buttons: 'Insert', 'Update', 'Delete', and 'Clear'. Underneath is a section titled 'Special Categories List' which displays a list of 9 items, each with a number and a name in Greek, followed by a right-pointing arrow icon. The list items are: 1. Αναζωογόνηση, 2. Ανοσοποιητικό Σύστημα, 3. Ανώτερα Ενεργειακά Σώματα, 4. Αποτοξίνωση, 5. Αυτο-ενδυνάμωση, 6. Αφαίρεση Ενεργειακών Επιρροών, 7. Δημιουργικότητα, 8. Ανώτατο και εσωτερικό σώμα, 9. Ανώτατο και εσωτερικό σώμα. At the bottom of the screen, there is a footer text: '2017 Georgios Tsilipakos, DIT MSc Student'.

Οι επιλογές που έχουν οι χρήστες επεξηγούνται παρακάτω.

- 9.1. «Insert»: Με την επιλογή αυτή αν έχει εισαχθεί πληροφορία στο «Special Category Name» εισάγεται μία νέα τιμή στο γνώρισμα «Special Category» με τις πληροφορίες που εισάγουν οι χρήστες και οι οποίες είναι κείμενο. Αν δεν έχει εισαχθεί πληροφορία, τότε δεν μπορεί να προχωρήσει η εισαγωγή και ενημερώνονται οι χρήστες με μήνυμα.
- 9.2. «Update»: Αν έχει επιλεγεί μία τιμή του γνωρίσματος «Special Category», μπορούν οι χρήστες να κάνουν αλλαγές στις πληροφορίες και με την επιλογή αυτή να ενημερωθεί η βάση δεδομένων. Στην περίπτωση αυτή αν δεν έχει γίνει κάποια επιλογή, δηλαδή αν το «Special Category ID» είναι κενό, δεν μπορεί να προχωρήσει η ενημέρωση και εμφανίζεται μήνυμα στους χρήστες. Η εικόνα όταν οι χρήστες επιλέγουν μία τιμή δίνεται στην παρακάτω εικόνα.
- 9.3. «Delete»: Όπως και παραπάνω όταν οι χρήστες έχουν επιλέξει μία τιμή του γνωρίσματος, μπορούν να κάνουν τη συγκεκριμένη επιλογή και να διαγράψουν την επιλεγμένη τιμή του γνωρίσματος. Και πάλι πρέπει να υπάρχει τιμή στο «Special Category ID» διαφορετικά εμφανίζεται μήνυμα λάθους στους χρήστες.
- 9.4. «Clear»: Με την επιλογή αυτή οι χρήστες καθαρίζουν την επιλογή συγκεκριμένης τιμής του γνωρίσματος «Special Category» και όλες οι πληροφορίες γίνονται κενές.

The screenshot displays the 'Tai Yi Treatments' application interface. At the top, there is a header bar with a 'Home' button, the title 'Tai Yi Treatments', and a 'Back' button. Below the header, the 'Special Categories' section contains a form with the following fields: 'Special Category ID' (with the value '6'), 'Special Category Name' (with the value 'Αφαίρεση Ενεργειακών Επιπρωών'), and 'Special Category Description' (empty). Below these fields are four buttons: 'Insert', 'Update', 'Delete', and 'Clear'. Underneath the form is a 'Special Categories List' which is a scrollable list of categories. The list contains the following items: 1. Αναζωογόνηση, 2. Ανοσοποιητικό Σύστημα, 3. Ανώτερα Ενεργειακά Σώματα, 4. Αποτοξίνωση, 5. Αυτο-ενδυνάμωση, 6. Αφαίρεση Ενεργειακών Επιπρωών (highlighted with a blue bar), 7. Δημιουργικότητα, and 8. Αιμόρροια και Ενεργειακά Επιπρωά. At the bottom of the application, there is a footer bar with the text '2017 Georgios Tsilipakos, DIT MSc Student'.

Εδώ ολοκληρώνονται οι οδηγίες χρήσης της εφαρμογής, αφού έχει περιγραφεί το σύνολο της λειτουργικότητας της εφαρμογής Tai Yi. Όπως ισχύει για κάθε εφαρμογή, με τη χρήση αυξάνεται η εξοικείωση και γίνεται αντιληπτή η λειτουργικότητά της, βοηθώντας να γίνονται οι απαιτούμενες ενέργειες με σωστό και αποτελεσματικό τρόπο. Σίγουρα υπάρχουν πολλά περιθώρια βελτίωσης και στη λειτουργικότητα και στην αισθητική της εφαρμογής και δοθείσης της ευκαιρίας σίγουρα θα μπορέσουν να αναπτυχθούν.

## ΑΝΑΦΟΡΕΣ

- [1] ITU, “ITU releases 2018 global and regional ICT estimates” Dec 2018; <https://www.itu.int/en/mediacentre/Pages/2018-PR40.aspx> [Προσπελάστηκε 11/9/2019]
- [2] Jonah Life Institute, “Life Energy Flow Tai Yi School of Healing”, 2019; <http://lifeenergyflowtaiyi.com/> [Προσπελάστηκε 11/9/2019]
- [3] A. Nayak, A. Poriya, D. Poojary, Type of NOSQL Databases and its Comparison with Relational Databases, *International Journal of Applied Information Systems (IJ AIS)*, vol. 5, no. 4, Mar. 2013, pp. 16-19.
- [4] C. Györfi, R. Györfi, R. Sotoc, A Comparative Study of Relational and Non-Relational Database Models in a Web-Based Application, *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 6, no. 11, 2015, pp. 78-83.
- [5] ObjectBox Limited, “Mobile databases: SQLite and SQLite alternatives for Android and iOS”, May 2017; <http://objectbox.io/sqlite-alternatives/> [Προσπελάστηκε 11/9/2019]
- [6] solid IT, “DB-Engines Ranking”, Sep. 2019; <https://db-engines.com/en/ranking> [Προσπελάστηκε 11/9/2019]
- [7] E. Malone, “The pros and cons of wireframing in a website redesign”, July 2017; <https://www.growthdrivendesign.com/blog/the-pros-and-cons-of-wireframing-in-a-website-redesign> [Προσπελάστηκε 11/9/2019]
- [8] G2 Crowd Inc., “Best Integrated Development Environment (IDE) Software”, 2019; <https://www.g2.com/categories/integrated-development-environment-ide> [Προσπελάστηκε 11/9/2019]
- [9] A. Majeed, I. Rauf, MVC Architecture: A Detailed Insight to the Modern Web Applications Development”, *Peer Review Journal of Solar & Photoenergy Systems*, vol. 1, issue 1, Sep. 2019; <https://crimsonpublishers.com/prsp/fulltext/PRSP.000505.php> [Προσπελάστηκε 11/9/2019]
- [10] M. Debnath, “A Step-by-step Guide to Creating NetBeans Web Apps”, Aug. 2017; <https://www.developer.com/java/ejb/a-step-by-step-guide-to-creating-netbeans-web-apps.html> [Προσπελάστηκε 11/9/2019]
- [11] S. Idesis, “Free Cross-Platform Mobile App Development Tools Compared - 2018”, May 2018; <https://www.outsystems.com/blog/free-cross-platform-mobile-app-development-tools-compared.html> [Προσπελάστηκε 11/9/2019]
- [12] Microsoft Visual Studio Docs, “Get started with Visual Studio Tools for Apache Cordova”, April 2016; <https://docs.microsoft.com/en-us/visualstudio/cross-platform/tools-for-cordova/first-steps/get-started-first-mobile-app?view=toolsforcordova-2015> [Προσπελάστηκε 11/9/2019]
- [13] GitHub, “SQLite compiled to JavaScript through Emscripten”, 2019; <https://github.com/kripken/sql.js> [Προσπελάστηκε 11/9/2019]
- [14] emscripten webpage, 2015; <https://emscripten.org/> [Προσπελάστηκε 11/9/2019]
- [15] GitHub, “Dropbox JavaScript SDK”, 2019; <https://github.com/dropbox/dropbox-sdk-js> [Προσπελάστηκε 11/9/2019]
- [16] Unicode Inc. 1991-2019, *Unicode Normalization Forms*, Unicode Standard Annex #15, 2019.
- [17] M. Χατζόπουλος, ed., *Θεμελιώδεις Αρχές Συστημάτων Βάσεων Δεδομένων*, Εκδόσεις ΔΙΑΥΛΟΣ, 2001.
- [18] Guru99, “20 Best SQL Management Tools in 2019”, 2019; <https://www.guru99.com/top-20-sql-management-tools.html> [Προσπελάστηκε 11/9/2019]
- [19] J. Vrána, “ADMINER - Database management in a single PHP file”, 2019; <https://www.adminer.org/> [Προσπελάστηκε 11/9/2019]
- [20] DB Browser for SQLite, “The Official home of the DB Browser for SQLite”, 2019; <https://sqlitebrowser.org/> [Προσπελάστηκε 11/9/2019]
- [21] DBeaver Community, DBeaver - Free Universal Database Tool, 2019; <https://dbeaver.io/> [Προσπελάστηκε 11/9/2019]
- [22] DbVis Software AB, DbVisualizer, 2019; <https://www.dbvis.com/> [Προσπελάστηκε 11/9/2019]
- [23] W. Lim, “A Beginner’s Guide to Wireframing”, June 2012; <https://webdesign.tutsplus.com/articles/a-beginners-guide-to-wireframing--webdesign-7399> [Προσπελάστηκε 11/9/2019]
- [24] Balsamiq Studios LLC 2008-2019, Balsamiq Wireframes, 2019; <https://balsamiq.com/index.html> [Προσπελάστηκε 11/9/2019]
- [25] Axure Software Solutions Inc. 2002-2019, Axure RP 9, 2019; <https://www.axure.com/> [Προσπελάστηκε 11/9/2019]
- [26] FlairBuilder 2009-2016, FlairBuilder, 2019; <http://flairbuilder.com/> [Προσπελάστηκε 11/9/2019]
- [27] Adobe, Fireworks CS6, 2019; [https://www.adobe.com/gr\\_en/products/fireworks.html](https://www.adobe.com/gr_en/products/fireworks.html) [Προσπελάστηκε 11/9/2019]
- [28] Adobe, InDesign, 2019; [https://www.adobe.com/gr\\_en/products/indesign.html](https://www.adobe.com/gr_en/products/indesign.html) [Προσπελάστηκε 11/9/2019]
- [29] Khella Productions Inc, Keynotopia, 2013; <https://keynotopia.com/> [Προσπελάστηκε 11/9/2019]

- [30] Evolus 2008-2012, Pencil Project, 2017; <https://pencil.evolus.vn/> [Προσπελάστηκε 11/9/2019]
- [31] A Produle Systems Pvt Ltd 2009-2019, MockFlow, 2019; <https://www.mockflow.com/> [Προσπελάστηκε 11/9/2019]
- [32] Some Character LLC, Mockingbird, 2019; <https://gomockingbird.com/home> [Προσπελάστηκε 11/9/2019]
- [33] Nulab Inc., Cacao, 2019; <https://cacao.com/> [Προσπελάστηκε 11/9/2019]
- [34] Microsoft Visual Studio Docs, "Install Visual Studio Tools for Apache Cordova", Oct. 2016; <https://docs.microsoft.com/en-us/visualstudio/cross-platform/tools-for-cordova/first-steps/installation?view=toolsforcordova-2015> [Προσπελάστηκε 11/9/2019]
- [35] The jQuery Foundation, «jQuery mobile Demos», 2014; <https://demos.jquerymobile.com/1.4.2/> [Προσπελάστηκε 11/9/2019]
- [36] Refsnes Data 1999-2019, w3schools.com, jQuery Tutorial, 2019; <https://www.w3schools.com/jquery/> [Προσπελάστηκε 11/9/2019]
- [37] Microsoft Visual Studio Docs, "Setup guide: Target iOS mobile devices in a Visual Studio Tools for Apache Cordova project", July 2015; <https://docs.microsoft.com/en-us/visualstudio/cross-platform/tools-for-cordova/first-steps/ios-guide?view=toolsforcordova-2015> [Προσπελάστηκε 11/9/2019]
- [38] Microsoft Visual Studio Docs, "Run your Apache Cordova app on Android", Aug. 2016; <https://docs.microsoft.com/en-us/visualstudio/cross-platform/tools-for-cordova/build-deploy/run-app-android-emulator?view=toolsforcordova-2015> [Προσπελάστηκε 11/9/2019]
- [39] Microsoft Visual Studio Docs, "Package a UWP app with Visual Studio", July 2019; [https://docs.microsoft.com/en-us/windows/msix/package/packageing-uwp-apps#sideload\\_package](https://docs.microsoft.com/en-us/windows/msix/package/packageing-uwp-apps#sideload_package) [Προσπελάστηκε 11/9/2019]
- [40] Microsoft Visual Studio Docs, "Enable your device for development", April 2019; <https://docs.microsoft.com/en-us/windows/uwp/get-started/enable-your-device-for-development> [Προσπελάστηκε 11/9/2019]
- [41] Γ. Σταμέλος, ed., *Τεχνολογία Λογισμικού, Θεωρία και Πράξη*, Εκδόσεις Κλειδάριθμος, 2003.
- [42] J. McCormack, D. Conway 1998-2005, Monash University, CSE2305 Object-Oriented Software Engineering course, 2005; <http://users.monash.edu/~jonmc/CSE2305/Topics/12.24.SWEng3/html/text.html> [Προσπελάστηκε 11/9/2019]
- [43] P. Upadhyay, The Role of Verification and Validation in System Development Life Cycle, *IOSR Journal of Computer Engineering (IOSRJCE)*, vol. 5, iss. 1, Sep.-Oct. 2012, pp. 17-20.
- [44] U. Eriksson, "Understanding the Difference Between Software Verification and Validation", May 2017; <https://reqtest.com/testing-blog/software-verification-and-validation-differences/> [Προσπελάστηκε 11/9/2019]