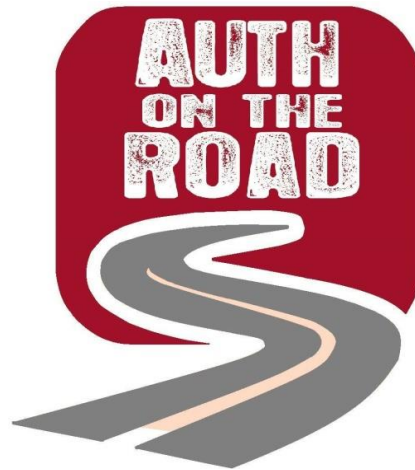




Τεχνολογία Λογισμικού  
Τομέας Ηλεκτρονικής και Υπολογιστών  
Τμήμα ΗΜΜΥ  
Α.Π.Θ.

8<sup>ο</sup> Εξάμηνο  
Άνοιξη 2016



**Dream, explore, discover!**

## **ΣΧΕΔΙΑΣΗ ΣΥΣΤΗΜΑΤΟΣ**

**Del.3.1**

**Version 1.0**

Αμπαρτσουμιάν Αλεξάνδρα [alexampa@auth.gr](mailto:alexampa@auth.gr)  
Δημαρίδου Βαλασία [valia.dimaridou@gmail.com](mailto:valia.dimaridou@gmail.com)  
Θεοδωρίδου Ιφιγένεια [ifigtheo@auth.gr](mailto:ifigtheo@auth.gr)  
Παπαδόπουλος Βασίλης [vasipapa@auth.gr](mailto:vasipapa@auth.gr)

**15/05/2016**



## Ιστορικό Αλλαγών

Όνομα	Ημ/νία	Περιγραφή Αλλαγής	Εκδ.
A. Συμεωνίδης	29/05/2009	Δημιουργία Εγγράφου Προσαρμογή του ESA software engineering standards guidelines (1991) και του εγγράφου SDD document, από τους Bruegge και Dutoit (2004).	0.1
Auth on the road	4/05/2016	Συγγραφή κεφαλαίων 1 και 2	0.2
Auth on the road	8/05/2016	Συγγραφή κεφαλαίου 3	0.3
Auth on the road	10/05/2016	Συγγραφή κεφαλαίου 4	0.4
Auth on the road	12/05/2016	Διόρθωση κεφαλαίου 3 και προσθήκες στο κεφάλαιο 4	0.5
Auth on the road	16/05/2016	Προσθήκη ανοικτών θεμάτων, και μορφοποίηση του εγγράφου	1.0

## Μέλη Ομάδας Ανάπτυξης

Όνομα	ΟΑ	Email
Χ. Ζολώτας	*	<a href="mailto:christopherzolotas@issel.ee.auth.gr">christopherzolotas@issel.ee.auth.gr</a>
Αμπαρτσουμιάν Αλεξάνδρα	Auth on the road	<a href="mailto:alexampa@auth.gr">alexampa@auth.gr</a>
Δημαρίδου Βαλασία	Auth on the road	<a href="mailto:valia.dimaridou@gmail.com">valia.dimaridou@gmail.com</a>
Θεοδωρίδου Ιφιγένεια	Auth on the road	<a href="mailto:ifigtheo@auth.gr">ifigtheo@auth.gr</a>
Παπαδόπουλος Βασίλης	Auth on the road	<a href="mailto:vasipapa@auth.gr">vasipapa@auth.gr</a>



## Πίνακας Περιεχομένων

Πίνακας Περιεχομένων .....	3
Λίστα Σχημάτων .....	4
1. Εισαγωγή .....	5
1.1 Στόχος του εγγράφου .....	5
1.2 Αντικείμενο του λογισμικού .....	6
1.3 Ορισμοί, Ακρωνύμια, Συντομεύσεις .....	6
1.4 Στόχοι Σχεδίασης .....	7
1.5 Αναγνωστικό Κοινό. Τρόπος Ανάγνωσης εγγράφου .....	8
1.6 Επισκόπηση Εγγράφου .....	9
2. Τρέχουσα Αρχιτεκτονική Λογισμικού .....	10
2.1 Αρχιτεκτονική πελάτη διακομιστή (client-server) .....	10
2.2 Αρχιτεκτονική τριών επιπέδων (three tier) .....	11
2.3 Η αρχιτεκτονική του συστήματος Auth On The Road .....	12
3. Προτεινόμενη Αρχιτεκτονική Λογισμικού .....	14
3.1 Αποδόμηση Συστήματος .....	14
3.1.1 Υποσύστημα LoginHandler .....	14
3.1.2 Υποσύστημα LoginGUI .....	14
3.1.3 Υποσύστημα ExcursionViewHandler .....	15
3.1.4 Υποσύστημα ExcursionViewGUI .....	15
3.1.5 Υποσύστημα ExcursionCreateHandler .....	16
3.1.6 Υποσύστημα ExcursionCreateGUI .....	16
3.1.7 Υποσύστημα AudioVisualHandler .....	17
3.1.8 Υποσύστημα AudioVisualGUI .....	17
3.1.9 Υποσύστημα ApplicationFormHandler .....	18
3.1.10 Υποσύστημα ApplicationFormGUI .....	18
3.1.11 Υποσύστημα Databases .....	19
3.2 Απεικόνιση Υλικού/Λογισμικού .....	20
3.2.1 Client Device .....	20
3.2.2 System Server .....	21
3.2.3 Συνολικό διάγραμμα ανάπτυξης .....	22
3.3 Διαχείριση Μόνιμων Δεδομένων .....	23
3.4 Έλεγχος Πρόσβασης και Ασφάλεια .....	24
3.5 Γενικός έλεγχος λογισμικού .....	26
3.6 Οριακές Συνθήκες .....	27
3.6.1 Έναρξη (start-up) λογισμικού .....	27
3.6.2 Τερματισμός (shutdown) λογισμικού .....	28
3.6.3 Διακοπή τροφοδοσίας .....	28
3.6.4 Σφάλματα λογισμικού .....	28
3.6.5 Ενημέρωση λογισμικού και συντήρηση συστήματος .....	29
3.6.6 Υπερφόρτωση συστήματος .....	29



4.	Προδιαγραφές Τμηματικού Σχεδιασμού (Component Design Specifications).....	30
4.1	Πρότυπα Σχεδιασμού που υιοθετήθηκαν.....	30
4.1.1	Δομικά πρότυπα.....	30
4.1.2	Πρότυπα συμπεριφοράς.....	32
5.	Πίνακας ιχνηλασιμότητας εγγράφων Σχεδίασης και Απαιτήσεων Λογισμικού.....	34
6.	Παράρτημα Ι – Ανοιχτά Θέματα.....	35

## Λίστα Σχημάτων

Σχήμα 1.	Επικοινωνία αυτόνομων υποσυστημάτων.....	5
Σχήμα 2.	Εκπαιδευτικές εκδρομές.....	6
Σχήμα 3.	Απαιτήσεις διαφορετικών ομάδων.....	7
Σχήμα 4.	Αναπαράσταση αρχιτεκτονικής πελάτη-διακομιστή.....	10
Σχήμα 5.	Αναπαράσταση μοντέλου τριών επιπέδων.....	11
Σχήμα 6.	Υποσύστημα LoginHandler.....	14
Σχήμα 7.	Υποσύστημα LoginGUI.....	15
Σχήμα 8.	Υποσύστημα ExcursionViewHandler.....	15
Σχήμα 9.	Υποσύστημα ExcursionViewGUI.....	16
Σχήμα 10.	Υποσύστημα ExcursionCreateHandler.....	16
Σχήμα 11.	Υποσύστημα ExcursionCreateGUI.....	17
Σχήμα 12.	Υποσύστημα AudioVisualHandler.....	17
Σχήμα 13.	Υποσύστημα AudioVisualGUI.....	18
Σχήμα 14.	Υποσύστημα ApplicationFormHandler.....	18
Σχήμα 15.	Υποσύστημα ApplicationFormGUI.....	19
Σχήμα 16.	Υποσύστημα Databases.....	19
Σχήμα 17.	Συνολικό διάγραμμα τμημάτων.....	20
Σχήμα 18.	Client Device.....	21
Σχήμα 19.	System Server.....	21
Σχήμα 20.	Διάγραμμα ανάπτυξης.....	22
Σχήμα 21.	Το πρότυπο Proxy όπως εφαρμόστηκε στο υποσύστημα AudioVisualHandler.....	31
Σχήμα 22.	Το πρότυπο Proxy.....	31
Σχήμα 23.	Το πρότυπο Bridge όπως εφαρμόστηκε στο υποσύστημα ExcursionCreateHandler.....	32
Σχήμα 24.	Το πρότυπο Bridge.....	32
Σχήμα 25.	Το πρότυπο Template όπως εφαρμόστηκε στο υποσύστημα ExcursionViewGUI.....	33
Σχήμα 26.	Το πρότυπο Template.....	33



## 1. Εισαγωγή

Η εφαρμογή *Auth on the road* αποτελεί ένα έργο προς ενημέρωση των μελών της πανεπιστημιακής κοινότητας σχετικά με εκπαιδευτικές εκδρομές, που πρόκειται να διεξαχθούν στα πλαίσια του Πανεπιστημίου. Στα έγγραφα που προηγήθηκαν, ολοκληρώθηκε η καταγραφή των απαιτήσεων χρηστών και ο καθορισμός των απαιτήσεων λογισμικού. Εν συνεχεία, στο παρόν έγγραφο, πραγματοποιείται η περιγραφή του συστήματος σε επίπεδο σχεδίασης και αρχιτεκτονικής.

### 1.1 Στόχος του εγγράφου

Σκοπός του παρόντος εγγράφου είναι η σχεδίαση του συστήματος της εφαρμογής, μέσω της επιλογής κατάλληλης αρχιτεκτονικής και προτύπων σχεδίασης. Προς επίτευξη του στόχου αυτού, είναι απαραίτητη η αποδόμηση του συστήματος σε επιμέρους αυτοτελή υποσυστήματα, τα οποία όμως συγχρονίζονται και επικοινωνούν αξιόπιστα και αποτελεσματικά μεταξύ τους.

Οι αλληλεπιδράσεις μεταξύ των υποσυστημάτων διαφαίνονται μέσα από τα διαγράμματα τμημάτων και τα διαγράμματα ανάπτυξης. Τα πρώτα αναλαμβάνουν την καταγραφή των εννοιολογικά συνεκτικών τμημάτων και των μεταξύ τους εξαρτήσεων, ενώ τα δεύτερα μοντελοποιούν τα τμήματα του λογισμικού, ούτως ώστε να επιτρέπεται ο έλεγχος του συστήματος σε θέματα επίδοσης και ασφάλειας και να καθίσταται δυνατή η εκτίμηση του κόστους του έργου. Σε αυτό το στάδιο, μπορεί να ελεγχθεί η ορθότητα της όποιας επιλογής αρχιτεκτονικής για το λογισμικό του συστήματος.

Προς αντιμετώπιση των όσων σχεδιαστικών προβλημάτων παρουσιάζονται κατά τη διαδικασία σχεδιασμού του συστήματος, παρατίθενται τα κατάλληλα πρότυπα σχεδίασης που τα επιλύουν. Κατ' αυτόν τον τρόπο, είναι πλέον, εφικτή η λήψη αποφάσεων για τα είδη λογισμικού που θα χρησιμοποιηθούν, η σχεδίαση του υλικού και η εισαγωγή στο στάδιο της ανάπτυξης του συστήματος.



Σχήμα 2. Επικοινωνία αυτόνομων υποσυστημάτων

## 1.2 Αντικείμενο του λογισμικού

- Το λογισμικό του συστήματος *Auth on the road* εξυπηρετεί την πραγματοποίηση εκπαιδευτικών εκδρομών στα πλαίσια του πανεπιστημίου. Επεξηγηματικά, αντικείμενο αυτού είναι η ενημέρωση των χρηστών σχετικά με εκδρομές, καθώς και ενημερωτικές εκδηλώσεις που τις αφορούν.
- Χρήστες του συστήματος μπορούν να αποτελέσουν όλα τα μέλη της πανεπιστημιακής κοινότητας, με δυνατότητες καταχώρησης εκδρομών, εκδηλώσεων, οπτικοακουστικού υλικού και προσωπικής αίτησης συμμετοχής, καθώς και προβολής του περιεχομένου όλων αυτών. Όσοι χρήστες προχωρούν στην καταχώρηση ενός στοιχείου από τα παραπάνω, έχουν επιπλέον το δικαίωμα τροποποίησης ή (και) διαγραφής αυτού. Επιπροσθέτως, ο δημιουργός της καταχώρησης μιας εκδρομής έχει το δικαίωμα αξιολόγησης των αιτήσεων συμμετοχής των υπόλοιπων χρηστών και της διαγραφής οποιουδήποτε αρχείου οπτικοακουστικού υλικού.
- Οι προαναφερθείσες λειτουργικότητες του συστήματος εξυπηρετούνται μέσω μιας απομακρυσμένης βάσης δεδομένων, η οποία αναλαμβάνει τη διαχείριση όλων των απαραίτητων δεδομένων, αλλά και την ανταλλαγή αυτών μεταξύ χρήστη και συστήματος.



Σχήμα 2. Εκπαιδευτικές εκδρομές

## 1.3 Ορισμοί, Ακρωνύμια, Συντομεύσεις

- ✓ Client : Πελάτης
- ✓ Server: Εξυπηρετητής
- ✓ GUI: Graphical User Interface (Γραφική διεπαφή χρήστη)
- ✓ DB: Database (Βάση Δεδομένων)
- ✓ API: Application Programming Interface (Διεπαφή Προγραμματισμού Εφαρμογών)
- ✓ HTTPS: Hypertext Transfer Protocol Secure
- ✓ URL: Σύνδεσμος
- ✓ SSL: Secure Sockets Layer
- ✓ LAN: Local Area Network
- ✓ Backup: Εφεδρικά δεδομένα
- ✓ MySQL: Σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων

## 1.4 Στόχοι Σχεδίασης

Στόχος της σχεδίασης του συστήματος είναι η επιτυχής και βέλτιστη υλοποίηση των λειτουργικών απαιτήσεων, μέσα από τους περιορισμούς που θέτουν οι μη λειτουργικές απαιτήσεις, όπως αυτές διατυπώθηκαν στο έγγραφο απαιτήσεων χρηστών. Επιπρόσθετα, εισάγονται νέοι περιορισμοί που αφορούν τις απαιτήσεις των ακόλουθων ομάδων φυσικών προσώπων:

➤ Ομάδα ανάπτυξης και συντήρησης (προγραμματιστές)

- ✓ Αξιοπιστία και λειτουργικότητα συστήματος
- ✓ Ελάχιστος αριθμός σφαλμάτων κώδικα, για εξασφάλιση ορθής λειτουργίας και ευρωστίας.
- ✓ Αναγνωσιμότητα κώδικα, για ευκολία συντήρησης από διαφορετικές ομάδες προγραμματιστών.
- ✓ Δυνατότητα εύκολης συντήρησης και επέκτασης του κώδικα, σε ενδεχόμενες μελλοντικές αναβαθμίσεις της εφαρμογής.

➤ Τελικοί χρήστες (μέλη της πανεπιστημιακής κοινότητας)

- ✓ Αξιοπιστία και ευρωστία συστήματος
- ✓ Φιλικό και εύχρηστο γραφικό περιβάλλον της εφαρμογής
- ✓ Χρηστικότητα και ευκολία εκμάθησης της εφαρμογής
- ✓ Απόδοση εκτέλεσης, με άμεσους χρόνους απόκρισης του συστήματος ανά ενέργεια χρήστη.
- ✓ Ελάχιστη επιβάρυνση συσκευής σε ισχύ και χρεώσεις δεδομένων.



Σχήμα 3. Απαιτήσεις διαφορετικών ομάδων

➤ Πανεπιστημιακές αρχές (πελάτες)

- ✓ Αξιοπιστία συστήματος
- ✓ Ελάχιστο δυνατό κόστος, ανάπτυξης και συντήρησης της εφαρμογής.
- ✓ Περάτωση της ανάπτυξης εντός σύντομου χρονικού διαστήματος.
- ✓ Αυξημένη παραγωγικότητα και διάδοση της εφαρμογής σε πλήθος τελικών χρηστών.
- ✓ Ικανοποιητική απόδοση εκτέλεσης του συστήματος.
- ✓ Παρακολούθηση των απαιτήσεων των χρηστών.



Ορισμένες από τις απαιτήσεις των παραπάνω ομάδων είναι κοινές, ενώ άλλες είναι αλληλοσυγκρουόμενες. Παραδείγματα αλληλοσυγκρουόμενων απαιτήσεων αποτελούν η λειτουργικότητα έναντι της χρηστικότητας και της γρήγορης ανάπτυξης, το κόστος έναντι της ευρωστίας κ.λπ.. Απαιτείται, επομένως, η ανεύρεση μιας χρυσής τομής στη σχεδίαση του συστήματος για την επίτευξη της μέγιστης δυνατής αποτελεσματικότητας και την επιτυχία της εφαρμογής.

Ένας συμβιβασμός μείζονος σημασίας για το σύστημα αφορά τη συνεκτικότητα και τη σύζευξη των υποσυστημάτων του λογισμικού. Η αποδόμηση του συστήματος σε επιμέρους τμήματα πρέπει να στοχεύει στη δημιουργία υποσυστημάτων με χαμηλή σύζευξη προκειμένου να μην επηρεάζει η αλλαγή ενός την λειτουργικότητα ενός άλλου. Ταυτόχρονα απαιτείται υψηλή συνεκτικότητα μεταξύ των κλάσεων κάθε υποσυστήματος ώστε αυτά να εκτελούν παρόμοια έργα και να είναι συσχετισμένα μεταξύ τους.

## 1.5 Αναγνωστικό Κοινό. Τρόπος Ανάγνωσης εγγράφου.

Για την πλήρη και σωστή κατανόηση του εγγράφου σχεδίασης λογισμικού, αλλά και την ορθή εκτίμηση αυτού, απαραίτητη κρίνεται η πρότερη μελέτη των εγγράφων απαιτήσεων χρηστών και λογισμικού. Το παρόν έγγραφο τηρεί μια συγκεκριμένη οργάνωση και ροή, την οποία καλούνται οι αναγνώστες να ακολουθήσουν προς διευκόλυνσή τους στην κατανόηση των περιεχομένων του. Αναγνωστικό κοινό του εγγράφου αποτελούν οι εξής ομάδες ατόμων:

- Μηχανικοί λογισμικού που θα αναλάβουν τη συγγραφή, δοκιμή και αποσφαλμάτωση του υπό ανάπτυξη λογισμικού.
- Μηχανικοί λογισμικού που θα αναλάβουν τη συντήρηση και επέκταση του συστήματος
- Προϊστάμενοι του Κέντρου Ηλεκτρονικής Διακυβέρνησης (ΚΗΔ) του Πανεπιστημίου, οι οποίοι θα αξιολογήσουν τη συγκεκριμένη πρόταση, θα αντιπροτείνουν αλλαγές και θα αναφέρουν τα συμπεράσματά τους στη διοίκηση του πανεπιστημίου.
- Τεχνικοί δικτύων που θα αναλάβουν το σχεδιασμό και την εγκατάσταση των βάσεων δεδομένων που απαιτούνται από το σύστημα.

Κάθε αναγνώστης, ανάλογα με την κατηγορία στην οποία ανήκει, οφείλει να εστιάσει τη μελέτη του σε διαφορετικά κεφάλαια του εγγράφου σχεδίασης. Κατ' αυτό τον τρόπο, για τους μηχανικούς ανάπτυξης του λογισμικού πρακτικό ενδιαφέρον έχουν τα κεφάλαια 2 και 3, προκειμένου να γίνουν κατανοητοί οι όροι της αρχιτεκτονικής του συστήματος και των υποσυστημάτων της εφαρμογής, ενώ για τους μηχανικούς συντήρησης του έργου τα κεφάλαια 3 και 4. Για τους τεχνικούς δικτύων ιδιάζουσας σημασίας αποτελούν τα κεφάλαια 3 και 4 και ιδιαιτέρως τα διαγράμματα ανάπτυξης, ώστε να γίνει σωστά η εγκατάσταση του συστήματος. Τέλος για τους προϊστάμενους του ΚΗΔ προτείνεται η μελέτη του 1<sup>ου</sup> και του 2<sup>ου</sup> κεφαλαίου για την κατανόηση των στόχων της σχεδίασης και της αρχιτεκτονικής του συστήματος.





## 1.6 Επισκόπηση Εγγράφου

Τα περιεχόμενα του παρόντος εγγράφου ορίζονται σύμφωνα με την ακόλουθη δομή:

- ΚΕΦΑΛΑΙΟ 1<sup>ο</sup>: Αποτελεί μια εισαγωγή στα θέματα που πραγματεύονται εντός του εγγράφου. Ορίζει περιληπτικά τους στόχους του εγγράφου, το αντικείμενο του λογισμικού του εφαρμογής, αλλά και τους στόχους της σχεδίασης του συστήματος. Επιπροσθέτως, παραθέτει ακρωνύμια, συντομεύσεις και γενικότερες παραδοχές που τηρούνται στο σώμα του εγγράφου και υποδεικνύει τον τρόπο ανάγνωσής του για κάθε ομάδα του αναγνωστικού κοινού. Εμπεριέχει, επιπλέον και μια σύντομη αναφορά στο περιεχόμενο του ιδίου και των επόμενων κεφαλαίων.
- ΚΕΦΑΛΑΙΟ 2<sup>ο</sup>: Περιλαμβάνει ανάλυση της αρχιτεκτονικής λογισμικού στην οποία βασίζεται η σχεδίαση του συστήματος, με αναφορά στη μορφή, στα πλεονεκτήματα, καθώς και στους λόγους επιλογής της.
- ΚΕΦΑΛΑΙΟ 3<sup>ο</sup>: Αναφέρεται στην αποδόμηση του συστήματος σε υποσυστήματα, σύμφωνα με την επιλεγμένη αρχιτεκτονική λογισμικού. Στο εσωτερικό του παρατίθενται τα αντίστοιχα διαγράμματα τμημάτων και ανάπτυξης, προς παρουσίαση της διασύνδεσης μεταξύ των υποσυστημάτων. Κατόπιν, τίγονται ζητήματα ελέγχου, ασφάλειας και πρόσβασης που αφορούν το λογισμικό και αναλύονται οι οριακές συνθήκες του συστήματος.
- ΚΕΦΑΛΑΙΟ 4<sup>ο</sup>: Αναλύει τα πρότυπα σχεδίασης που υιοθετούνται και εφαρμόζονται στα υποσυστήματα του Auth on the road. Η χρησιμότητά τους έγκειται στην προσφορά γενικών σχεδιαστικών λύσεων σε προβλήματα και ελλείψεις της εφαρμογής. Εν συνεχεία, παρουσιάζει πίνακα ιχνηλασιμότητας αλλαγών και λίστα ανοιχτών θεμάτων.

## 2. Τρέχουσα Αρχιτεκτονική Λογισμικού

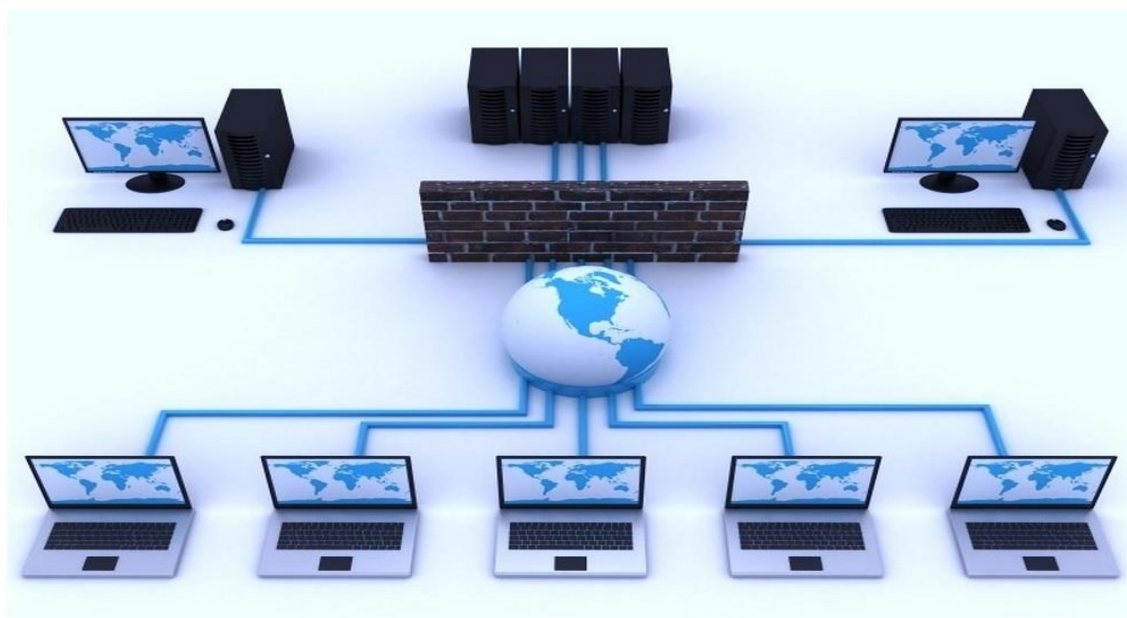
Η εφαρμογή *Auth on the road*, η οποία αναπτύσσεται στο παρόν έγγραφο, έχει ως βασική προϋπόθεση για τη λειτουργία της την συνεχή επικοινωνία του εκάστοτε χρήστη με αυτή. Συνεπώς εξετάζοντας τις υπάρχουσες αρχιτεκτονικές βρέθηκε ότι η κατάλληλη αρχιτεκτονική για το σύστημα προς ανάπτυξη είναι η αρχιτεκτονική πελάτη-διακομιστή (client-server). Επιπρόσθετα, θα χρησιμοποιηθεί και μία ακόμη αρχιτεκτονική, η αρχιτεκτονική τριών επιπέδων (three tier architecture).

### 2.1 Αρχιτεκτονική πελάτη-διακομιστή (client-server)

Η αρχιτεκτονική πελάτη-διακομιστή είναι μία αρχιτεκτονική η οποία διαχωρίζει τα καθήκοντα ενός συστήματος στους πελάτες (clients) οι οποίοι αιτούνται κάποια υπηρεσίας ή πληροφορίας και στους διακομιστές (server) οι οποίοι ανταποκρίνονται σε αυτό το αίτημα.

Πιο συγκεκριμένα, ο πελάτης χρησιμοποιώντας το λογισμικό των γραφικών διεπαφών χρήστη είναι σε θέση να δημιουργήσει αιτήσεις για πληροφορίες και να τις αποστείλει στο διακομιστή. Εν συνεχεία οι επιστροφόμενες από το διακομιστή πληροφορίες αποθηκεύονται από τον πελάτη.

Ένας διακομιστής σε αντίθεση με τον πελάτη, δεν ξεκινάει κάποια διαδικασία, αντίθετα είναι σε συνεχή λειτουργία έτσι ώστε να μπορεί να λαμβάνει και να επεξεργαστεί σε οποιαδήποτε στιγμή κάποιο αίτημα ενός πελάτη. Ένας διακομιστής μπορεί εκτός από το να λαμβάνει τα αιτήματα του πελάτη, να διαχειρίζεται, να αποθηκεύει και να ανακτά πληροφορίες μέσω εφαρμογών διαχείρισης πληροφοριών. Ακόμη αναλαμβάνει την προστασία και την ακεραιότητα των δεδομένων που διαχειρίζεται με μεθόδους όπως είναι η δημιουργία αντιγράφων ασφαλείας.



Σχήμα 4. Αναπαράσταση αρχιτεκτονικής πελάτη-διακομιστή

Ο διακομιστής είναι ένα σύστημα το οποίο είναι σε θέση να λαμβάνει ταυτόχρονα αιτήματα από πολλούς πελάτες. Η επιλογή του κατάλληλου διακομιστή είναι μείζονος σημασίας για ένα τέτοιο σύστημα, και κάθε διαφορετική εφαρμογή είναι απαραίτητο να επιλέγει το κατάλληλο είδος διακομιστή για τον αριθμό των πελατών αλλά και τον υπολογιστικό φόρτο που αναλαμβάνει να διεκπεραιώσει. Παρόλα αυτά ακόμη και η ορθή επιλογή διακομιστή δεν μπορεί να εγγυηθεί ότι βέβαιη ανταπόκριση καθώς τα αιτήματα μπορεί να υπερβούν την διαθέσιμη υπολογιστική ισχύ του. Για αυτό το λόγο είναι απαραίτητο ο διακομιστής να διαθέτει κάποια τακτική που δεν επιτρέπει σε μεμονωμένους χρήστες να εκμετελεύονται όλη την ισχύ του, αλλά και να διαθέτει κάποια τακτική προτεραιότητας εξυπηρέτησης.

Η αρχιτεκτονική πελάτη διακομιστή είναι πολύ δημοφιλής καθώς προσφέρει πολλά πλεονεκτήματα. Αυτό φαίνεται από το γεγονός ότι πρόκειται για μία αρχιτεκτονική που προσφέρει υψηλό επίπεδο αξιοπιστίας καθώς όλη η υπολογιστική διαδικασία λαμβάνει χώρα σε ένα σωστά οργανωμένο, ασφαλές σύστημα και τακτά συντηρούμενο σύστημα το οποίο μπορεί να χρησιμοποιήσει αποτελεσματικά την υπολογιστική ισχύ του.

## 2.2 Αρχιτεκτονική τριών επιπέδων (three tier)

Η αρχιτεκτονική τριών επιπέδων είναι αρχιτεκτονική πολλαπλών επιπέδων (n-tier architecture). Πρόκειται για έναν τύπο αρχιτεκτονικής στην οποία τα τρία επίπεδα αποτελούν η παρουσίαση, η επεξεργασία καθώς και η διαχείριση δεδομένων.

Τα επίπεδα αυτά μπορούν να διατηρηθούν ως ανεξάρτητες υπομονάδες και τις περισσότερες φορές να υλοποιηθούν σε διαφορετικές πλατφόρμες. Γενικά η ανεξαρτησία των επιπέδων σημαίνει ότι είναι δυνατό, λόγω της ραγδαίας ανάπτυξης της τεχνολογίας, να μεταβληθεί ολοκληρωτικά ένα επίπεδο χωρίς να επηρεαστούν τα υπόλοιπα. Ακόμη κατά τον ίδιο τρόπο είναι δυνατή η προσθήκη ενός εντελώς νέου επιπέδου χωρίς την παραμικρή αλλαγή στα προηγούμενα.



Σχήμα 5. Αναπαράσταση μοντέλου τριών επιπέδων



Γενικά σε αυτή την αρχιτεκτονική, κάθε επίπεδο εξαρτάται από τα χαμηλότερα του επίπεδα, ενώ παρέχει υπηρεσίες και πληροφορίες στα ανώτερα του, χωρίς όμως να έχει γνώση για αυτά. Πιο συγκεκριμένα τα επίπεδα αναλύονται παρακάτω:

➤ **Πρώτο επίπεδο – επίπεδο διαχείρισης δεδομένων (Data Tier)**

Αποτελώντας το βασικότερο επίπεδο του συστήματος, ο Database Server παρέχει όλες τις απαραίτητες λειτουργίες για την αποθήκευση, ανάκτηση, ενημέρωση και συντήρηση των δεδομένων του συστήματος καθώς επίσης και όλους τους απαραίτητους μηχανισμούς για την ακεραιότητα και ασφάλεια των δεδομένων.

➤ **Δεύτερο επίπεδο – επίπεδο επεξεργασίας (Logic Tier)**

Αποτελεί το κύριο τμήμα του λογισμικού, στο οποίο εκτελούνται οι περισσότερες λειτουργίες, εκτός εκείνων που σχετίζονται με τη διαμόρφωση των οθονών εργασίας. Υπάρχει δυνατότητα εγκατάστασης περισσότερων του ενός Application Servers σε διαφορετικά μηχανήματα, αξιοποιώντας, με τον τρόπο αυτό, οποιαδήποτε διαθέσιμη υπολογιστική ισχύ και εξασφαλίζοντας εξαιρετικά αποτελέσματα ανταπόκρισης, αξιοπιστίας και επεκτασιμότητας.

➤ **Τρίτο επίπεδο – επίπεδο παρουσίασης (Presentation Tier)**

Το τρίτο επίπεδο του λογισμικού αποτελεί τη επαφή του χρήστη με το σύστημα (User Interface). Στο επίπεδο αυτό, πραγματοποιείται η διαχείριση των Οθονών Εργασίας (User Screens) καθώς επίσης και η μορφοποίηση των δεδομένων που εμφανίζονται.

## 2.3 Η αρχιτεκτονική του συστήματος Auth On The Road

Όπως ήδη αναφέρθηκε το σύστημα Auth On The Road θα ακολουθήσει την αρχιτεκτονική client-server καθώς και την αρχιτεκτονική τριών επιπέδων. Η επιλογή αυτή υπαγορεύεται από τα πλεονεκτήματα που διαθέτουν αυτά τα είδη αρχιτεκτονικής, όπως αναφέρθηκαν στις προηγούμενες δύο παραγράφους. Πιο συγκεκριμένα τα τρία επίπεδα που αναφέρθηκαν στην προηγούμενη παράγραφο θα αναλυθούν σύμφωνα με τις απαιτήσεις του παρόντος συστήματος:

➤ **Πρώτο επίπεδο – επίπεδο διαχείρισης δεδομένων (Data Tier)**

Το πρώτο επίπεδο περιλαμβάνει ένα υποσύστημα που αποτελείται από όλους τους εντολοδόχους των βάσεων δεδομένων. Σε αυτό το επίπεδο γίνεται εφικτή η επικοινωνία του συστήματος *Auth On The Road* με τις βάσεις δεδομένων του, έτσι ώστε να πραγματοποιηθούν οι εντολές εγγραφής και ανάγνωσης. Τελικά το υποσύστημα που αποτελεί αυτό το επίπεδο είναι το:

[Databases](#)

➤ **Δεύτερο επίπεδο – επίπεδο επεξεργασίας (Logic Tier)**

Το επίπεδο αυτό περιλαμβάνει όλους τους ελεγκτές που εκτελούν τις λειτουργίες με το κύριο υπολογιστικό κόστος του συστήματος. Πιο συγκεκριμένα τα υποσυστήματα του παρόντος επιπέδου παραλαμβάνουν τα αιτήματα των χρηστών από το πρώτο επίπεδο και με κατάλληλη επεξεργασία των δεδομένων προβαίνει στις κατάλληλες ενέργειες έτσι ώστε να αυτά τα αιτήματα να ικανοποιηθούν. Αναλυτικά τα υποσυστήματα που αποτελούν αυτό το επίπεδο είναι τα:

[LoginHandler,ExcursionCreateHandler,ApplicationFormHandler,ExcursionViewHandler,AudioVisualHandler](#)



➤ **Τρίτο επίπεδο – επίπεδο παρουσίασης (Presentation Tier)**

Το τρίτο και τελευταίο επίπεδο αποτελείται από όλα εκείνα τα υποσυστήματα που συνιστούν τη διεπαφή που είναι ορατή στο χρήστη. Τα υποσυστήματα του επιπέδου αυτού σε συνδυασμένη χρήση με τα υποσυστήματα των δύο επόμενων επιπέδων συνιστούν το σύνολο της εφαρμογής. Κάθε φορά ανάλογα με τη λειτουργικότητα που επιθυμεί ο χρήστης εμφανίζεται η ανάλογη γραφική διεπαφή και ανάλογα με την είσοδο που εισάγει ο χρήστης επιτυγχάνεται η συγκεκριμένη αυτή λειτουργία. Συγκεκριμένα τα υποσυστήματα που αποτελούν αυτό το επίπεδο είναι τα:

[LoginGUI,ExcursionCreateGUI,ApplicationFormGUI,ExcursionViewGUI,AudioVisualGUI](#)

### 3. Προτεινόμενη Αρχιτεκτονική Λογισμικού

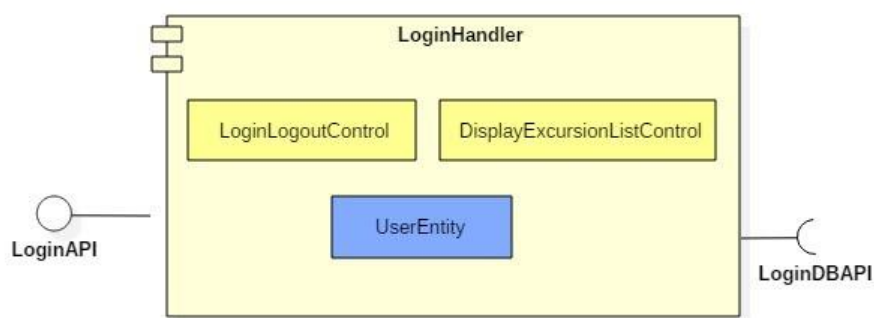
Το παρόν κεφάλαιο περιγράφει το μοντέλο σχεδίασης του συστήματος της εφαρμογής. Τα μοντέλα σχεδίασης είναι αφαιρετικά και έχουν οργανωτική δομή, με στόχο τη μείωση της πολυπλοκότητας του συστήματος μετατρέποντας σύνθετα προβλήματα σε απλά.

#### 3.1 Αποδόμηση Συστήματος

Πρόκειται για το διαχωρισμό του συστήματος σε επιμέρους υποσυστήματα και τμήματα, τα οποία φέρουν ξεχωριστές ευθύνες και λειτουργίες. Στην παράγραφο που ακολουθεί ορίζονται τα υποσυστήματα με τα τμήματα που εμπεριέχουν και αναλύεται η λειτουργικότητα καθενός από αυτά.

##### 3.1.1 Υποσύστημα LoginHandler

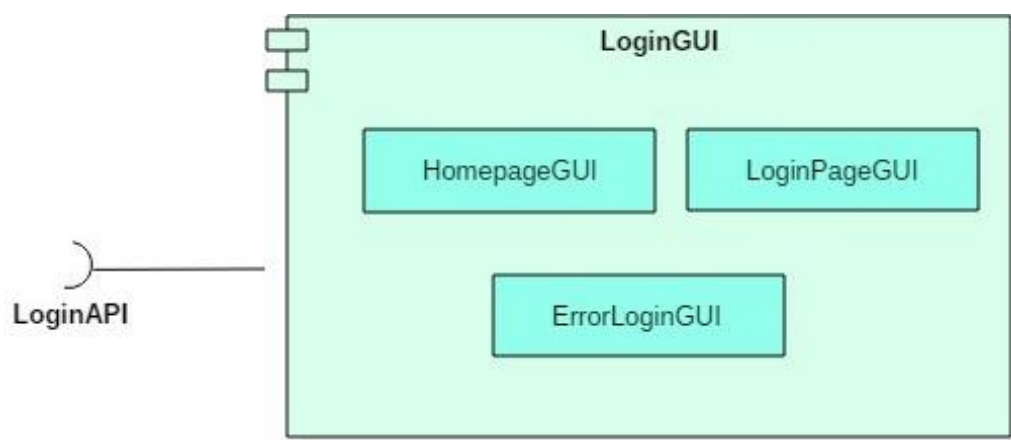
Το συγκεκριμένο υποσύστημα εμπεριέχει ως τμήματά του τον ελεγκτή που ελέγχει τη σύνδεση/αποσύνδεση του χρήστη στο σύστημα, τον ελεγκτή που χειρίζεται την αρχική σελίδα της εφαρμογής, καθώς και την οντότητα με τις πληροφορίες που αφορούν το χρήστη. Συγκεκριμένα, οι κλάσεις που περιλαμβάνει το υποσύστημα είναι οι LoginLogoutControl, DisplayExcursionListControl και η UserEntity.



Σχήμα 6.Υποσύστημα LoginHandler

##### 3.1.2 Υποσύστημα LoginGUI

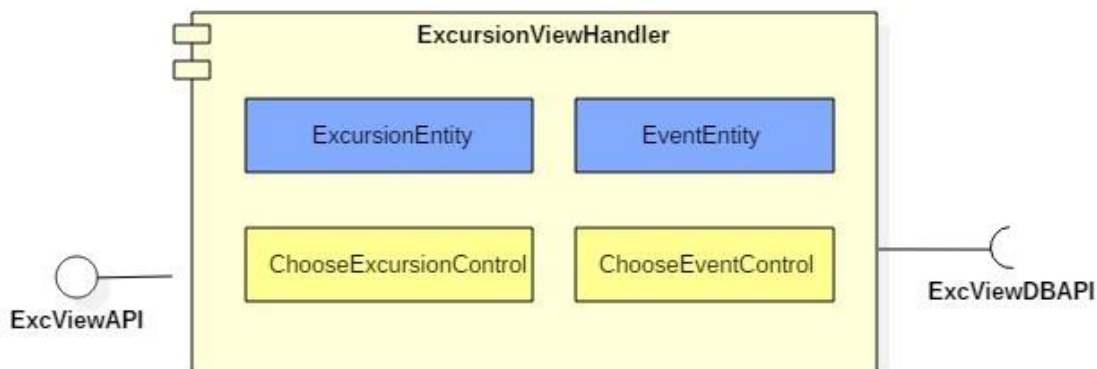
Το υποσύστημα αυτό αποτελείται από τις διεπαφές που συναντά ένας χρήστης πριν και αμέσως μετά την είσοδό του στο σύστημα. Αναλυτικότερα, οι κλάσεις που το απαρτίζουν είναι η LoginPageGUI, δηλαδή η σελίδα που εξυπηρετεί την είσοδο του χρήστη στο σύστημα, η ErrorGUI, που αφορά το μήνυμα σφάλματος το οποίο εμφανίζεται σε περίπτωση εισαγωγής λανθασμένων στοιχείων στα πεδία της LoginPageGUI και η HomepageGUI, που αποτελεί την αρχική σελίδα της εφαρμογής.



Σχήμα 7.Υποσύστημα LoginGUI

### 3.1.3 Υποσύστημα ExcursionViewHandler

Το παρόν υποσύστημα περιέχει τις κλάσεις που είναι απαραίτητες για την προβολή της σελίδας μιας εκδρομής και μιας εκδήλωσης. Οι κλάσεις αυτές είναι η *ChooseExcursionControl*, η *ChooseEventControl*, καθώς και οι αντίστοιχες κλάσεις οντοτήτων *ExcursionEntity* και *EventEntity*, οι οποίες περιλαμβάνουν στοιχεία που αφορούν τις εκδρομές και τις εκδηλώσεις.



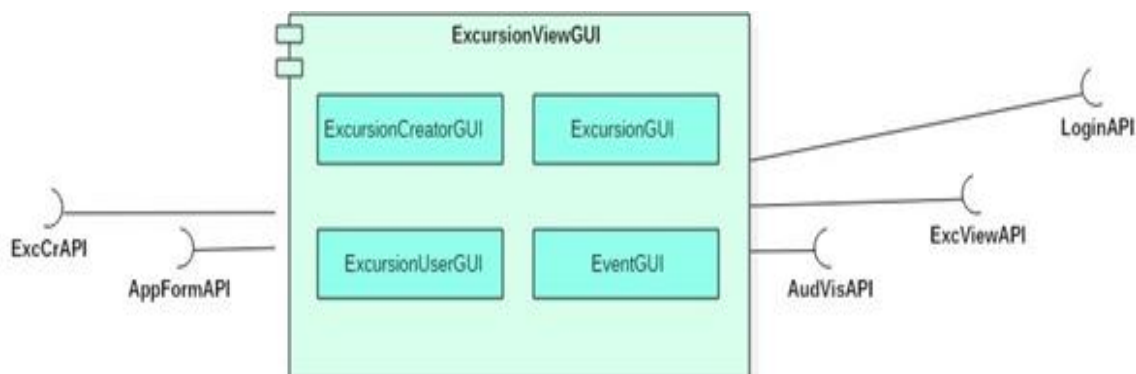
Σχήμα 8.Υποσύστημα ExcursionViewHandler

### 3.1.4 Υποσύστημα ExcursionViewGUI

Το υποσύστημα *ExcursionViewGUI* απαρτίζεται από τις διεπαφές προβολής των εκδρομών και των εκδηλώσεων, ανάλογα με την ιδιότητα του χρήστη που έχει συνδεθεί. Τα τμήματα του υποσυστήματος είναι:



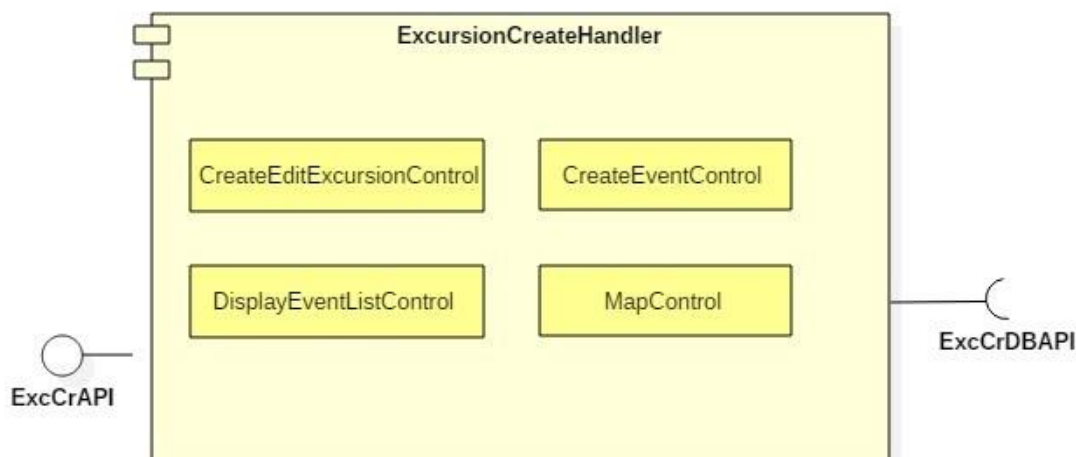
**1)**η κλάση ExcursionGUI, που κληρονομείται από τις **2)**ExcursionUserGUI και **3)**ExcursionCreatorGUI, οι οποίες αποτελούν αντίστοιχα τη σελίδα προβολής μιας εκδρομής από έναν απλό χρήστη και από το χρήστη-δημιουργό της και η **4)**EventGUI που αναπαριστά τη διεπαφή προβολής μιας εκδήλωσης.



Σχήμα 9.Υποσύστημα ExcursionViewGUI

### 3.1.5 Υποσύστημα ExcursionCreateHandler

Το υποσύστημα ExcursionCreateHandle περιέχει τους ελεγκτές που διαχειρίζονται τη δημιουργία και την τροποποίηση μιας εκδρομής (CreateEditExcursionControl) και μιας εκδήλωσης (CreateEventControl), την προβολή της λίστας διαθέσιμων εκδηλώσεων μιας εκδρομής (DisplayEventListControl) και την τοποθεσία μιας εκδήλωσης στο χάρτη (MapControl).

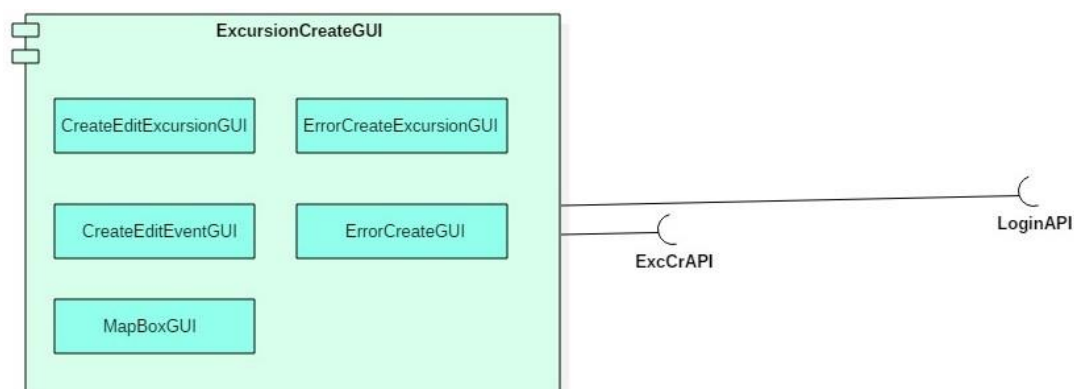


Σχήμα 10.Υποσύστημα ExcursionCreateHandler

### 3.1.6 Υποσύστημα ExcursionCreateGUI

Το συγκεκριμένο υποσύστημα αποτελείται από τις αντίστοιχες διεπαφές του ανωτέρου υποσυστήματος, ExcursionCreateHandle. Πιο συγκεκριμένα, περιέχει τις διεπαφές για τη δημιουργία και τροποποίηση εκδρομών και εκδηλώσεων (CreateEditExcursionGUI, CreateEditEventGUI), καθώς και τις αντίστοιχες διεπαφές για τα μηνύματα σφαλμάτων που διενεργούνται από το χρήστη κατά τη διαδικασία δημιουργίας/τροποποίησης (ErrorCreateExcursionGUI, ErrorCreateGUI). Επιπρόσθετως, συνυπάρχει στο ίδιο υποσύστημα και η διεπαφή προβολής της τοποθεσίας στο χάρτη, δηλαδή η κλάση MapBoxGUI.

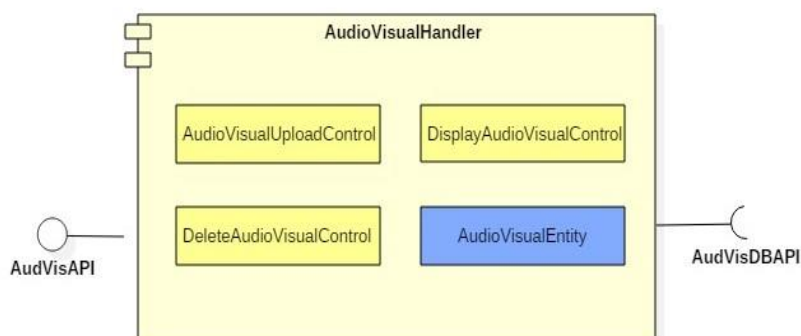




Σχήμα 11.Υποσύστημα ExcursionCreateGUI

### 3.1.7 Υποσύστημα AudioVisualHandler

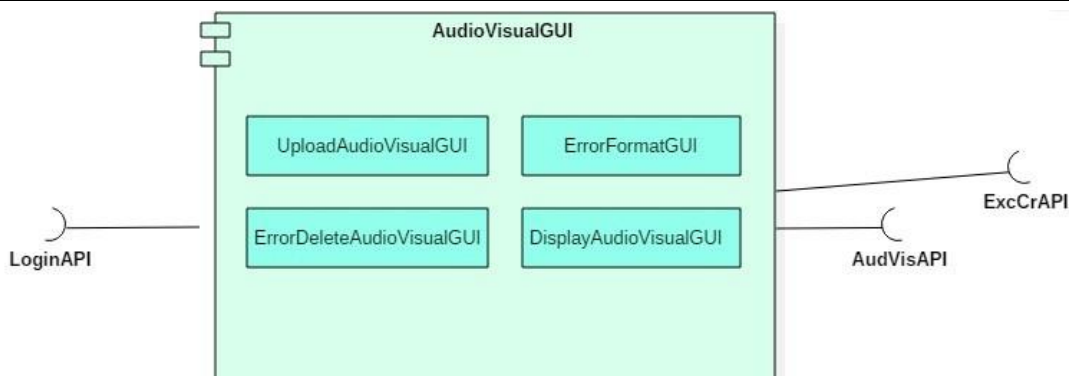
Πρόκειται για το υποσύστημα που αναλαμβάνει τη διαχείριση οπτικοακουστικού υλικού των εκδρομών. Οι ελεγκτές που το απαρτίζουν είναι υπεύθυνοι για την ανάρτηση αρχείων οπτικοακουστικού υλικού (**AudioVisualUploadControl**), για την προβολή του περιεχομένου τους (**DisplayAudioVisualControl**) και για τη διαγραφή τους (**DeleteAudioVisualControl**). Οι ελεγκτές αυτοί φέρουν εις πέρας τις αρμοδιότητές τους μέσω των στοιχείων που συλλέγουν από την κλάση οντότητας **AudioVisualEntity**.



Σχήμα 12.Υποσύστημα AudioVisualHandler

### 3.1.8 Υποσύστημα AudioVisualGUI

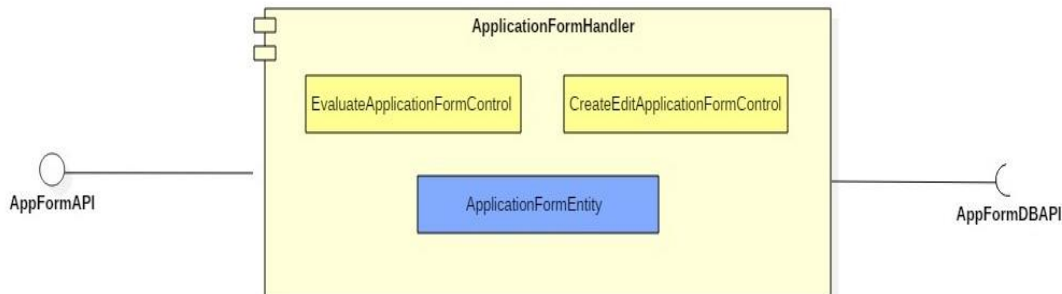
Πρόκειται για το υποσύστημα κλάσεων διεπαφών που σχετίζονται με αρχεία οπτικοακουστικού υλικού. Η διεπαφή **UploadAudioVisualGUI** αναπαριστά τη σελίδα που προβάλλεται στο χρήστη κατά τη διαδικασία επιλογής ενός αρχείου οπτικοακουστικού υλικού προς ανάρτηση στη σελίδα μιας εκδρομής, η διεπαφή **ErrorFormatGUI** αφορά το μήνυμα σφάλματος που εμφανίζεται κατόπιν επιλογής μη συμβατού τύπου αρχείου από το χρήστη, η **DisplayAudioVisualGUI** αναλαμβάνει την προβολή του περιεχομένου μιας εικόνας ή ενός βίντεο, ενώ η **ErrorDeleteAudioVisualGUI** εμφανίζει μήνυμα σφάλματος σε περίπτωση που ο χρήστης δεν έχει το δικαίωμα διαγραφής του αρχείου που επιθυμεί.



Σχήμα 13.Υποσύστημα AudioVisualGUI

### 3.1.9 Υποσύστημα ApplicationFormHandler

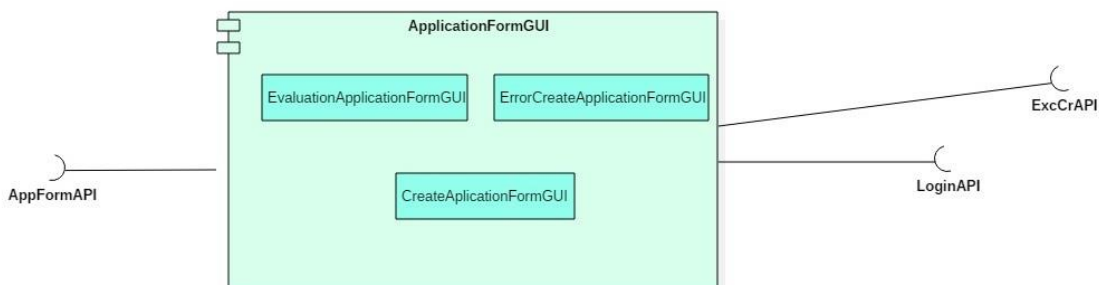
Το υποσύστημα αυτό είναι υπεύθυνο για τις αιτήσεις συμμετοχής σε μια εκδρομή, χειριζόμενο μέσω των τμημάτων του τη δημιουργία/αλλαγή μιας αίτησης συμμετοχής από έναν απλό χρήστη και την αξιολόγηση των αιτήσεων μιας εκδρομής από το χρήστη-δημιουργό της. Οι ελεγκτές που φέρουν τις προαναφερθείσες αρμοδιότητες είναι αντίστοιχα ο **CreateEditApplicationFormControl** και ο **EvaluateApplicationFormControl**, οι οποίοι λαμβάνουν όλες τις απαραίτητες πληροφορίες των αιτήσεων από την κλάση οντότητας **ApplicationFormEntity**.



Σχήμα 14.Υποσύστημα ApplicationFormHandler

### 3.1.10 Υποσύστημα ApplicationFormGUI

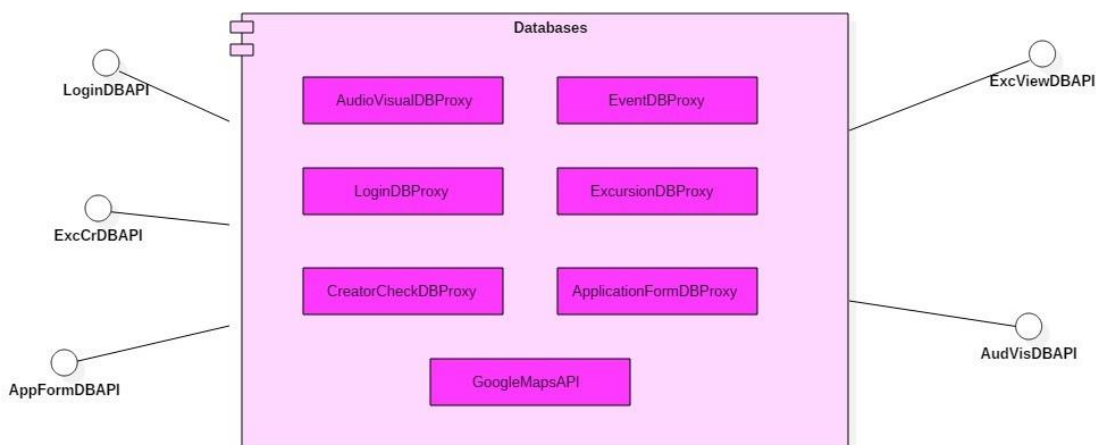
Οι διεπαφές, οι οποίες αναπαριστούν τις σχετιζόμενες με τις αιτήσεις συμμετοχής λειτουργίες, εμπεριέχονται στο υποσύστημα αυτό. Ονομαστικά, οι κλάσεις αυτές είναι η **CreateApplicationFormGUI**, που αφορά τη δημιουργία/αλλαγή μιας αίτησης συμμετοχής, η **ErrorCreateApplicationFormGUI**, για σχετικά μηνύματα ασφαμάτων προς το χρήστη και η **EvaluateApplicationFormGUI**, για τη σελίδα αξιολόγησης των αιτήσεων από το χρήστη-δημιουργό μιας εκδρομής.



Σχήμα 15.Υποσύστημα ApplicationFormGUI

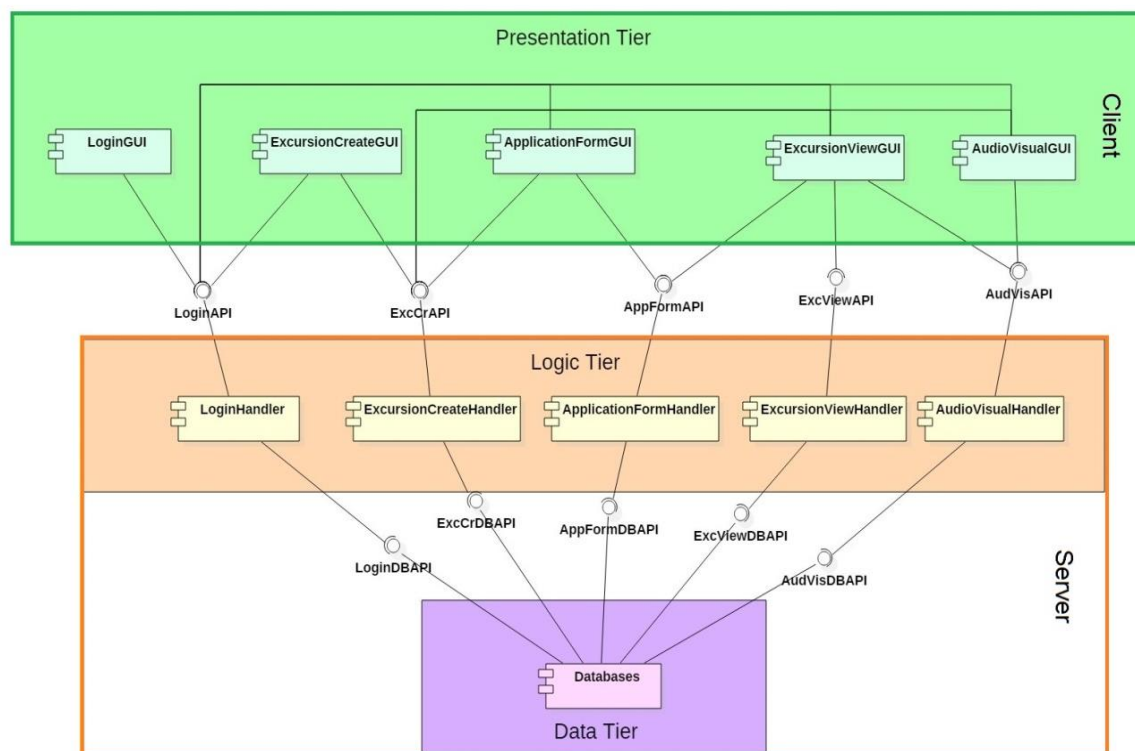
### 3.1.11 Υποσύστημα Databases

Όλα τα παραπάνω υποσυστήματα, που διαχειρίζονται λειτουργίες της εφαρμογής, επικοινωνούν με αντίστοιχες βάσεις δεδομένων. Οι τελευταίες με τη σειρά τους διαθέτουν αποθηκευμένες όλες τις πληροφορίες, που είναι απαραίτητες για την περάτωση των προαναφερθέντων λειτουργιών και συνεπώς, την ορθή εκτέλεση της εφαρμογής. Πιο αναλυτικά, οι βάσεις αυτές εξυπηρετούν την είσοδο του χρήστη στο σύστημα (**LoginDBProxy**), τον προσδιορισμό της ιδιότητάς του ως προς τα δικαιώματά του σε μια εκδρομή (**CreatorCheckDBProxy**), καθώς και τις δυνατότητες που αφορούν εκδρομές (**ExcursionDBProxy**), εκδηλώσεις (**EventDBProxy**), οπτικοακουστικό υλικό εκδρομών (**AudioVisualDBProxy**), αιτήσεις συμμετοχής (**ApplicationFormDBProxy**), αλλά και την απομακρυσμένη σύνδεση με εφαρμογή χαρτών (**GoogleMapsAPI**).



Σχήμα 16.Υποσύστημα Databases

Εν συνεχεία, παρατίθεται το διάγραμμα τμημάτων που παρουσιάζει τις εξαρτήσεις μεταξύ των επιμέρους υποσυστημάτων. Τα διαγράμματα τμημάτων αξιοποιούνται για επιμερισμό τμημάτων του συστήματος σε ομάδες ανάπτυξης, αλλά και προς έλεγχο για το αν παρέχεται η απαιτούμενη λειτουργικότητα από το σύστημα.



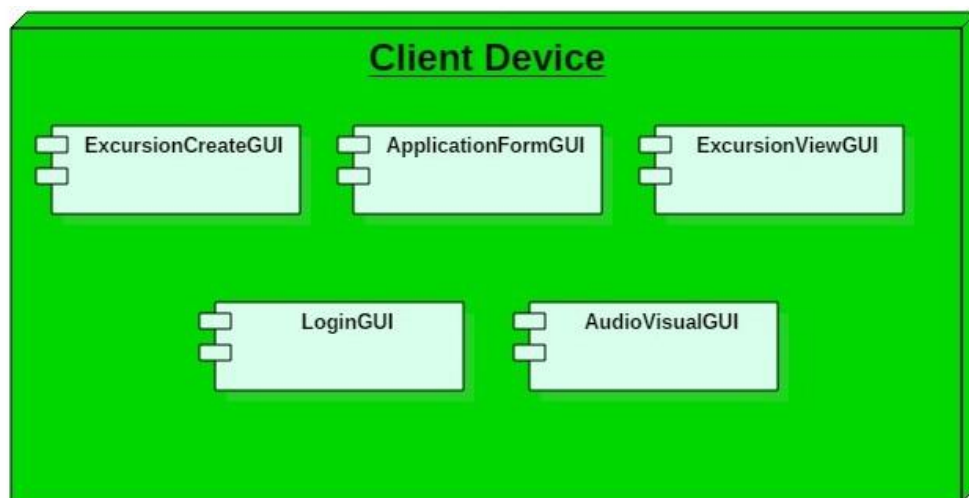
Σχήμα 17.Συνολικό διάγραμμα τμημάτων

## 3.2 Απεικόνιση Υλικού/Λογισμικού

Στη συνέχεια θα παρουσιαστούν με τα κατάλληλα διαγράμματα UML η απεικόνιση Υλικού/Λογισμικού. Πιο συγκεκριμένα, σκοπός αυτής της παραγράφου είναι η κατανόηση του τρόπου σύνδεσης των τμημάτων του συστήματος, καθώς και το πως συσχετίζονται τα τμήματα αυτά με το υλικό του συστήματος.

### 3.2.1 Client Device

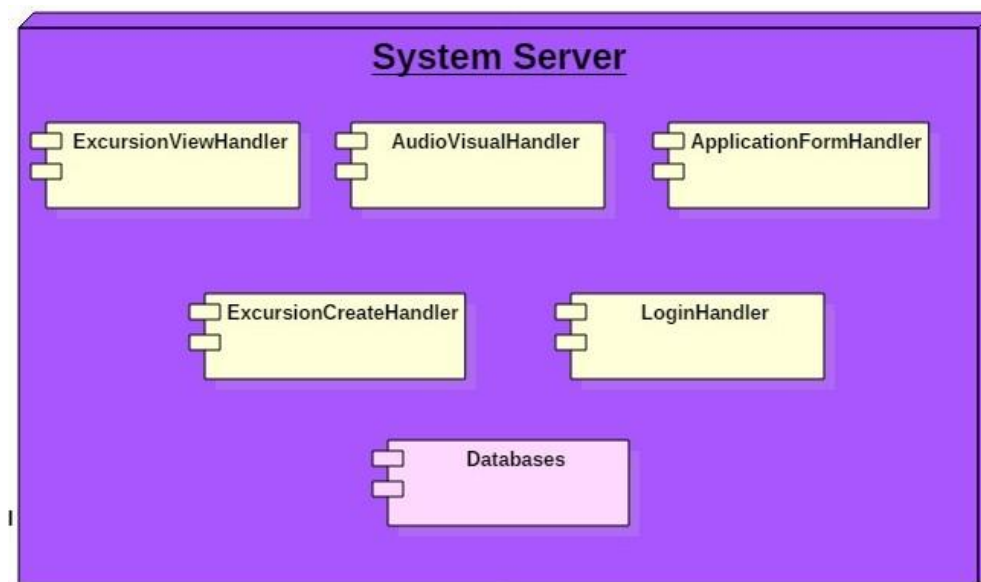
Πρόκειται για το κόμβο του συστήματος που αφορά αποκλειστικά τις διεπαφές που απαιτούνται από το χρήστη προκειμένου να χρησιμοποιηθεί η εφαρμογή. Η χρήση του γίνεται μέσω συγκεκριμένου υλικού, η οποία θα μπορούσε να είναι οποιαδήποτε φορητή συσκευή που διαθέτει σύνδεση στο διαδίκτυο. Η τελευταία είναι απαραίτητη προκειμένου ο χρήστης να μπορέσει να επικοινωνήσει με το κόμβο του server έτσι ώστε να παραλάβει ή να αποστείλει δεδομένα απαραίτητα για την εκπλήρωση του αιτήματος του χρήστη. Γενικά ο κόμβος αυτός αποτελείται αποκλειστικά από το γραφικό κομμάτι της εφαρμογής.



Σχήμα 18.Client Device

### 3.2.2 System Server

Σε αυτό το κόμβο ανήκουν όλα εκείνα τα υποσυστήματα που αναλαμβάνουν να επικοινωνήσουν με τον server του συστήματος έτσι ώστε να ικανοποιηθούν τα αιτήματα των χρηστών. Γενικά ο κόμβος αυτός οργανώνει την ορθή λειτουργία όλου του συστήματος (μέσω των handler υποσυστημάτων του) και την επικοινωνία με την εξωτερική βάση δεδομένων του συστήματος (μέσω των proxies).

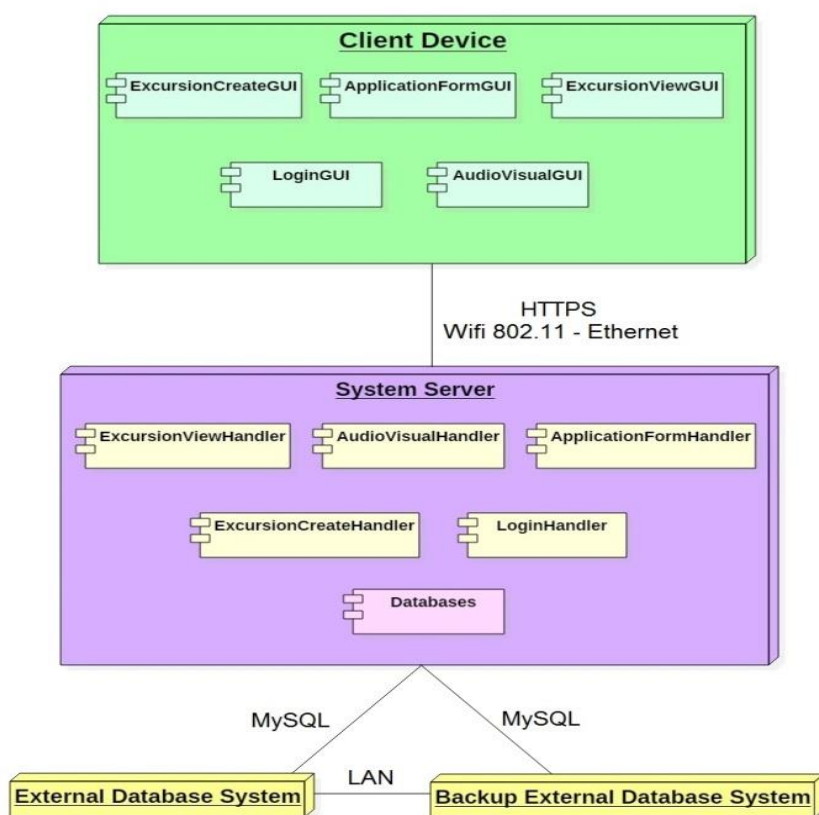


Σχήμα 19.System Server

### 3.2.3 Συνολικό διάγραμμα ανάπτυξης

Εν συνεχεία θα παρουσιαστεί ο τρόπος με τον οποίο επικοινωνούν οι διάφοροι κόμβοι του συστήματος. Αρχικά, ο κόμβος Client device και ο κόμβος System Server επικοινωνούν μέσω μέσω διαδικτύου. Το πρωτόκολλο με το οποίο επικοινωνούν οι δύο κόμβοι είναι το HTTPS. Το πρωτόκολλο HTTPS χρησιμοποιείται στην πληροφορική για να δηλώσει μία ασφαλή δικτυακή σύνδεση http. Ένας σύνδεσμος (URL) που αρχίζει με το πρόθεμα https υποδηλώνει ότι θα χρησιμοποιηθεί κανονικά το πρωτόκολλο HTTP, αλλά η σύνδεση θα γίνει σε διαφορετική πόρτα (443 αντί 80) και τα δεδομένα θα ανταλλάσσονται κρυπτογραφημένα. Ακόμη πρέπει να τονιστεί ότι αναφέρεται στον συνδυασμό του απλού HTTP πρωτοκόλλου και των δυνατοτήτων κρυπτογράφησης που παρέχει το πρωτόκολλο Secure Sockets Layer (SSL). Η κρυπτογράφηση που χρησιμοποιείται διασφαλίζει ότι τα κρυπτογραφημένα δεδομένα δεν θα μπορούν να υποκλαπούν από άλλους κακόβουλους χρήστες ή από επιθέσεις man-in-the-middle. Τέλος, η σύνδεση μεταξύ client και server γίνεται είτε με WiFi είτε με Ethernet.

Συνεχίζοντας την ανάλυση στο παρακάτω διάγραμμα έχει προσθεθεί η εξωτερική βάση δεδομένων. Όπως φαίνεται σε αυτό η επικοινωνία μεταξύ των συστημάτων γίνεται μέσω της MySQL, ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων ιδιαίτερα δημοφιλές για διαδικτυακά προγράμματα και ιστοσελίδες. Επιπλέον κρίθηκε αναγκαία η ύπαρξη ενός backup εξωτερικού συστήματος, για περιπτώσεις απώλειας δεδομένων. Η επικοινωνία μεταξύ του εξωτερικού συστήματος και του backup του γίνεται μέσω LAN (Local Area Network). Παρακάτω δίνεται το αντίστοιχο διάγραμμα:



Σχήμα 20. Διάγραμμα ανάπτυξης



**Στη συνέχεια παρατίθεται οι ελάχιστες και οι προτεινόμενες απαιτήσεις σε επίπεδο χρήστη και server :**

	Minimum Requirements	Recommended Requirements
Client	CPU: Dual core 1.9 GHZ	CPU: Dual core 2.5 GHZ
	RAM:1 GB	RAM:2 GB
	Video Card: nVidia 9600GT	Video Card: nVidia Geforce GTX 660
Server	CPU: i3 2.7 GHZ	CPU: i7 4.4 GHZ
	RAM:4 GB	RAM:8 GB
	Video Card: nVidia Geforce GTX 970	Video Card: nVidia Geforce GTX Titan X

### 3.3 Διαχείριση Μόνιμων Δεδομένων

Η διαχείριση μόνιμων δεδομένων περιγράφει τα δεδομένα που αποθηκεύονται από το σύστημα και την υποδομή που χρειάζεται για την αποθήκευσή τους.

Η διαχείριση των μόνιμων δεδομένων γίνεται με την χρήση σχεσιακών βάσεων δεδομένων, καθώς επιτρέπουν την ταυτόχρονη πρόσβαση και αποθήκευση των δεδομένων από πολλούς χρήστες καθώς και την μεταφορά των δεδομένων σε διαφορετικούς τύπους πλατφόρμας. Η υλοποίηση της βάσης δεδομένων γίνεται με τη MySQL, καθώς είναι εύκολη στη χρήση της και μπορεί να ικανοποιήσει τις απαιτήσεις του συστήματος.

Η επικοινωνία του συστήματος με τις βάσεις δεδομένων, οι οποίες είναι εξωτερικές του συστήματος γίνεται μέσω των κατάλληλων κλάσεων διεπαφής, οι οποίες είναι οι ακόλουθες:

- **LoginDBProxy:** Η κλάση αυτή χρησιμοποιείται για επικοινωνία με τη βάση δεδομένων χρηστών. Στη βάση αυτή βρίσκονται τα στοιχεία όλων των χρηστών, οι οποίοι ανήκουν στην πανεπιστημιακή κοινότητα και έχουν δικαίωμα πρόσβασης στο σύστημα. Χρησιμοποιείται επομένως για την ταυτοποίηση των στοιχείων που δίνονται κατά τη διαδικασία της σύνδεσης στην εφαρμογή.
- **CreatorCheckDBProxy:** Η κλάση αυτή επικοινωνεί με τη βάση δεδομένων εκδρομών και χρησιμοποιείται προκειμένου να ελεγχθεί εάν ο συνδεδεμένος χρήστης είναι ο χρήστης-δημιουργός μιας εκδρομής.
- **ExcursionDBProxy:** Η κλάση αυτή χρησιμοποιείται για επικοινωνία με τη βάση δεδομένων των εκδρομών.
- **EventDBProxy:** Αυτή η κλάση χρησιμοποιείται για την επικοινωνία με τη βάση δεδομένων των εκδηλώσεων.
- **ApplicationFormDBProxy:** Η κλάση αυτή επικοινωνεί με απομακρυσμένη βάση δεδομένων όπου είναι αποθηκευμένες οι αιτήσεις συμμετοχής των συμμετεχόντων.
- **AudioVisualDBProxy:** Η κλάση αυτή επικοινωνεί με απομακρυσμένη βάση δεδομένων όπου είναι αποθηκευμένο το οπτικοακουστικό υλικό των εκδρομών.





### 3.4 Έλεγχος Πρόσβασης και Ασφάλεια

Ο έλεγχος πρόσβασης στο σύστημα γίνεται με την επιβεβαίωση ενός username και ενός password, τα οποία εισάγει ο χρήστης στην σελίδα σύνδεσης και, εφόσον είναι έγκυρα, εισέρχεται στην εφαρμογή. Το username και το password τα έχει ορίσει ο χρήστης κατά την εγγραφή του στη σχολή και την απόκτηση πανεπιστημιακού λογαριασμού. Τα δεδομένα χρηστών διατηρούνται στη βάση δεδομένων κρυπτογραφημένα.

Το σύστημα περιλαμβάνει δύο είδη χρηστών, τον απλό χρήστη και τον χρήστη δημιουργό. Πρόκειται επομένως για σύστημα πολλών χρηστών και διαφορετικοί δράστες έχουν δικαίωμα πρόσβασης σε διαφορετική λειτουργικότητα και διαφορετικά δεδομένα. Για να εξασφαλιστεί η εκτέλεση ενεργειών από κάθε χρήστη στις οποίες έχει εξουσιοδότηση, το σύστημα εμφανίζει για κάθε χρήστη το κατάλληλο γραφικό περιβάλλον που του επιτρέπει να πραγματοποιήσει μόνον ενέργειες για τις οποίες έχει δικαίωμα.

Ο έλεγχος πρόσβασης και η ασφάλεια περιγράφουν τη μοντελοποίηση των χρηστών του συστήματος με τη μορφή ενός πίνακα πρόσβασης (access matrix), ο οποίος παρουσιάζεται παρακάτω.

User	LoginPageGUI	ErrorLoginGUI	HomePageGUI
	loginPageGUI()	display()	HomePageGUI()
	loginClick()	backClick()	logoutClick()
			createExcursionClick()
			chooseExcursionClick()
			scrollClick()
			display()
	<b>ExcursionUserGUI*</b>	<b>ExcursionGUI</b>	<b>EventGUI</b>
	ExcursionUserGUI()	ExcursionGUI()	EventGUI()
	CreateEditApplicationFormClick()	AudioVisualClick()	CancelClick()
		UploadAudioVisualClick()	MapClick()
		ChooseEventsClick()	HomeClick()
		HomeClick()	
		ScrollClick()	
		display()	
	<b>CreateApplicationFormGUI*</b>	<b>ErrorCreateApplicationFormGUI*</b>	<b>UploadAudioVisualGUI</b>





	CreateApplicationFormGUI()	display()	UploadAudioVisualGUI()
	preFillFields()	backClick()	UploadClick()
	SubmitClick()		CancelClick()
	DeleteClick()		BrowseClick()
	CancelClick()		display()
	ScrollClick()		
	display()		
	<b>ErrorFormatGUI</b>	<b>DisplayAudioVisualGUI</b>	<b>MapBoxGUI</b>
	display()	DisplayAudioVisulaGUI()	showMap()
	backClick()	PlayClick()	
		SoundClick()	
		DeleteClick()	
		CancelClick()	
		HomeClick()	
		display()	
	<b>ErrorDeleteAudioVisualGUI</b>		
	display()		
	backClick()		

<b>Creator</b>	<b>HomePageGUI</b>	<b>ExcursionCreatorGUI</b>	<b>CreateEditExcursionGUI</b>
	editExcursionClick()	ExcursionCreatorGUI()	CreateEditExcursionGUI()
		EvaluateApplicationsClick()	preFillFields()
			CreateEventClick()
			SubmitClick()
			CancelClick()



			EditEventClick()
			displayExcursionPage()
	<b>ErrorCreateExcursionGUI</b>	<b>CreateEditEventGUI</b>	<b>ErrorCreateEventGUI</b>
	display()	CreateEditEventGUI()	display()
	backClick()	preFillFields()	backClick()
		SubmitClick()	
		CancelClick()	
		display()	
	<b>EvaluationApplicationFormGUI</b>		
	EvaluationApplicationFormGUI()		
	ApprovalClick()		
	DeclineClick()		
	NextClick()		
	HomeClick()		

Στο σύστημα υπάρχει ένα είδος ιεραρχίας και κληρονομικότητας. Έτσι λοιπόν, ο User είναι ο απλούστερος χρήστης του συστήματος, ο Creator κληρονομεί τις λειτουργίες του User εκτός από τις λειτουργίες του χρήστη που βρίσκονται στις κλάσεις που δηλώνονται με \* στον πίνακα πρόσβασης και έχει κάποιες επιπλέον ιδιότητες διαχείρισης.

### 3.5 Γενικός έλεγχος λογισμικού

Στο μέρος αυτό θα περιγράψει πως τα αιτήματα ξεκινούν και πως τα υποσυστήματα συγχρονίζονται. Επίσης αναγράφονται θέματα συγχρονισμού και νημάτων.

Ο χρήστης μπορεί μέσω των γραφικών διεπαφών να επιλέξει κάποια λειτουργία του συστήματος. Για την υλοποίηση της λειτουργίας καλείται κάποιο αντικείμενο ελέγχου που είναι υπεύθυνο γι' αυτή και στη συνέχεια εκτελείται η λειτουργία ή μπορεί να προηγηθεί η κλήση ενός νέου αντικείμενου ελέγχου. Κατά τη διάρκεια εκτέλεσης υπάρχει συχνά επικοινωνία με ορισμένα αντικείμενα Proxy για την ανάκτηση ή την αποθήκευση επιθυμητών δεδομένων.

Στο σύστημα δεν υπάρχει ένα μόνο αντικείμενο ελέγχου αλλά πολλαπλά αντικείμενα ελέγχου, που το καθένα είναι υπεύθυνο για κάποια λειτουργία, επομένως ο έλεγχος είναι καταναμημένος. Τα διαγράμματα ακολουθιών έχουν κυρίως μορφή σκάλας και επομένως επιλέγεται το αποκεντρωμένο μοντέλο σχεδίασης για τον γενικό έλεγχο του λογισμικού.



Στο σύστημα υπάρχουν δύο χρήστες, ο απλός χρήστης και ο χρήστης δημιουργός κάποιας εκδρομής. Δικαίωμα στην τροποποίηση των δεδομένων μίας εκδρομής ή εκδήλωσης έχει μόνον ο δημιουργός της και αντίστοιχα δικαίωμα στην τροποποίηση μίας αίτησης συμμετοχής έχει μόνο αυτός που την υπέβαλε, επομένως δεν υπάρχει περίπτωση ταυτόχρονης μεταβολής των στοιχείων από δύο χρήστες, δηλαδή ταυτόχρονη εγγραφή στην ίδια θέση μνήμης. Συνεπώς η εφαρμογή χρειάζεται δημιουργία πολλαπλών νημάτων για να υποστηρίξει την ταυτόχρονη αποστολή αιτημάτων από τους χρήστες. Επίσης δεν υπάρχουν προβλήματα συγχρονισμού που θα έπρεπε να διευθετηθούν για την σωστή λειτουργία της εφαρμογής.

### 3.6 Οριακές Συνθήκες

Για την εξασφάλιση της σωστής λειτουργίας και της αξιοπιστίας του συστήματος πρέπει να προσδιοριστεί πλήρως η απόκριση του στις οριακές συνθήκες.

#### 3.6.1 Έναρξη (start-up) λογισμικού

Η εκκίνηση του συστήματος απαιτεί την σύνδεση στο διαδίκτυο για την χρήση λογισμικού **Auth On The Road** στις διαφορετικές συσκευές που θα χρησιμοποιούν οι χρήστες. Η είσοδος στο σύστημα γίνεται με την εισαγωγή username και password, τα οποία έχουν δοθεί στους χρήστες κατά την εγγραφή τους στην πανεπιστημιακή κοινότητα του ΑΠΘ. Για τη σωστή λειτουργία της εφαρμογής, κατά την εκκίνηση της θα πρέπει η εφαρμογή να συνδεθεί με επιτυχία με τις βάσεις δεδομένων, μέσω του διαδικτύου, και με την σειρά του θα πρέπει οι βάσεις δεδομένων να συνδεθούν με τα συνεργαζόμενα εξωτερικά συστήματα όπως το Google Maps.

<b>Σύντομη περιγραφή:</b>	Στόχος του σεναρίου χρήσης είναι η εκκίνηση του συστήματος.
<b>Πυροδότηση Δραστηριότητας:</b>	Επιλογή του εικονιδίου της εφαρμογής.
<b>Προϋποθέσεις:</b>	Ο χρήστης πρέπει να έχει εγκαταστήσει την εφαρμογή στη συσκευή του.

<b>Βασική Ροή:</b>		
<b>Γραμμή</b>	<b>Ενέργεια χρήστη συστήματος</b>	<b>Απάντηση συστήματος</b>
1.	Ο χρήστης πατάει το εικονίδιο της εφαρμογής.	Το σύστημα ελέγχει αν υπάρχει σύνδεση στο διαδίκτυο.
2.		Το σύστημα συνδέεται με τις βάσεις δεδομένων και το εξωτερικό σύστημα Google Maps.
<b>Μετάπειτα κατάσταση:</b>	Το σύστημα εμφανίζει τη σελίδα σύνδεσης.	

<b>Εναλλακτική Ροή (EP-1): Το σύστημα δεν μπορεί να συνδεθεί με τις βάσεις δεδομένων</b>		
Εφόσον το σύστημα στη γραμμή 2 στη Βασική Ροή δεν συνδεθεί με τα εξωτερικά συστήματα, πρέπει να επιστρέψει σχετικό μήνυμα σφάλματος		
<b>Γραμμή</b>	<b>Ενέργεια χρήστη συστήματος</b>	<b>Απάντηση συστήματος</b>
1.		Το σύστημα εμφανίζει μήνυμα σφάλματος
Το σενάριο χρήσης επιστρέφει στη γραμμή 1 της βασικής ροής.		



### 3.6.2 Τερματισμός (shutdown) λογισμικού

Η εφαρμογή **Auth On The Road** τερματίζεται όταν πατηθεί το πλήκτρο «X». Κατά τον τερματισμό του συστήματος ελέγχεται αν υπάρχουν διαδικασίες που είναι σε εξέλιξη και αναμένεται ο τερματισμός τους. Κατά τον τερματισμό τους αποθηκεύονται προσωρινά δεδομένα που κρίνονται απαραίτητα στην μνήμη και τέλος αποδεσμεύονται οι πόροι που δέσμευε η εφαρμογή, συμπεριλαμβανομένου και του χώρου που καταλάμβανε στην μνήμη RAM. Κατά τον τερματισμό της εφαρμογής ο χρήστης αποσυνδέεται αυτόματα.

<b>Σύντομη περιγραφή:</b>	Στόχος του σεναρίου χρήσης είναι ο τερματισμός του συστήματος.
<b>Πυροδότηση Δραστηριότητας:</b>	Επιλογή του κουμπιού «X».
<b>Προϋποθέσεις:</b>	Ο χρήστης πρέπει να έχει εκκινήσει την εφαρμογή.

<b>Βασική Ροή:</b>		
<b>Γραμμή</b>	<b>Ενέργεια χρήστη συστήματος</b>	<b>Απάντηση συστήματος</b>
<b>1.</b>	Ο χρήστης πατάει το «X».	Το σύστημα αποσυνδέεται από τα εξωτερικά συστήματα.
<b>2.</b>		Το σύστημα αποσυνδέει τους πόρους που δέσμευε.
<b>Μετάπειτα κατάσταση:</b>	Κλείνει η σελίδα της εφαρμογής.	

### 3.6.3 Διακοπή τροφοδοσίας

Στην περίπτωση διακοπής τροφοδοσίας της συσκευής δεν προβλέπεται ανάκτηση των δεδομένων που είχαν υποβληθεί χωρίς να γίνει μόνιμη αποθήκευση τους, καθώς δεν υπάρχει δυνατότητα αποθήκευσης στοιχείων χωρίς την αποστολή τους στην βάση δεδομένων.

### 3.6.4 Σφάλματα Λογισμικού

Σφάλματα λογισμικού μπορούν να συμβούν κατά την διάρκεια προσπέλασης μίας βάσης δεδομένων για ενημέρωσή της ή για ανάκτηση δεδομένων από αυτή. Επίσης μπορούν να συμβούν κατά τη διάρκεια της εκτέλεσης κάποιας ενέργειας η οποία απαιτεί επικοινωνία του συστήματος με ένα εξωτερικό σύστημα, όπως το GoogleMaps. Τα σφάλματα λογισμικού προκαλούνται όταν προκύπτει σφάλμα κατά την σύνδεση και την επικοινωνία της εφαρμογής με τα εξωτερικά συστήματα.

Επομένως σε όλες τις περιπτώσεις που προκύπτει σφάλμα πρόσβασης στη βάση δεδομένων ή σε κάποιο εξωτερικό σύστημα πρέπει να γίνουν τρεις βασικές ενέργειες. Το σύστημα εμφανίζει μήνυμα σφάλματος, το σύστημα κάνει επανεκκίνηση για να αποκαταστήσει το σφάλμα και τέλος επιστρέφει στην ίδια σελίδα που ήταν ο χρήστης όταν προέκυψε το σφάλμα.



<b>Σύντομη περιγραφή:</b>	Στόχος του σεναρίου χρήσης είναι η αντιμετώπιση σφαλμάτων λογισμικού.
<b>Πυροδότηση Δραστηριότητας:</b>	Το σύστημα δεν μπορεί να συνδεθεί με τα εξωτερικά συστήματα.
<b>Προϋποθέσεις:</b>	Ο χρήστης πρέπει να έχει εκκινήσει την εφαρμογή.

<b>Βασική Ροή:</b>		
<b>Γραμμή</b>	<b>Ενέργεια χρήστη συστήματος</b>	<b>Απάντηση συστήματος</b>
<b>1.</b>	Ο χρήστης ζητάει αποθήκευση δεδομένων στη βάση ή ανάκτηση δεδομένων από τη βάση είτε από το εξωτερικό σύστημα Google Maps.	Το σύστημα αδυνατεί να επικοινωνήσει με το εξωτερικό σύστημα.
<b>2.</b>		Το σύστημα εμφανίζει μήνυμα σφάλματος.
<b>3.</b>		Γίνεται επανεκκίνηση του συστήματος.
<b>Μετάπειτα κατάσταση:</b>	Το σύστημα μεταβαίνει στη σελίδα που ήταν ο χρήστης όταν συνέβη το σφάλμα.	

### 3.6.5 Ενημέρωση λογισμικού και συντήρηση συστήματος

Κατά καιρούς η ομάδα ανάπτυξης θα προχωρά σε ενημερώσεις του λογισμικού, με στόχο τη βελτίωση πιθανών αδυναμιών ή ακόμα προσθήκη χαρακτηριστικών. Όσο καιρό διαρκεί αυτή η ενημέρωση του συστήματος κανένας χρήστης δεν θα μπορεί να χρησιμοποιήσει την εφαρμογή. Θα ενημερώνεται όμως με σχετικό μήνυμα για τον λόγο που δεν μπορεί να χρησιμοποιήσει την εφαρμογή και τότε θα μπορεί και πάλι να την χρησιμοποιήσει, κατά την διαδικασία σύνδεσης του (login).

Επομένως στο σενάριο χρήσης Συνδέομαι στο σύστημα πρέπει να προστεθεί η παρακάτω εναλλακτική ροή.

<b>Εναλλακτική Ροή (EP-2):</b> Γίνεται ενημέρωση λογισμικού.		
Εάν στη γραμμή 1 στη Βασική Ροή ο χρήστης επιλέξει «σύνδεση» κατά τη χρονική περίοδο στην οποία γίνεται ενημέρωση του λογισμικού.		
<b>Γραμμή</b>	<b>Ενέργεια χρήστη συστήματος</b>	<b>Απάντηση συστήματος</b>
<b>1.</b>		Το σύστημα εμφανίζει ενημερωτικό μήνυμα.
<b>Μετάπειτα κατάσταση:</b>	Το σύστημα παραμένει στη σελίδα σύνδεσης της εφαρμογής.	

### 3.6.6 Υπερφόρτωση συστήματος

Σε περιπτώσεις όπου συνδέονται πάρα πολλοί χρήστες στο σύστημα, ή όταν πολλοί χρήστες προσπαθούν ταυτόχρονα να εκτελέσουν κάποιες ενέργειες, υπάρχει η πιθανότητα να υπερφορτωθεί το σύστημα. Σε μια τέτοια περίπτωση, για να αποφευχθεί η κατάρρευση του συστήματος αλλά και για να επιτευχθεί μια λύση με το λιγότερο δυνατό οικονομικό βάρος, το σύστημα αυτόματα θα μειώνει την ποιότητα των παρεχόμενων υπηρεσιών.