



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

8ο Εξάμηνο
Άνοιξη 2020



Απαιτήσεις Χρηστών και Προδιαγραφές Συστήματος

Del.1.1

Version 0.1

Κούτρα Χαριτίνη charitink@ece.auth.gr
Κωστοπούλου Ναταλία nkostopou@ece.auth.gr
Νάστος Βίκτωρ viktorna@ece.auth.gr
Πορτοκαλίδης Σταύρος stavport@ece.auth.gr

06/04/2020



Μέλη της Ομάδας Ανάπτυξης

Όνομα	ΟΑ	Email
Α. Συμεωνίδης		asymeon@issel.ee.auth.gr
Κούτρα Χαριτίνη	ETHER (Ομάδα 18)	charitink@ece.auth.gr
Κωστοπούλου Ναταλία	ETHER (Ομάδα 18)	nkostopou@ece.auth.gr
Νάστος Βίκτωρ	ETHER (Ομάδα 18)	viktorna@ece.auth.gr
Πορτοκαλίδης Σταύρος	ETHER (Ομάδα 18)	stavport@ece.auth.gr



ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

Πίνακας Περιεχομένων	3
Λίστα Σχημάτων	4
Εισαγωγικά.....	5
1 Απαιτήσεις Συστήματος.....	6
1.1 Λειτουργικές απαιτήσεις (Σενάρια χρήσης)	6
1.2 Χρήστες και εξωτερικά συστήματα	9
1.2.1 <Χρήστης 1>	9
1.2.2 <Εξωτερικό Σύστημα 1>.....	9
1.3 Σημαντικές μη λειτουργικές απαιτήσεις	10
1.1 Απαιτήσεις επίδοσης.....	10
1.2 Απαιτήσεις ασφάλειας (Security)	10
1.3 Απαιτήσεις Χρηστικότητας (Usability)	10
1.4 Απαιτήσεις Φορητότητας (Portability)	11
1.5 Απαιτήσεις Αξιοπιστίας (Reliability).....	11
1.6 Τεχνικές Απαιτήσεις περιβάλλοντος.....	11
1.7 Απαιτήσεις πολιτικής (Policy - legal, corporate)	12
1.10 Λεξιλογικοί Προσδιορισμοί	13
2 Σενάρια Χρήσης	14
2.1 Διάγραμμα σεναρίων χρήσης.....	14
2.2 <Όνομα πακέτου σεναρίων χρήσης > (Feature με βάση την ορολογία Gherkin).....	15
2.2.1<Feature: Search flights >.....	15
2.2.2<Feature: Sort results>	16
2.2.3<Feature: Apply filters>	17
2.2.4<Feature: Choose flight>	18
2.2.5<Feature: Check price changes >	19
2.2.6<Feature: Evaluate airlines and travel agencies >.....	20
2.2.7<Feature: Check evaluation of airlines and travel agencies>.....	21
2.2.8<Feature: Search trip within a budget >	22
2.2.9<Feature: Connect with airlines and travel agencies >.....	23
2.2.10<Feature: Propose destinations to logged-in user >	24
3 Επιδεικτικά γραφικά παράθυρα διεπαφής.....	26
3.1 <Παράθυρο αναζήτησης>	26
3.2 < Παράθυρο Αναζήτησης με βάση budget και Παράθυρο Αποτελεσμάτων >.....	27
3.3 < Παράθυρα επιλογής φίλτρων >.....	28
3.4 < Παράθυρο επιλεγμένης πτήσης και Παράθυρο αξιολόγησης του παρόχου>.....	29
3.5 < Παράθυρα αξιολόγησης παρόχων >.....	30
3.6 < Παράθυρο παρακολούθησης μεταβολής τιμών >.....	31
3.7 < Παράθυρο Σφάλματος και Παράθυρο φόρτωσης σελίδας>.....	32
3.8 < Παράθυρο αναζήτησης πτήσης για web browser >.....	33
4 Στατική Μοντελοποίηση	34
4.1 <Πακέτα λεξιλογίου σεναρίων υψηλής προτεραιότητας>	34
4.1.1 Πακέτο SearchFlight	34



4.1.2 Πακέτο ResultHandle	41
4.2 <Πακέτο λεξιλογίου σεναρίων μέσης προτεραιότητας>	49
4.2.1 Πακέτο SupplierHandle	49
4.3 <Πακέτα λεξιλογίου σεναρίων χαμηλής προτεραιότητας >	57
4.3.1 Πακέτο ProposeTravel	57
4.3.2 Πακέτο PriceChange	62
Παράρτημα Ι – Γλωσσάριο	69
Παράρτημα ΙΙ – Ανοιχτά Θέματα	70



ΛΙΣΤΑ ΣΧΗΜΑΤΩΝ

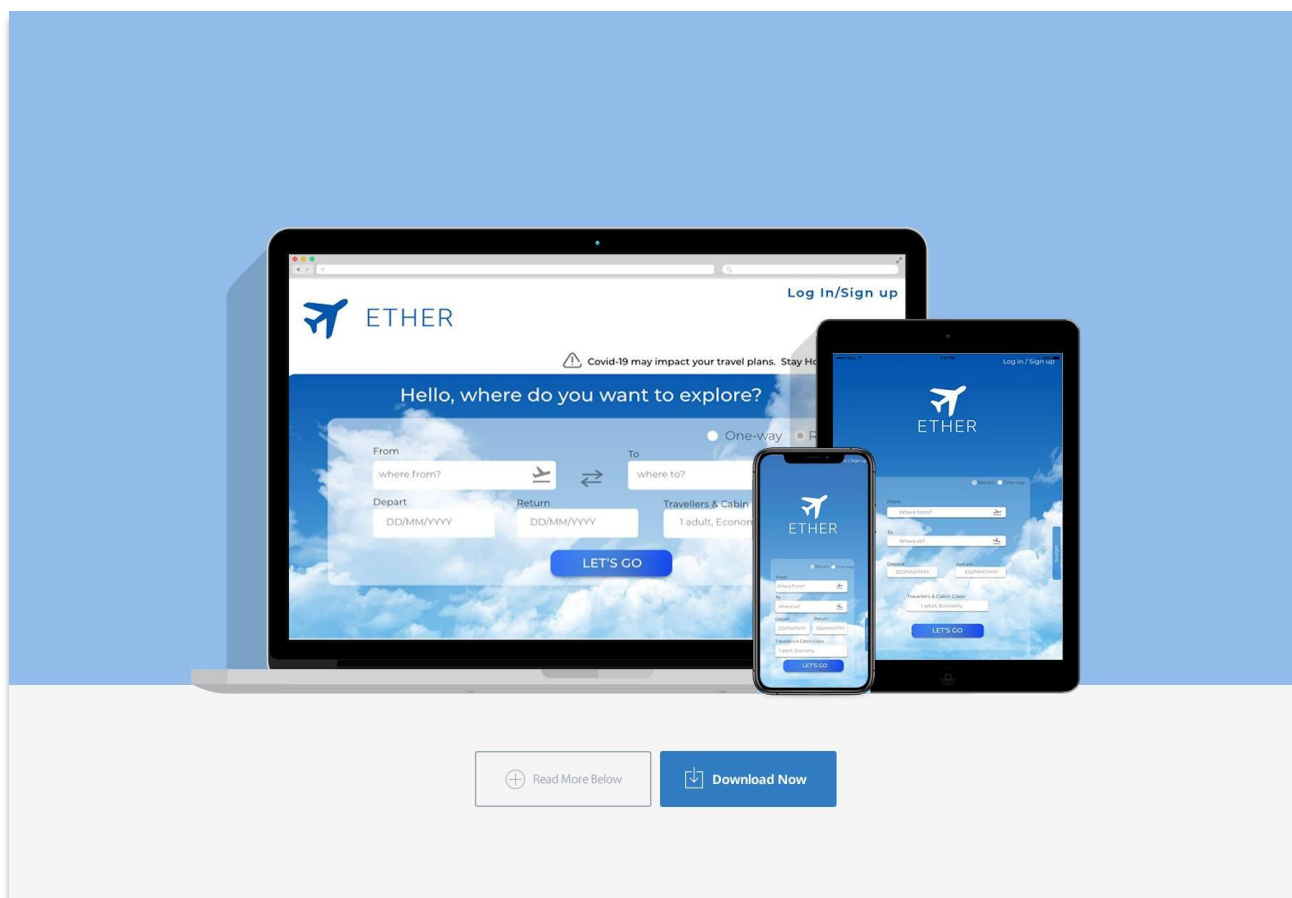
Σχήμα 1. Μοντέλο Σεναρίων Χρήσης
Σχήμα 2. Ο χρήστης αναζητά πτήσεις
Σχήμα 3. Ο χρήστης ταξινομεί πτήσεις
Σχήμα 4. Ο χρήστης εφαρμόζει φίλτρα
Σχήμα 5. Ο χρήστης επιλέγει μία πτήση
Σχήμα 6. Ο χρήστης βλέπει τις μεταβολές των τιμών
Σχήμα 7. Ο χρήστης καταχωρεί μία αξιολόγηση
Σχήμα 8. Ο χρήστης βλέπει τις αξιολογήσεις των παρόχων
Σχήμα 9. Ο χρήστης αναζητεί ταξίδια με βάση ένα budget
Σχήμα 10. Ο χρήστης συνδέεται με την αεροπορική εταιρεία ή ταξιδιωτικά γραφεία
Σχήμα 11. Ο εγγεγραμμένος χρήστης βλέπει προτάσεις για ταξίδια
Σχήμα 12. Παράθυρο Αναζήτησης
Σχήμα 13. Παράθυρο Αναζήτησης εγγεγραμμένου χρήστη
Σχήμα 14. Παράθυρο Αναζήτησης με βάση budget
Σχήμα 15. Παράθυρο Αποτελεσμάτων
Σχήμα 16-17. Παράθυρο επιλογής φίλτρων
Σχήμα 18. Παράθυρο επιλεγμένης πτήσης
Σχήμα 19. Παράθυρο εμφάνισης της αξιολόγησης του παρόχου
Σχήματα 20-22. Παράθυρα αξιολόγησης παρόχων
Σχήματα 23-25. Παράθυρο παρακολούθησης μεταβολής τιμών
Σχήμα 26. Παράθυρο Σφάλματος(περιέχει διαφορετικό μήνυμα σε κάθε περίπτωση)
Σχήμα 27. Παράθυρο φόρτωσης σελίδας
Σχήμα 28. Παράθυρο αναζήτησης πτήσης για web browser
Σχήμα 29. SearchFlight Package
Σχήμα 30. Διάγραμμα Κλάσεων SearchFlight Package
Σχήμα 31. ResultHandle Package
Σχήμα 32. Διάγραμμα Κλάσεων ResultHandle Package
Σχήμα 33. SupplierHandle Package
Σχήμα 34. Διάγραμμα Κλάσεων SupplierHandle Package
Σχήμα 35. ProposeTravel Package
Σχήμα 36. Διάγραμμα Κλάσεων ProposeTravel Package
Σχήμα 37. Διάγραμμα Κλάσεων PriceChange Package
Σχήμα 38. Διάγραμμα Κλάσεων PriceChange Package



ΕΙΣΑΓΩΓΗ

Τα ταξίδια, είτε επαγγελματικά είτε αναψυχής, ως μία πράξη απόδρασης από την καθημερινότητα, αποτελούν πλέον μία από τις ουσιαστικές πτυχές της ζωής των ανθρώπων. Σκέψη μας πίσω από το ETHER ήταν να συγκεντρώσουμε όλες τις εναλλακτικές πτήσεων σε ένα σημείο, ώστε να βοηθάμε τον κάθε ταξιδιώτη να βρει τις καλύτερες επιλογές σε πτήσεις

Ο ταξιδιώτης διαλέγει πού θέλει να πάει, πότε θέλει να πραγματοποιήσει το ταξίδι του και για να βρει με ένα απλό κλικ τις καλύτερες ευκαιρίες πτήσεων. Επιπλέον, γίνονται προτάσεις με βάση το ταξιδιωτικό γούστο, ενώ παράλληλα βοηθάμε τον ταξιδιώτη να οργανώσει το ταξίδι με βάση το δικό του budget. Τα έξυπνα φίλτρα αναζήτησης, όπως ο αριθμός των στάσεων και η ώρα αναχώρησης, είναι εκεί για να τον βοηθήσουν στην επιλογή του. Και όταν πια ξέρει που πηγαίνει και πότε φεύγει μένει μόνο η σύνδεση με τον πάροχο που επέλεξε ώστε να μπορέσει να αγοράσει το εισιτήριο του.





1. ΑΠΑΙΤΗΣΕΙΣ ΣΥΣΤΗΜΑΤΟΣ

1.1 ΛΕΙΤΟΥΡΓΙΚΕΣ ΑΠΑΙΤΗΣΕΙΣ

<ΛΑ-1>

Ο χρήστης πρέπει να μπορεί να αναζητήσει πτήσεις.

Περιγραφή: Ο χρήστης πρέπει να μπορεί να επιλέξει ανάμεσα σε μετάβαση απλή ή με επιστροφή. Έπειτα πρέπει να μπορεί εισάγοντας το αεροδρόμιο/πόλη/χώρα εκκίνησης και του προορισμού αντίστοιχα, ορίζοντας τις ημερομηνίες αναχώρησης, επιστροφής, κατηγορία ναύλου και αριθμό ταξιδιωτών να αναζητεί την πτήση που επιθυμεί.

User Priority (5/5): Η αναζήτηση πτήσης είναι η βασικότερη λειτουργία για τον χρήστη, καθώς χωρίς αυτή δεν έχει νόημα ύπαρξης η πλατφόρμα.

Technical Priority (5/5): Η απαίτηση αυτή είναι πολύ υψηλής προτεραιότητας για το σύστημα.

<ΛΑ-2>

Ο χρήστης πρέπει να μπορεί να επιλέξει την ταξινόμηση των αποτελεσμάτων.

Περιγραφή: Ο χρήστης εφόσον έχει ολοκληρώσει την αναζήτησή του πρέπει να μπορεί να επιλέξει την μορφή ταξινόμησης των αποτελεσμάτων σε αύξουσα τιμή, ελάχιστη διάρκεια ή καλύτερη επιλογή, δηλαδή ο συνδυασμός των δύο.

User Priority (4/5): Η ταξινόμηση των αποτελεσμάτων είναι μια χρήσιμη λειτουργία για τον χρήστη, καθώς διευκολύνει την αναζήτηση του.

Technical Priority (3/5): Η απαίτηση αυτή είναι σχετικά μεσαίας προτεραιότητας για το σύστημα.

<ΛΑ-3>

Ο χρήστης πρέπει να μπορεί να επιλέξει φίλτρα για τα αποτελέσματα.

Περιγραφή: Ο χρήστης εφόσον έχει ολοκληρώσει την αναζήτηση πρέπει να μπορεί να περιορίσει τα αποτελέσματα επιλέγοντας φίλτρα όπως ώρες αναχώρησης, αεροπορική εταιρεία, διάρκεια ταξιδιού κλπ.

User Priority (4/5): Το φιλτράρισμα των αποτελεσμάτων είναι μια χρήσιμη λειτουργία για τον χρήστη, καθώς τον διευκολύνει στην επιλογή του.

Technical Priority (3/5): Η απαίτηση αυτή είναι σχετικά μεσαίας προτεραιότητας για το σύστημα.



<ΛΑ-4>

Ο χρήστης πρέπει να μπορεί να επιλέξει μια πτήση.

Περιγραφή: Ο χρήστης εφόσον έχει ολοκληρώσει την αναζήτηση πρέπει να μπορεί να επιλέξει την πτήση που ανταποκρίνεται καλύτερα στις ανάγκες του για το ταξίδι που αναζητά.

User Priority (5/5): Η επιλογή της πτήσης είναι απαραίτητη προϋπόθεση για την επίτευξη της λειτουργίας του συστήματος.

Technical Priority (5/5): Η απαίτηση αυτή είναι υψηλής προτεραιότητας, καθώς αποτελεί βασικό στόχο του συστήματος.

<ΛΑ-5>

Ο χρήστης πρέπει να μπορεί να συνδεθεί με τον πάροχο των εισιτηρίων που επέλεξε.

Περιγραφή: Όταν ο χρήστης επιλέγει μια πτήση, από κάποιον πάροχο, το σύστημα πρέπει να μπορεί να τον συνδέει με το site του παρόχου.

User Priority (5/5) : Η σύνδεση του χρήστη με τον πάροχο είναι απαραίτητη προϋπόθεση για την ολοκλήρωση της αγοράς εισιτηρίου.

Technical Priority (5/5) : Η απαίτηση αυτή είναι ιδιαίτερα σημαντική για το σύστημα.

<ΛΑ-6>

Ο εγγεγραμμένος χρήστης θα έπρεπε να μπορεί να αξιολογήσει τον πάροχο των εισιτηρίων.

Περιγραφή : Ο χρήστης, εφόσον είναι εγγεγραμμένος, θα έπρεπε να μπορεί να αξιολογεί τον πάροχο σε διάφορους τομείς των υπηρεσιών του, όπως Ακρίβεια Τιμής, Εξυπηρέτηση Πελατών κλπ.

User Priority (3/5) : Η δυνατότητα αξιολόγησης του χρήστη είναι μια επιθυμητή λειτουργία αλλά όχι και αναγκαία.

Technical Priority (3/5) : Η απαίτηση αυτή είναι μεσαίας κατηγορίας για το σύστημα.

<ΛΑ-7>

Ο χρήστης πρέπει να μπορεί να δει την αξιολόγηση των παρόχων.

Περιγραφή : Ο χρήστης πρέπει να μπορεί να βλέπει την αξιολόγηση των παρόχων των άλλων χρηστών, όταν επιλέξει ένα αποτέλεσμα της αναζήτησης του.

User Priority (3/5) : Η εμφάνιση της αξιολόγησης των παρόχων είναι μια επιθυμητή λειτουργία για τον χρήστη, καθώς γίνεται πιο εύκολη η σύγκριση των παρόχων.

Technical Priority (3/5) : Η απαίτηση αυτή είναι ιδιαίτερα σημαντική για το σύστημα, καθώς δίνει μια καλύτερη εικόνα των παρόχων στον χρήστη.



<ΛΑ-8>

Ο χρήστης θα έπρεπε να μπορεί να αναζητήσει ταξίδια αναλόγως το budget.

Περιγραφή : Όταν ο χρήστης δεν μπορεί να αποφασίσει προορισμό, το σύστημα πρέπει να μπορεί να του δείξει προτάσεις για πτήσεις μέσα σε ένα συγκεκριμένο προϋπολογισμό χρημάτων.

User Priority (3/5): Η πρόταση πτήσεων δεν είναι τόσο σημαντική για τον χρήστη.

Technical Priority (2/5) : Η απαίτηση αυτή δεν είναι ιδιαίτερα σημαντική για το σύστημα.

<ΛΑ-9>

Ο εγγεγραμμένος χρήστης θα έπρεπε να μπορεί να παρακολουθεί τη μεταβολή των τιμών.

Περιγραφή : Ο χρήστης, εφόσον είναι εγγεγραμμένος, θα έπρεπε να μπορεί να παρακολουθεί τη μεταβολή των τιμών (είτε αύξηση είτε μείωση) των εισιτηρίων.

User Priority (2/5) : Η δυνατότητα να παρακολουθεί ο χρήστης τη μεταβολή των τιμών είναι μια επιθυμητή λειτουργία, αλλά όχι και αναγκαία.

Technical Priority (3/5) : Η απαίτηση αυτή είναι σχετικά μεσαίας προτεραιότητας για το σύστημα, καθώς είναι υπεύθυνο για την άρτια παρακολούθηση μεταβολών των τιμών.

<ΛΑ-10>

Ο εγγεγραμμένος χρήστης πρέπει να μπορεί να βλέπει ταξίδια σύμφωνα με τις προτιμήσεις του.

Περιγραφή: Το σύστημα αποθηκεύει τους προορισμούς του εγγεγραμμένου χρήστη και του προτείνει νέα ταξίδια σύμφωνα με ιστορικό των ταξιδιών του και τις συχνές αναζητήσεις του.

User Priority(3/5): Η απαίτηση είναι μεσαίας κατηγορίας για τον χρήστη επειδή τον διευκολύνει στην αναζήτηση ταξιδιών.

Technical Priority (4/5): Η απαίτηση αυτή είναι ιδιαίτερα σημαντική για το σύστημα, καθώς είναι υπεύθυνο για τη διατήρηση του ιστορικού ταξιδιών.



1.2 ΧΡΗΣΤΕΣ ΚΑΙ ΕΞΩΤΕΡΙΚΑ ΣΥΣΤΗΜΑΤΑ

Στην παρούσα πλατφόρμα δεν είναι απαραίτητη η ύπαρξη λογαριασμού για την είσοδο, δίνοντας ωστόσο τη δυνατότητα σε αυτούς που επιθυμούν να εγγραφούν, ώστε να έχουν επιπλέον δυνατότητες στο σύστημα. Οι χρήστες λοιπόν χωρίζονται στους α) Απλούς Χρήστες και β) Εγγεγραμμένους Χρήστες.

1.2.1 Κατηγορίες χρηστών

Απλός χρήστης: Στην κατηγορία αυτή ανήκουν οι χρήστες οι οποίοι μπορούν να περιηγηθούν στο σύστημα περιορίζοντας τους στις βασικές λειτουργίες του. Συγκεκριμένα η αναζήτηση πτήσης και ταξινόμηση των αποτελεσμάτων αυτής της αναζήτησης καθώς και η επιλογή παρόχου.

Εγγεγραμμένος χρήστης: Στην κατηγορία αυτή ανήκουν οι χρήστες οι οποίοι κληρονομούν όλες τις παροχές του απλού χρήστη με κάποιες επιπρόσθετες. Τέτοιες είναι η αξιολόγηση των παρόχων, η πρόταση πτήσεων/ταξιδιών με βάση τις προτιμήσεις τους ή ενημέρωση της μεταβολής των τιμών.

1.2.2 Εξωτερικά συστήματα

Βάση δεδομένων: Είναι ένα εξωτερικό σύστημα απαραίτητο για τη διατήρηση όλων των στοιχείων των εγγεγραμμένων χρηστών. Αυτή η βάση δεδομένων είναι άμεσα συνδεδεμένη με το σύστημα καθώς ενημερώνεται για κάθε εγγραφή νέου χρήστη και διατηρώντας τις προτιμήσεις και το ιστορικό των χρηστών δημιουργεί εξατομικευμένες προτάσεις για την καλύτερη εξυπηρέτησή τους. Ακόμα, αποτελεί μία βάση δεδομένων για την διατήρηση όλων των στοιχείων των πτήσεων, των αεροπορικών εταιρειών και των ταξιδιωτικών παρόχων. Τέλος, η βάση δεδομένων του εξωτερικού συστήματος καθίσταται δυναμική, καθώς μπορούν να υπάρξουν προσθήκες και αφαιρέσεις παρόχων, αλλά και ενημέρωση των τιμών για τυχόν μεταβολή από τους παρόχους.

Αεροπορικές εταιρείες / ταξιδιωτικοί πάροχοι: Για την επίτευξη της βασικότερης λειτουργίας του συστήματος, δηλαδή τη σύνδεση χρήστη-παρόχου για την ολοκλήρωση της αγοράς εισιτηρίων, απαιτείται η συνεχής επικοινωνία του κεντρικού συστήματος με τους εξωτερικούς παρόχους.



1.3 ΣΗΜΑΝΤΙΚΕΣ ΜΗ ΛΕΙΤΟΥΡΓΙΚΕΣ ΑΠΑΙΤΗΣΕΙΣ

1.1 Απαιτήσεις επίδοσης

<ΜΛΑ-1>

Το σύστημα πρέπει να μπορεί να υποστηρίξει 20 εκατομμύρια χρήστες, χωρίς να επηρεάζεται η απόδοσή του.

Περιγραφή: Το σύστημα πρέπει να υποστηρίζει ταυτόχρονα 20 εκατομμύρια επισκέπτες χωρίς να επηρεάζεται το conventional rate της εφαρμογής, δηλαδή ταχύτητα απόκρισης 1-3 δευτερόλεπτα (ώστε ο σέρβερ να μπορεί να τους εξυπηρετήσει όλους με τη μέγιστη ταχύτητα φόρτωσης), εμφάνισης αποτελεσμάτων αναζήτησης, φόρτωσης audiovisual υλικού κλπ.

User Priority (5/5): Η γρήγορη και αποτελεσματική απόκριση του συστήματος είναι πολύ σημαντική για το σύνολο των χρηστών, καθώς κάθε χρήστης επιθυμεί μία εύχρηστη και άμεσα αποκρίσιμη αναζήτηση.

Technical Priority (5/5): Η απαίτηση αυτή είναι πολύ σημαντική για το σύστημα το οποίο πρέπει να είναι εκσυγχρονισμένο και κατάλληλα διαμορφωμένο ώστε να μπορεί να ανταπεξέλθει στις απαιτήσεις και στις ανάγκες μεγάλου ταυτόχρονου αριθμού χρηστών.

1.2 Απαιτήσεις Ασφάλειας (Security)

<ΜΛΑ-2>

Το σύστημα πρέπει να μην επιτρέπει οποιονδήποτε χρήστη να επεμβαίνει στις πληροφορίες, εκτός από τους διαχειριστές ασφάλειας (security administrator).

Περιγραφή: Το σύστημα πρέπει να μη επιτρέπει οποιονδήποτε να επεμβαίνει στις πληροφορίες του συστήματος για λόγους ασφάλειας. Πρόσβαση στις πληροφορίες του συστήματος πρέπει να έχουν μόνο οι διαχειριστές.

User Priority (1/5): Η ασφάλεια του συστήματος δεν είναι κάτι που αφορά τον χρήστη

Technical Priority (5/5): Είναι σημαντικό για το σύστημα να προστατεύει τα δεδομένα του και τις πληροφορίες του.

1.3 Απαιτήσεις χρηστικότητας (Usability)

<ΜΛΑ-3>

Το σύστημα πρέπει να είναι εύχρηστο σε χρήστες χωρίς μεγάλη εξοικείωση με άλλες μηχανές αναζήτησης.

Περιγραφή: Το σύστημα πρέπει να μπορεί να εξυπηρετεί άτομα, τα οποία δεν είναι εξοικειωμένα με άλλες μηχανές αναζήτησης, ή ακόμα και άτομα που δυσκολεύονται στη χρήση του διαδικτύου γενικότερα. Τέτοια άτομα είναι συνήθως μεγαλύτερης ηλικίας, τα οποία είναι εξίσου σημαντικά στην παγκόσμια αγορά.



User Priority (5/5): Ο χρήστης για να συνεχίσει να χρησιμοποιεί ένα σύστημα είναι απαραίτητο να το θεωρεί εύκολο στη χρήση. Συνεπώς αυτή η απαίτηση καθίσταται σημαντική για τον ίδιο.

Technical Priority (2/5): Η απαίτηση αυτή αν και χρήσιμη για το μέσο χρήστη δεν είναι απαραίτητη προϋπόθεση για τη λειτουργία του συστήματος.

1.4 Απαιτήσεις φορητότητας (portability)

<ΜΛΑ-4>

Το σύστημα πρέπει να είναι φορητό (portable).

Περιγραφή: Το σύστημα πρέπει να μένει αναλλοίωτο, δηλαδή μεταφερόμενο από ένα λειτουργικό σύστημα σε ένα άλλο να μην δημιουργεί προβλήματα στον χρήστη.

User Priority (3/5): Ο χρήστης αλλάζοντας λειτουργικό σύστημα θα πρέπει να είναι ικανός να χρησιμοποιεί το σύστημα χωρίς δυσκολία. Συνεπώς είναι μεσαίας σημασίας για τον χρήστη.

Technical Priority (5/5): Η απαίτηση αυτή είναι πολύ σημαντική για το σύστημα γιατί αφορά την αξιοπιστία του και την ικανότητα του να προσφέρει σε όλους του χρήστες, ανεξαρτήτως του λειτουργικού συστήματος, τις ίδιες παροχές.

1.5 Απαιτήσεις Αξιοπιστίας (Reliability)

<ΜΛΑ-5>

Το σύστημα πρέπει να ελέγχει την αξιοπιστία των παρόχων.

Περιγραφή: Το σύστημα πρέπει να ελέγχει την αξιοπιστία των παρόχων του, ώστε να γίνεται σωστά η εξυπηρέτηση και η ολοκλήρωση της αγοράς του χρήστη. Η εξαπάτηση ή κακή εξυπηρέτηση του χρήστη από τον πάροχο, θέτει ζήτημα αξιοπιστίας του συστήματος.

User Priority (5/5): Η αξιοπιστία είναι από τους σημαντικότερους παράγοντες για τον χρήστη.

Technical Priority (5/5): Είναι σημαντικό για το σύστημα να μπορεί να ελέγχει τους συνεργάτες του και τις υπηρεσίες που προσφέρουν.

1.6 Τεχνικές Απαιτήσεις Περιβάλλοντος

<ΜΛΑ-6>

Το σύστημα πρέπει να επιστρέφει στην αρχική σελίδα μετά από 20 λεπτά παραμονή σε μια αναζήτηση.

Περιγραφή: Το σύστημα δίνει την δυνατότητα στον χρήστη να παραμείνει σε μια αναζήτηση για χρονικό διάστημα 20 λεπτών. Στην συνέχεια τον επαναφέρει την αρχική σελίδα ώστε να ενημερωθεί το σύστημα.

User Priority (1/5): Η απαίτηση δεν είναι σημαντική για τον χρήστη.



Technical Priority (4/5): Η απαίτηση είναι σημαντική για το σύστημα, καθώς πρέπει να ενημερώνει τα δεδομένα που παρέχει στους χρήστες σύμφωνα με τις μεταβολές των παρόχων και να αποφεύγεται η υπερφόρτωση της συγκεκριμένης αναζήτησης.

1.7 Απαιτήσεις πολιτικής (Policy)

<ΜΛΑ-6>

Το σύστημα πρέπει να ρωτάει τον χρήστη εάν θέλει να συμμορφωθεί με την πολιτική του συστήματος.

Περιγραφή: Το σύστημα πρέπει να μπορεί να ρωτάει τον χρήστη, προτού κάνει οποιαδήποτε ενέργεια στο σύστημα, μέσω ενός αναδυόμενου παραθύρου εάν συμφωνεί με τους όρους χρήσης, προϋποθέσεις και πολιτικής του συστήματος. Με αυτόν τον τρόπο ο χρήστης γνωρίζει τα δικαιώματά του.

User Priority (4/5): Ο χρήστης είναι αναγκαίο να δηλώνει εάν συμφωνεί ή όχι με την πολιτική του συστήματος, συνεπώς είναι μια επιθυμητή απαίτηση για το χρήστη.

Technical Priority (5/5): Η απαίτηση αυτή είναι σημαντική για το σύστημα, καθώς πρέπει να γνωρίζει αν ο χρήστης συμμορφώνεται με τους κανόνες του συστήματος, για να συνεχιστεί η οποιαδήποτε ενέργεια.



1.4 ΛΕΞΙΚΟΓΡΑΦΙΚΟΙ ΠΡΟΣΔΙΟΡΙΣΜΟΙ

Πτήσεις: Οι αεροπορικές συνδέσεις μεταξύ των σημείων αναχώρησης και προορισμού.

Ταξινόμηση: Η διαδικασία επιλογής διάταξης των αποτελεσμάτων. Συνίσταται στην ταξινόμηση σε αύξουσα τιμή, ελάχιστη διάρκεια και καλύτερη επιλογή που αποτελεί συνδυασμό των δυο.

Αξιολόγηση: Περιλαμβάνει τις κριτικές των εγγεγραμμένων χρηστών για τους παρόχους των εισιτηρίων με βάση των χαρακτηριστικών price accuracy, customer service, refund policy, ease of booking και other.

Φίλτρα: Η διαδικασία επιλογής από τον χρήστη συγκεκριμένων χαρακτηριστικών για το ταξίδι του, όπως ώρα αναχώρησης, αεροπορική εταιρεία και διάρκεια ταξιδιού.

Αναζήτηση: Πράξη κατά την οποία ο χρήστης επιλέγει απλή μετάβαση ή με επιστροφή, χώρα εκκίνησης και προορισμού, ημερομηνίες αναχώρησης, επιστροφής, κατηγορία ναύλου και αριθμό ταξιδιωτών και βλέπει τα αποτελέσματα.

Χρήστης: Ο άνθρωπος που χρησιμοποιεί την πλατφόρμα.

Εγγεγραμμένος χρήστης: Ο χρήστης που εισέρχεται στην πλατφόρμα, ο οποίος επιτρέπει στο σύστημα να κρατάει κάποια δεδομένα του, όπως Όνομα ή ταξίδια τα οποία έχει πραγματοποιήσει στο παρελθόν. Για να συμβεί αυτό, πρέπει ο συγκεκριμένος χρήστης να έχει εγγραφεί στην πλατφόρμα.

Πάροχος: Είναι όλοι οι συνεργάτες του συστήματος όπως ταξιδιωτικοί πράκτορες και αεροπορικές εταιρείες που έχουν στην διάθεσή τους τα εισιτήρια και μπορούν να τα παρέχουν στους χρήστες του συστήματος.

Σύνδεση με πάροχο: Πράξη κατά την οποία ο χρήστης συνδέεται με τον πάροχο που έχει επιλέξει.

Budget: Ένα συγκεκριμένο όριο χρημάτων.

Ταξίδια: Περιλαμβάνει προτάσεις προορισμών για τους χρήστες

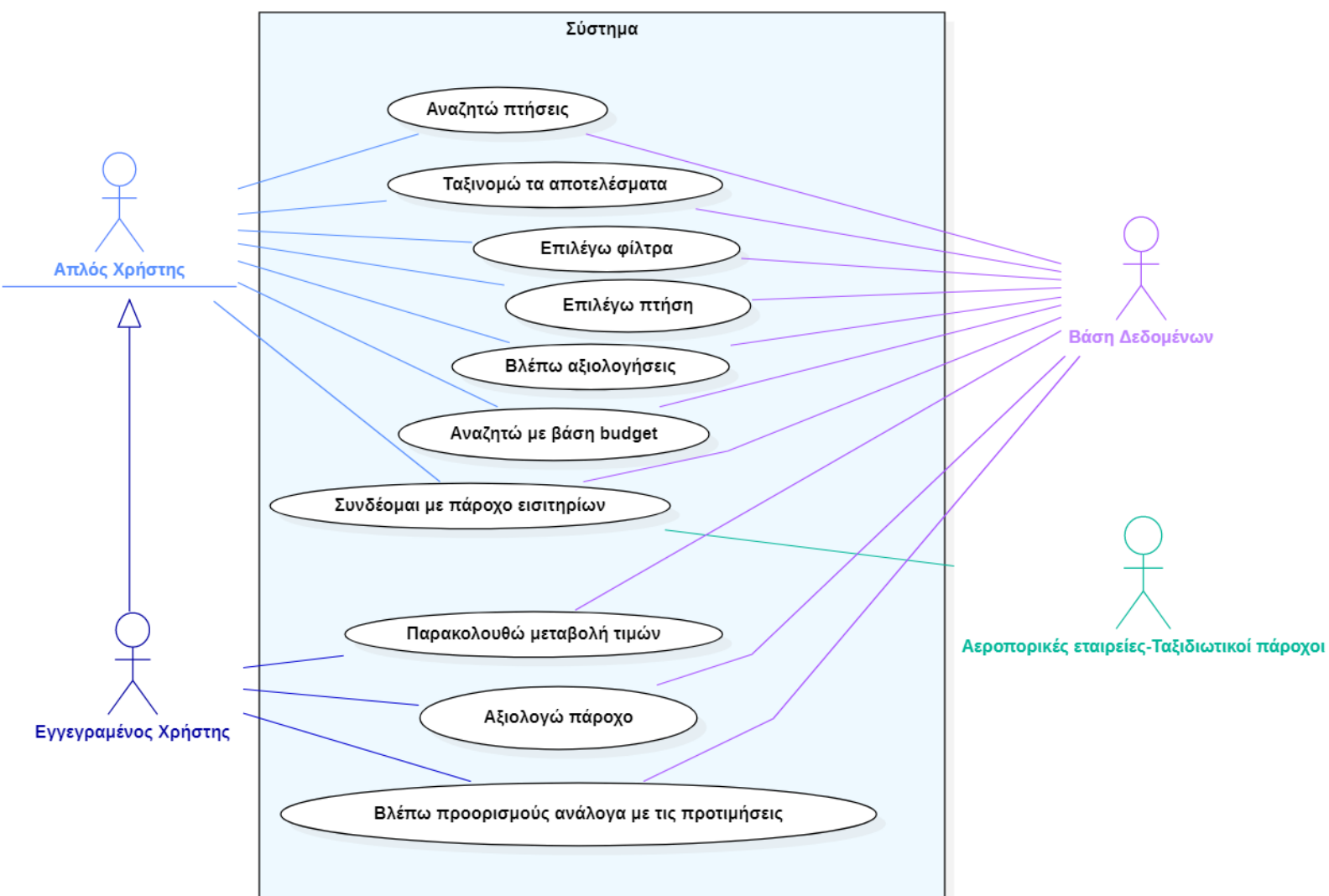
Μεταβολή των τιμών: Η οποιαδήποτε αλλαγή συμβαίνει στις τιμές των πτήσεων ή ταξιδιών.

Αποτελέσματα: Είναι όλες οι δυνατές πτήσεις που εμφανίζει το σύστημα και πληρούν τις απαιτήσεις του χρήστη σε μια αναζήτηση του.

Προτιμήσεις του χρήστη: Είναι η τάση που εμφανίζει ένας χρήστης του συστήματος να επιλέγει ταξίδια με παρόμοια χαρακτηριστικά. Το σύστημα κρατάει ιστορικό με την συγκατάθεση του χρήστη για να μπορεί να του προτείνει ταξίδια και να τον εξυπηρετεί με το καλύτερο δυνατό τρόπο.

2. ΣΕΝΑΡΙΑ ΧΡΗΣΗΣ

2.1 ΔΙΑΓΡΑΜΜΑ ΣΕΝΑΡΙΩΝ ΧΡΗΣΗΣ



Σχήμα 1. Μοντέλο Σεναρίων Χρήσης



2.2 ΣΕΝΑΡΙΑ ΧΡΗΣΗΣ GHERKIN

2.2.1 Feature: Search flights

Scenario: Available flights

When I choose return or one-way flight

When I choose country city or airport for departure/landing

When I choose date for departure/landing

When I choose cabin class and number of travelers

Then I click Let's go

And I am informed about the available flights

Scenario: Unavailable flights because of dates

When I choose return or one-way flight

When I choose country city or airport for departure/landing

When I choose date for departure/landing

When I choose cabin class and number of travelers

Then I click Let's go

Then I am informed that there are not available flights in these dates

And I return to home page(yes/no)

Then I edit dates

And I click Let's go

Scenario: Unavailable flights because of number of travelers

When I choose return or one-way flight

When I choose country city or airport for departure/landing

When I choose date for departure/landing

When I choose cabin class and number of travelers

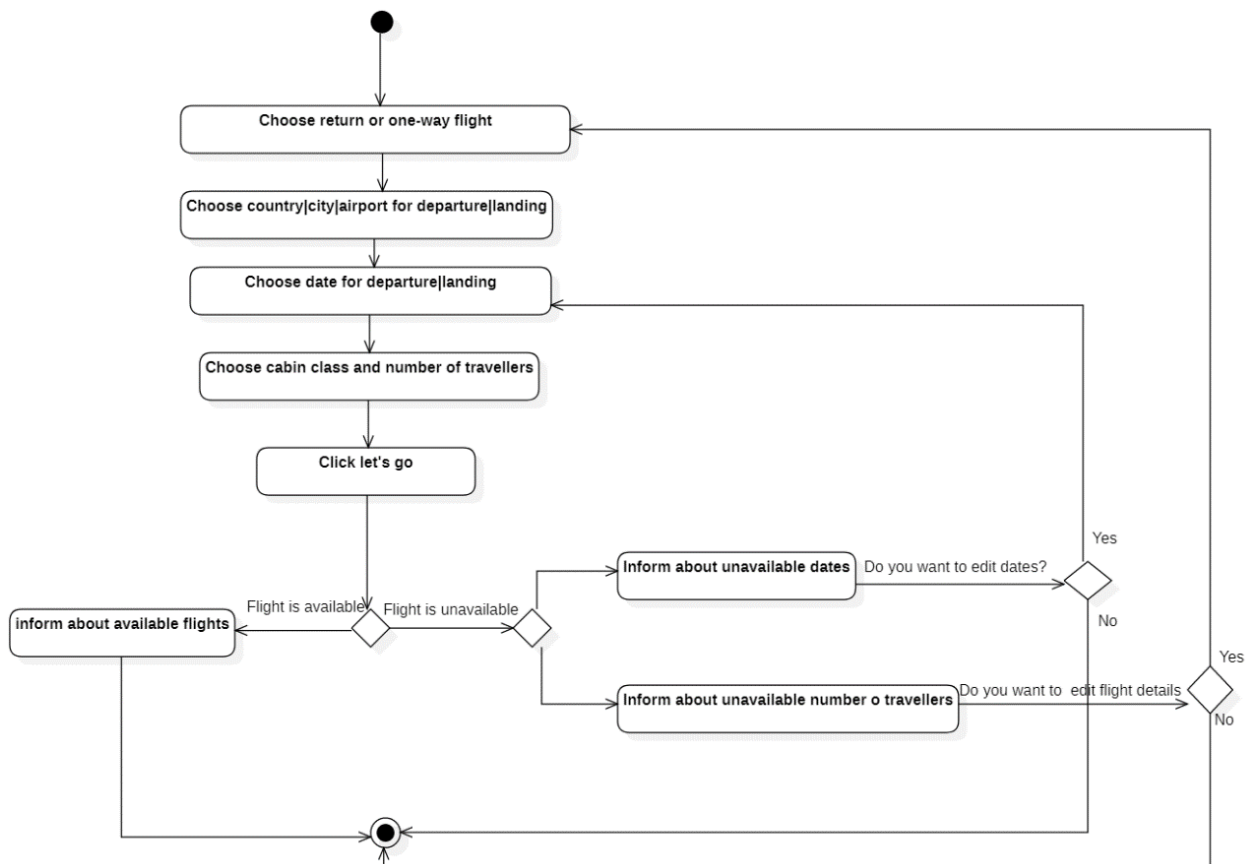
Then I click Let's go

Then I am informed that there are not available flights with that number of travelers

And I return to home page(yes/no)

Then I edit number of travelers

And I click Let's go



Σχήμα 2. Ο χρήστης αναζητά πτήσεις

2.2.2 Feature: Sort results

Scenario: Assortment based on cheapest flight first

Given that I have successfully searched for a flight

Then by default I see the results sorted by best flight first

Then I am given the possibility to sort the results based on cheapest price first

When I click on it

Then I have the assortment i chose

Scenario: Assortment based on fastest flight first

Given that I have successfully searched for a flight

Then by default I see the results sorted by best flight first

Then I am given the possibility to sort the results based on fastest flight first

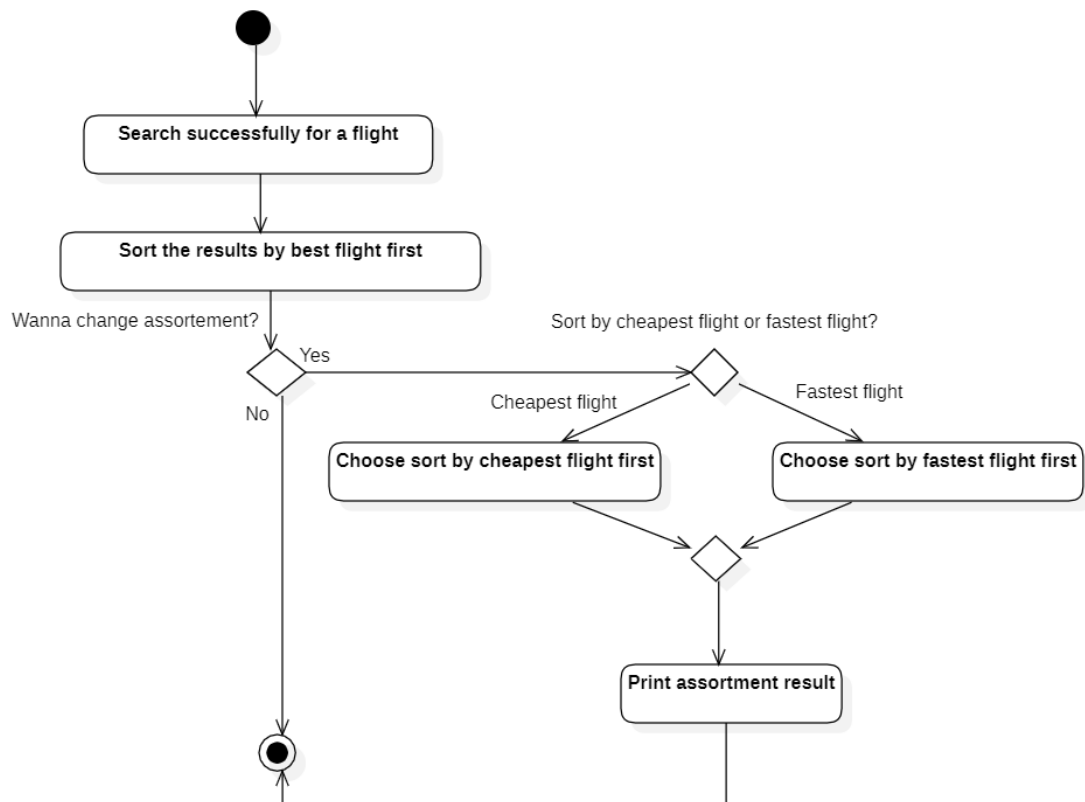
When I click on it

Then I have the assortment I chose

Scenario: Assortment based on best(cheap and fast) flight first

Given that I have successfully searched for a flight

Then by default I see the results sorted by best flight first



Σχήμα 3. Ο χρήστης ταξινομεί πτήσεις

2.2.3 Feature: Apply filters

Scenario: Choice of number of Stops

Given that I have successfully searched for a flight

When I choose how many stops my flight should have

Then I click on it

And I have the results I chose

Scenario: Choice of Departure time

Given that I have successfully searched for a flight

When I choose between what time range my flight will depart

| Outbound | [00:00 , 23:59] |

| Return | [00:00 , 23:59] |

Then I click on it

And I have the results I chose

Scenario: Choice of Journey duration

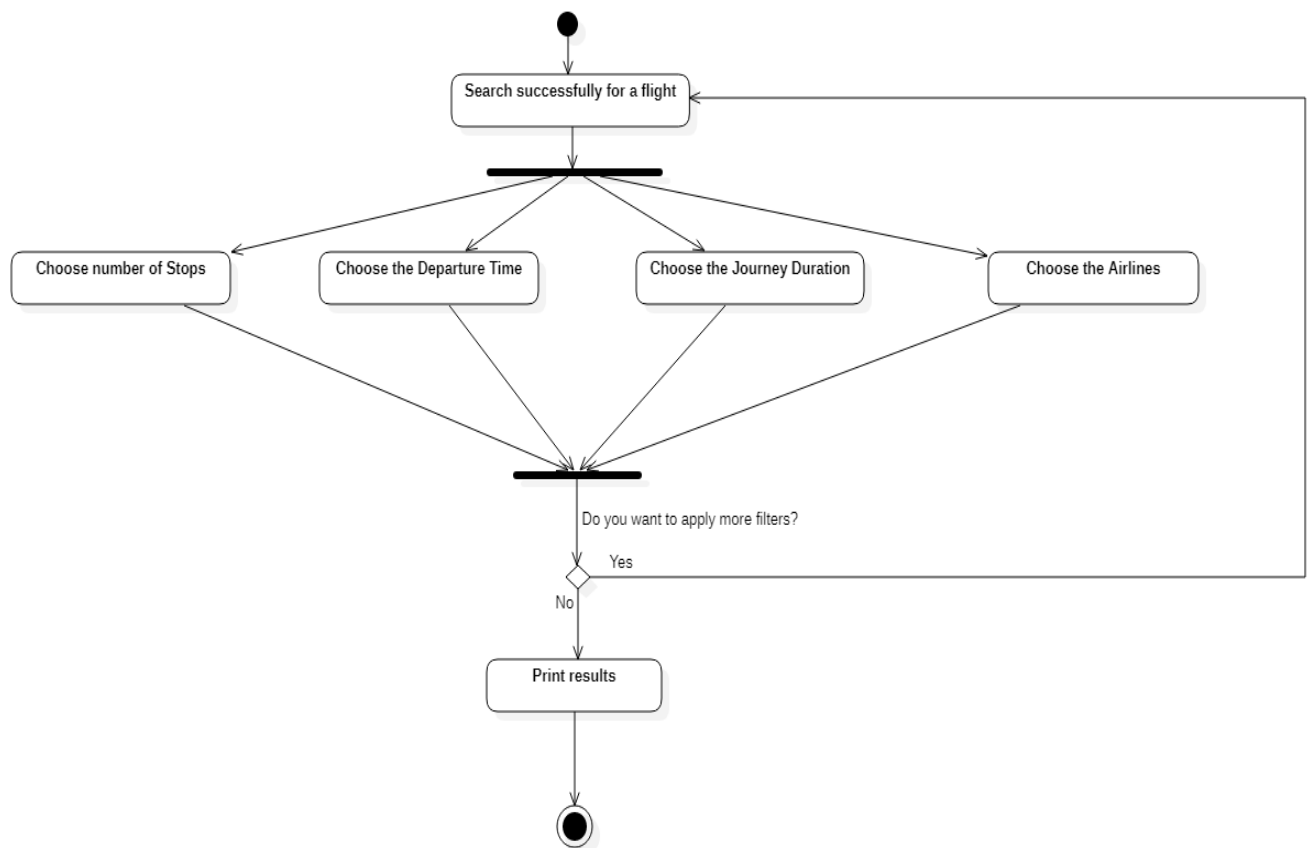
Given that I have successfully searched for a flight

When I choose how much duration my journey should have

| Duration | [0 , 35] |

Then I click on it
And I have the results I chose

Scenario: Choice of Airlines
Given that I have successfully searched for a flight
When I choose which airlines I prefer for my flight
Then I click on it
And I have the results I chose



Σχήμα 4. Ο χρήστης εφαρμόζει φίλτρα

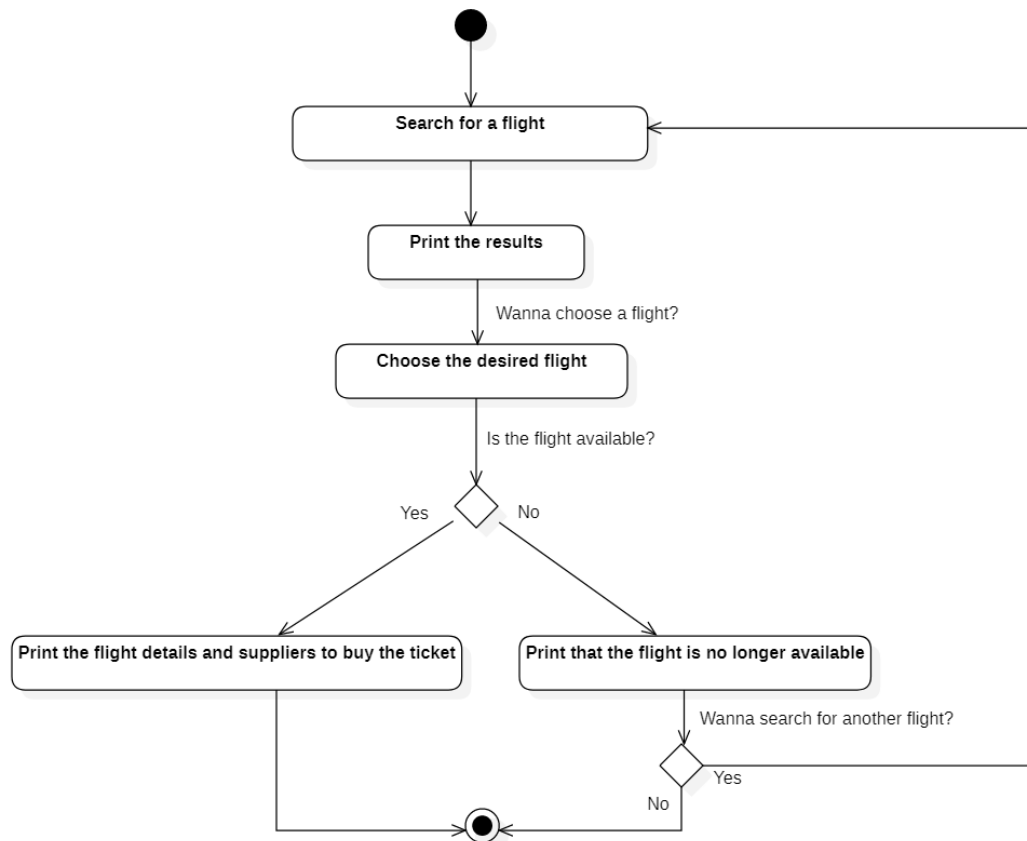
2.2.4 Feature: Choose flight

Scenario: Successfully pick a flight
Given that I have successfully searched for a flight
Then I see the results
Then I am given the possibility to choose the desired one
When I click on it
Then I see the flight details and suppliers to buy the ticket

Scenario: Unsuccessfully pick a flight due to sudden unavailability



Given that I have successfully searched for a flight
Then I see the results
Then I am given the possibility to choose the desired one
When I click on it
Then I see that the flight is no longer available
And I am asked if I want to choose another flight (yes/no)



Σχήμα 5. Ο χρήστης επιλέγει μία πτήση

2.2.5 Feature: Check price changes

Scenario: Login User search for price changes

Given that I am a login user and I have successfully searched for a flight
When I search for price changes
Then the system show me a graph with price changes

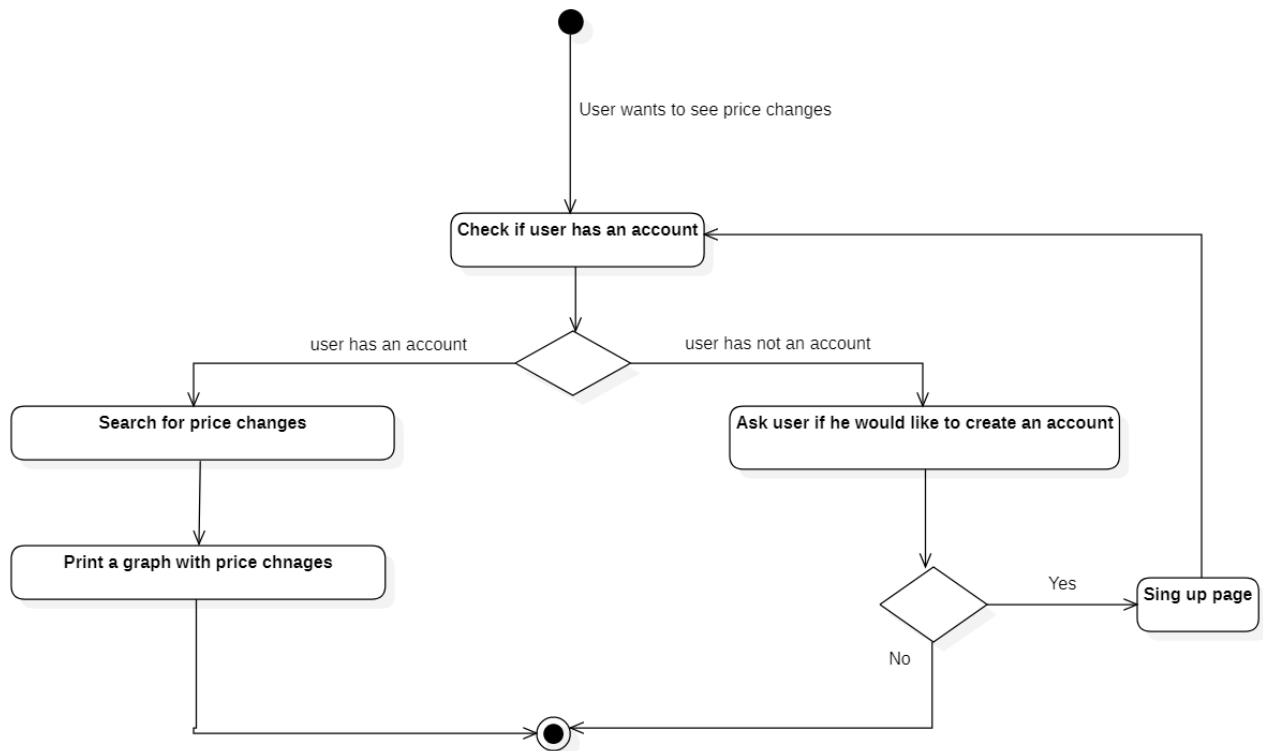
Scenario: User search for price changes

Given that I am a user and I have successfully searched for a flight
When search for price changes



Then the system show me a message that I can not check for price changes, because I am not a Login user

And the system ask me, if I would like to create an account or to continue



Σχήμα 6. Ο χρήστης βλέπει τις μεταβολές των τιμών

2.2.6 Feature: Evaluate airlines and travel agencies

Scenario: Login User evaluate airlines and travel agencies

Given that I am a login user

When I evaluate airlines and travel agencies

Then the system asks me to give an evaluation

| Evaluation | [1 , 5] |

When I choose a number

Then I click ok

And I return to Home

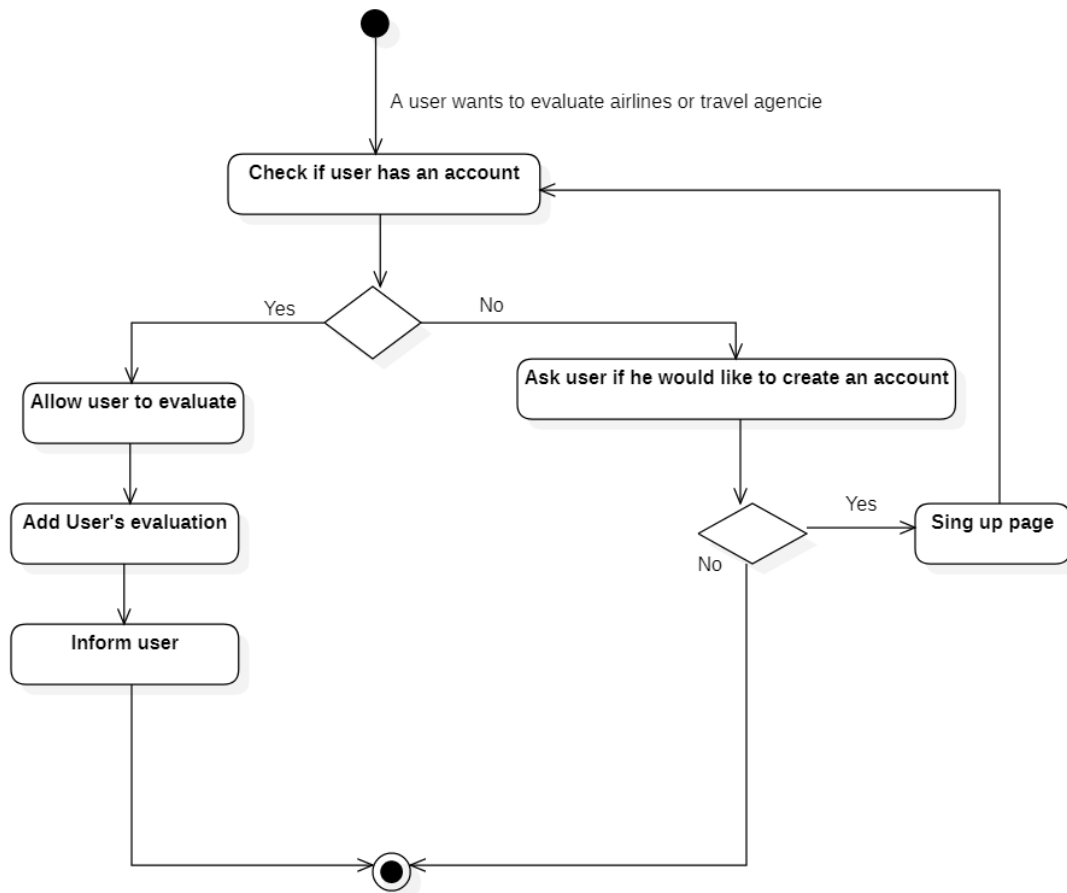
Scenario: User try to evaluate airlines and travel agencies

Given that I am a user

When try to evaluate airlines and travel agencies

Then the system show me a message that I can not evaluate airlines and travel agencies, because I am not a Login user

And the system ask me, if I would like to create an account or to continue



Σχήμα 7. Ο χρήστης καταχωρεί μία αξιολόγηση

2.2.7 Feature: Check evaluation of airlines and travel agencies

Scenario: Ratings exist

Given that I have successfully searched for a flight

When I choose a specific flight

Then I see a list of airlines and travel agencies

When I choose a specific travel agency/ airline

Then I see the list of ratings

Scenario: Ratings do not exist

Given that I have successfully searched for a flight

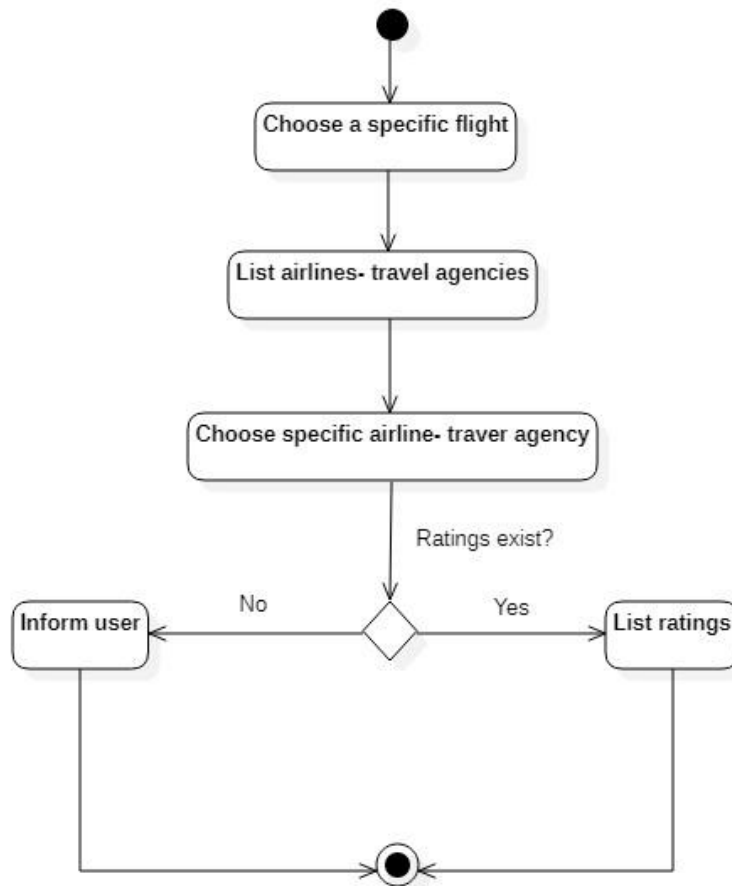
When I choose a specific flight

Then I see a list of airlines and travel agencies

When I choose a specific travel agency/ airline

But this specific travel agency/ airline does not have ratings

Then I am informed that ratings do not exist



Σχήμα 8. Ο χρήστης βλέπει τις αξιολογήσεις των παρόχων

2.2.8 Feature: Search trip within a budget

Scenario: Flights for the trip exist

Given that I can not decide a destination

When I click Budget button

Then I am asked to give my budget for the trip

When I give my budget

Then the system search for flights and destinations

And provide me the results

Scenario: Flights for the trip do not exist

Given that I can not decide a destination

When I click Budget button

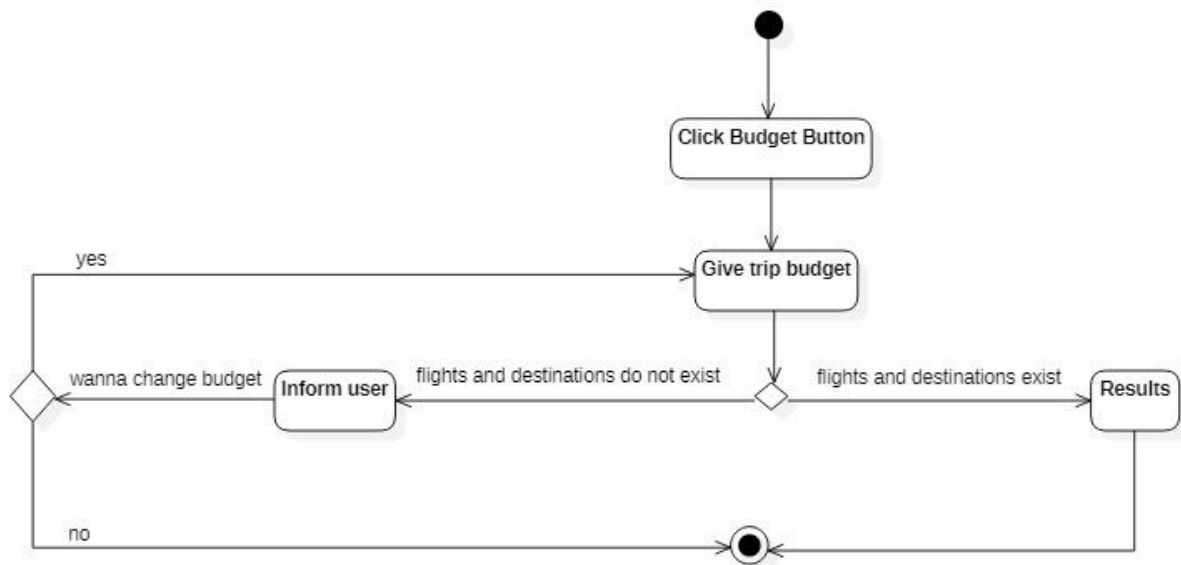
Then I am asked to give my budget for the trip

When I give my budget

Then the system search for flights and destinations

Then I am informed that flights for my budget do not exist

And I'm asked to choose a different budget



Σχήμα 9. Ο χρήστης αναζητεί ταξίδια με βάση ένα budget

2.2.9 Feature: Connect with airlines and travel agencies

Scenario: Successful connection

Given that I have picked the flight

When I see a list of airlines and travel agencies

Then I pick the airline or the travel agent

And I am transferred to their environment to proceed the purchase

Scenario: Unsuccessful connection

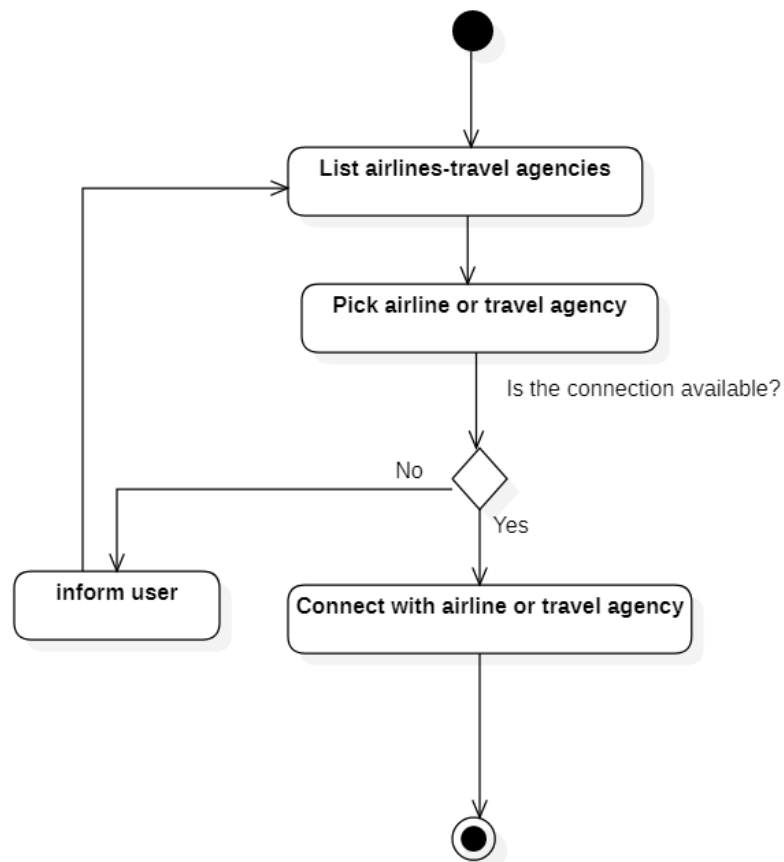
Given that I have picked the flight

When I see a list of airlines and travel agencies

Then I pick the airline or the travel agent

Then I am informed that the connection is unavailable

And I return to the previous page



Σχήμα 10. Ο χρήστης συνδέεται με την αεροπορική εταιρεία ή ταξιδιωτικά γραφεία

2.2.10 Feature: Propose destinations to logged-in user

Scenario: Existence of travel history in database

Given that I am logged in user

When I enter the Homepage

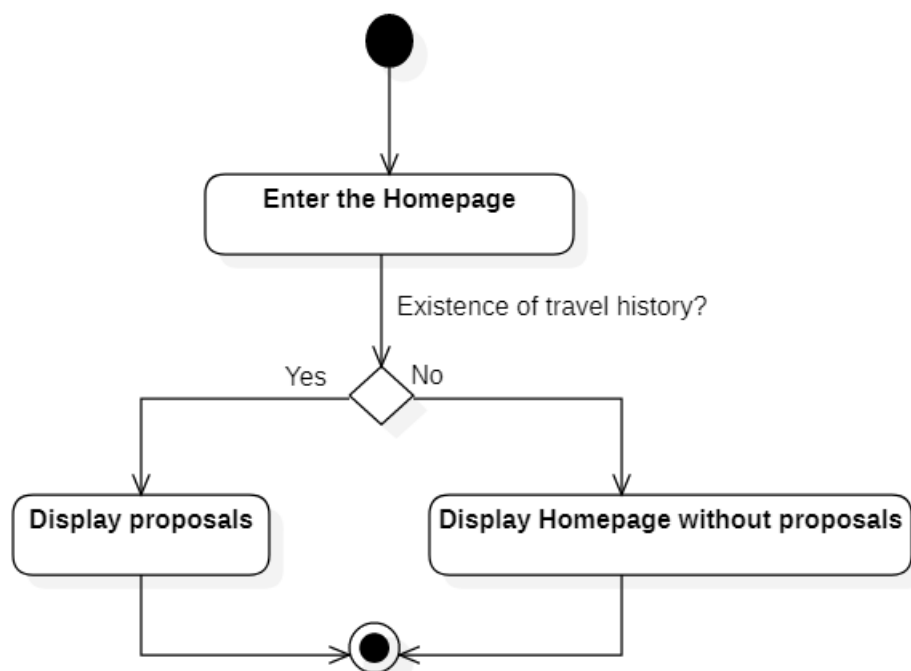
Then I see the proposals based on my preferences

Scenario: Non-existence of travel history in database

Given that I am logged in user

When I enter the Homepage

Then I see the Homepage without any proposal



Σχήμα 11. Ο εγγεγραμμένος χρήστης βλέπει προτάσεις για ταξίδια

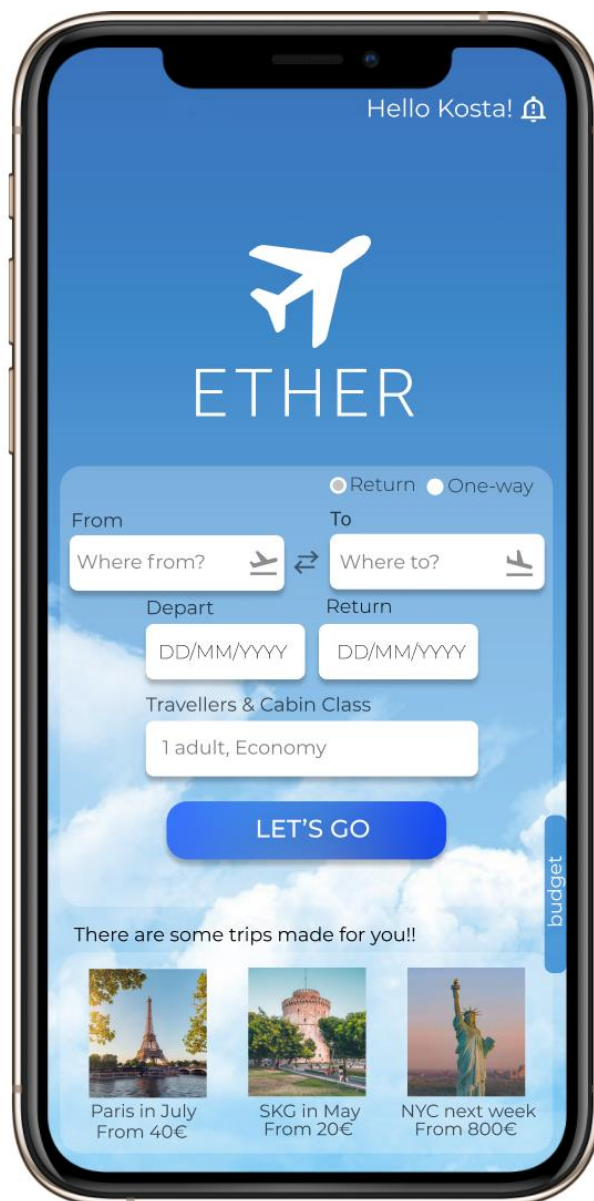


ΕΠΙΔΕΙΚΤΙΚΑ ΓΡΑΦΙΚΑ ΠΑΡΑΘΥΡΑ ΔΙΕΠΑΦΗΣ

3.1 ΠΑΡΑΘΥΡΑ ΑΝΑΖΗΤΗΣΗΣ

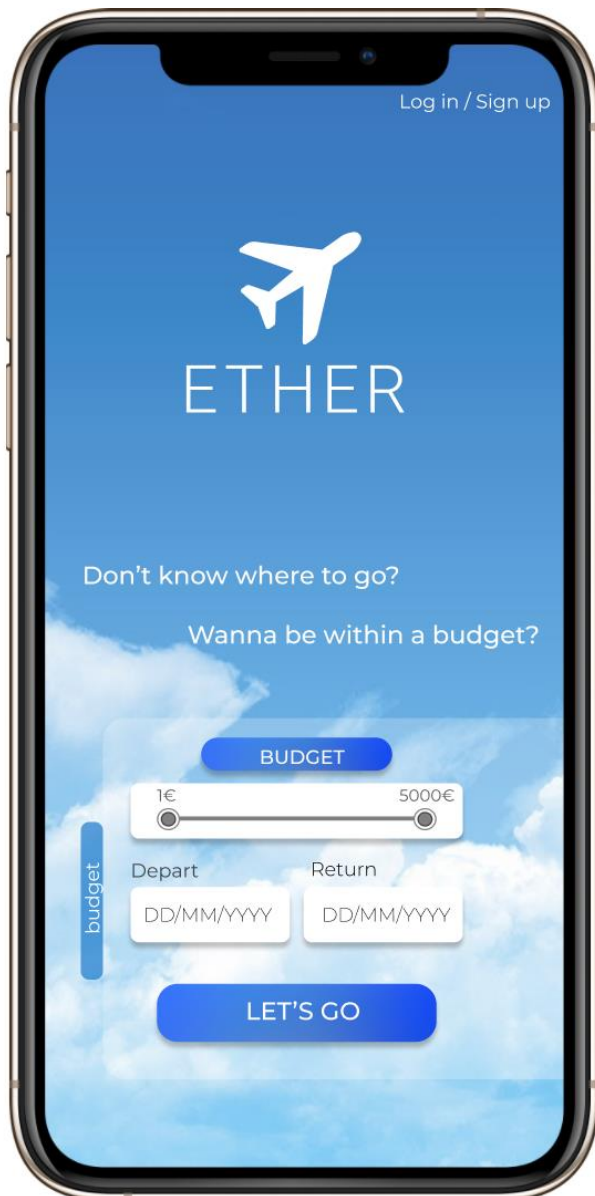


Σχήμα 12. Παράθυρο Αναζήτησης

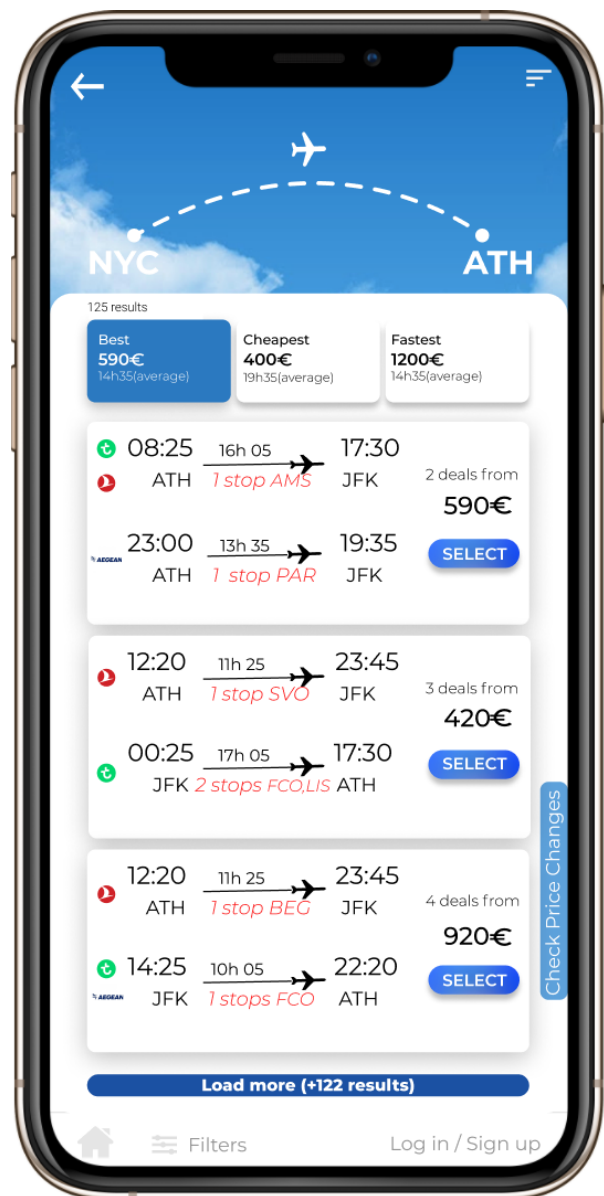


Σχήμα 13. Παράθυρο Αναζήτησης
εγγεγραμμένου χρήστη

3.2 ΠΑΡΑΘΥΡΟ ΑΝΑΖΗΤΗΣΗΣ ΜΕ ΒΑΣΗ BUDJET ΚΑΙ ΠΑΡΑΘΥΡΟ ΑΠΟΤΕΛΕΣΜΑΤΩΝ

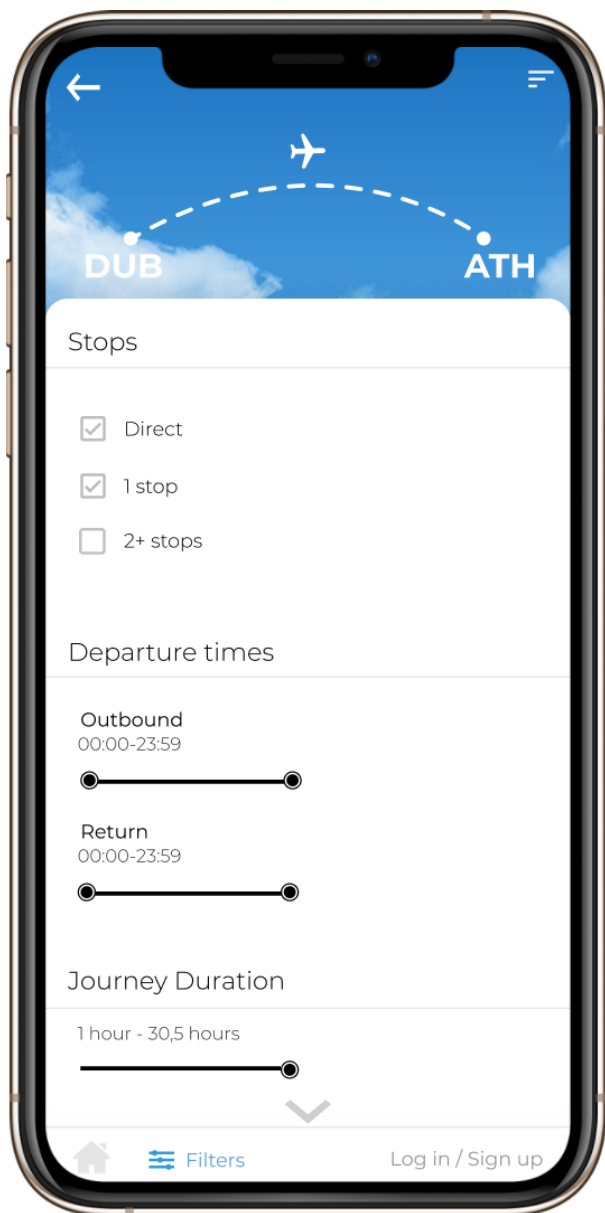


Σχήμα 14. Παράθυρο Αναζήτησης με βάση budget

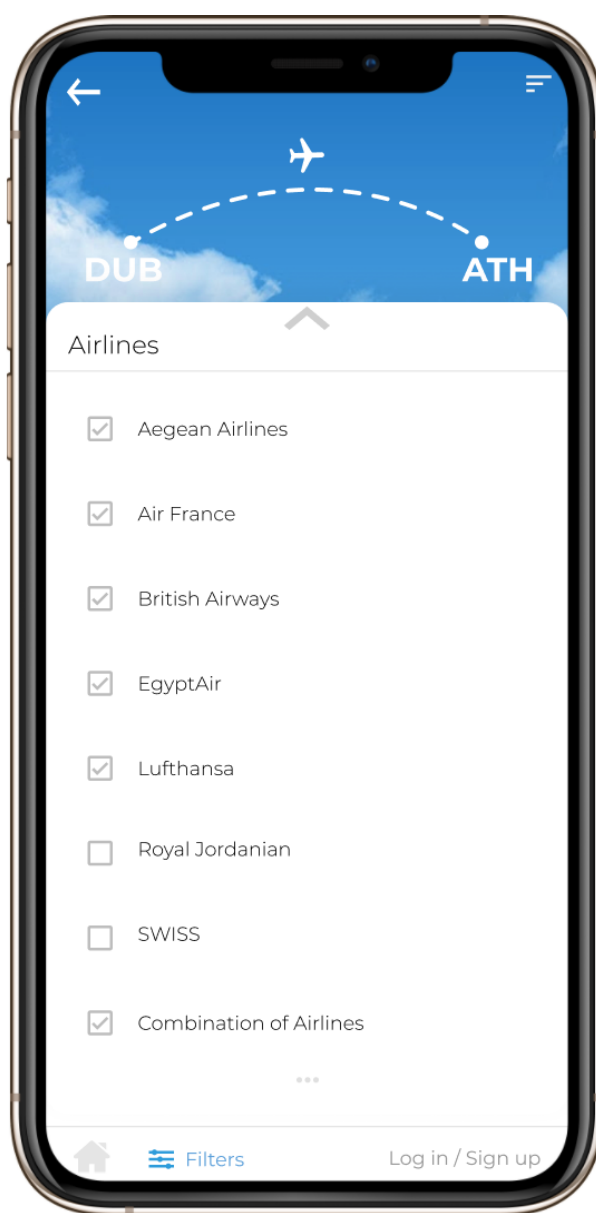


Σχήμα 15. Παράθυρο Αποτελεσμάτων

3.3 ΠΑΡΑΘΥΡΑ ΕΠΙΛΟΓΗΣ ΦΙΛΤΡΩΝ

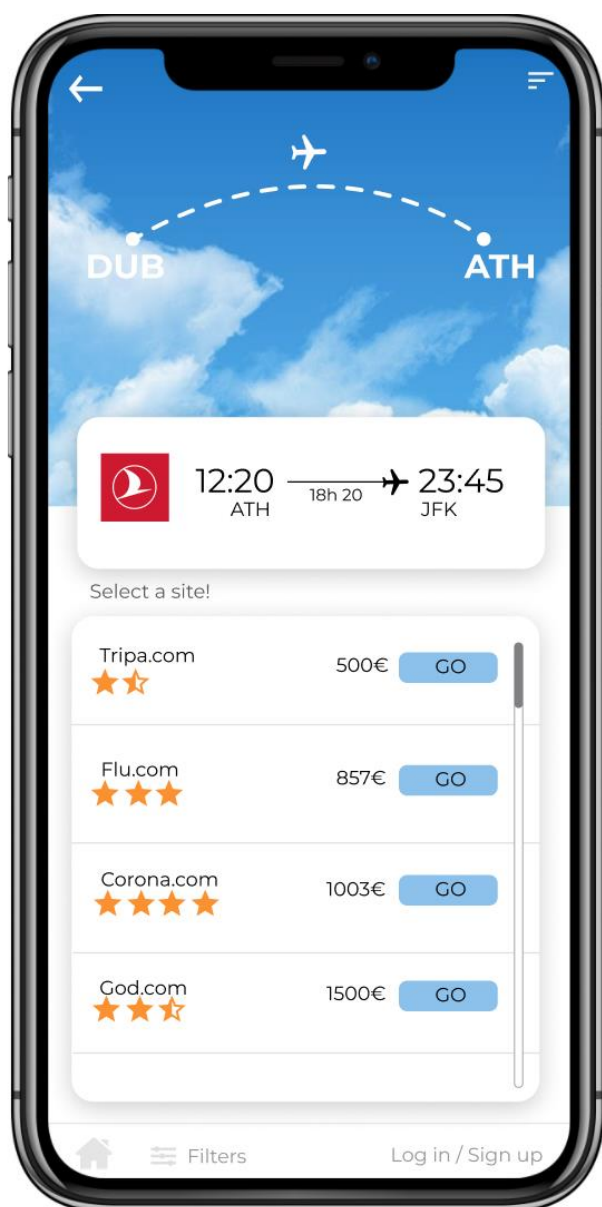


Σχήμα 16. Παράθυρο επιλογής φίλτρων

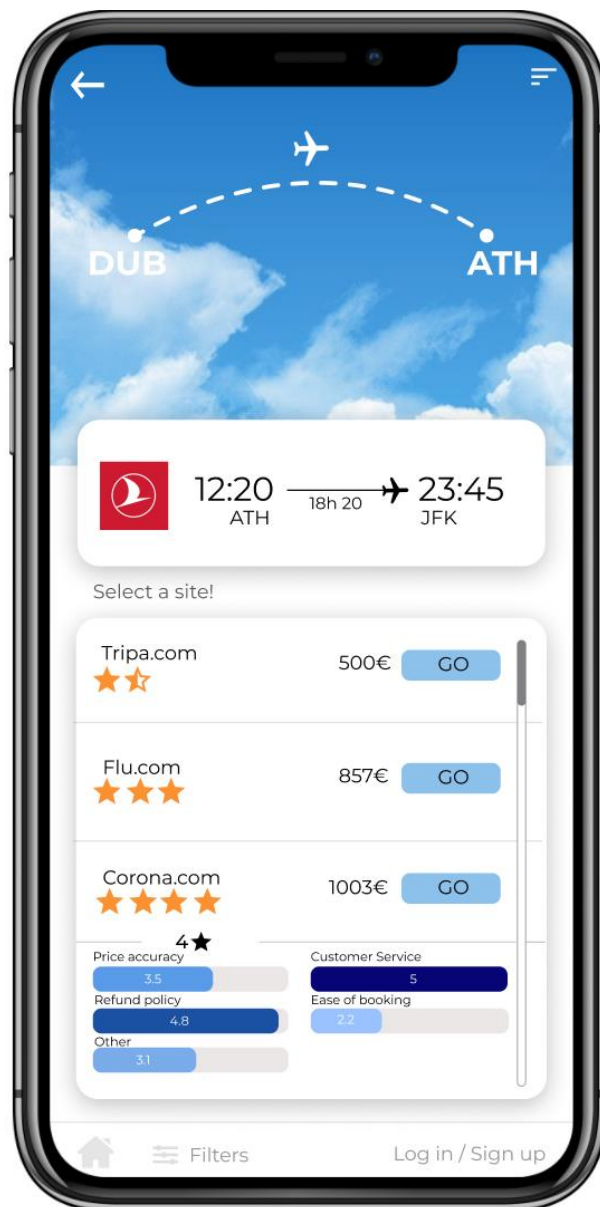


Σχήμα 17. Παράθυρο επιλογής φίλτρων

3.4 ΠΑΡΑΘΥΡΟ ΕΠΙΛΕΓΜΕΝΗΣ ΠΤΗΣΗΣ ΚΑΙ ΠΑΡΑΘΥΡΟ ΕΜΦΑΝΙΣΗΣ ΑΞΙΟΛΟΓΗΣΗΣ



Σχήμα 18. Παράθυρο επιλεγμένης πτήσης



Σχήμα 19. Παράθυρο εμφάνισης της αξιολόγησης του παρόχου



3.5 ΠΑΡΑΘΥΡΑ ΑΞΙΟΛΟΓΗΣΗΣ ΤΩΝ ΠΑΡΟΧΩΝ



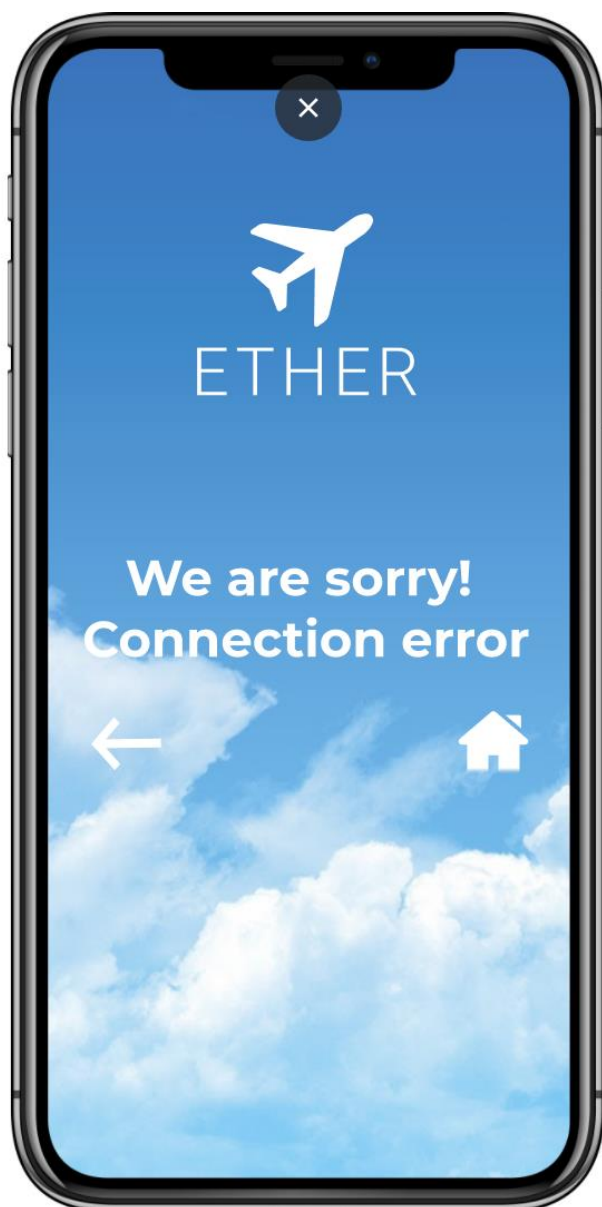
Σχήματα 20-22. Παράθυρα αξιολόγησης παρόχων

3.6 ΠΑΡΑΘΥΡΟ ΠΑΡΑΚΟΛΟΥΘΗΣΗΣ ΜΕΤΑΒΟΛΗΣ ΤΙΜΩΝ



Σχήματα 23-25. Παράθυρο παρακολούθησης μεταβολής τιμών

3.7 ΠΑΡΑΘΥΡΟ ΣΦΑΛΜΑΤΟΣ ΚΑΙ ΠΑΡΑΘΥΡΟ ΦΟΡΤΩΣΗΣ ΣΕΛΙΔΑΣ



Σχήμα 26. Παράθυρο Σφάλματος(περιέχει διαφορετικό μήνυμα σε κάθε περίπτωση)



Σχήμα 27. Παράθυρο φόρτωσης σελίδας



3.8 ΠΑΡΑΔΕΙΓΜΑ ΠΑΡΑΘΥΡΟΥ ΔΙΕΠΑΦΗΣ ΣΕ BROWSER ΑΠΟ DESKTOP

Log In/Sign up

ETHER

⚠ Covid-19 may impact your travel plans. Stay Home ! Stay Safe !

Hello, where do you want to explore?

From where from? To where to?

Depart DD/MM/YYYY Return DD/MM/YYYY Travellers & Cabin Class 1 adult, Economy

One-way Return

Budget

LET'S GO

Download on the App Store

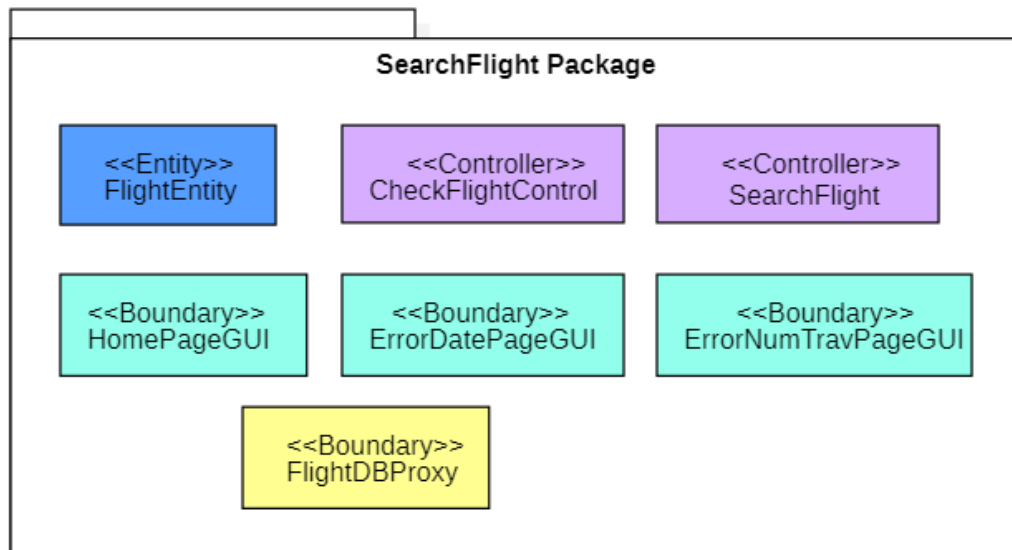
Σχήμα 28. Παράθυρο αναζήτησης πτήσης για web browser

4. ΣΤΑΤΙΚΗ ΜΟΝΤΕΛΟΠΟΙΗΣΗ

4.1 ΠΑΚΕΤΑ ΛΕΞΙΛΟΓΙΟΥ ΥΨΗΛΗΣ ΠΡΟΤΕΡΑΙΟΤΗΤΑΣ

4.1.1 Πακέτο SearchFlight

Το πακέτο αυτό περιλαμβάνει την αναζήτηση των πτήσεων, καθώς και τον έλεγχο διαθεσιμότητας των συγκεκριμένων δεδομένων της αναζήτησης. Επιπλέον περιλαμβάνει την αρχική οθόνη αναζήτησης που βλέπει ο χρήστης. Παρακάτω παρουσιάζονται οι κλάσεις που διαχειρίζονται τις παραπάνω επιλογές.



Σχήμα 29. SearchFlight Package

Entity Flight

«Entity» FlightEntity
-flightID: integer -date: string -startLocation: string -endLocation: string -numberOfTravellers: integer -cabinClass: string
+FlightEntity(flightID integer, date string, startLocation string, endLocation string, numberOfTravellers integer, CabinClass string): void +setFlightID(flightID: integer): void +setDate(date: string): void +setStartLocation(startLocation: string): void +setEndLocation(endLocation: string): void +setNumberOfTravellers(numberOfTravellers: integer): void +setCabinClass(cabinClass: string): void +getFlightID(): integer +getDate(): string +getStartLocation(): string +getEndLocation(): string +getNumberOfTravellers(): integer +getCabinClass(): string



Αυτή είναι η κλάση οντότητας της πτήσης, η οποία περιέχει τα δεδομένα που είναι απαραίτητα για μια αναζήτηση πτήσης.

Τα χαρακτηριστικά της κλάσης SearchFlight :

- flightID: Ο μοναδικός χαρακτηριστικός αριθμός για κάθε πτήση.
- date: Η ημερομηνία αναζήτησης πτήσης.
- startLocation: Το σημείο αναχώρησης του ταξιδιού.
- endLocation: Το σημείο επιστροφής του ταξιδιού.
- numberOfTravellers: Ο αριθμός ταξιδιωτών που επιλέγεται σε κάθε αναζήτηση.
- cabinClass: Η κατηγορία ναύλου που επιλέγεται στην αναζήτηση.

Οι μέθοδοι της κλάσης SearchFlight είναι :

- FlightEntity(flightID integer, date string, startLocation string, endLocation string, numberOfTravellers integer, CabinClass string): Η συνάρτηση αυτή είναι ένας δομητής της οντότητας FlightEntity.
- setFlightID(flightID: integer): Η συνάρτηση αυτή δέχεται σαν όρισμα τον μοναδικό χαρακτηριστικό αριθμό κάθε πτήσης.
- setDate(date: string): Η συνάρτηση αυτή δίνει στην ημερομηνία τη συμβολοσειρά που λαμβάνει ως όρισμα.
- setStartLocation(startLocation: string): Η συγκεκριμένη συνάρτηση δίνει στο σημείο εκκίνησης τη συμβολοσειρά που λαμβάνει ως όρισμα.
- setEndLocation(endLocation: string): Η συνάρτηση αυτή δίνει στο σημείο αναχώρησης τη συμβολοσειρά που λαμβάνει ως όρισμα.
- setCabinClass(cabinClass: string): Η συγκεκριμένη συνάρτηση δίνει στην κατηγορία ναύλου τη συμβολοσειρά που λαμβάνει ως όρισμα.
- getFlightID(): Η συνάρτηση αυτή επιστρέφει τον μοναδικό χαρακτηριστικό αριθμό για κάθε πτήση.
- getStartLocation(): Η συγκεκριμένη συνάρτηση επιστρέφει το σημείο εκκίνησης του ταξιδιού.
- getEndLocation(): Η συνάρτηση αυτή επιστρέφει το σημείο επιστροφής του ταξιδιού.
- getNumberOfTravellers(): Η συνάρτηση αυτή επιστρέφει τον αριθμό των ταξιδιωτών που επιλέγονται σε κάθε αναζήτηση.
- getCabinClass(): Η συγκεκριμένη συνάρτηση επιστρέφει την κατηγορία ναύλου που επέλεξε ο χρήστης.

Controller CheckFlightControl

«Controller» CheckFlightControl
-flightID: integer
+checkNumberOfTravellers(flightID: integer): boolean +checkDate(flightID: integer): boolean +returnError(checkFlag: boolean): string +returnHomepage(checkFlag: boolean): void +displayResults(flightID: integer): void



Ο ελεγκτής αυτός περιέχει όλες τις κατάλληλες συναρτήσεις που κρίνουν ποια θα είναι η επόμενη σελίδα του αποτελέσματος της αναζήτησης.

Χαρακτηριστικά της κλάσης :

- flightID: Ο μοναδικός χαρακτηριστικός αριθμός για κάθε πτήση.

Μέθοδοι της κλάσης :

- checkNumberOfTravellers(flightID: integer): Η μέθοδος αυτή ελέγχει τη διαθεσιμότητα του ζητούμενου αριθμού ταξιδιωτών, λαμβάνοντας τα δεδομένα από την κλάση FlightDBProxy.
- checkDate(flightID: integer): Η μέθοδος αυτή ελέγχει τη διαθεσιμότητα της ζητούμενης ημερομηνίας, λαμβάνοντας τα δεδομένα από την κλάση FlightDBProxy.
- returnError(checkFlag: boolean): Η συνάρτηση αυτή καλεί το ErrorDatePageGUI ή ErrorNumTravellersGUI σε περίπτωση εύρεσης σφάλματος.
- returnHomepage(checkFlag: boolean): Η συνάρτηση αυτή επιστρέφει τον χρήστη στην αρχική σελίδα μέσω του HomePageGUI, εφόσον υπάρξει αδυναμία εύρεσης πτήσης.
- displayResults(flightID: integer): Η συγκεκριμένη συνάρτηση είναι υπεύθυνη για τη διεξαγωγή αποτελεσμάτων των διαθέσιμων αναζητήσεων, και καλεί τον controller ResultsControl για την εμφάνιση τους.

Controller SearchFlight

«Controller» SearchFlight
+searchFlights(flightResult FlightEntity): void +budgetButtonSearch(): void

Ο ελεγκτής αυτός περιέχει την κατάλληλη συνάρτηση που είναι υπεύθυνη για την αναζήτηση των πτήσεων.

Μέθοδοι της κλάσης :

- searchFlights(flightResult FlightEntity): Η συνάρτηση αυτή είναι υπεύθυνη για την αναζήτηση των πτήσεων. Λαμβάνει τα δεδομένα από την οντότητα FlightEntity και καλεί την βάση δεδομένων να επεξεργαστεί τα στοιχεία.
- budgetButtonSearch(): Η συνάρτηση αυτή επικοινωνεί με την διεπαφή SearchButtonClick() του πακέτου ProposeTravel, ώστε η αναζήτηση να πραγματοποιηθεί μέσα σε ένα προεπιλεγμένο budget.



Boundary HomePageGUI

«Boundary» HomePageGUI
-lblLogo: Label -btnLogIn: Button -btnSignUp: Button -lblReturn: Label -btnReturn: Button -lblOneWay: Label -btnOneWay: Button -btnLetsGo: Button -lblStartLocation: Label -txtStartLocation: TextBox -lblEndLocation: Label -txtEndLocation: TextBox -lblDepartDate: Label -txtDepartDate: TextBox -lblReturnDate: Label -txtReturnDate: TextBox -lblTravellersAndClass: Label -txtTravellersAndClass: TextBox -lblReturnSymbol: Label -btnBudget: Button
+clickLetsGo(btnLetsGo: Button): void +clickBudgetButton(btnBudget: Button): void

Η κλάση αυτή εκφράζει τη διεπαφή της αρχικής σελίδας του χρήστη.

Χαρακτηριστικά της κλάσης :

- lblLogo: Η ετικέτα η οποία δίνει το λογότυπο της εφαρμογής.
- btnLogIn: Το κουμπί με το οποίο μπορεί να συνδεθεί ο χρήστης στο σύστημα.
- btnSignUp: Το κουμπί με το οποίο μπορεί να κάνει εγγραφή ένας χρήστης στο σύστημα.
- lblReturn: Η ετικέτα η οποία δείχνει την επιλογή «Μετ' επιστροφής» στον χρήστη.
- btnReturn: Το κουμπί το οποίο δίνει την επιλογή «Μετ' επιστροφής» στον χρήστη.
- lblOneWay: Η ετικέτα η οποία δείχνει την επιλογή «Απλή Μετάβαση» στον χρήστη.
- btnOneWay: Το κουμπί το οποίο δίνει την επιλογή «Απλή Μετάβαση» στον χρήστη.
- btnLetsGo: Το κουμπί το οποίο μεταφέρει τον χρήστη στην επόμενη σελίδα.
- lblStartLocation: Η ετικέτα η οποία βρίσκεται πάνω από το πεδίο συμπλήρωσης του σημείου εκκίνησης.
- txtStartLocation: Είναι το πεδίο συμπλήρωσης του σημείου εκκίνησης.
- lblEndLocation: Η ετικέτα η οποία βρίσκεται πάνω από το πεδίο συμπλήρωσης του σημείου επιστροφής.
- txtEndLocation: Είναι το πεδίο συμπλήρωσης του σημείου επιστροφής.
- lblDepartDate: Η ετικέτα η οποία βρίσκεται πάνω από το πεδίο συμπλήρωσης της ημερομηνίας αναχώρησης.
- txtDepartDate: Είναι το πεδίο συμπλήρωσης της ημερομηνίας αναχώρησης.
- lblReturnDate: Η ετικέτα η οποία βρίσκεται πάνω από το πεδίο συμπλήρωσης της ημερομηνίας επιστροφής.
- txtReturnDate: Είναι το πεδίο συμπλήρωσης του σημείου επιστροφής.



- `lblTravellersAndClass`: Η ετικέτα που βρίσκεται πάνω από το πεδίο συμπλήρωσης του αριθμού ταξιδιωτών και της κατηγορίας ναύλου.
- `txtTravellersAndClass`: Είναι το πεδίο συμπλήρωσης του αριθμού ταξιδιωτών και της κατηγορίας ναύλου.
- `lblReturnSymbol`: Η ετικέτα αυτή περιέχει ένα σύμβολο μεταξύ των πεδίων συμπλήρωσης αναχώρησης και επιστροφής.

Μέθοδοι της κλάσης :

- `clickLetsGo(btnLetsGo: Button)`: Η μέθοδος αυτή καλεί τον controller `SearchFlight`, ο οποίος ελέγχει το κουμπί που πάτησε ο χρήστης και ξεκινάει την αναζήτηση πτήσης. Για να ξεκινήσει η δράση της συνάρτησης αυτής πρέπει να συμπληρωθούν κατάλληλα τα πλαίσια της ημερομηνίας, των σημείων αναχώρησης και επιστροφής, του αριθμού ταξιδιωτών, καθώς και της κατηγορίας ναύλου.
- `clickBudgetButton(btnBudget: Button)`: Η συνάρτηση αυτή είναι υπεύθυνη για την επιλογή του κουμπιού `Budget` που πατάει ο χρήστης.

Boundary ErrorDatePageGUI

«Boundary» ErrorDatePageGUI
-errorDate: string -btnHomepage: Button
+display(error: string): void +homepageClick(): void

Πρόκειται για τη διεπαφή που είναι υπεύθυνη για την επιστροφή σφαλμάτων στον χρήστη, όσον αφορά το πεδίο των ημερομηνιών.

Χαρακτηριστικά της κλάσης:

- `errorDate`: το μήνυμα σφάλματος που προβάλλεται στον χρήστη, λόγω μη διαθέσιμου αποτελέσματος συγκεκριμένης ημερομηνίας.
- `btnHomepage`: Είναι το κουμπί το οποίο μπορεί να πατήσει ο χρήστης ώστε να επιστρέψει στην αρχική οθόνη.

Οι μέθοδοι της κλάσης είναι:

- `display(error: string)`: Η μέθοδος αυτή είναι υπεύθυνη για την προβολή του μηνύματος σφάλματος στο χρήστη. Η συνάρτηση αυτή καλεί την `returnError` της `CheckFlightControl`.
- `homepageClick()`: Η συγκεκριμένη συνάρτηση καλεί τη μέθοδο `returnHomepage` του controller `CheckFlightControl`, και μεταφέρει τον χρήστη στην αρχική οθόνη.



Boundary ErrorNumTravellersGUI

«Boundary» ErrorNumTravellersGUI
-errorNumTravellers: string -btnHomepage: Button
+display(error: string): void +homepageClick(): void

Πρόκειται για τη διεπαφή που είναι υπεύθυνη για την επιστροφή σφαλμάτων στον χρήστη, όσον αφορά το πεδίο του αριθμού ταξιδιωτών.

Χαρακτηριστικά της κλάσης:

- errorNumTravellers: το μήνυμα σφάλματος που προβάλλεται στον χρήστη, λόγω μη διαθέσιμου αποτελέσματος συγκεκριμένου αριθμού ταξιδιωτών
- btnHomepage: Είναι το κουμπί το οποίο μπορεί να πατήσει ο χρήστης ώστε να επιστρέψει στην αρχική οθόνη.

Οι μέθοδοι της κλάσης είναι:

- display(error: string): Η μέθοδος αυτή είναι υπεύθυνη για την προβολή του μηνύματος σφάλματος στο χρήστη. Η συνάρτηση αυτή καλεί την returnError της CheckFlightControl.
- homepageClick(): Η συγκεκριμένη συνάρτηση καλεί τη μέθοδο returnHomepage του controller CheckFlightControl, και μεταφέρει τον χρήστη στην αρχική οθόνη.

Boundary FlightDBProxy

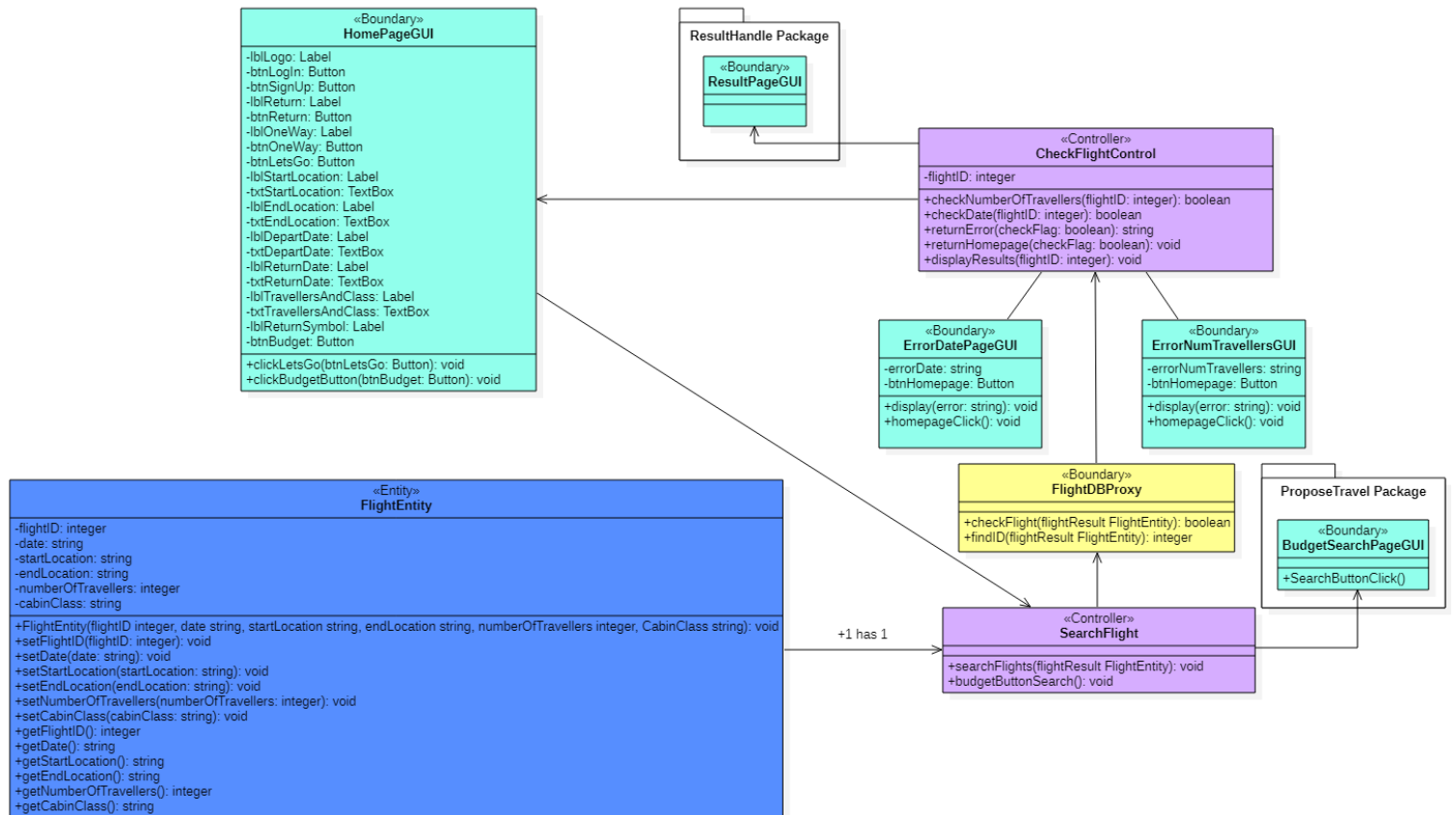
«Boundary» FlightDBProxy
+checkFlight(flightResult FlightEntity): boolean +findID(flightResult FlightEntity): integer

Αυτή η κλάση χρησιμοποιείται για την εύρεση και αποθήκευση πτήσεων. Αποτελεί δηλαδή μια βάση δεδομένων των αναζητήσεων.

Οι μέθοδοι της κλάσης είναι:

- checkFlight(date: string, startLocation: string, endLocation: string, numberOfTravellers: integer, cabinClass: string): Η μέθοδος αυτή ελέγχει αν τα στοιχεία της αναζήτησης είναι υπαρκτά. Σε περίπτωση ορθότητας επιστρέφεται η λογική τιμή true, ειδάλλως επιστρέφεται η λογική τιμή false.
- findID(date: string, startLocation: string, endLocation: string, numberOfTravellers: integer, cabinClass: string): Η μέθοδος αυτή επιστρέφει τον μοναδικό χαρακτηριστικό αριθμό της πτήσης.

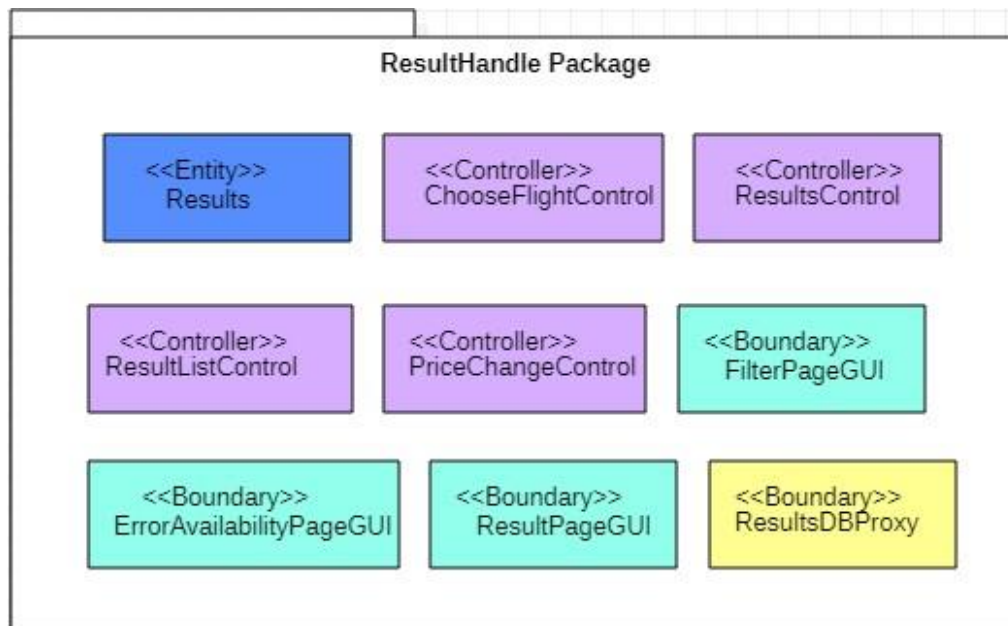
ΔΙΑΓΡΑΜΜΑ ΚΛΑΣΕΩΝ ΠΑΚΕΤΟΥ SearchFlight



Σχήμα 30. Διάγραμμα Κλάσεων SearchFlight Package

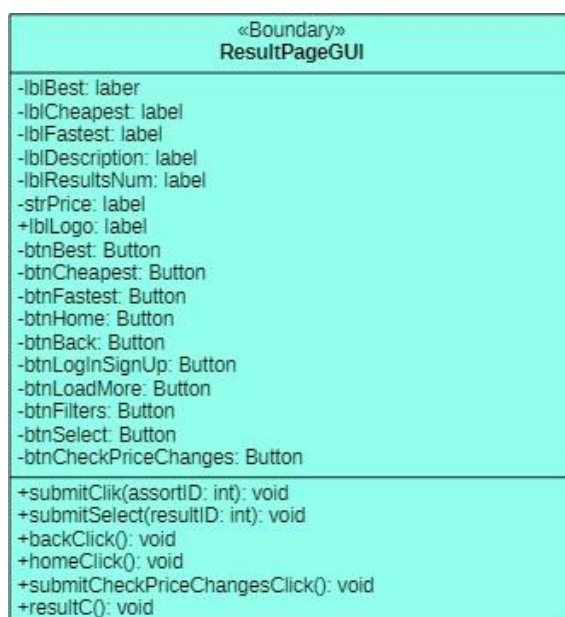
4.1.2 Πακέτο ResultHandle

Το πακέτο αναφέρεται στις επιλογές του χρήστη να επιλέξει μια πτήση , να επιλέξει φίλτρα καθώς και να ορίσει την ταξινόμηση των αποτελεσμάτων. Επιπλέον, περιλαμβάνει την οθόνη των αποτελεσμάτων που βλέπει ο χρήστης αφού έχει κάνει αναζήτηση για μία πτήση και την οθόνη που εμφανίζεται όταν θέλει να επιλέξει ένα ή περισσότερα φίλτρα. Παρακάτω παρουσιάζονται οι κλάσεις που διαχειρίζονται τις παραπάνω επιλογές.



Σχήμα 31. ResultHandle Package

Boundary ResultPage





Η κλάση αυτή είναι υπεύθυνη για την προβολή των αποτελεσμάτων αναζήτησης του χρήστη καθώς και την επιλογή ταξινόμησης τους.

Χαρακτηριστικά της κλάσης:

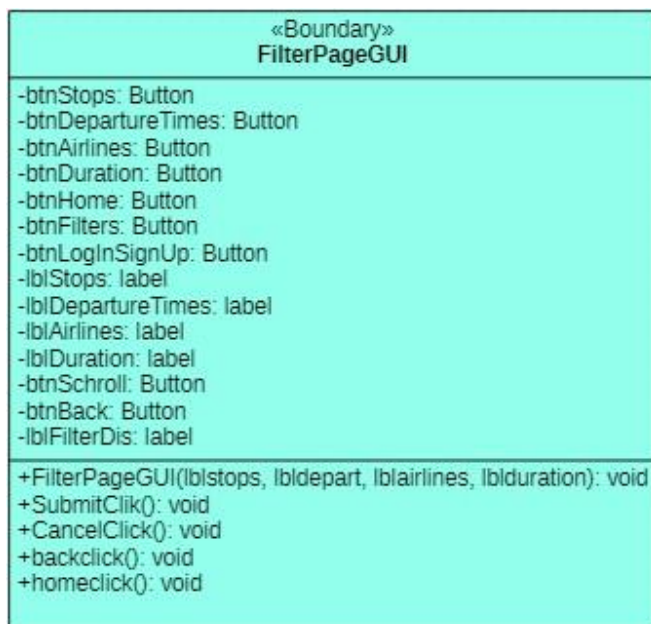
- **lblBest:** Πρόκειται για την εμφάνιση της έκφρασης <<Best>> που αποτελεί κατηγορία ταξινόμησης.
- **lblCheapest:** Πρόκειται για την εμφάνιση της έκφρασης <<Cheapest>> που αποτελεί κατηγορία ταξινόμησης.
- **lblFastest:** Πρόκειται για την εμφάνιση της έκφρασης <<Fastest>> που αποτελεί κατηγορία ταξινόμησης.
- **lblDescription:** Πρόκειται για την εμφάνιση των χαρακτηριστικών της πτήσης. Συγκεκριμένα <<ώρα αναχώρησης>> , <<ώρα άφιξης>> , <<διάρκεια ταξιδιού>> , logo αεροπορικής εταιρείας.
- **lblResultsNum:** Πρόκειται για την εμφάνιση του αριθμού των αποτελεσμάτων.
- **strPice:** Πρόκειται για την εμφάνιση της τιμής σε κάθε κατηγορία ταξινόμησης.
- **lblLogo:** Λογότυπο της εφαρμογής.
- **btnBest:** Κουμπί για την ταξινόμηση των αποτελεσμάτων με την καλύτερη επιλογή.
- **btnCheapest:** Κουμπί για την ταξινόμηση των αποτελεσμάτων κατά αύξουσα τιμή.
- **btnFastest:** Κουμπί για την ταξινόμηση των αποτελεσμάτων με την ελάχιστη διάρκεια.
- **btnHome:** Κουμπί για την επιστροφή του χρήστη στην αρχική σελίδα
- **btnBack:** Κουμπί για την επιστροφή του χρήστη στην προηγούμενη σελίδα
- **btnLogInSignUp:** Κουμπί για την είσοδο ή την εγγραφή στο σύστημα της εφαρμογής
- **btnLoadMore:** Κουμπί για την φόρτωση περισσότερων αποτελεσμάτων.
- **btnFilters:** Κουμπί για την επιλογή φίλτρων
- **btnSelect:** Κουμπί για την επιλογή ενός αποτελέσματος
- **btnCheckPriceChanges:** Κουμπί για την παρακολούθηση της μεταβολής των τιμών

Μέθοδοι της κλάσης:

- **submitClick(assortID: int):** Η συνάρτηση αυτή καλεί την συνάρτηση **resultsEdit(num:integer)** του ελεγκτή **ResultListControl** ώστε μετά την επιλογή μιας ταξινόμησης με συγκεκριμένο id να γίνει ανακατανομή των αποτελεσμάτων που παρουσιάζονται στον χρήστη.
- **submitSelect(resultID: int):** Πρόκειται για τη συνάρτηση επιλογής ενός συγκεκριμένου αποτελέσματος που συνδέεται με την κλάση έλεγχου **ResultsListControl** ().
- **homeClick():** Η συνάρτηση επιστρέφει στην αρχική σελίδα της εφαρμογής και συγκεκριμένα στον **CheckFlightControl** ().
- **backClick():** Η συνάρτηση αυτή επιστρέφει στην προηγούμενη σελίδα που σε αυτή την περίπτωση ταυτίζεται με την αρχική σελίδα της εφαρμογής.
- **submitCheckPriceChangesClick():** Η συνάρτηση αυτή καλεί την συνάρτηση **priceChangeControlClick()** του ελεγκτή **PriceChangeControl** για την παρακολούθηση της μεταβολής των τιμών.
- **resultC():** Η συνάρτηση δείχνει τα αποτελέσματα μετά την επιβολή του φίλτρου ή της ταξινόμησης



Boundary FilterPage



Η κλάση αυτή είναι υπεύθυνη για την επιλογή ενός φίλτρου και την προβολή των φιλτραρισμένων αποτελεσμάτων στον χρήστη

Χαρακτηριστικά της κλάσης:

- lblStops: Πρόκειται για την εμφάνιση της έκφρασης <<Stops>> που αποτελεί κατηγορία φίλτρου.
- lblDepartureTimes: Πρόκειται για την εμφάνιση της έκφρασης <<Departure Times>> που αποτελεί κατηγορία φίλτρου.
- lblAirlines: Πρόκειται για την εμφάνιση της έκφρασης <<Airlines>> που αποτελεί κατηγορία φίλτρου.
- lblDuration: Πρόκειται για την εμφάνιση της έκφρασης <<Duration>> που αποτελεί κατηγορία φίλτρου.
- lblFilterDis: Πρόκειται για την εμφάνιση των επιλογών του κάθε φίλτρου.
Όπως τα ονόματα των αεροπορικών εταιριών, την εμφάνιση των αριθμών των στάσεων, το φάσμα επιλογής των ωρών αναχώρησης και επιστροφής, καθώς και την επιλογή της διάρκειας.
- btnStops: Κουμπί για την επιλογή των επιθυμητών στάσεων.
- btnDepartureTimes: Κουμπί για την επιλογή της ώρας αναχώρησης.
- btnAirlines: Κουμπί για την επιλογή μιας ή πολλών αεροπορικών εταιριών .
- btnDuration: Κουμπί για την επιλογή της διάρκειας.
- btnHome: Κουμπί για την επιστροφή του χρήστη στην αρχική σελίδα
- btnBack: Κουμπί για την επιστροφή του χρήστη στην προηγούμενη σελίδα
- btnLogInSignUp: Κουμπί για την είσοδο ή την εγγραφή στο σύστημα της εφαρμογής
- btnScroll: Κουμπί για την προβολή περισσότερων φίλτρων .
- btnFilters: Κουμπί για την επιλογή φίλτρων

Μέθοδοι της κλάσης:

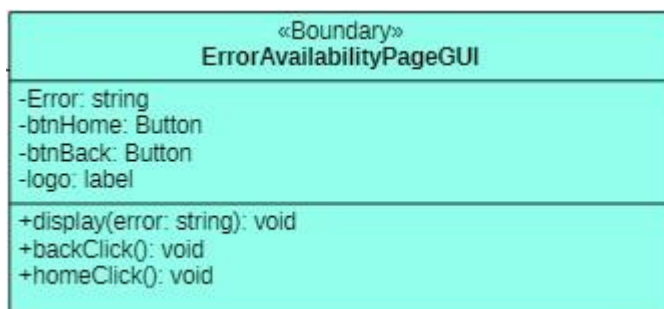
- FilterPageGUI(in lblstops, in lbldepart, in lblairlines, in lblduration): Συναρτηση δόμησης που αρχικοποιεί τις τιμές των φίλτρων.
- SubmitClick(filterID: int): Η συνάρτηση αυτή κάλει την συνάρτηση resultsEdit(num:integer) του ελεγκτή ResultListControl ώστε μετά την επιλογή ενός



φίλτρου με συγκεκριμένο id να γίνει σωστή ανακατανομή των αποτελεσμάτων που παρουσιάζονται στον χρήστη.

- `CanelClick(filterID: int)` : Η συνάρτηση αυτή κάλει την συνάρτηση και `resultsEdit(num:integer)` του ελεγκτή `ResultListControl` σε περίπτωση που ο χρήστης θελήσει να αφαιρέσει την επιλογή ενός φίλου.
- `homeClick()`: Η συνάρτηση επιστρέφει στην αρχική σελίδα της εφαρμογής.
- `backClick()`: Η συνάρτηση αυτή επιστρέφει στην προηγούμενη σελίδα που σε αυτή την περίπτωση είναι η σελίδα για την επιλογή πτήσης.

Boundary ErrorAvailabilityPage



Η κλάση αυτή είναι υπεύθυνη για να προβάλει στον χρήστη το σφάλμα κατά την επιλογή μιας πτήσης.

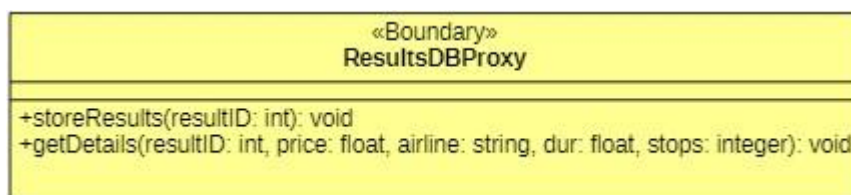
Χαρακτηριστικά της κλάσης:

- `Error`: Πρόκειται για την εμφάνιση του μηνύματος σφάλματος « Sorry this flight is no longer available »
- `btnHome`: Κουμπι για επιστροφη στην αρχιλη σελιδα
- `btnBack`: Κουμπη για επιστροφη πισω στα αποτελεσματα.
- `Logo`: το λογότυπο της εφαρμογής.

Μέθοδοι της κλάσης:

- `display(error: string)` : Η συνάρτηση αυτή εμφανίζει το μήνυμα λάθους.
- `backClick()`: Η συνάρτηση αυτή καλεί τον `ChooseFlightControl` για την επιστροφή στην σελίδα των αποτελεσμάτων.
- `homeClick()`: Η συνάρτηση αυτή επιστρέφει στην αρχική σελίδα της εφαρμογής.

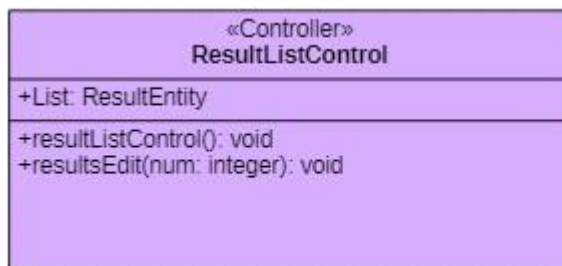
Boundary ResultsDBProxy



Η κλάση χρησιμοποιείται για την επικοινωνία με την βάση δεδομένων των αποτελεσμάτων της αναζήτησης

**Μέθοδοι της κλάσης:**

- `storeResults(resultID: int)`: Πρόκειται για τη συνάρτηση που αποθηκεύει τα αποτελέσματα στην βάση δεδομένων
- `getDitails (resultID: int)`: Πρόκειται για τη συνάρτηση που δοσμένου του id του αποτελέσματος επιστρέφει τις πληροφορίες του (τιμή, ώρα αναχώρησης κτλπ)

Controller ResultsListControl

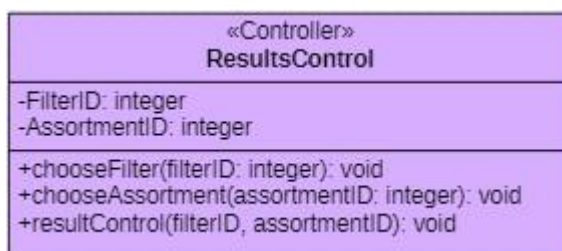
Η κλάση ελέγχου εμφανίζει την λίστα με τα αποτελέσματα της αναζήτησης του χρήστη.

Χαρακτηριστικά της κλάσης:

- List: Πρόκειται την λίστα των αποτελεσμάτων που πρόκειται να εμφανιστούν στο ResultPageGUI.

Μέθοδοι της κλάσης:

- `resultListControl()`: Η συνάρτηση αυτή αρχικοποιεί την λίστα με τα αποτελέσματα.
- `resultsEdit()`: Η συνάρτηση αυτή ταξινομεί τα αποτελέσματα μετά την επιβολή ενός φίλτρου ή μιας ταξινόμησης από τον χρήστη και καλεί την `resultC()` του FilterPageGUI ώστε να εμφανίσει στον χρήστη τα αποτελέσματα σωστά ταξινομημένα.

Controller ResultsControl

Η κλάση έλεγχου είναι υπεύθυνη για την επιβολή ενός φίλτρου ή ταξινόμησης στα αποτελέσματα.

Χαρακτηριστικά της κλάσης:

- FilterID: Πρόκειται για τον χαρακτηριστικό αριθμό του φίλτρου.
- AssortmentID: Πρόκειται για τον χαρακτηριστικό αριθμό της ταξινόμησης.

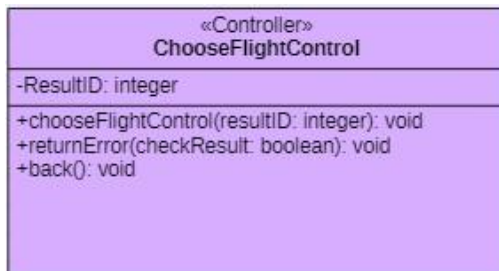
Μέθοδοι της κλάσης:

- `chooseFilter(filterID:integer)`: Η συνάρτηση αυτή καλείται σε περίπτωση που ο χρήστης θέλει να προσθέσει ένα φίλτρο.
- `chooseAssortment(assortmentID:integer)`: Η συνάρτηση αυτή καλείται σ περίπτωση που ο χρήστης θέλει να επιλέξει μια ταξινόμηση .



- `resultControl(filterID, assortmentID)`: Η μέθοδος αυτή καλεί την αντίστοιχη συνάρτηση στο proxy αντικείμενο, βάσει του μοναδικού id του φίλτρου και της ταξινόμησης ώστε να πάρει τα χαρακτηριστικά των αποτελεσμάτων.

Controller ChooseFlightControl



Η κλάση έλεγχου είναι υπεύθυνη για την επιλογή μιας πτήσης.

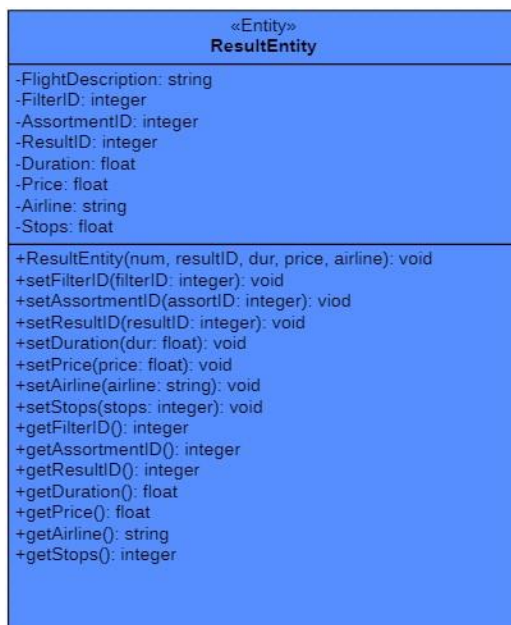
Χαρακτηριστικά της κλάσης:

- ResultID: Πρόκειται για τον χαρακτηριστικό αριθμό του αποτελέσματος .

Μέθοδοι της κλάσης:

- `chooseFlightControl (resultID:integer)`: Η συνάρτηση αυτή καλείται όταν ο χρήστης επιλέγει μια πτήση και συνδέεται με τον `ChosenFlightPropertiesGUI()` για την μετέπειτα επιλογή παρόχου.
- `returnError(checkResult: boolean)`: Η συνάρτηση αυτή καλεί το `ErrorAvailabilityPageGUI` σε περίπτωση σφάλματος κατά την επιλογή της πτήσης.
- `back()`: η συνάρτηση αυτή καλεί σε περίπτωση σφάλματος για την επιλογή άλλης πτήσης.

Entity Supplier





Αυτή είναι η κλάση οντότητας των αποτελεσμάτων και περιέχει τα στοιχεία για τον ορισμό της και τα δεδομένα για κάθε αποτέλεσμα

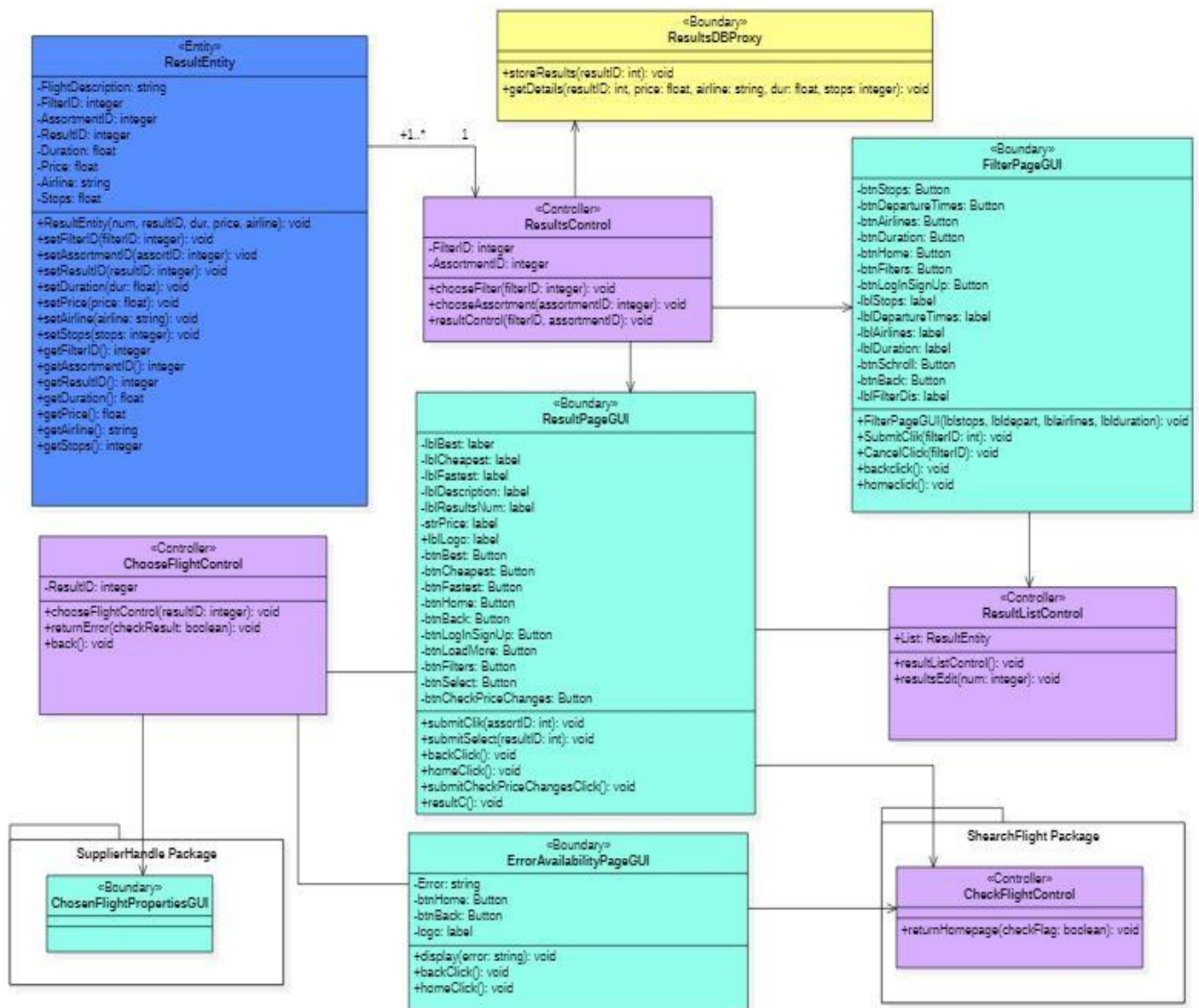
Χαρακτηριστικά της κλάσης:

- FilterID: Πρόκειται για τον μοναδικό αριθμό που χαρακτηρίζει κάθε φίλτρο.
- FlightDescription: Πρόκειται για τα χαρακτηριστικά της πτήσης όπως χώρα, αεροδρόμιο κτλπ
- AssortmentID: Πρόκειται για τον μοναδικό αριθμό που χαρακτηρίζει κάθε ταξινόμηση.
- ResultID: Πρόκειται για τον μοναδικό αριθμό που χαρακτηρίζει κάθε αποτέλεσμα.
- Duration: Η διάρκεια του ταξιδιού.
- Price: Η τιμή του αποτελέσματος.
- Airline: Η αεροπορική εταιρία κάθε αποτελέσματος.
- Stops: Ο αριθμός των στάσεων κάθε πτήσης.

Μέθοδοι της κλάσης:

- ResultEntity(num, resulted, dur, price, airline): Η συνάρτηση αυτή είναι ένας δομητής της οντότητας ResultEntity
- setFilterID(filterID:integer): Η συνάρτηση αυτή θέτει το όρισμα που δέχεται στο μοναδικό χαρακτηριστικό αριθμό χρήστη που χαρακτηρίζει κάθε φίλτρο.
- setAssortmentID(assortID:integer): Η συνάρτηση αυτή θέτει το όρισμα που δέχεται στο μοναδικό χαρακτηριστικό αριθμό χρήστη που χαρακτηρίζει κάθε ταξινόμησης.
- setResultID(resultID:integer): Η συνάρτηση αυτή θέτει το όρισμα που δέχεται στο μοναδικό χαρακτηριστικό αριθμό χρήστη που χαρακτηρίζει κάθε αποτέλεσμα.
- setDuration(dur:float): Η συνάρτηση αυτή θέτει την διάρκεια μιας πτήσης.
- setPrice(prise:float): Η συνάρτηση αυτή θέτει την τιμή μιας πτήσης.
- setAirline(airline: string): Η συνάρτηση αυτή θέτει το όνομα της αεροπορικής εταιρίας του ταξιδιού.
- setStops(stops: integer): Η συνάρτηση αυτή θέτει τις στάσεις ενός ταξιδιού.
- setFlightDescription(desc: string): Η συνάρτηση θέτει την περιγραφή της πτήσης
- getFilterID(): Η συνάρτηση αυτή επιστρέφει τον μοναδικό αριθμό που χαρακτηρίζει κάθε φίλτρο.
- getAssortmentID(): Η συνάρτηση αυτή επιστρέφει τον μοναδικό αριθμό που χαρακτηρίζει κάθε ταξινόμηση.
- getResultID(): Η συνάρτηση αυτή επιστρέφει τον μοναδικό αριθμό που χαρακτηρίζει κάθε αποτέλεσμα.
- getDuration(): Η συνάρτηση αυτή επιστρέφει την διάρκεια μιας πτήσης.
- getPrice(): Η συνάρτηση αυτή επιστρέφει την τιμή του ταξιδιού.
- getAirline(): Η συνάρτηση αυτή επιστρέφει το όνομα της αεροπορικής εταιρίας
- getStops(): Η συνάρτηση αυτή επιστρέφει τον αριθμό των ενδιάμεσων στάσεων.
- getFlightDescription(): Η συνάρτηση αυτή επιστρέφει τα γονικά χαρακτηριστικά του ταξιδιού.

ΔΙΑΓΡΑΜΜΑ ΚΛΑΣΕΩΝ ΠΑΚΕΤΟΥ ResultHandle

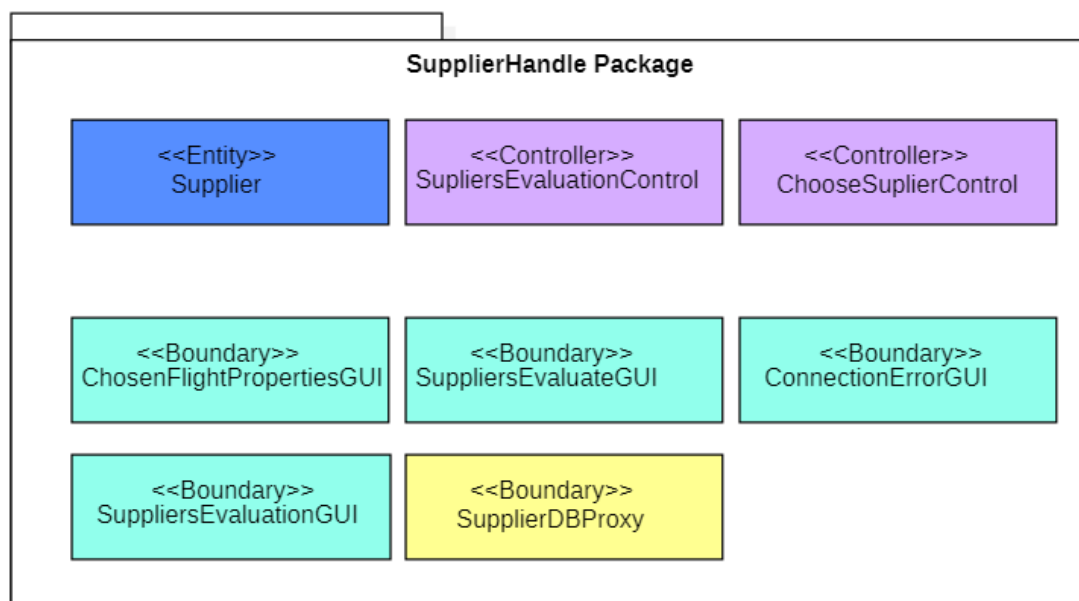


Σχήμα 32. Διάγραμμα Κλάσεων ResultHandle Package

4.2 ΠΑΚΕΤΑ ΛΕΞΙΛΟΓΙΟΥ ΜΕΣΑΙΑΣ ΠΡΟΤΕΡΑΙΟΤΗΤΑΣ

4.2.1 Πακέτο SupplierHandle

Το πακέτο αυτό αναφέρεται στις επιλογές του χρήστη να αξιολογήσει παρόχους (αεροπορικές εταιρείες και ταξιδιωτικά γραφεία), να δει αξιολογήσεις παρόχων και να επιλέξει πάροχο για αγορά των εισιτηρίων. Επιπροσθέτως, περιλαμβάνει την οθόνη που βλέπει ο χρήστης αφού έχει επιλέξει την πτήση που επιθυμεί, την οθόνη που βλέπει τις αξιολογήσεις των παρόχων καθώς και την οθόνη που καταθέτουν αξιολογήσεις οι εγγεγραμμένοι χρήστες. Παρακάτω παρουσιάζονται οι κλάσεις που διαχειρίζονται τις παραπάνω επιλογές.



Σχήμα 33. SupplierHandle Package

Boundary ChosenFlightProperties

«Boundary» ChosenFlightPropertiesGUI
-LblChosenFlightProperties: label -LblSelectSite: label -BtnHome: button -BtnBack: button -LblLogo: Label -BtnLogin: button -BtnSignup: button -BtnGo: button -supplierList: comboBox -BtnShowMore: button
+homeClick(): void +backClick(): void +LoginClick(): Void +SignupClick(): void +goClick(): void



Η διεπαφή αυτή είναι υπεύθυνη για την προβολή των χαρακτηριστικών της επιλεγμένης πτήσης καθώς και την λίστα πιθανών παρόχων εισιτηρίων (αεροπορικές εταιρείες, ταξιδιωτικά γραφεία)

Χαρακτηριστικά της κλάσης:

- LblChosenFlightProperties: Πρόκειται για την εμφάνιση των χαρακτηριστικών της επιλεγμένης πτήσης. Συγκεκριμένα <<ώρα αναχώρησης>>, <<ώρα άφιξης>>, <<διάρκεια ταξιδιού>>, logo αεροπορικής εταιρείας
- LblSelectSite: Πρόκειται για την εμφάνιση της έκφρασης <<Select a site!>>
- BtnHome: Κουμπί για την επιστροφή του χρήστη στην αρχική σελίδα
- BtnBack: Κουμπί για την επιστροφή του χρήστη στην προηγούμενη σελίδα
- LblLogo: Λογότυπο της εφαρμογής
- BtnLogin: Κουμπί για την είσοδο στο σύστημα της εφαρμογής
- BtnSignup: Κουμπί για την εγγραφή χρήστη στο σύστημα της εφαρμογής
- BtnGo: Κουμπί για την μετάβαση του χρήστη στο site του παρόχου
- supplierList: Λίστα όλων των παρόχων από τους οποίους μπορεί ο χρήστης να προβεί στην αγορά του εισιτηρίου. Συγκεκριμένα περιλαμβάνει τα ονόματα των παρόχων, την τιμή του εισιτηρίου, την συνολική αξιολόγηση (αν υπάρχει).
- BtnShowMore: Κουμπί για την προβολή της αξιολόγησης του παρόχου

Μέθοδοι της κλάσης:

- homeClick(): Η μέθοδος αυτή καλεί την συνάρτηση returnHomepage του Controller CheckFlightControl για να επιστρέψει ο χρήστης στην αρχική σελίδα
- backClick(): Η μέθοδος αυτή επιστρέφει τον χρήστη στην προηγούμενη σελίδα
- goClick(): Η μέθοδος αυτή καλεί την συνάρτηση ChooseSupplier() του ελεγκτή ChooseSupplierControl, ο οποίος είναι υπεύθυνος για την επιλογή και την σύδεση του χρήστη με το site του παρόχου.

Boundary SuppliersEvaluation

«Boundary» SuppliersEvaluationGUI
-LblMeanEvaluation: label
-LblPriceAccuracy: label
-LblPriceAccuracyEvaluation: label
-LblCostumerService: label
-LblCostumerServiceEvaluation: label
-LblRefundPolicy: label
-LblRefundPolicyEvaluation: label
-LblEaseBooking: label
-LblEaseBookingEvaluation: label
-LblOther: label
-LblOtherEvaluation: label
+supplierSelected(): void

Αυτή είναι η κλάση για την προβολή των αξιολογήσεων του κάθε παρόχου. Επιπλέον κληρονομεί όλα τα χαρακτηριστικά της κλάσης ChosenFlightProperties

Χαρακτηριστικά της κλάσης:

- LblMeanEvaluation: Πρόκειται για την εμφάνιση της μέσης τιμής της αξιολόγησης του παρόχου
- LblPriceAccuracy: Πρόκειται για την εμφάνιση της έκφρασης <<price accuracy>> που αποτελεί κατηγορία αξιολόγησης



- LblPriceAccuracyEvaluation: Πρόκειται για την εμφάνιση της αξιολόγησης στην συγκεκριμένη κατηγορία
- LblCostumerService: Πρόκειται για την εμφάνιση της έκφρασης <<costumer service>> που αποτελεί κατηγορία αξιολόγησης
- LblCostumerServiceEvaluation: Πρόκειται για την εμφάνιση της αξιολόγησης στην συγκεκριμένη κατηγορία
- LblRefundPolicy: Πρόκειται για την εμφάνιση της έκφρασης <<refund policy>> που αποτελεί κατηγορία αξιολόγησης
- LblRefundPolicyEvaluation: Πρόκειται για την εμφάνιση της αξιολόγησης στην συγκεκριμένη κατηγορία
- LblEaseBooking: Πρόκειται για την εμφάνιση της έκφρασης <<ease of booking>> που αποτελεί κατηγορία αξιολόγησης
- LblEaseBookingEvaluation: Πρόκειται για την εμφάνιση της αξιολόγησης στην συγκεκριμένη κατηγορία
- LblOther: Πρόκειται για την εμφάνιση της έκφρασης <<other>> που αποτελεί κατηγορία αξιολόγησης
- LblOtherEvaluation: Πρόκειται για την εμφάνιση της αξιολόγησης στην συγκεκριμένη κατηγορία

Μέθοδοι της κλάσης:

- supplierSelected(): η συνάρτηση αυτή καλεί την συνάρτηση loadSupplierEvaluation() του ελεγκτή SuppliersEvaluationControl, για την προβολή της αξιολόγησης του επιλεγμένου παρόχου

Boundary SuppliersEvaluate

«Boundary» SuppliersEvaluateGUI
-logo: label -LblChooseAirAgen: label -LblPriceAccuracy: label -BtnPriceAccuracyStars: button -LblCostumerService: label -BtnCostumerServiceStars: button -LblRefundPolicy: label -BtnRefundPolicyStars: button -LblEaseBooking: label -BtnEaseBookingStars: button -LblOther: label -BtnOtherStars: button -chooseAirAgenList: comboBox -BtnAdd: button
+airAgenSelected(): void +addClick(): void

Αυτή είναι η κλάση για την κατάθεση αξιολογήσεων στον κάθε πάροχο από έναν εγγεγραμμένο χρήστη.

Χαρακτηριστικά της κλάσης:

- Logo: Λογότυπο της εφαρμογής
- LblChooseAirAgen: Πρόκειται για την εμφάνιση της έκφρασης <<Choose an airline/travel agency>>
- LblPriceAccuracy: Πρόκειται για την εμφάνιση της έκφρασης <<price accuracy>> που αποτελεί κατηγορία αξιολόγησης
- BtnPriceAccuracyStars: Κουμπί υπό μορφή αστεριών για την για την καταχώρηση της αξιολόγησης με βάση την τιμή

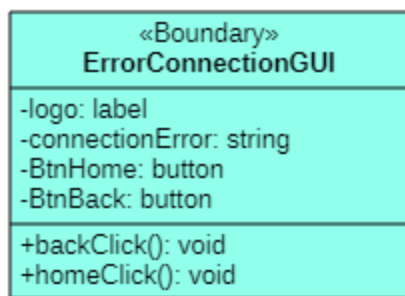


- LblCostumerService:Πρόκειται για την εμφάνιση της έκφρασης <<costumer service>> που αποτελεί κατηγορία αξιολόγησης
- BtnCostumerServiceStars:Κουμπί υπό μορφή αστεριών για την για την καταχώρηση της αξιολόγησης με βάση την προμήθεια υπηρεσιών
- LblRefundPolicy:Πρόκειται για την εμφάνιση της έκφρασης <<refund policy>> που αποτελεί κατηγορία αξιολόγησης
- BtnRefundPolicyStars:Κουμπί υπό μορφή αστεριών για την για την καταχώρηση της αξιολόγησης με βάση την πολιτική επιστροφής χρημάτων
- LblEaseBooking:Πρόκειται για την εμφάνιση της έκφρασης <<ease of booking>> που αποτελεί κατηγορία αξιολόγησης
- BtnEaseBookingStars:Κουμπί υπό μορφή αστεριών για την για την καταχώρηση της αξιολόγησης με βάση την διαδικασία κράτησης
- LblOther:Πρόκειται για την εμφάνιση της έκφρασης <<other>> που αποτελεί κατηγορία αξιολόγησης
- BtnOtherStars:Κουμπί υπό μορφή αστεριών για την για την καταχώρηση της αξιολόγησης με βάση άλλους από τους προαναφερθέντες λόγους
- chooseAirAgenList: Λίστα επιλογής από όλες τις αεροπορικές εταιρείες και τα ταξιδιωτικά γραφεία με σκοπό την κατάθεση κριτικής
- BtnAdd: κουμπί για την προσθήκη κριτικής σε πάροχο

Μέθοδοι της κλάσης:

- airAgenSelected(): η συνάρτηση αυτή καλεί την συνάρτηση LoadSupplier() του ελεγκτή SuppliersEvaluationControl, για την προβολή του παρόχου προς αξιολόγηση
- addClick(): Η μέθοδος αυτή καλεί την συνάρτηση addEvaluation() του ελεγκτή SuppliersEvaluationControl, και προσθέτει μία αξιολόγηση στον πάροχο με βάσει τις επιλογές που έχουν γίνει στα priceAccuracyStars, costumerServiceStars, refundPolicyStars, easeBookingStars, otherStars.

Boundary ErrorConnection



Αυτή είναι η κλάση είναι υπεύθυνη για την προβολή στον χρήστη σφάλμα σύνδεσης με πάροχο

Χαρακτηριστικά της κλάσης:

- Logo: Λογότυπο της εφαρμογής
- connectionError: Πρόκειται για την εμφάνιση της έκφρασης <<We are sorry! Connection error with *supplier*>>
- BtnHome: Κουμπί για την επιστροφή του χρήστη στην αρχική σελίδα
- BtnBack: Κουμπί για την επιστροφή του χρήστη στην προηγούμενη σελίδα

Μέθοδοι της κλάσης:

- homeClick(): Η μέθοδος αυτή καλεί την συνάρτηση returnHomepage του Controller CheckFlightControl για να επιστρέψει ο χρήστης στην αρχική σελίδα



- backClick(): Η μέθοδος αυτή επιστρέφει τον χρήστη στην προηγούμενη σελίδα

Boundary SuppliersDBProxy

«Boundary» SuppliersDBProxy
+addSupplier(supplier SupplierEntity): void +loadSupplier(supplierID: integer): SupplierEntity +loadSupplierEvaluation(supplierID: integer): SupplierEntity

Αυτή η κλάση χρησιμοποιείται για την επικοινωνία με τη βάση δεδομένων των παρόχων

Μέθοδοι της κλάσης:

- addSupplier(supplier SupplierEntity): Πρόκειται για τη συνάρτηση που αποθηκεύει έναν πάροχο στη βάση δεδομένων
- loadSupplier(supplierID: integer): Πρόκειται για τη συνάρτηση που φορτώνει τα δεδομένα του ζητούμενου παρόχου
- loadSupplierEvaluation(supplierID: integer): Πρόκειται για τη συνάρτηση που δοσμένου του id του παρόχου επιστρέφει την αξιολόγηση που έχει δεχθεί

Controller SupplierEvaluationControl

«Controller» SuppliersEvaluationControl
+addSupplier(supplier SupplierEntity): void +loadSupplier(supplierID: integer): SupplierEntity +loadSupplierEvaluation(supplierID: integer): SupplierEntity +addEvaluation(supplierID: integer): void +meanEvaluation(numEval: integer, eval: float, addEvaluation: void): float +editEvaluationStatus(evaluationExist: boolean): boolean

Ο ελεγκτής αυτός είναι υπεύθυνος για την την προβολή, την προσθήκη και τον υπολογισμό της αξιολόγησης του κάθε παρόχου καθώς επίσης να τροποποιεί κάποια δεδομένα των παρόχων.

Μέθοδοι της κλάσης:

- addSupplier(supplier SupplierEntity): Η μέθοδος αυτή καλεί την αντίστοιχη συνάρτηση αποθήκευσης παρόχου στο proxy αντικείμενο με σκοπό την αποθήκευση του νέου παρόχου
- loadSupplier(supplierID: integer) : Η μέθοδος αυτή καλεί την αντίστοιχη συνάρτηση φόρτωσης παρόχου στο proxy αντικείμενο, βάσει του μοναδικού αναγνωριστικού του εν λόγω παρόχου
- loadSupplierEvaluation(supplierID: integer): Η μέθοδος αυτή καλεί την αντίστοιχη συνάρτηση φόρτωσης της αξιολόγησης του παρόχου στο proxy αντικείμενο, βάσει του μοναδικού αναγνωριστικού του εν λόγω παρόχου για την προβολή της αξιολόγησης του
- addEvaluation(supplierID: integer): Η μέθοδος καλείται κάθε φορά που ο χρήστης επιθυμεί να προσθέσει μία κριτική



- `meanEvaluation(numEval: integer, eval:float)`: Η μέθοδος αυτή καλείται όταν γίνεται η `addEvaluation(supplierID: integer)` για να υπολογιστεί ο νέος μέσος όρος των αξιολογήσεων που προβάλλονται στον χρήστη
- `editEvaluationStatus(evaluationExist: boolean)`: Η μέθοδος αυτή καλεί την συνάρτηση `setEvaluationExistence(evalExist:boolean)` όταν γίνεται η πρώτη αξιολόγηση του χρήστη για να θέσει την τιμή `boolean` σε `true`

Controller `SupplierEvaluationControl`

«Controller» ChooseSupplierControl
<code>+ChooseSupplier(supplierID: integer): void</code> <code>+ErrorConnection(supplierID: integer, checkFlag: boolean): void</code>

Ο ελεγκτής αυτός είναι υπεύθυνος για την την σύνδεση του χρήστη με τον πάροχο αλλά και την προβολή της σελίδα σφάλματος σε περίπτωση αποτυχίας σύνδεσης

Μέθοδοι της κλάσης:

- `ChooseSupplier(supplierID: integer)`: Η μέθοδος αυτή καλεί την αντίστοιχη συνάρτηση φόρτωσης παρόχου στο `procy` αντικείμενο, βάσει του μοναδικού αναγνωριστικού του εν λόγω παρόχου και χρησιμοποιεί το `SupplierURL` για να συνδέσει το χρήστη με τον πάροχο
- `ErrorConnection(supplierID: integer,checkFlag: boolean)`: Η παρούσα συνάρτηση καλεί το `ErrorConnectionGUI` σε περίπτωση σφάλματος φόρτωσης της σελίδας του παρόχου

Entity `Supplier`

«Entity» Supplier
<code>-supplierID: integer</code> <code>-supplierName: string</code> <code>-numOfEvaluation: integer</code> <code>-evaluation: float</code> <code>-evaluationExistence: boolean</code> <code>-supplierURL: string</code>
<code>+SupplierEntity(int id, string name, int numOfEval, float eval, boolean evalExist, string supURL)</code> <code>+setSupplierID(id: integer): void</code> <code>+setSupplierName(name: string): void</code> <code>+setNumOfEvaluation(numOfEval: integer): void</code> <code>+setEvaluation(eval: float): void</code> <code>+setEvaluationExistence(evalExist: boolean): void</code> <code>+setSupplierURL(supURL: string): void</code> <code>+getSupplierID(): integer</code> <code>+getSupplierName(): string</code> <code>+getNumOfEvaluation(): integer</code> <code>+getEvaluation(): float</code> <code>+getEvaluationExistence(): boolean</code> <code>+getSupplierURL(): string</code>

Αυτή είναι η κλάση οντότητας του παρόχου και περιέχει τα στοιχεία για τον ορισμό της και τα δεδομένα για την αξιολόγηση που δέχεται

**Χαρακτηριστικά της κλάσης:**

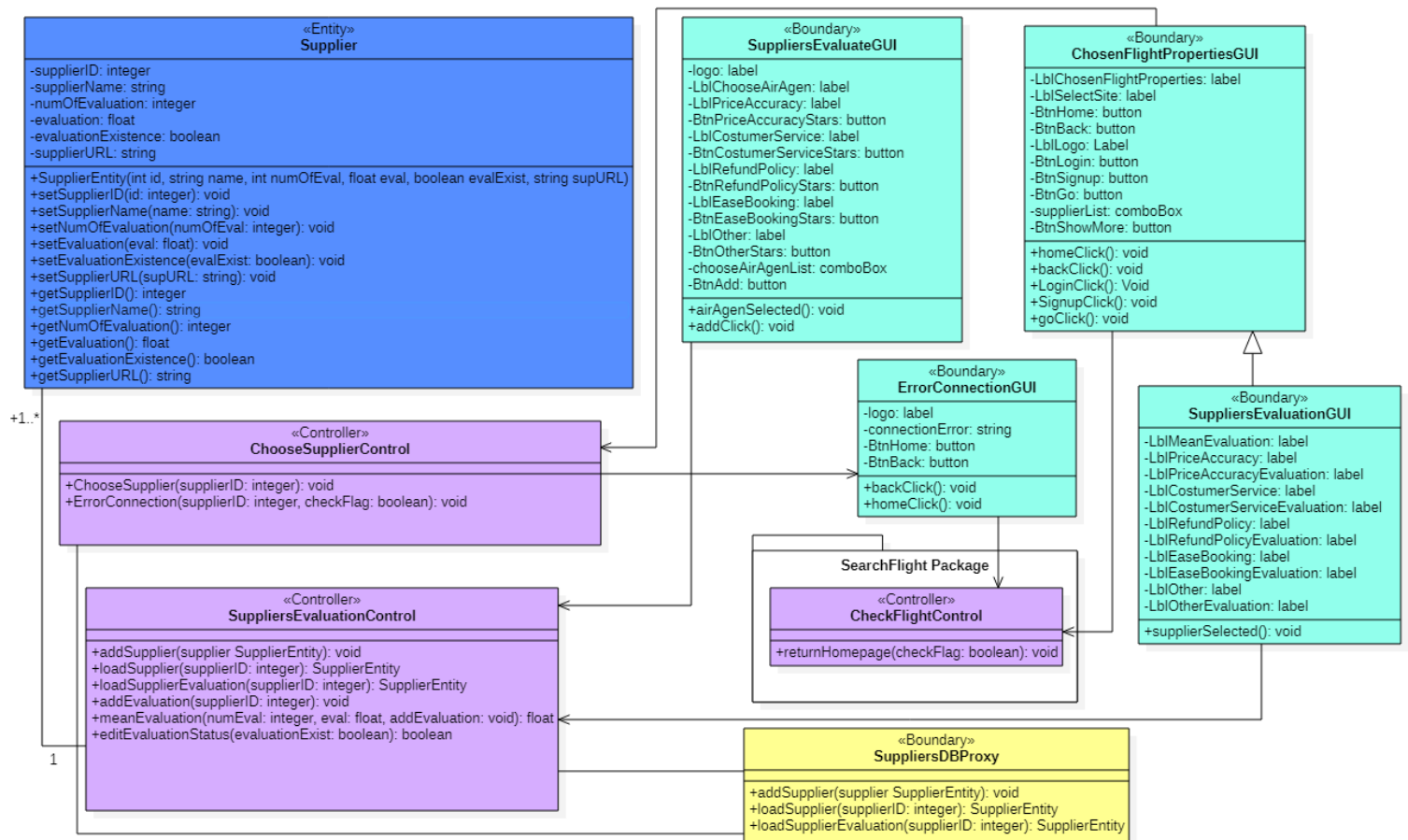
- **SupplierID:** Πρόκειται για τον μοναδικό αριθμό που χαρακτηρίζει κάθε πάροχο.
- **SupplierName:** Πρόκειται για το όνομα του παρόχου
- **numOfEvaluation:** Ο αριθμός των αξιολογήσεων που έχει κάθε πάροχος
- **Evaluation:** Η αξιολόγηση που έχει κάθε πάροχος
- **EvaluationExistence:** Η μεταβλητή που δείχνει την ύπαρξη ή όχι αξιολόγησης σε ένα πάροχο
- **SupplierURL:** Ο URL σύνδεσμος ώστε να συνδέει το σύστημα τον χρήστη με τον πάροχο

Μέθοδοι της κλάσης:

- **SupplierEntity(int id, string name, int numOfEval, float eval, boolean evalExist, string supURL):** Η συνάρτηση αυτή είναι ένας δομητής της οντότητας SupplierEntity
- **setSupplierID(in id:integer):** Η συνάρτηση αυτή θέτει το όρισμα που δέχεται στο μοναδικό χαρακτηριστικό αριθμό χρήστη που χαρακτηρίζει κάθε πάροχο
- **setSupplierName(name:string):** Η συνάρτηση αυτή θέτει το όνομα του παρόχου την συμβολοσειρά που λαμβάνει ως όρισμα
- **setNumOfEvaluation(numOfEval:integer):** Η συνάρτηση αυτή θέτει το όρισμα που παίρνει ως τον αριθμό των αξιολογήσεων
- **setEvaluation(eval:float):** Η συνάρτηση αυτή θέτει το όρισμα που λαμβάνει ως την αξιολόγηση που χαρακτηρίζει κάθε πάροχο
- **setEvaluationExistence(evalExist:boolean):** Η συνάρτηση αυτή θέτει το όρισμα που λαμβάνει ως την μεταβλητή που δείχνει την ύπαρξη ή όχι αξιολόγησης σε ένα πάροχο
- **setSupplierURL(supURL:string):** Η συνάρτηση αυτή θέτει το URL του παρόχου την συμβολοσειρά που λαμβάνει ως όρισμα
- **getSupplierID():** Η συνάρτηση αυτή επιστρέφει τον μοναδικό αριθμό που χαρακτηρίζει κάθε πάροχο
- **getSupplierName():** Η συνάρτηση αυτή επιστρέφει το όνομα του παρόχου
- **getNumOfEvaluation():** Η συνάρτηση αυτή επιστρέφει τον αριθμό των αξιολογήσεων που έχει κάθε πάροχος
- **getEvaluation():** Η συνάρτηση αυτή επιστρέφει την αξιολόγηση που έχει κάθε πάροχος
- **getEvaluationExistence():** Η συνάρτηση αυτή επιστρέφει την μεταβλητή που δείχνει την ύπαρξη ή όχι αξιολόγησης σε ένα πάροχο
- **getSupplierURL():** Η συνάρτηση αυτή επιστρέφει τον URL σύνδεσμο ώστε να συνδέει το σύστημα τον χρήστη με τον πάροχο



ΔΙΑΓΡΑΜΜΑ ΚΛΑΣΕΩΝ ΠΑΚΕΤΟΥ SupplierHandle

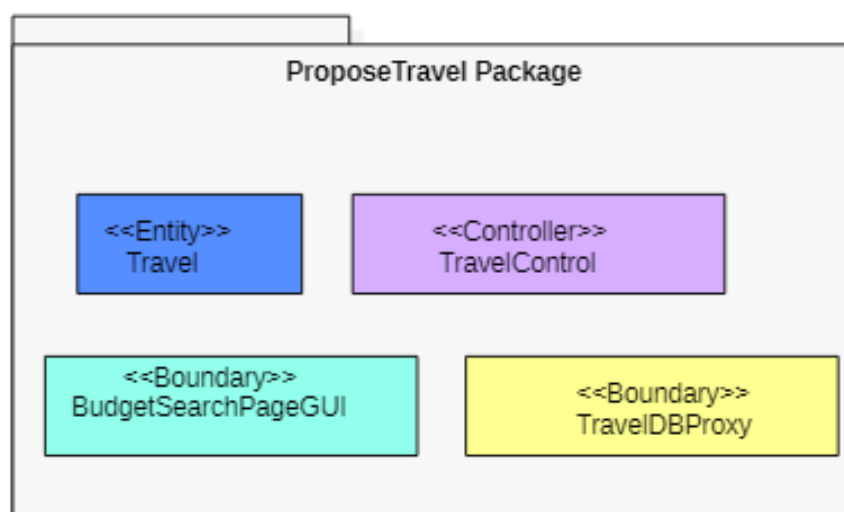


Σχήμα 34. Διάγραμμα Κλάσεων SupplierHandle Package

4.3 ΠΑΚΕΤΑ ΛΕΞΙΛΟΓΙΟΥ ΧΑΜΗΛΗΣ ΠΡΟΤΕΡΑΙΟΤΗΤΑΣ

4.3.1 Πακέτο ProposeTravel

Το πακέτο αυτό αναφέρεται στην δυνατότητα του χρήστη να αναζητήσει μία πτήση με βάση το budget που μπορεί να διαθέσει. Επιπλέον, περιλαμβάνει την οθόνη που βλέπει ο χρήστης όταν κάνει αυτής της μορφής αναζήτησης.



Σχήμα 35. ProposeTravel Package

Boundary BudgetSearchPageGUI

«Boundary» BudgetSearchPageGUI
-MinBudget: Button -MaxBudget: Button -DepDate: TextBox -ReturnDate: TextBox -LblDepartDate: Label -LblReturnDate: Label -LblDepart: Label -LblReturn: Label -SearchButton: Button -ButtonLogin: Button -ButtonSignup: Button -Logo: Image -LblTravelQuestion: Label -LblBudgetQuestion: Label -LblBudget: Label -ButtonBudget: Button
+SearchButtonClick() +BudgetButtonClick()

Πρόκειται για την διεπαφή της σελίδας όπου ο χρήστης μπορεί να επιλέξει να κάνει αναζήτηση πτήσεων ανάλογα με το Budget του.



Χαρακτηριστικά κλάσης:

- **MinBudgetButton:** Είναι το κουμπί που χρησιμοποιεί ο χρήστης κάνοντας scrolling, για να επιλέξει την ελάχιστη τιμή.
- **MaxBudgetButton:** Είναι το κουμπί που χρησιμοποιεί ο χρήστης κάνοντας scrolling, για να επιλέξει την μέγιστη τιμή.
- **DepDate:** Είναι το text Box όπου ο χρήστης γράφει την ημερομηνία αναχώρησης.
- **ReturnDate:** Είναι το text Box όπου ο χρήστης γράφει την ημερομηνία επιστροφής.
- **LblDepartDate:** Είναι το Label που βρίσκεται μέσα στο text box DepDate και δείχνει την μορφή που θα έχει η ημερομηνία που θα βάλει ο χρήστης.
- **LblReturnDate:** Είναι το Label που βρίσκεται μέσα στο text box ReturnDate και δείχνει την μορφή που θα έχει η ημερομηνία που θα βάλει ο χρήστης.
- **LblDepart:** Είναι το Label που βρίσκεται πάνω από το text box DepDate και γράφει Depart.
- **LblReturn:** Είναι το Label που βρίσκεται πάνω από το text box ReturnDate και γράφει Return.
- **ButtonLogin:** Είναι το κουμπί που βρίσκεται στο πάνω μέρος της σελίδας και γράφει Login. Ο χρήστης το χρησιμοποιεί αν θέλει να συνδεθεί στο λογαριασμό του.
- **ButtonSignUp:** Είναι το κουμπί που βρίσκεται στο πάνω μέρος της σελίδας και γράφει Signup. Ο χρήστης το χρησιμοποιεί αν θέλει να δημιουργήσει νέο λογαριασμό.
- **Logo:** Είναι το Logo της εφαρμογής που βρίσκεται στο πάνω μέρος της σελίδας.
- **LblTravelQuestion:** Είναι το Label που εμφανίζεται κάτω από το Logo και γράφει την ερώτηση: ' Don't know where to go? '
- **LblBudgetQuestion:** Είναι το Label που εμφανίζεται κάτω από το Logo και γράφει την ερώτηση: ' Wanna be within a budget? '
- **LblBudget:** Είναι το Label που γράφει budget πάνω από το πλαίσιο για επιλογή του budget.
- **ButtonBudget:** Είναι το κουμπί που πάτησε ο χρήστης για να βρεθεί στην BudgetSearchPageGUI.
- **SearchButton:** Είναι το κουμπί που βρίσκεται στο κάτω μέρος της σελίδας και γράφει Let's Go.

Μέθοδοι της κλάσης:

- **SearchButtonClick():** Η μέθοδος αυτή ενεργοποιείται όταν πατηθεί το κουμπί Search Button και καλεί την μέθοδο SearchTravel του controller Travel Controller.
- **BudgetButtonClick():** Η μέθοδος αυτή ενεργοποιείται όταν πατηθεί το κουμπί BudgetButton και καλεί την μέθοδο Return Home του controller Travel Controller.

Boundary TravelDBProxy

«Boundary» TravelDBProxy
+SearchTravels(TravelObject: Travel): Boolean

Η κλάση αυτή επικοινωνεί με τη βάση δεδομένων και χρησιμοποιείται για να μπορέσουμε να βρούμε τα ταξίδια ανάλογα με της προτιμήσεις του χρήστη.



Μέθοδοι της κλάσης:

SearchTravels(TravelObject: Travel): Η μέθοδος αυτή παίρνει σαν όρισμα ένα Travel Object και αναζητάει ταξίδια στη βάση δεδομένων με αυτά τα χαρακτηριστικά. Επιστρέφει Boolean, True αν υπάρχουν ταξίδια με αυτά τα χαρακτηριστικά, και False αν δεν υπάρχουν ταξίδια με αυτά τα χαρακτηριστικά.

Entity Travel

«Entity» Travel
-MinBudget: Integer -MaxBudget: Integer -DepDate: String -ReturnDate: String -TravelID: Int
+Travel(MinBudget: Integer, MaxBudget: Integer, DepDate: String, ReturnDate: String) +getMinBudget(): Integer +getMaxBudget(): Integer +getDepDate(): String +getReturnDate(): String +getTravelID(): Integer +setMinBudget(MinBudgetAmount: Integer): void +setMaxBudget(MaxBudgetAmount: Integer): void +setDepDate(DepDate: String): void +setReturnDate(ReturnDate: String): void +setTravelID(TravelID: Integer): void

Πρόκειται για την οντότητα των ταξιδιών που επιλέγει ο χρήστης ανάλογα με το budget που θέλει να διαθέσει.

Χαρακτηριστικά κλάσης:

- MinBudget: Είναι ένας ακέραιος που εκφράζει το ελάχιστο ποσό που έχει δηλώσει ο χρήστης, για το ταξίδι του.
- Maxbudget: Είναι ένας ακέραιος που εκφράζει το μέγιστο ποσό που έχει δηλώσει ο χρήστης, για το ταξίδι του.
- DepDate: Ένα string όπου ο χρήστης έχει δηλώσει την ημερομηνία που θέλει να είναι η πτήση αναχώρησης.
- RetunDate: Ένα string όπου ο χρήστης έχει δηλώσει την ημερομηνία που θέλει να είναι η πτήση επιστροφής.
- TravelID: Ένας Integer που είναι ο αναγνωριστικός αριθμός για το ταξίδι.

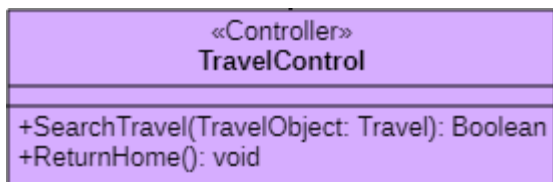
Μέθοδοι της κλάσης:

- Travel(MinBudget: Integer, MaxBudget: Integer, DepDate: String, ReturnDate: String): είναι ο constructor αντικειμένων της κλάσης. Παίρνει για ορίσματα τα χαρακτηριστικά της κλάσης και αρχικοποιεί τα αντικείμενα όταν καλείται.
- getMinBudget(): Getter του χαρακτηριστικού MinBudget, χρησιμοποιείται για να πάρει την τιμή του.
- getMaxBudget(): Getter του χαρακτηριστικού MaxBudget, χρησιμοποιείται για να πάρει την τιμή του.
- getDepDate(): Getter του χαρακτηριστικού DepDate, χρησιμοποιείται για να πάρει την τιμή του.



- `getReturnDate()`: Getter του χαρακτηριστικού `ReturnDate`, χρησιμοποιείται για να πάρει την τιμή του.
- `getTravelID()`: Getter του χαρακτηριστικού `TravelID`, χρησιμοποιείται για να πάρει την τιμή του.
- `setMinBudget(in MinBudgetAmount: Integer)`: Setter του χαρακτηριστικού `MinBudget`, χρησιμοποιείται για να αλλάξει την τιμή του.
- `setMaxBudget(in MaxBudgetAmount: Integer)`: Setter του χαρακτηριστικού `MaxBudget`, χρησιμοποιείται για να αλλάξει την τιμή του.
- `setDepDate(in DepDate: String)`: Setter του χαρακτηριστικού `DepDate`, χρησιμοποιείται για να αλλάξει την τιμή του.
- `setReturnDate(in ReturnDate: String)`: Setter του χαρακτηριστικού `ReturnDate`, χρησιμοποιείται για να αλλάξει την τιμή του.
- `setTravelID(in TravelID: Integer)`: Setter του χαρακτηριστικού `TravelID`, χρησιμοποιείται για να αλλάξει την τιμή του.

Controller `TravelControl`



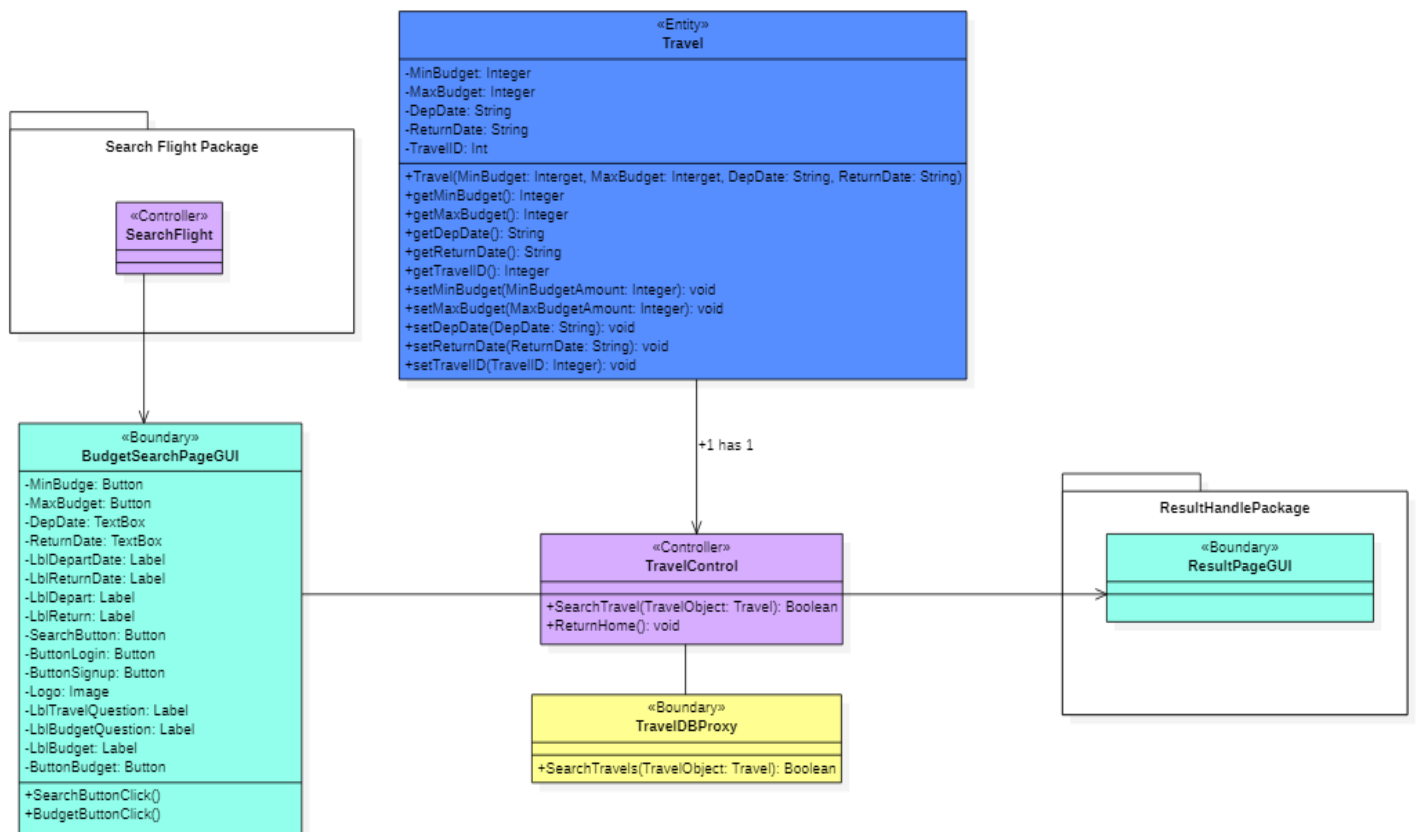
Ο ελεγκτής αυτός είναι υπεύθυνος για την αναζήτηση των ταξιδιών με βάση το budget του επιθυμεί ο χρήστης.

Μέθοδοι της κλάσης:

`SearchTravel(TravelObject: Travel)`: Η μέθοδος αυτή ενεργοποιείται όταν πατηθεί το κουμπί `Search Button`. Συνδέεται και με την βάση δεδομένων για να αναζητήσει ταξίδια με αυτά τα χαρακτηριστικά. . Επιστρέφει `Boolean True` αν υπάρχουν ταξίδια με αυτά τα χαρακτηριστικά, και `False` αν δεν υπάρχουν ταξίδια με αυτά τα χαρακτηριστικά.

`ReturnHome()`: Η μέθοδος αυτή ενεργοποιείται όταν πατηθεί το `ButtonBudget` και επιστρέφει στο `HomePage`.

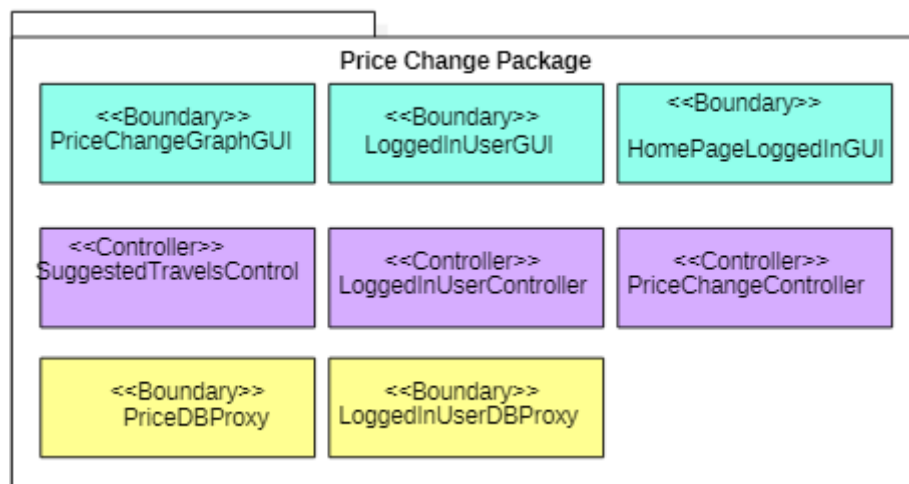
ΔΙΑΓΡΑΜΜΑ ΚΛΑΣΕΩΝ ΠΑΚΕΤΟΥ ProposeTravel



Σχήμα 36. Διάγραμμα Κλάσεων ProposeTravel Package

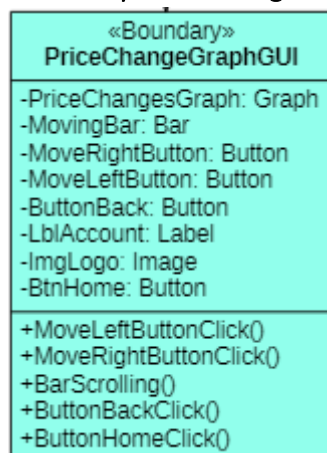
4.3.2 Πακέτο PriceChange

Το πακέτο αυτό περιλαμβάνει τις απαραίτητες κλάσεις για την δημιουργία του διαγράμματος για την εξέλιξη των τιμών των πτήσεων, όπως επίσης και για τον έλεγχο της σύνδεσης του χρήστη στο σύστημα που είναι απαραίτητος για να μπορεί ένας χρήστης να δει το διάγραμμα. Επιπλέον, περιλαμβάνει τις κλάσεις για να μπορεί το σύστημα να προτείνει ταξίδια στο χρήστη. Παρακάτω φαίνεται το πακέτο κλάσεων.



Σχήμα 37. Διάγραμμα Κλάσεων PriceChange Package

Boundary PriceChangeGraphGUI



Πρόκειται για την διεπαφή της σελίδας όπου ο χρήστης μπορεί δει το διάγραμμα με την εξέλιξη της τιμής των πτήσεων.

Χαρακτηριστικά κλάσης:

- PriceChangesGraph: Είναι το γράφημα που εξεικονίζει την εξέλιξη της τιμής για την πτήση.



- MovingBar: Είναι η μπάρα που ο χρήστης κάνει scrolling για να μετακινηθεί δεξιά και αριστερά στο γράφημα.
- MoveRightButton: Είναι ένα κουμπί με σχήμα βελάκι που μετακινεί την εικόνα προς τα δεξιά.
- MoveLeftButton: Είναι ένα κουμπί με σχήμα βελάκι που μετακινεί την εικόνα προς τα αριστερά.
- ButtonBack: Ένα κουμπί στο πάνω μέρος της σελίδας που επιστρέφει στην προηγούμενη σελίδα.
- LblAccount: Είναι το Label στο πάνω μέρος της σελίδας που δείχνει το User name του χρήστη.
- ImgLogo: Είναι το Logo της εφαρμογής.
- BtnHome: Κουμπί που σε μεταφέρει στο HomePage.

Μέθοδοι κλάσης:

- MoveLeftButtonClick(): Η μέθοδος αυτή ενεργοποιείται όταν πατηθεί το κουμπί για να πάει αριστερά το διάγραμμα.
- MoveRightButtonClick(): Η μέθοδος αυτή ενεργοποιείται όταν πατηθεί το κουμπί για να πάει δεξιά το διάγραμμα.
- BarScrolling(): Η μέθοδος αυτή ενεργοποιείται όταν ο χρήστης κάνει scroll την μπάρα.
- ButtonBackClick(): Η μέθοδος αυτή ενεργοποιείται όταν πατηθεί το κουμπί BackButton. Συνδέεται με τη μέθοδο GoBack του controller.
- ButtonHomeClick(): Η μέθοδος αυτή ενεργοποιείται όταν πατηθεί το κουμπί HomeButton. Συνδέεται με τη μέθοδο ReturnHome του controller.

Boundary LoggedInUserGUI

«Boundary» LoggedInUserGUI
-LoginButton: Button -SignUpButton: Button -GoBackButton: Button
+LoginButtonClick() +SignUpButtonClick() +GoBackButtonClick()

Πρόκειται για την διεπαφή της σελίδας όπου ο χρήστης μπορεί να κάνει Login ή Sign up στο σύστημα.

Χαρακτηριστικά κλάσης:

- LoginButton: κουμπί που πατάει ο χρήστης για να κάνει Login.
- SignUpButton: κουμπί που πατάει ο χρήστης για να κάνει Signup.
- GoBackButton: κουμπί που πατάει ο χρήστης για να πίσω.

Μέθοδοι κλάσης:

- LoginButtonClick(): Μέθοδος που ενεργοποιείται όταν πατηθεί το κουμπί στη Login και συνδέεται με την αντίστοιχη μέθοδο στο <<Controller>>LoggedInUserController.



- SignupButtonClick(): Μέθοδος που ενεργοποιείται όταν πατηθεί το κουμπί στη SignUp και συνδέεται με την αντίστοιχη μέθοδο στο <<Controller>>LoggedInUserController.
- GoBackButtonClick(): Μέθοδος που ενεργοποιείται όταν πατηθεί το κουμπί στη GoBack και συνδέεται με την αντίστοιχη μέθοδο στο <<Controller>>LoggedInUserController.

Controller Price Change Controller

«Controller» PriceChangeController
-PricesArray: Float[0..*] -DateArray[]: Strings[0..*]
+PriceChangeController(PriceArray: float[1..*], DateArray[1..*]) +CreateCraph(PricesArray: Float[0..*], DateArray: String[0..*]) +setPricesArray(PricesArray: Float[0..*]) +setDateArray(Dates: String[0..*]) +GoBack(): void +ReturnHome(): void

Ο ελεγκτής αυτός είναι υπεύθυνος για να πάρει τα δεδομένα των τιμών και ημερομηνιών από την DB και να δημιουργήσει το διάγραμμα εξέλιξης τιμής.

Χαρακτηριστικά κλάσης:

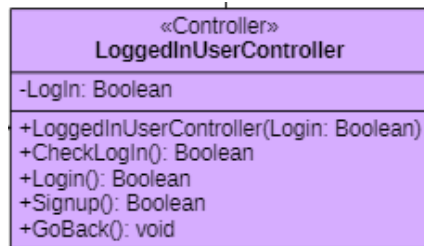
- PricesArray: Είναι ένας πίνακας από Floats, είναι η παλιές τιμές του ταξιδιού (άξονας y στο γράφημα).
- DateArray[]: Είναι ένας πίνακας από strings, έχει το ίδιο μέγεθος με τον PriceArray και σε κάθε θέση έχει την ημερομηνία για την αντίστοιχη τιμή που βρίσκεται στο πίνακα PriceArray (άξονας x στο γράφημα).

Μέθοδοι της κλάσης:

- CreateCraph(PricesArray:Float[0..*], DateArray:String[0..*]): μέθοδος που παίρνει σαν ορίσματα τις τιμές και τις ημερομηνίες και φτιάχνει το γράφημα.
- setPricesArray(PricesArray:Float[0..*]): μέθοδος setter που βάζει τιμές στο πίνακα PriceArray.
- setDateArray(in Dates:String[0..*]): μέθοδος setter που βάζει τιμές στο πίνακα DateArray.
- GoBack(): Μέθοδος που μας επιστρέφει στη προηγούμενη σελίδα.
- ReturnHome(): μέθοδος που μας επιστρέφει στο HomePage.



Controller LoggedInUserController



Ο ελεγκτής αυτός είναι υπεύθυνος για να ελέγχει αν ο χρήστης είναι συνδεδεμένος ή όχι στο σύστημα.

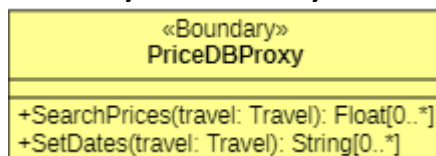
Χαρακτηριστικά κλάσης:

- Login: True αν ο χρήστης είναι login, false αν δεν είναι.

Μέθοδοι της κλάσης:

- CheckLogin(): Μέθοδος που ελέγχει αν ο user είναι login ή όχι. Επιστρέφει Boolean.
- Login(): Μέθοδος που ενεργοποιείται αν ο χρήστης επιλέξει να κάνει Login. Επιστρέφει true αν γίνει με επιτυχία.
- Signup(): Μέθοδος που ενεργοποιείται αν ο χρήστης επιλέξει να κάνει SignUp. Επιστρέφει true αν γίνει με επιτυχία.
- GoBack(): Μέθοδος που μας επιστρέφει στη προηγούμενη σελίδα.

Boundary PriceDBProxy

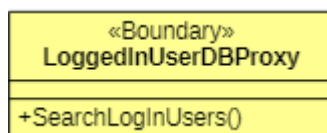


Η κλάση αυτή επικοινωνεί με τη βάση δεδομένων και χρησιμοποιείται για να μπορέσουμε να βρούμε τις τιμές και τις αντίστοιχες ημερομηνίες των πτήσεων.

Μέθοδοι κλάσης:

- SearchPrices(travel: Travel): μέθοδος που αναζητεί στη βάση δεδομένων της τιμές των παλιών πτήσεων.
- SetDates(travel: Travel): μέθοδος που αναζητεί στη βάση δεδομένων της ημερομηνίες των παλιών πτήσεων.

Boundary LoggedInUserDBProxy



Η κλάση αυτή επικοινωνεί με τη βάση δεδομένων και χρησιμοποιείται για να μπορέσουμε να βρούμε τους εγγεγραμμένους χρήστες στο σύστημα.

**Μέθοδοι κλάσης:**

- SearchLoginUsers(): Μέθοδος που αναζητεί στη βάση δεδομένων τους χρήστες που είναι Login.

Boundary LoggedInPageGUI

«Boundary» HomeLoggedInPageGUI
-LblAccount: String -BtnTravel1: Button -LblTravel1: Label -BtnTravel2: Button -LblTravel2: Label -BtnTravel3: Button -LblTravel3: Label -ImageTravel1: Image -ImageTravel2: Image -ImageTravel3: Image
+BtnTravel1Click() +BtnTravel2Click() +BtnTravel3Click()

Πρόκειται για την διεπαφή της σελίδας όπου ο εγγεγραμμένος χρήστης μπορεί να κάνει αναζήτηση μιας πτήσης ή να επιλέξει ένα ταξίδι που του προτείνεται. Η κλάση αυτή κληρονομεί την απλή Home Page και επιπλέον προστίθενται η επιλογή του προτεινόμενου ταξιδιού.

Χαρακτηριστικά κλάσης:

- LblAccount: Το label που βρίσκεται στο άνω μέρος της σελίδας και αναγραφεί το username του χρήστη.
- BtnTravel1: Το κουμπί που επιλέγει ο χρήστης το 1^ο από τα 3 ταξίδια που προτείνονται.
- LblTravel1: Το Label με την περιγραφή του 1^{ου} από τα 3 ταξίδια που προτείνονται.
- ImageTravel1: Η φωτογραφία πάνω στο κουμπί του 1^{ου} από 3 ταξίδια που προτείνονται.
- BtnTravel2: Το κουμπί που επιλέγει ο χρήστης το 2^ο από τα 3 ταξίδια που προτείνονται.
- LblTravel2: Το Label με την περιγραφή του 2^{ου} από τα 3 ταξίδια που προτείνονται.
- ImageTravel2: Η φωτογραφία πάνω στο κουμπί του 2^{ου} από 3 ταξίδια που προτείνονται.
- BtnTravel3: Το κουμπί που επιλέγει ο χρήστης το 3^ο από τα 3 ταξίδια που προτείνονται.
- LblTravel3: Το Label με την περιγραφή του 3^{ου} από τα 3 ταξίδια που προτείνονται.
- ImageTravel3: Η φωτογραφία πάνω στο κουμπί του 3^{ου} από 3 ταξίδια που προτείνονται.

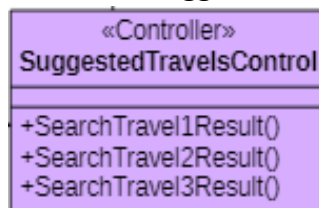
Μέθοδοι κλάσης:

- BtnTravel1Click(): Η μέθοδος αυτή ενεργοποιείται όταν ο χρήστης επιλέγει το κουμπί BtnTravel1, δηλαδή το 1^ο προτεινόμενο ταξίδι. Η μέθοδος αυτή συνδέεται με την αντίστοιχη μέθοδο στη κλάση SuggestedTravelController.



- BtnTravel2Click(): Η μέθοδος αυτή ενεργοποιείται όταν ο χρήστης επιλέγει το κουμπί BtnTravel2, δηλαδή το 2^ο προτεινόμενο ταξίδι. Η μέθοδος αυτή συνδέεται με την αντίστοιχη μέθοδο στη κλάση SuggestedTravelController.
- BtnTravel2Click(): Η μέθοδος αυτή ενεργοποιείται όταν ο χρήστης επιλέγει το κουμπί BtnTravel2, δηλαδή το 2^ο προτεινόμενο ταξίδι. Η μέθοδος αυτή συνδέεται με την αντίστοιχη μέθοδο στη κλάση SuggestedTravelController.

Controller SuggestedTravelsControl

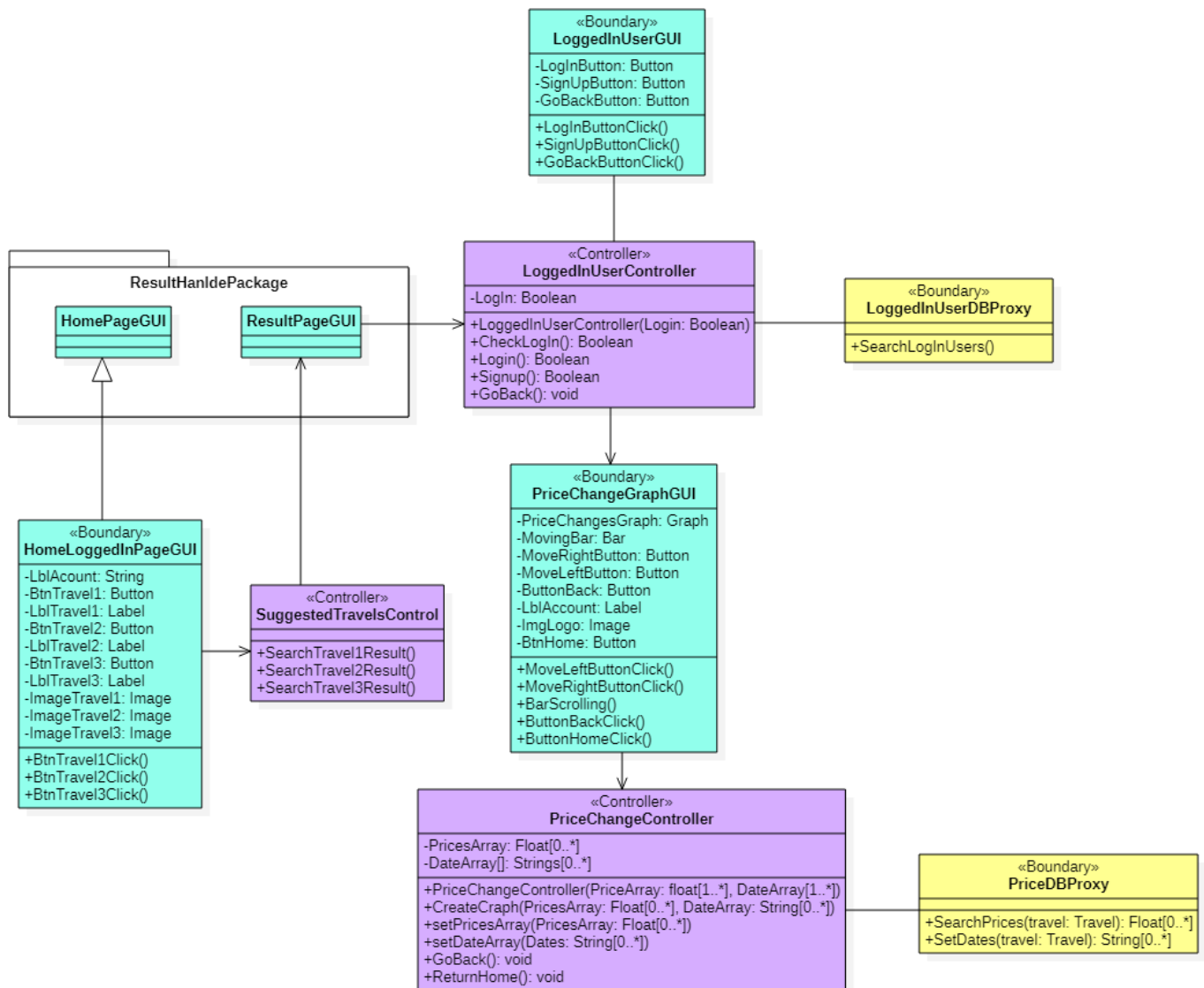


Ο ελεγκτής αυτός είναι υπεύθυνος για να ελέγχει το πάτημα ενός από τα τρία κουμπιά που συνδέονται με τα τρία προτεινόμενα ταξίδια στον εγγεγραμμένο χρήστη από το σύστημα.

Μέθοδοι κλάσης:

- SearchTravel1Results(): Η μέθοδος αυτή ενεργοποιείται όταν ο χρήστης επιλέγει το 1^ο προτεινόμενο ταξίδι.
- SearchTravel1Results(): Η μέθοδος αυτή ενεργοποιείται όταν ο χρήστης επιλέγει το 2^ο προτεινόμενο ταξίδι.
- SearchTravel1Results(): Η μέθοδος αυτή ενεργοποιείται όταν ο χρήστης επιλέγει το 3^ο προτεινόμενο ταξίδι.

ΔΙΑΓΡΑΜΜΑ ΚΛΑΣΕΩΝ ΠΑΚΕΤΟΥ PriceChange



Σχήμα 38. Διάγραμμα Κλάσεων PriceChange Package



ΠΑΡΑΡΤΗΜΑ Ι - ΓΛΩΣΣΑΡΙΟ

ΛΑ : Λειτουργική Απαίτηση

ΜΛΑ : Μη Λειτουργική Απαίτηση

ΟΑ : Ομάδα Ανάπτυξης

ΣΧ : Σενάριο Χρήσης

Del : Deliverable



ΠΑΡΑΡΤΗΜΑ II – ΑΝΟΙΧΤΑ ΘΕΜΑΤΑ

- Δεν υλοποιούνται οι σελίδες σύνδεσης και εγγραφής του χρήστη καθώς ζητήθηκε να μην περιλαμβάνονται στις λειτουργικές απαιτήσεις.