

1. Για την υλοποίηση της εφαρμογής χρησιμοποιήθηκε το πλαίσιο JavaFx και το IDE IntelliJ IDEA. Το γραφικό περιβάλλον της εφαρμογής σχεδιάστηκε με τη βοήθεια του Scene Builder.
2. Κατασκεύασα την κλάση Flight η οποία αποτελεί το μοντέλο όλων των πτήσεων και τα βασικά πεδία που περιέχει είναι το id της πτήσης, η πόλη αναχώρησης, ο τύπος πτήσης, ο τύπος του αεροπλάνου κι ο προβλεπόμενος χρόνος αναχώρησης. Στη συνέχεια της ανάπτυξης του κώδικα, δημιούργησα επίπροσθετα βοηθητικά πεδία.
3. Κατασκεύασα την κλάση Parking η οποία αποτελεί το μοντέλο για κάθε είδος χώρου στάθμευσης. Τα βασικά πεδία που περιέχει είναι ο τύπος του χώρου στάθμευσης, το πλήθος των θέσεων του συγκεκριμένου χώρου, το κόστος στάθμευσης στις θέσεις της συγκεκριμένης κατηγορίας και το πρόθεμα για τα μοναδικά αναγνωριστικά των θέσεων του συγκεκριμένου χώρου.
4. Όταν από το μενού της εφαρμογής επιλέγεται το start ξεκινά η εφαρμογή διαβάζοντας τα κατάλληλα αρχεία και αρχικοποιεί δυο βασικές δομές. Πρώτον, αρχικοποιεί τον πίνακα αντικειμένων τύπου Parking (our\_park), ο οποίος περιέχει τις πληροφορίες για όλους τους χώρους στάθμευσης. Κατόπιν, διαβάζει από το αρχείο των πτήσεων και δημιουργεί τα αντίστοιχα αντικείμενα τύπου Flight, τα οποία τα προσθέτει στην ArrayList our\_flights. Σε αυτήν τη δομή εντάσσει και όποια άλλη πτήση εισαχθεί από το χρήστη της εφαρμογής.
5. Στο κάτω μέρος της οθόνης εμφανίζω διάφορα μηνύματα σχετικά με τις πτήσεις και συγκεκριμένα με τις μεταβάσεις της κατάστασης του. Για παράδειγμα, όταν η πτήση γίνεται PARKED εμφανίζω μήνυμα ότι τώρα η συγκεκριμένη βρίσκεται σε αυτήν την κατάσταση και τότε προβλέπεται ότι θα αναχωρήσει. Ένα ακόμη παράδειγμα είναι ότι εμφανίζω μήνυμα για το πότε φεύγει κάθε πτήση κι επισημαίνω μαζί αν έφυγε στην ώρα της, νωρίτερα ή καθυστερημένη, καθώς και το κόστος της κι αν έλαβε κάποια έκπτωση. Για την υλοποίηση της συγκεκριμένης λειτουργίας ανακατεύθυνα την έξοδο της κονσόλας σε ένα TextArea στοιχείο που δημιούργησα στο scene buider.
6. Ο τρόπος που έχω θέσει την εφαρμογή να αποφασίζει πότε μια πτήση θα αναχωρεί είναι ο εξής: όταν η πτήση γίνεται parked προσθέτω στη συγκεκριμένη χρονική στιγμή έναν τυχαίο αριθμό (θετικός ή αρνητικός), ο οποίος δεν ξεπερνά κατά απόλυτη τιμή τον προβλεπόμενο χρόνο αναχώρησης. Έτσι, η εφαρμογή αποφασίζει τυχαία πότε μια πτήση θα αναχωρήσει κι έτσι υλοποιούνται στην εφαρμογή και τα 3 ενδεχόμενα, δηλαδή να αναχωρήσει μια πτήση έγκαιρα, νωρίτερα ή με καθυστέρηση.
7. Παρόλο που εγώ γνωρίζω εκ των προτέρων πότε μια πτήση θα αναχωρήσει στην πραγματικότητα αυτό δεν το λαμβάνω υπόψιν μου σε σημεία όπως για παράδειγμα στο κουμπί του μενού Delayed. Δηλαδή, εκεί εμφανίζω τις πτήσεις που είχαν σχεδιασμένη αναχώρηση νωρίτερα από τον τρέχοντα χρόνο. Άλλο ένα παράδειγμα, στο κουμπί του μενού Next Departures, αν μια πτήση έχει σχεδιασμένη αναχώρηση στο χρόνο 00:17, ο χρόνος την στιγμή που πατιέται το κουμπί είναι 00:12 κι ο χρόνος που πραγματικά θα αναχωρήσει είναι 00:35, τότε η πτήση θα εμφανιστεί στον πίνακα των next departures.
8. Στον πίνακα που εμφανίζεται πατώντας το κουμπί Delayed, στην τελευταία στήλη εμφανίζεται ο χρόνος τελικής αναχώρησης (δηλαδή συμπεριλαμβανομένου της καθυστέρησης).
9. Στο πάνω μέρος της οθόνης που εμφανίζονται οι συγκεκριμένες πληροφορίες στην ετικέτα Arrived Flights (ο αριθμός πτήσεων σε κατάσταση άφιξης) εμφανίζω τον αριθμό όλων των πτήσεων που έχουν κάνει αίτημα να εισέλθουν στο αεροδρόμιο, δηλαδή κι αυτές που είναι HOLDING, LANDING, PARKED.
10. Στην κλάση Controller.java κάθε public μέθοδος που περιέχει είναι τεκμηριωμένη σύμφωνα με τις προδιαγραφές του εργαλείου javadoc Error! Reference source not found..

11. Όταν ο χρήστης πατήσει το κουμπί Load, ενώ η εφαρμογή ήδη τρέχει, τότε αν υπάρχει το αρχείο που θα επιλέξει, η εφαρμογή ξεκινά από την αρχή με βάση τα αρχεία που επέλεξε ο χρήστης. Αν επιλέξει αρχείο που δεν υπάρχει τότε η εφαρμογή συνεχίζει να τρέχει κανονικά.
12. Όταν ο χρήστης πατήσει το κουμπί Start, ενώ η εφαρμογή ήδη τρέχει, τότε η εφαρμογή ξεκινά από την αρχή.
13. Τα αρχεία που μπορεί να κάνει load η εφαρμογή θα πρέπει να βρίσκονται μέσα στον φάκελο medialab (όπου βρίσκονται και τα αρχεία airport\_SCENARIO-ID.txt και setup\_SCENARIO-ID.txt) . Επίσης, θα πρέπει να υπάρχουν και τα 2 αρχεία, να είναι στη μορφή αυτή και το μόνο που να αλλάζει να είναι το SCENARIO-ID.
14. Η κλάση Gates αποτελεί ένα μοντέλο για κάθε μεμονωμένη θέση στάθμευσης που υπάρχει και συνδέει μια μεμονωμένη θέση με την πτήση που είναι παρκαρισμένη σε αυτήν.