



ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

“PRIMON: Hotel Price Monitoring Android App”



Φοιτητής
Καραγιαννόπουλος Πάρις
Αρ. Μητρώου: it144250

Επιβλέπων
Σαλαμπάσης Μιχάλης
Βαθμίδα Καθηγητής

Ημερομηνία 31/01/2024

Τίτλος Δ.Ε. Android App for Hotel Price Monitoring

Κωδικός Δ.Ε. [23280](#)

Όνοματεπώνυμο φοιτητή Πάρις Καραγιαννόπουλος

Όνοματεπώνυμο εισηγητή Σαλαμπάσης Μιχάλης

Ημερομηνία ανάληψης Π.Ε. 19/10/2023

Ημερομηνία περάτωσης Π.Ε. 31/01/2024

Βεβαιώνω ότι είμαι ο συγγραφέας αυτής της εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, έχω καταγράψει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών, εικόνων και κειμένου, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επιπλέον, βεβαιώνω ότι αυτή η εργασία προετοιμάστηκε από εμένα προσωπικά, ειδικά ως διπλωματική εργασία, στο Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του ΔΙ.ΠΑ.Ε.

Η παρούσα εργασία αποτελεί πνευματική ιδιοκτησία του φοιτητή Καραγιαννόπουλου Πάρι που την εκπόνησε/αν. Στο πλαίσιο της πολιτικής ανοικτής πρόσβασης, ο συγγραφέας/δημιουργός εκχωρεί στο Διεθνές Πανεπιστήμιο της Ελλάδος άδεια χρήσης του δικαιώματος αναπαραγωγής, δανεισμού, παρουσίασης στο κοινό και ψηφιακής διάχυσης της εργασίας διεθνώς, σε ηλεκτρονική μορφή και σε οποιοδήποτε μέσο, για διδακτικούς και ερευνητικούς σκοπούς, άνευ ανταλλάγματος. Η ανοικτή πρόσβαση στο πλήρες κείμενο της εργασίας, δεν σημαίνει καθ' οιονδήποτε τρόπο παραχώρηση δικαιωμάτων διανοητικής ιδιοκτησίας του συγγραφέα/δημιουργού, ούτε επιτρέπει την αναπαραγωγή, αναδημοσίευση, αντιγραφή, πώληση, εμπορική χρήση, διανομή, έκδοση, μεταφόρτωση (downloading), ανάρτηση (uploading), μετάφραση, τροποποίηση με οποιονδήποτε τρόπο, τμηματικά ή περιληπτικά της εργασίας, χωρίς τη ρητή προηγούμενη έγγραφη συναίνεση του συγγραφέα/δημιουργού.

Η έγκριση της διπλωματικής εργασίας από το Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του Διεθνούς Πανεπιστημίου της Ελλάδος, δεν υποδηλώνει απαραίτητως και αποδοχή των απόψεων του συγγραφέα, εκ μέρους του Τμήματος.

Πρόλογος

Η περάτωση αυτής της πτυχιακής εργασίας αποτελεί το τελευταίο έργο και σταθμό των σπουδών μου στο ΔΠΠΑΕ Θεσσαλονίκης. Η επιλογή της εν λόγω πτυχιακής έγινε γύρω από τρεις πρακτικούς πυλώνες. Για αρχή, οι Android εφαρμογές έχουν την κυρίαρχη θέση στον κόσμο των Mobile applications. Επίσης, η υλοποίηση αυτής της εφαρμογής έγινε με τις τελευταίες διαθέσιμες τεχνολογίες που προσφέρει η πλατφόρμα ανάπτυξης android εφαρμογών, ώστε μετά την ολοκλήρωση της, να έχω αποκομίσει μία γνώση που να είναι εφαρμόσιμη και χρήσιμη στη σημερινή επαγγελματική πραγματικότητα. Στόχος ήταν η υλοποίηση της πτυχιακής να είναι όσο το δυνατό πιο κοντά σε ένα πραγματικό εργασιακό περιβάλλον. Για αυτόν το λόγο, πέρα από το περιβάλλον ανάπτυξης android studio χρησιμοποιήθηκαν και επιπρόσθετες τεχνολογίες που χρησιμοποιούνται σε πραγματικές συνθήκες εργασίες στον εν λόγω τομέα όπως είναι το εργαλείο σχεδίασης Figma, το εργαλείο οργάνωσης Trello και το σύστημα υλοποίησης Agile. Τέλος, το περιεχόμενο της εφαρμογής αποτελεί τον τρίτο πρακτικό πυλώνα της. Τις τελευταίες δεκαετίες, και ιδιαίτερα μετά την περίοδο της καραντίνας, η αγορά του τουρισμού και των ταξιδιών εντός και εκτός της χώρας έχει αυξηθεί σημαντικά. Κατ' επέκταση, αυτό έχει ως αποτέλεσμα πλέον η παρακολούθηση των μεταβαλλόμενων τιμών των διαθέσιμων δωματίων να είναι εξαιρετικά σημαντική για τους ίδιους τους επιχειρηματίες του τουρισμού. Από την στιγμή ανάληψης της εν λόγω εργασίας, αρκετές ιστοσελίδες έχουν δημιουργηθεί και η οργάνωση αυτών αποτελεί γεγονός που κάνει πιο ισχυρή την ανάγκη μιας εφαρμογής για την κάλυψη αυτών των αναγκών. Αυτή την ανάγκη έρχεται να καλύψει η συγκεκριμένη πτυχιακή με την ανάπτυξη μιας Android App για την παρακολούθηση των τιμών των ξενοδοχείων, την ανάδειξη στατιστικών, καθώς και τη λήψη προσποιημένων ειδοποιήσεων για τους χρήστες.

Περίληψη

Στην παρακάτω πτυχιακή εργασία παρουσιάζεται η υλοποίηση μιας Android εφαρμογής, με θέμα την παρακολούθηση των τιμών ξενοδοχείων/δωματίων, μέσα από αντίστοιχες ιστοσελίδες, τη παρουσίαση γενικότερων στατιστικών αυτού του τομέα καθώς και τη δημιουργία και αποστολή προσωποποιημένων ειδοποιήσεων. Αρχικά γίνεται μία αναφορά και μικρή έρευνα γύρω από τη βιομηχανία του τουρισμού. Μετά, το ενδιαφέρον δίνεται στις έννοιες των Business Intelligence, Hotel Analytics και το Hotel Pricing. Παρουσιάζονται στατιστικά στοιχεία από δημόσια δεδομένα (open data) ώστε να υπάρχει ένα πλαίσιο αναφοράς για τις λειτουργίες και τη χρησιμότητα της εφαρμογής που αναπτύχθηκε στον πραγματικό κόσμο ενώ παράλληλα τονίζεται και η χρηστικότητα την ανάλυσης των ξενοδοχειακών δεδομένων και συγκεκριμένα η παρακολούθηση των δεδομένων των τιμών. Σε τεχνολογικό επίπεδο η παρούσα εργασία αφορά μία “native Android” εφαρμογή, για την οποία χρησιμοποιήθηκε ως γλώσσα προγραμματισμού η Kotlin σε συνδυασμό με την νέα UI βιβλιοθήκη που προσφέρει η Google, την Jetpack Compose. Σε αυτό το σύνολο σύγχρονων τεχνολογιών προστέθηκε και η Retrofit, μια βιβλιοθήκη ευρείας χρήσεως, για να αποκτήσει η εφαρμογή σύνδεση με τον server. Το scrapping των δεδομένων και το backend της εν λόγω android υλοποίησης έχει δημιουργηθεί από άλλες πτυχιακές που εκπονούνται στο τμήμα. Για την αρχιτεκτονική της εφαρμογής χρησιμοποιήθηκε το μοντέλο MVVM (Model-View-ViewModel) σε συνδυασμό με τα μοτίβα μιας καθαρής αρχιτεκτονικής (clean architecture). Η MVVM αρχιτεκτονική δίνει την δυνατότητα διαχωρισμού του UI από τη λογική. Εφαρμόζοντας και τις αρχές της καθαρής Αρχιτεκτονικής, ο κώδικας της εφαρμογής γίνεται πιο διαχειρίσιμος και διατηρήσιμος, ενώ η αξιοποίηση της βιβλιοθήκης Hilt διευκόλυνε τη διαμόρφωση του κώδικα. Πέρα από το πρακτικό αποτέλεσμα της εφαρμογής με την ολοκλήρωση αυτής της εργασίας δημιουργείται και μια βάση κατά την οποία, με την μελλοντική εξέλιξη του backend, μπορεί να εξελιχθεί μελλοντικά και η ίδια η εφαρμογή με τον εμπλουτισμό επιπρόσθετων λειτουργιών.

Abstract

The following thesis presents the implementation of an Android application focusing on monitoring hotel/room prices through corresponding websites and, and creating personalized notifications according to user preferences. Initially, a reference and brief research conducted around the sites and applications hosting hotels and rooms. Subsequently, the focus shifts to Android applications of this kind. Statistical data from public sources are presented to provide knowledge and a frame of reference for the functions and utility of the application in the real world. Moreover, the importance of hotel analytics, especially hotel price monitoring, in this reference context is evident. On a practical level, this work is an Android application in which Kotlin was used as the programming language in conjunction with Google's new UI library, Jetpack Compose. Retrofit, a widely used library, was added to this set of modern technologies set to enable the application's connection with the server. It should be stated here that fellow IHU students developed the data scrapping and the backend used in the app as part of their thesis. The application's architecture employed the MVVM (Model-View-ViewModel) model in combination with the patterns of a clean architecture. The MVVM architecture facilitates the separation of UI from logic. Applying the principles of Clean Architecture makes the code more manageable and maintainable, while using the Hilt library facilitates the code configuration. Beyond the practical outcome of the application, completing this work also creates a foundation on which, with the future development of the backend, the application itself can evolve by enriching additional functions.

Ευχαριστίες

Για την ολοκλήρωση αυτής της πτυχιακής θα ήθελα να ευχαριστήσω, πρωτίστως τους γονείς μου Χρήστο και Μαριάννα και τον αδελφό μου Αλέξανδρο για την υπομονή και την υποστήριξη 27 χρόνων, την Χριστίνα για όλη την υπομονή και στήριξη της τα τελευταία 5 χρόνια. Στη συνέχεια θα ήθελα να ευχαριστήσω τον φίλο μου Ανδρέα για την πολύτιμη βοήθεια του. Τέλος θα ήθελα να ευχαριστήσω θερμά τον επιβλέπων καθηγητή της πτυχιακής εργασίας, Μιχάλη Σαλαμπάση για την άμεση ανταπόκριση, την εξαιρετική συνεργασία και την καθοδήγηση του δημιουργώντας με αυτόν τον τρόπο έναν ιδανικό επίλογο.

Περιεχόμενα

Περίληψη.....	iv
Ευχαριστίες	vi
Περιεχόμενα	vii
Κατάλογος Σχημάτων	ix
Κατάλογος Πινάκων.....	x
Συνομογραφίες.....	xi
Κεφάλαιο 1ο: Εισαγωγή.....	12
1.1 Βιομηχανία του Hotel Booking	12
1.2 Business Intelligence.....	15
1.3 Hotel Analytics.....	18
1.4 Hotel Price Monitoring.....	20
1.5 Hotel Pricing.....	21
1.6 Κίνητρο Πτυχιακής Εργασίας	23
Κεφάλαιο 2ο: Τεχνολογίες	25
2.1 Android.....	25
2.1.1 Ιστορικό Android-based εφαρμογών.....	25
2.1.2 Περιβάλλον Android Studio	28
2.1.3 Kotlin: Γλώσσα Προγραμματισμού	30
2.1.4 Jetpack Compose: UI Library.....	31
2.2 Trello: Οργάνωση Διεργασιών	31
2.3 Agile πλάνο της υλοποίησης της εφαρμογής	33
2.4 GitHub: Οργάνωση κώδικα.....	34
2.5 Figma: Εργαλείο Σχεδίασης και αλληλεπίδρασης	35
Κεφάλαιο 3ο: PRIMON- Hotel Price Monitoring Android App.....	38
3.1 Λειτουργία.....	38
3.2 Σύνδεση με Backend	38
3.3 Ανάπτυξη Front-End	42
3.3.1 Splash Screen	43
3.3.2 Σελίδα Login	44
3.3.3 Σελίδα Sign Up.....	46
3.3.4 Σελίδα επιβεβαίωσης Email	47

3.3.5	Σελίδα Dashboard.....	47
3.3.6	Σελίδα Προφίλ.....	53
Κεφάλαιο 4ο:	Συμπεράσματα και προτάσεις βελτίωσης.....	58
Κεφάλαιο 5ο:	Βιβλιογραφία.....	59

Κατάλογος Σχημάτων

Σχήμα 1.1: Ποσοστό δισεκατομμυρίων που συνέβαλε η ταξιδιωτική και τουριστική βιομηχανία στον ΑΕΠ της Ευρώπης μεταξύ 2019 και 2022	13
Σχήμα 1.2: Το συνολικό ποσοστό των πωλήσεων/κρατήσεων που πραγματοποιήθηκαν online	15
Σχήμα 1.3 Δομή εργασίας των σύγχρονων δομών ανάλυσης	15
Σχήμα 2.1: Ποσοστά της αγοράς Λειτουργικών Συστημάτων κινητών συσκευών το 2023	25
Σχήμα 2.2: Περιβάλλον ανάπτυξης Android Studio	30
Σχήμα 2.3: Στιγμιότυπο και τμήμα των πινάκων εργασιών για τη Hotel Price Monitoring android εφαρμογής	33
Σχήμα 2.4: Γράφημα Figma της Hotel Price Monitoring Web εφαρμογής.....	36
Σχήμα 2.5: Γράφημα Figma της Hotel Price Monitoring android εφαρμογής.....	37
Σχήμα 3.1: Κλάση ApiModule.....	39
Σχήμα 3.2: Στιγμιότυπο της κλάσης MonitorApiService με τα backend endpoints	40
Σχήμα 3.3: Monitor List Backend Endpoint.....	41
Σχήμα 3.4: Hotel Info Backend Endpoint	42
Σχήμα 3.5: Αρχιτεκτονική Εφαρμογής	43
Σχήμα 3.6: Splash Screen.....	44
Σχήμα 3.7: Login Page	45
Σχήμα 3.8: Wrong Credentials	46
Σχήμα 3.9: Sign Up & OTP page	47
Σχήμα 3.10: Dashboard Page	48
Σχήμα 3.11	49
Σχήμα 3.12: Delete list gesture action.....	49
Σχήμα 3.13: Λίστα δωματίων.....	50
Σχήμα 3.14: Σχεδιάγραμμα μέσης τιμής δωματίου ανά μήνα.....	50
Σχήμα 3.15: Κώδικας μετατροπής της απάντησης του Server σε γράφημα.....	51
Σχήμα 3.16: Search Page.....	51
Σχήμα 3.17: Εκχώρηση δωματίου σε λίστα παρακολούθησης	52
Σχήμα 3.18: Share map component.....	53
Σχήμα 3.19: Profile page	54
Σχήμα 3.20: Push Notification	55
Σχήμα 3.21: About option at Profile Page.....	55
Σχήμα 3.22: Share option at profile page & Share Message	56
Σχήμα 3.23: SessionManager Class	57

Κατάλογος Πινάκων

Πίνακας 1: Phases of PRIMON development	34
---	----

Συντομογραφίες

ΔΠΠΑΕ	Διεθνές Πανεπιστήμιο Ελλάδος
IHU	International Hellenic University
Π.Ε.	Πτυχιακή Εργασία
API	Application Programming Interface
MVVM	Model View ViewModel
UI	Use Interface
PRIMON	Hotel Price Monitoring Android App
OTP	One Time Password
BI	Business Intelligence
AI	Artificial Intelligence

Κεφάλαιο 1ο: Εισαγωγή

Στη σύγχρονη εποχή, η τεχνολογία κινητών τηλεφώνων αναπτύσσεται ραγδαία και έχει ένα μεγάλο αντίκτυπο στις ζωές μας. Επιπρόσθετα, με την πληθώρα και ποικιλία τους οι εφαρμογές παίζουν σημαντικό ρόλο στην καθημερινότητά μας. Η επικοινωνία μεταξύ ανθρώπων και πελατών, η διάδραση μεταξύ ανθρώπων και υπηρεσιών, για οικονομικά και επιχειρησιακά προβλήματα, η διευθέτηση γίνεται μέσω των εφαρμογών κινητών συσκευών [1]. Μία από τις μεγαλύτερες και σημαντικότερες βιομηχανίες τον 21αι. είναι αδιαμφισβήτητα η **βιομηχανία του τουρισμού**. Η ευκολία μαζικής και οικονομικής μετακίνησης σε άλλες χώρες, με εξαιρετικά μεγαλύτερη ασφάλεια, η επιρροή των μέσων μαζικής ενημέρωσης στη καθημερινότητα μας και κυρίως η πληθώρα εργαλείων που είναι ελεύθερα διαθέσιμα ή με ελάχιστο κόστος, είναι κάποιοι από τους λόγους που δικαιολογούν αυτή τη θέση.

Τα κυριότερα προϊόντα της βιομηχανίας του τουρισμού είναι:

- **Μεταφορά:** Μέσω αέρος, μέσω θαλάσσης, μέσω ξηράς
- **Διαμονή-Καταλύματα:** Ξενοδοχεία και Αλυσίδες Ξενοδοχείων, Bed and Breakfast (B & Bs), Καταλύματα αγροτουρισμού, Κάμπινγκ, Κοινόχρηστου Καταλύματα & Airbnb
- **Υποστηρικτικές Υπηρεσίες:** Διαδικτυακό περιεχόμενο, Δημόσιες υπηρεσίες, Οικονομικές υπηρεσίες, Διατροφή, Διασκέδαση, Εμπορικές Εγκαταστάσεις, Εκπαίδευση και Κατάρτιση, Τουριστικοί Οδηγοί - Ξεναγοί
- **Πωλήσεις και Διανομή:** Ταξιδιωτικά γραφεία, Τουρ διαφόρων τύπων [2]

Γίνεται, με αυτόν τον τρόπο, εμφανές ότι η αναζήτηση διαμονής ή καταλύματος αποτελούν ένα από τα πλέον δομικά συστατικά αυτής της τεράστιας βιομηχανίας. Μέσα σε αυτόν τον τομέα η παρακολούθηση των τιμών των ξενοδοχείων καθώς και η ίδια η τιμολόγηση τους είναι εξίσου σημαντικές και οι εφαρμογές που δίνουν αυτές τις λύσεις γίνονται αναπόσπαστο κομμάτι στη καθημερινότητα των ξενοδόχων και της βιωσιμότητας αυτής της βιομηχανίας.

1.1 Βιομηχανία του Hotel Booking

Πριν την εποχή του διαδικτύου, οι κρατήσεις για ένα ξενοδοχείου μπορούσαν να πραγματοποιηθούν μόνο με αναλογικά μέσα όπως τα ταξιδιωτικά γραφεία. Οι χρήστες μπορούσαν να επικοινωνήσουν στο φυσικό χώρο ή μέσω τηλεφώνου για να πραγματοποιήσει ο εκάστοτε πράκτορας την κράτηση, ή να επικοινωνήσουν άμεσα με το ξενοδοχείο, κατά την οποία ο ενδιαφερόμενος έκανε ο ίδιος την κράτηση μαθαίνοντας από το ξενοδοχείο τις τιμές και την διαθεσιμότητα. Υπήρχε και η επιλογή της επιτόπου κράτησης πηγαίνοντας στο ξενοδοχείο χωρίς καμία προηγούμενη επικοινωνία ή μέσα από έντυπα μέσα όπως είναι οι εφημερίδες, τα περιοδικά και τα διαφημιστικά φυλλάδια. Τέλος οι προτάσεις άλλων ταξιδιωτών από δικές τους εμπειρίες και ενέργειες ήταν ένας επιπλέον τρόπος. Τα περισσότερα μέσα πλέον ακούγονται δύσχρηστα και ξεπερασμένα.

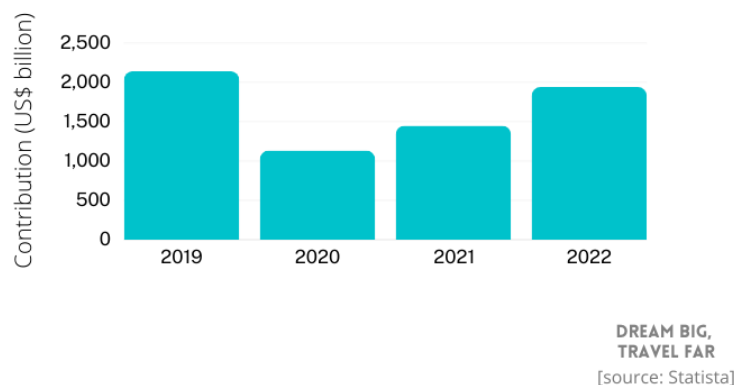
Με την αρχική εμφάνιση του διαδικτύου και κυρίως με την εμφάνιση των πρώτων ιστοσελίδων booking όπως ήταν το [Expedia](#), τα κυρίαρχα μέσα έγιναν η κράτηση μέσω διαδικτύου και τηλεφωνική επικοινωνία, με την δεύτερη να επικρατεί για μία δεκαετία.

Σήμερα, το 73% του συνόλου των τουριστών προχωρούν σε κράτηση δωματίων στα ξενοδοχεία μέσω του Διαδικτύου (βλ. Σχήμα 1.2), το οποίο έχει γίνει ένας από τους κύριους τρόπους πραγματοποίησης κρατήσεων [3]. Οι ταξιδιώτες έχουν αναδείξει το Διαδίκτυο ως τον κύριο μηχανισμό για τον εντοπισμό και την κράτηση δωματίων ξενοδοχείων. [4]

Πριν από μερικά χρόνια αναφέρθηκε ότι ο τουρισμός ήταν η βιομηχανία με τον υψηλότερο όγκο online συναλλαγών, ενώ το περιοδικό "The Economist" (2004) προέβλεψε ότι ο τομέας του ταξιδιωτικού τουρισμού θα μπορούσε να είναι ο πρώτος με την πλειοψηφία των πωλήσεών του να πραγματοποιούνται online. [5] [6]

Στην Εικόνα 1.2 μπορούμε να διακρίνουμε την πλέον υπεροχή των διαδικτυακών κρατήσεων έναντι των υπόλοιπων τρόπων που έχουν αναφερθεί στην αρχή του κεφαλαίου. Το 2022 το 68% των συνολικών κρατήσεων ξενοδοχείων ή καταλυμάτων πραγματοποιήθηκε διαδικτυακά αφήνοντας πίσω τον μέχρι τώρα κυρίαρχο τρόπο επικοινωνίας μέσω τηλεφώνου.

In 2022, travel and tourism contributed \$1,939 billion to the overall GDP in Europe.



Σχήμα 1.1: Ποσοστό δισεκατομμυρίων που συνέβαλε η ταξιδιωτική και τουριστική βιομηχανία στον ΑΕΠ της Ευρώπης μεταξύ 2019 και 2022 [7]

Η πρώτη γενιά ιστοσελίδων και εφαρμογών κράτησης ξενοδοχείων παρείχε ορισμένες πληροφορίες όπως η διεύθυνση, οι υπηρεσίες και οι εκάστοτε εγκαταστάσεις. Πλέον, μέσω των ιστοσελίδων και

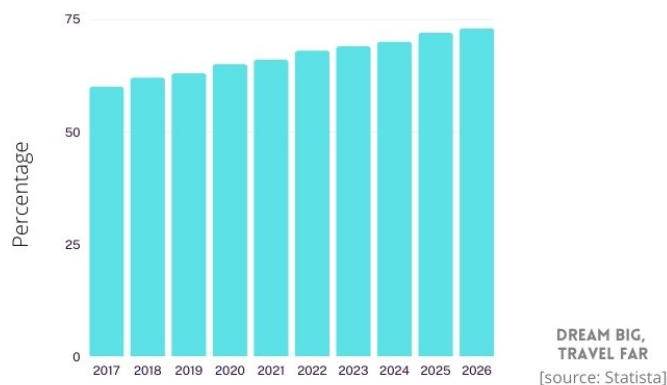
εφαρμογών ξενοδοχείων οι ταξιδιώτες όχι μόνο έχουν πρόσβαση σε πληροφορίες και υπηρεσίες του ξενοδοχείου, αλλά επίσης τους δίνεται η δυνατότητα να κλείσουν το δωμάτιό τους “εν κινήσει”.

Οι εφαρμογές κρατήσεων μέσω κινητού μπορεί να αποτελούσαν μέχρι τώρα ένα μικρό μέρος των συνολικών κρατήσεων ξενοδοχείων, αλλά η αποδοχή και η δημοφιλία τους αυξάνονται εντυπωσιακά καθώς οι περισσότεροι άνθρωποι απομακρύνονται από τους υπολογιστές και συγκεντρώνονται στις κινητές συσκευές. [8]

Στατιστικά η χρήση εφαρμογών για διαδικτυακές κρατήσεις έχουν αυξηθεί σημαντικά τα τελευταία χρόνια. Αυτό οφείλεται σε έναν αριθμό παραγόντων. Ενδεικτικά:

- **Αύξηση χρήσης κινητών τηλεφώνων:** Οι κρατήσεις μέσω κινητών συσκευών έχουν αυξηθεί, με όλο και περισσότερους ταξιδιώτες να χρησιμοποιούν smartphones και tablets για τον προγραμματισμό και την κράτηση των ταξιδιών τους.
- **Χρήση Εφαρμογών:** Πολλοί ταξιδιώτες προτιμούν τη χρήση εφαρμογών για τις κρατήσεις ξενοδοχείων. Τα ξενοδοχεία και οι ταξιδιωτικές εταιρείες έχουν επενδύσει στην ανάπτυξη και βελτιστοποίηση αντίστοιχων εφαρμογών .
- **Δυνατότητα κράτησης τελευταίας στιγμής:** Η ευκολία της χρήσης και καταχώρησης κράτησης ενισχύει αυτήν την τάση.
- **Παγκόσμια Κάλυψη:** Η εύκολη και άμεση πρόσβαση σε παγκόσμια εμβέλεια κατατάσσει τις online εφαρμογές κράτησης αρκετά δημοφιλείς στις προτιμήσεις των ταξιδιωτών.
- **Online Πληρωμή:** Η ενσωμάτωση επιλογών πληρωμής μέσω κινητού έχει διευκολύνει τις οικονομικές συναλλαγές, παρέχοντας στους χρήστες εύκολες και ασφαλείς μεθόδους πληρωμής.
- **Ευέλικτος Σχεδιασμός:** Οι ιστοσελίδες κρατήσεων ξενοδοχείων έχουν υιοθετήσει σχεδιαστικές αρχές που εξασφαλίζουν μια ομαλή εμπειρία σε διάφορες συσκευές. [9]

68% of all sales in travel & tourism are made online in 2022.



Σχήμα 1.2: Το συνολικό ποσοστό των πωλήσεων/κρατήσεων που πραγματοποιήθηκαν online [10]

1.2 Business Intelligence

Η business intelligence συνδυάζει επιχειρηματικές στατιστικές αναλύσεις, εξόρυξη δεδομένων, απεικόνιση δεδομένων, εργαλεία και δομή δεδομένων, και πρακτικές για να βοηθά τις οργανώσεις να παίρνουν αποφάσεις με βάση τα δεδομένα. Πρακτικά η BI παρέχει τη συνολική εικόνα των δεδομένων της εκάστοτε οργάνωσης και της χρήσης των δεδομένων που οδηγούν σε αλλαγή, εξάλειψη της ανεπάρκειας και γρήγορη εφαρμογή στην αγορά ή προμήθεια αλλαγών. Οι νέες BI λύσεις δίνουν προτεραιότητα στην ευέλικτη αυτοεξυπηρετούμενη ανάλυση, στα ελεγχόμενα δεδομένα σε έμπιστες πλατφόρμες, σε δυναμικούς χρήστες επιχειρήσεων και στην ταχύτητα της πληροφορίας. [11]



Σχήμα 1.3 Δομή εργασίας των σύγχρονων δομών ανάλυσης [12]

Κεφάλαιο 1

Οι επιχειρήσεις και οι οργανώσεις έχουν ερωτήματα και στόχους. Για να απαντήσουν τις ερωτήσεις αυτές και να παρακολουθούν την πορεία τους προς αυτούς τους στόχους, συγκεντρώνουν τα απαραίτητα δεδομένα, τα αναλύουν και αποφασίζουν σε τι ενέργειες θα προβούν για να επιτύχουν τους στόχους τους.

Στο τεχνικό κομμάτι συλλέγονται μη επεξεργασμένα δεδομένα από τα συστήματα των επιχειρήσεων. Επεξεργάζονται τα δεδομένα και τα αποθηκεύουν σε βάσεις δεδομένων, στο cloud, σε εφαρμογές και αρχεία. Αφού τα αποθηκεύσουν, οι χρήστες μπορούν να έχουν πρόσβαση στα δεδομένα και να ξεκινήσουν τη διαδικασία ανάλυσης για να απαντηθούν τα ερωτήματα των επιχειρήσεων. Οι πλατφόρμες της BI μπορούν, επίσης, να προσφέρουν εργαλεία οπτικοποίησης δεδομένων, τα οποία μετατρέπουν τα δεδομένα σε διαγράμματα ή γραφήματα, όπως επίσης και να τα παρουσιάσουν στους ενδιαφερόμενους ή σε αυτούς που παίρνουν τις αποφάσεις.

Η Business Intelligence είναι ένας όρος που περιλαμβάνει τις διαδικασίες και τις μεθόδους της συλλογής, αποθήκευσης και επεξεργασίας δεδομένων από τις λειτουργίες και τις δραστηριότητες των επιχειρήσεων προκειμένου να βελτιώσουν την αποδοτικότητά τους. Όλα αυτά τα πράγματα ενώνονται για να δημιουργήσουν μια συνολική εικόνα μιας εταιρίας για να βοηθήσουν τους ανθρώπους να πάρουν καλύτερες αποφάσεις. Τα τελευταία χρόνια, η BI έχει εξελιχθεί σε τέτοιο βαθμό που περιλαμβάνει περισσότερες διεργασίες και δραστηριότητες για να βοηθάει στην βελτίωση της αποδοτικότητας. Αυτές οι διεργασίες περιλαμβάνουν:

Εξόρυξη δεδομένων: Χρήση βάσεων δεδομένων, στατιστικών και machine learning (ML), με σκοπό να ανακαλυφθούν νέες τάσεις σε μεγάλα data sets.

Μετάδοση: Διαμοιρασμός αναλυτικών στοιχείων των δεδομένων σε ενδιαφερόμενους, ώστε να βγάλουν συμπεράσματα και να πάρουν αποφάσεις.

Μέθοδοι μέτρησης της αποδοτικότητας και συγκριτική αξιολόγηση: Σύγκριση πρόσφατων δεδομένων της αποδοτικότητας σε σχέση με παλαιότερα δεδομένα, με σκοπό την παρακολούθηση της απόδοσης σε συσχετισμό με τους στόχους, με τη χρήση εξατομικευμένων ταμπλό.

Περιγραφική ανάλυση: Χρήση των αρχικών αναλύσεων των δεδομένων ώστε να βρεθεί τι συνέβη.

Ερωτήματα: Η BI βρίσκει απαντήσεις από τα data sets με σκοπό την απάντηση συγκεκριμένων ερωτημάτων.

Ανάλυση στατιστικών: Παίρνει τα αποτελέσματα από περιγραφικές αναλύσεις και εξερευνά περαιτέρω τα δεδομένα, χρησιμοποιώντας στατιστικές, όπως “πώς ήρθε αυτή η τάση και γιατί”.

Οπτικοποίηση δεδομένων: Μετατροπή ανάλυσης δεδομένων σε οπτικές παρουσιάσεις, όπως γραφήματα, διαγράμματα και ιστογράμματα, με σκοπό την ευκολότερη κατανάλωση δεδομένων.

Οπτική ανάλυση: Εξερευνά τα δεδομένα μέσω οπτικής αφήγησης για να επικοινωνήσουν γνώσεις ταξιδιού στη ροή της ανάλυσης.

Προετοιμασία δεδομένων: Συλλογή πολλαπλών πηγών δεδομένων, για την αναγνώριση των διαστάσεων και των μετρήσεων, και την προετοιμασία για ανάλυση δεδομένων.

Η BI περιλαμβάνει αναλύσεις δεδομένων και αναλυτικά στοιχεία εταιρειών, αλλά τα χρησιμοποιεί μόνο ως μέρη της όλης διαδικασίας. Η BI βοηθά τους χρήστες να βγάλουν συμπεράσματα από τα αναλυτικά στοιχεία. Οι επιστήμονες δεδομένων εισχωρούν στις λεπτομέρειες των δεδομένων, χρησιμοποιώντας προχωρημένες στατιστικές και προγνωστικά αναλυτικά στοιχεία για να ανακαλύψουν μοτίβα και να προβλέψουν μελλοντικά μοτίβα.

Η Business Intelligence παίρνει αυτά τα μοντέλα και τους αλγορίθμους και αναλύει τα αποτελέσματα σε πραγματοποιήσιμη γλώσσα. Σύμφωνα με το Gartner's IT γλωσσάρι, τα αναλυτικά στοιχεία των εταιρειών περιλαμβάνουν εξόρυξη δεδομένων, προγνωστικές αναλύσεις, εφαρμοσμένες αναλύσεις και στατιστικές. Εν συντομία, οι οργανώσεις διενεργούν αναλύσεις στοιχείων των εταιρειών, ως μέρος της μεγαλύτερης BI στρατηγικής τους.

Η BI είναι σχεδιασμένη να απαντά συγκεκριμένα ερωτήματα και να παρέχει αναλύσεις, με μια ματιά, για αποφάσεις ή σχεδιασμό. Ωστόσο, οι εταιρείες μπορούν να χρησιμοποιούν τις διεργασίες των αναλύσεων για να βελτιώνουν συνεχώς τις επιπλέον ερωτήσεις και την επανάληψη. Τα αναλυτικά στοιχεία των εταιρειών δεν πρέπει να είναι μια διαδικασία σε σειρά, επειδή η απάντηση μιας ερώτησης, είναι πιθανό, να οδηγήσει σε επιπλέον ερωτήσεις και επανάληψη. Αντ' αυτού, σκέψου ότι αυτή η διαδικασία είναι ένας κύκλος πρόσβασης σε δεδομένα, ανακάλυψης, εξερεύνησης και διαμοιρασμού πληροφοριών.

Αυτό αποκαλείται "ο κύκλος των αναλυτικών στοιχείων", ένας όρος που εξηγεί πως οι επιχειρήσεις χρησιμοποιούν αναλυτικά στοιχεία για να αντιδρούν στην αλλαγή ερωτημάτων και τις προσδοκίες.

Στο παρελθόν, τα εργαλεία της BI ήταν βασισμένα στο μοντέλο της παραδοσιακής BI. Αυτό ήταν μια προσέγγιση όπου η business intelligence οδηγούνταν από την οργάνωση του IT και τα περισσότερα, αν όχι όλα, τα ερωτήματα των αναλύσεων μπορούσαν να απαντηθούν μέσω στατιστικών μεταδόσεων. Αυτό σήμαινε, ότι αν κάποιος είχε ένα επιπλέον ερώτημα για την αναφορά που παρέλαβε, το αίτημα του θα πήγαινε στη βάση της σειράς των μεταδόσεων και θα ξεκινούσε ξανά η διαδικασία. Αυτό οδήγησε σε αργούς κύκλους μεταδόσεων και οι άνθρωποι δεν μπορούσαν να αξιοποιήσουν τα πρόσφατα δεδομένα για να πάρουν αποφάσεις. Η παραδοσιακή BI είναι ακόμα μια συνήθης προσέγγιση για τακτική μετάδοση και απάντηση αμετάβλητων ερωτημάτων. Ωστόσο, η νέα BI είναι διαδραστική και προσιτή. Ενώ τα τμήματα IT είναι ακόμα ένα σημαντικό κομμάτι στη διαχείριση της πρόσβασης των δεδομένων, πολλαπλά επίπεδα χρηστών μπορούν να προσαρμόσουν τα ταμπλό και να δημιουργήσουν αναφορές σε μικρό χρονικό διάστημα. Με το σωστό λογισμικό, οι χρήστες έχουν τη δύναμη να οπτικοποιήσουν τα δεδομένα και να απαντήσουν τις δικές τους ερωτήσεις.

Η BI είναι κάτι παραπάνω από ένα απλό λογισμικό, είναι ένας τρόπος να διατηρείς μια ολιστική και σε πραγματικό χρόνο άποψη όλων των σχετικών εταιρικών δεδομένων.

Η εφαρμογή της BI προσφέρει πολλά πλεονεκτήματα, από καλύτερη ανάλυση μέχρι αύξηση στο πλεονέκτημα του ανταγωνισμού. Μερικά από τα πλεονεκτήματα της ανώτερης BI περιλαμβάνουν:

- Σαφήνεια δεδομένων
- Αυξημένη αποδοτικότητα
- Καλύτερη εμπειρία πελατών
- Μεγαλύτερη ευχαρίστηση υπαλλήλων [12]

1.3 Hotel Analytics

Στο σημερινό δυναμικό και ανταγωνιστικό γίνεσθαι η σημασία των hotel analytics (ανάλυση ξενοδοχειακών δεδομένων) δεν μπορεί να περάσει απαρατήρητη. Η βιομηχανία της φιλοξενίας βρίσκεται σε ένα μεταβατικό στάδιο, όπως δείχνουν και οι παραπάνω αναλύσεις, και αλλάζει σύμφωνα με τις συμπεριφορές των καταναλωτών και την περαιτέρω ανάπτυξη της τεχνολογίας. Η ανάλυση των ξενοδοχειακών δεδομένων αποτελεί, αν όχι τον θεμελιακό, έναν από τους πιο βασικούς ρόλους σε αυτή την εξέλιξη, προσφέροντας οφέλη που επηρεάζουν, τόσο την ικανοποίηση των ταξιδιωτών όσο και την αποδοτικότερη λειτουργία των επιχειρήσεων. Εν συνεχεία, παρουσιάζονται σύντομα κάποιοι λόγοι που τονίζουν τον λόγο που τα hotel analytics επηρεάζουν και διαμορφώνουν την εν λόγω βιομηχανία.

Προσωποποιημένη εμπειρία πελάτη: Μέσω της λεπτομερούς ανάλυσης δεδομένων των φιλοξενούμενων πελατών, τα ξενοδοχεία μπορούν να δημιουργήσουν μοναδικές υπηρεσίες και να προσαρμόσουν τις υπηρεσίες τους στις προτιμήσεις του εκάστοτε φιλοξενούμενου, ενισχύοντας την αίσθηση οικειότητας και προσωποποίησης ακόμα και σε κλίμακα πιο απαιτητικών πελατών.

Βελτίωση τιμών και τιμολόγησης: Η ανάλυση της τιμολόγησης επιτρέπει στα ξενοδοχεία να προσαρμόζουν δυναμικά τις τιμές τους με βάση παράγοντες όπως η ζήτηση, οι εποχιακές σεζόν και οι τιμές των ανταγωνιστών, βελτιστοποιώντας, με αυτό τον τρόπο, τα έσοδα και εξασφαλίζοντας ανταγωνιστικές τιμές στην αγορά.

Βελτιωμένο μάρκετινγκ: Το βασισμένο, σε δεδομένα, μάρκετινγκ επιτρέπει στα ξενοδοχεία να στοχεύουν πιο αποτελεσματικά στο επιθυμητό κοινό μέσα από προσωποποιημένες καμπάνιες και αντίστοιχες προωθήσεις.

Λειτουργική αποδοτικότητα: Η ανάλυση ξενοδοχειακών δεδομένων παρέχει πληροφορίες, όπως την διαχείριση πόρων, την απόδοση του προσωπικού και της συνολικής αποδοτικότητας επιτρέποντας στο εκάστοτε ξενοδοχείο να πραγματοποιεί αποδοτικές αποφάσεις που ενισχύουν την αποδοτικότητα καθώς και την ικανοποίηση του πελάτη.

Προβλέψεις και στρατηγικός σχεδιασμός: Με την ανάλυση δεδομένων ιστορικού και τις τάσεις της αγοράς, τα ξενοδοχεία μπορούν να προχωρήσουν σε προληπτικές προβλέψεις και στρατηγικό σχεδιασμό, διασφαλίζοντας την προετοιμασία τους σε τυχόν μεταβολές, μελλοντικές αλλαγές καθώς και για την περίπτωση εμφάνισης ευκαιριών.

Διαχείριση Φήμης: Η διαχείριση της επωνυμίας-φήμης είναι εξαιρετικά σημαντική για τα ξενοδοχεία. Μέσω της ανάλυσης δεδομένων η παρακολούθηση των εμπειριών των πελατών και των κριτικών μέσα από τα κοινωνικά μέσα είναι ζωτικής σημασίας για την ανάπτυξη και διατήρηση της φήμης των Ξενοδοχείων.

Ανάλυση ανταγωνιστών: Η συνεχής ανάλυση των ανταγωνιστών παρέχει, στα ξενοδοχεία, στρατηγική εποπτεία επιτρέποντας την αναγνώριση των ισχυρών και των αδύναμων σημείων του ανταγωνισμού, οδηγώντας έτσι στην μελέτη και στο σχεδιασμό τη κατάλληλης στρατηγικής με απώτερο αποτέλεσμα την βελτίωση της θέσης τους στην αγορά.

Διαχείριση ρίσκου: Η διαχείριση των ρίσκων μέσω της ανάλυσης δεδομένων επιτρέπει στα ξενοδοχεία να προβλέπουν και να αντιμετωπίζουν πιθανά προβλήματα που μπορεί να εμφανιστούν. Αυτό έχει ως αποτέλεσμα να εξασφαλίσουν την ανθεκτικότητα της επιχείρησης σε απρόβλεπτες καταστάσεις.

Πρωτοβουλίες βιωσιμότητας: Οι πρωτοβουλίες βιωσιμότητας ενισχύονται με τη χρήση αναλυτικών δεδομένων, επιτρέποντας στα ξενοδοχεία να παρακολουθούν και να βελτιστοποιούν την κατανάλωση πόρων πετυχαίνοντας βιώσιμες πρακτικές για το περιβάλλον. [13]

Για να κατανοήσουμε εις βάθος τον τρόπο λειτουργίας και παρουσίας των πληροφοριών μέσα στην εφαρμογή PRIMON πρέπει να γίνει κατανοητό ότι τα data analytics στον χώρο των ξενοδοχείων για πρακτικούς λόγους κατηγοριοποιούνται με βάση κάποιες κατηγορίες. Κάθε κατηγορία εξυπηρετεί ένα μοναδικό σκοπό και προσφέρει διαφορετικού τύπου πληροφορίες. Παρακάτω παρουσιάζονται οι κύριες κατηγορίες των data analytics.

Περιγραφική Ανάλυση (Descriptive Analytics): Περιλαμβάνει την εξέταση ιστορικών δεδομένων ώστε να χαρτογραφηθεί η παρελθοντική πορεία των ζητούμενων δεδομένων. Παρουσιάζει τάσεις, μοτίβα και δείκτες απόδοσης χωρίς να πραγματοποιεί προβλέψεις.

Διαγνωστική Ανάλυση (Diagnostic Analytics): Επικεντρώνεται στην αναγνώριση των λόγων, των οποίων κάποια γεγονότα πραγματοποιήθηκαν ή πραγματοποιούνται. Εξερευνά τις σχέσεις αιτίου-αποτελέσματος μέσα στα δεδομένα και δίνει μία εξήγηση για το αποτέλεσμα.

Προβλεπτική Ανάλυση (Predictive Analytics): Χρησιμοποιεί στατιστικούς αλγόριθμους και τεχνική μηχανικής μάθησης για να αναλύσει ιστορικά δεδομένα και να κάνει προβλέψεις για μελλοντικά γεγονότα. Ο σκοπός της είναι να προβλέψει τάσεις καθώς και τα μελλοντικά αποτελέσματα.

Προγραμματιστική Ανάλυση (Prescriptive Analytics): Πηγαίνοντας ένα βήμα πέρα από τα μελλοντικά αποτελέσματα, η προγραμματιστική ανάλυση προτείνει ενέργειες βελτιστοποίησης των αποτελεσμάτων αυτών. Προσφέρει μηχανισμούς λήψης αποφάσεων με συγκεκριμένες στρατηγικές για την επίτευξη συγκεκριμένων στόχων βασισμένη σε διάφορα σενάρια και την πιθανή επίδρασή τους.

Ανάλυση σε Πραγματικό Χρόνο (Real-time Analytics): Περιλαμβάνει την επεξεργασία και την ανάλυση δεδομένων προσφέροντας άμεσες απαντήσεις και λήψεις αποφάσεων. Είναι ζωτικής σημασίας

για εφαρμογές που ο χρόνος απόκρισης πρέπει να ναι μικρός και για βιομηχανίες που απαιτούν άμεση ανταπόκριση σε μεταβαλλόμενες συνθήκες.

Ανάλυση Αισθήματος (Sentiment Analysis): Γνωστό και σαν “εξόρυξη γνώμης”, περιλαμβάνει αξιολόγηση και την ερμηνεία απόψεων, συναισθημάτων και συμπεριφορών εκφρασμένων σε γραπτά δεδομένα. Βοηθάει τις επιχειρήσεις να κατανοήσουν την αντίληψη του κοινού και τα συναισθήματα απέναντι σε ένα προϊόν, μια υπηρεσία ή μία επωνυμία. [13]

1.4 Hotel Price Monitoring

Σε συνέχεια των hotel analytics, η εφαρμογή που αποτελεί το κεντρικό θέμα αυτής της πτυχιακής εργασίας επικεντρώνεται στο κομμάτι του παρακολούθηση τιμής (price monitoring). **Η διαδικασία της παρακολούθησης των τιμών των ξενοδοχείων περιλαμβάνει την συστηματική παρακολούθηση και ανάλυση των τιμολογήσεων δωματίων και υπηρεσιών των ξενοδοχείων.** Η διαδικασία δεν αφορά μόνο την παρακολούθηση της δομής την τιμολόγησης του εκάστοτε ξενοδοχείου αλλά και την παρακολούθηση των τιμών των ανταγωνιστών του, των τάσεων της αγοράς και τα μοτίβα ζήτησης των πελατών.

Η **σημασία** της παρακολούθησης των τιμών του ξενοδοχείου συναντάται στα παρακάτω σημεία.

Πλεονέκτημα ανταγωνισμού: Η κατανόηση των στρατηγικών τιμολόγησης των ανταγωνιστών επιτρέπει στα ξενοδοχεία να θέσουν τις τιμές τους πιο ανταγωνιστικά. Αυτό μπορεί να βοηθήσει στην προσέλκυση περισσότερων επισκεπτών, προσφέροντας μεγαλύτερη αξία ή να χαρακτηρίσει τις υπηρεσίες καλύτερες σε σύγκριση με άλλες.

Δυναμική τιμολόγηση: Η ξενοδοχειακή βιομηχανία είναι εξαιρετικά δυναμική, με την ζήτηση δωματίων να κυμαίνεται λόγω διαφόρων παραγόντων, όπως η εποχή, οι τοπικές εκδηλώσεις ή ακόμη και οι αλλαγές στα δρομολόγια των πτήσεων. Η παρακολούθηση τιμών επιτρέπει στα ξενοδοχεία να εφαρμόζουν δυναμικές στρατηγικές τιμολόγησης, προσαρμόζοντας τις τιμές σε πραγματικό χρόνο, έτσι ώστε να ταιριάζουν με τη ζήτηση και αυτόν τον τρόπο να μεγιστοποιούν τα έσοδα.

Γνώση της αγοράς: Η τακτική παρακολούθηση των τιμών παρέχει γνώσεις στις τάσεις των αγορών, βοηθώντας τα ξενοδοχεία να κατανοούν πότε να αναμένουν υψηλότερη ή χαμηλότερη ζήτηση. Αυτή η γνώση μπορεί να καθορίζει διάφορες λειτουργικές αποφάσεις από την επάνδρωση μέχρι τις προσπάθειες προώθησης.

Βελτιστοποίηση εσόδων: Με τη στρατηγική προσαρμογή των τιμών, με βάση τη ζήτηση και τον ανταγωνισμό, τα ξενοδοχεία μπορούν να αυξήσουν τα έσοδα τους. Η κανονική τιμολόγηση είναι απαραίτητη, όχι μόνο για την προσέλκυση πελατών, αλλά και για την εξασφάλιση της αποδοτικότητας.

Διαχείριση των εντυπώσεων των πελατών: Η τιμολόγηση επηρεάζει την εντύπωση που έχουν οι πελάτες για ένα ξενοδοχείο. Η τακτική παρακολούθηση και προσαρμογή των τιμών βοηθάει στη

διατήρηση της ισορροπίας ανάμεσα στο να θεωρηθεί οικονομικό ένα ξενοδοχείο αλλά ταυτόχρονα και ποιοτικό.

Διαχείριση διαδικτυακής κατανομής: Είναι σημαντικό, για τα ξενοδοχεία που βρίσκονται σε λίστες διαφόρων διαδικτυακών ταξιδιωτικών πρακτορείων να διατηρούν μία ισότητα στις τιμές (σταθερή τιμολόγηση σε διάφορες πλατφόρμες). Η παρακολούθηση τιμών βοηθάει αποτελεσματικά στη διαχείριση αυτών των σχέσεων και στην αποφυγή των συγκρούσεων ή της δυσανασχέτησης των πελατών, λόγω απόκλισης τιμών.

Πρόγνωση και σχεδιασμός: Η μακροχρόνια παρακολούθηση τιμών μπορεί να βοηθήσει στην πρόγνωση των μελλοντικών τάσεων της αγοράς, οδηγώντας σε έναν πιο ακριβή σχεδιασμό επιχειρηματικής στρατηγικής, από την κατανομή του προϋπολογισμού μέχρι και σχέδια επέκτασης.

Προσαρμογή στη συμπεριφορά των πελατών: Στην ψηφιακή εποχή, οι καταναλωτές είναι ικανοί να συγκρίνουν τιμές διαδικτυακά, πριν αποφασίσουν να κάνουν μία κράτηση. Η τακτική παρακολούθηση εξασφαλίζει, η στρατηγική ενός ξενοδοχείου να συμφωνεί με τη τωρινή συμπεριφορά και τις προσδοκίες των πελατών.

Εν κατακλείδι το hotel price monitoring είναι μία διαδικασία υψηλής σημασίας για τη διασφάλιση της οικονομικής σταθερότητας και του υγιούς ανταγωνισμού στον χώρο της αγοράς των ξενοδοχείων. Επιτρέπει στους ξενοδόχους να παίρνουν αποφάσεις βασισμένες σε δεδομένα και τις τρέχουσες τάσεις της αγοράς και συμπεριφορά των πελατών. Μια στρατηγική τιμολόγησης με βάσει τους παραπάνω παράγοντες μπορεί να οδηγήσει σε αύξηση των εσόδων και την απόκτηση ή να διατήρηση μιας καλής θέσης στην αγορά γεγονός που έρχεται να επιβεβαιώσει η έρευνα και δημοσίευση του Girish. K. Nair πάνω στη δυναμική τιμολόγηση των ξενοδοχείων [14]. Όσο η βιομηχανία συνεχίζει να εξελίσσεται με την παρουσία νέων τεχνολογιών και με τις προσδοκίες των πελατών να συνεχίζουν να αυξάνονται, ο ρόλος του price monitoring γίνεται ολοένα και πιο ζωτικός για της επίτευξη και διατήρηση της ανάπτυξης και της επιτυχίας.

1.5 Hotel Pricing

Με τον όρο ‘‘τιμολόγηση ξενοδοχείου’’ αναφερόμαστε στον τρόπο με τον οποίο βελτιστοποιούνται οι τιμές των καταλυμάτων, με στόχο την αύξηση, τόσο της πελατείας όσο και των συνολικών εσόδων. Η παροχή υπηρεσιών με την κατάλληλη και πιο αποδοτική κοστολόγηση, με την προσπάθεια διατήρησης μια σταθερής οικονομικής βάσης καθώς και την αποφυγή των υπονομεύσεων από τους ανταγωνιστές είναι μία πολύπλοκη εργασία που απαιτεί συνεχή προσοχή και ενημέρωση. Οι ξενοδόχοι πρέπει να είναι διορατικοί με την κοστολόγηση των δωματίων τους. Η συλλογή πραγματικών δεδομένων για τις τάσεις της αγοράς και των επιχειρήσεων είναι ζωτικής σημασίας για τη δημιουργία και τη διαχείριση ενός βέλτιστου μοντέλου τιμολόγησης.

Η τιμολόγηση των καταλυμάτων των ξενοδοχείων βασίζεται πάνω στη βασική αρχή προσφοράς και ζήτησης. Κατά περιόδους μεγάλης αιχμής η ζήτηση είναι μεγάλη οπότε οι τιμές έχουν ανοδική τάση. Αντιστοίχως σε περιόδους χαμηλής ζήτησης οι τιμές τείνουν να 'ναι χαμηλότερες ώστε να προσελκύσουν περισσότερους επισκέπτες. Ωστόσο, δεν είναι μόνο η ανάλογη περίοδος που επηρεάζει αυτή τη μεταβλητή. Παράγοντές, όπως οι τιμές των ανταγωνιστών, οι κριτικές των επισκεπτών, τα λειτουργικά κόστη και τα επιθυμητά, απαραίτητα περιθώρια κέρδους πρέπει να ληφθούν υπόψιν από τους επαγγελματίες. Η πετυχημένη τιμολόγηση πρέπει να ισορροπεί ανάμεσα στη στρατηγική, την έρευνα αγοράς και τις προσδοκίες των πελατών.

Ο κεντρικός στόχος της τιμολόγησης είναι η εξασφάλιση του μέγιστου κέρδους από κάθε ένα ξεχωριστό δωμάτιο. Αυτό επιτυγχάνεται από το πόσο κοστολογείται δυναμικά το κάθε δωμάτιο και όχι ποια είναι η αντικειμενική αξία του. Ένας καλεσμένος συχνά μπορεί να πληρώσει παραπάνω από αυτή την αντικειμενική αξία με αν νιώσει ότι θα αποκομίσει κάποιο επιπλέον όφελος ή υπηρεσία από αυτό.

Σημαντικός παράγοντας όπως αναφέρθηκε και παραπάνω είναι και η πληρότητα, διαθεσιμότητα κάθε ξενοδοχείου. Με τη μη διάθεση όλων των δωματίων σε επισκέπτες, οι πιθανότητες επίτευξης κέρδους μειώνονται. Οπότε, η επιτυχία της τιμολόγησης έγκειται στο πάντρεμα της καλύτερης δυνατής τιμής και τη μεγιστοποίηση της πληρότητας [15].

Μια στρατηγική τιμολόγησης προκειμένου να 'ναι αποδοτική και επιτυχημένη πρέπει τόσο ο επιχειρηματίας όσο και αυτός που καλείται να υλοποιήσει την εν λόγω στρατηγική να κατανοήσουν τις βασικές αρχές της τιμολόγησης. Αυτές είναι η γνώση του κοινού-στόχου, η κατανόηση των απαιτήσεων των επισκεπτών και η ανάλυση των δεδομένων για τη δημιουργία της στρατηγικής. Με αυτές τις γνώσεις γίνεται πολύ εύκολη η επιλογή της εκάστοτε στρατηγικής [16].

Σαφώς όπως και σε κάθε άλλη βιομηχανία υπάρχουν κάποια πρότυπα και κανόνες που πρέπει να τηρούνται προκειμένου τα ξενοδοχεία να ακολουθούν τη νόμιμη οδό. Στον τομέα την τιμολόγησης οι κανόνες που πρέπει να τηρούνται είναι οι εξής:

- Διαφανείς και ακριβείς τιμές: Οι επισκέπτες πληρώνουν αυτά που πιστεύουν ότι πρέπει να πληρώσουν.
- Μη ψευδείς διαφημίσεις: Οι επισκέπτες πληρώνουν για κάποια συγκεκριμένα προϊόντα και υπηρεσίες. Ο εκάστοτε επιχειρηματίας καλείται με τη σειρά του να παρέχει ακριβώς αυτά τα προϊόντα και υπηρεσίες.
- Ανάλυση των επιπρόσθετων χρεώσεων: Κάποιες πρόσθετες χρεώσεις μπορεί να είναι τοπικοί φόροι (που κυμαίνονται ανάλογα τη χώρα, πόλη) και επιπρόσθετες υπηρεσίες [15].

Εν συνεχεία παρουσιάζονται οι τρεις κοινές προσεγγίσεις στην τιμολόγηση:

- Η τιμολόγηση βάσει κόστους (Cost-based pricing) : Μια προσέγγιση τιμολόγησης με οικονομική κατεύθυνση, όπου τα προϊόντα τιμολογούνται για να αποφέρουν ένα δίκαιο

κέρδος πέρα και πάνω από όλα τα κόστη που σχετίζονται με την παραγωγή του προϊόντος.

- Τιμολόγηση με βάση τον πελάτη (Customer-driven pricing): Μια προσέγγιση τιμολόγησης καθοδηγούμενη από την αγορά, όπου οι τιμές καθορίζονται από το ποσό που οι πελάτες είναι διατεθειμένοι να πληρώσουν για το προϊόν.
- Τιμολόγηση βασισμένη στον ανταγωνισμό (Competition-driven pricing): Μια προσέγγιση τιμολόγησης καθοδηγούμενη από την αγορά, όπου οι τιμές καθορίζονται από το επίπεδο τιμολόγησης, το οποίο επιτυγχάνει ένα συγκεκριμένο επίπεδο μεριδίου αγοράς από την εταιρεία. [17]

Αν και πλέον υπάρχει ένας αριθμός διαθέσιμων εφαρμογών για hotel pricing & hotel price monitoring, όπως είναι τα price2spy.com, prisync.com, d-edge.com, είναι εμφανές ότι δεν υπάρχει ένας αντίστοιχος αριθμός όπως είναι οι διαθέσιμες εφαρμογές για online booking. Αυτό έχει ως αποτέλεσμα να υπάρχει ένα γενναίο περιθώριο για ανάπτυξη επιπροσθέτων εφαρμογών ώστε να καλυφθεί περαιτέρω το υπάρχον φάσμα.

1.6 Κίνητρο Πτυχιακής Εργασίας

Όπως έχει γίνει κατανοητό εξαιτίας του τεράστιου ρόλου της τουριστικής βιομηχανίας, και πιο συγκεκριμένα της βιομηχανίας ξενοδοχείων-καταλυμάτων, και του καθοριστικού ρόλου της τεχνολογίας στην καθημερινότητα μας, υπάρχει μια συνεργασία που συναντάται ανάμεσα σε αυτούς τους κλάδους, το οποίο είναι και ο κύριος ρόλος της εφαρμογής της πτυχιακής αυτής.

Και στις δυο περιπτώσεις καθίσταται επιτακτική η ανάγκη για ομαδική συλλογή πληροφοριών από το μεγαλύτερο ποσοστό online εφαρμογών και ιστοσελίδων booking. Ο μεγάλος όγκος καλείται να καταστεί διαχειρίσιμος. Στη συνέχεια, αναλαμβάνει η **PRIMON: Hotel Price Monitoring** εφαρμογή, η οποία με τη σειρά της καλείται να κάνει αυτήν την πληροφορία παρουσιάσιμη και κατανοητή στους επιχειρηματίες του ξενοδοχειακού χώρου. Η ανάπτυξη αυτού του είδους εφαρμογών μπορεί θεωρηθεί επιτακτική για την βιομηχανία των καταλυμάτων, τόσο για τους ξενοδόχους όσο και γενικότερα για τους επαγγελματίες τους χώρου. Παρουσιάζονται μερικοί από τους πλέον σημαντικούς λόγους για ένα έργο.

Μεγάλη ζήτηση στις βιομηχανίες των ξενοδοχείων: Σε αυτόν το χώρο ο ανταγωνισμός έχει πρωταρχικό λόγο. Όποτε η συνεχής βελτιστοποίηση των στρατηγικών τιμολόγησης είναι απαραίτητη. Κατ' επέκταση ένας μηχανισμός που συμβάλλει σε αυτή τη βελτίωση θα έχει αρκετή ζήτηση στον χώρο.

Διαχείριση των εσοδών: Η αποτελεσματική τιμολόγηση είναι εξίσου ζωτικής σημασίας για την επιβίωση ενός ξενοδοχείου. Μία εφαρμογή τέτοιας χρήσης μπορεί να βοηθήσει τους ξενοδόχους να προσαρμόζουν τις τιμές ανάλογα με τη ζήτηση, τις ανταγωνιστικές τιμές, τις τάσεις της αγοράς για την επίτευξη του μεγαλύτερου δυνατού κέρδους.

Αυτοματισμός: Στο παρελθόν η συνεχής παρακολούθηση τιμών ήταν μια χρονοβόρα χειρωνακτική εργασία η οποία δεν είχε πάντα το καλύτερο, ποιοτικά, αποτέλεσμα. Πολλές φορές, οι τιμές σύγκρισης δεν ήταν ενημερωμένες και πολλές φορές υπήρχε και ο ανθρώπινος παράγοντας που μπορούσε να κάνει κάποιο λάθος κατά τη μεταφορά και τη σύγκριση δεδομένων. Πλέον όλα τα παραπάνω προβλήματα μπορούν να λυθούν με τις αυτοματοποιημένες εργασίες που ήδη υπάρχουν και συνεχίζουν να αναπτύσσονται.

Λήψη αποφάσεων μέσω δεδομένων: Στον επιχειρηματικό τομέα η λήψη αποφάσεων συνδέεται ολοένα και περισσότερο με την ανάλυση δεδομένων. Μία εφαρμογή παρακολούθησης τιμών ξενοδοχείων μπορεί να παρέχει πολύτιμες πληροφορίες ώστε οι ξενοδόχοι να παίρνουν ενημερωμένες αποφάσεις για τις στρατηγικές τιμολόγησής τους.

Κενό στην αγορά και η καινοτομία: Όπως αναφέρθηκε και στις παραπάνω ενότητες στον χώρο των εφαρμογών που αφορούν τους επαγγελματίες του τουρισμού υπάρχει μια εμφανής αριθμητική έλλειψη στο hotel price monitoring. Αυτό καθιστά ιδανικό αυτόν τον χώρο για περαιτέρω καινοτομία και νέες ιδέες για τυχόν κενά που μπορεί να μην έχουν καλυφθεί ακόμα.

Η PRIMON αποτελεί μία hotel analytics και price monitoring android-based εφαρμογή. Πιο συγκεκριμένα είναι real-time analytics εφαρμογή, με χρήση πραγματικών δεδομένων και στο κομμάτι του hotel analytics συναντάται στις κατηγορίες της προσωποποιημένης εμπειρίας του πελάτη, της δυναμικής τιμολόγησης και της ανάλυσης ανταγωνιστών. Όπως αναφέρθηκε και παραπάνω, το κύριο κοινό μιας τέτοιας εφαρμογής μπορεί να είναι οι ιδιοκτήτες ξενοδοχείων αλλά και εξειδικευμένοι επαγγελματίες και υπάλληλοί του χώρου.

Η μεγάλη σημασία της ύπαρξης μιας τέτοιου είδους εφαρμογής αυξάνεται μέσα στο πλαίσιο της μετά Covid περιόδου, κατά τη διάρκεια της οποίας, όπως αναφέρθηκε, όλη η βιομηχανία του τουρισμού υπέστη ένα διακριτό πλήγμα. Έχοντας παρέλθει αυτό το στάδιο και με τον τομέα του τουρισμού να ξανά-αναπτύσσεται και να ακμάζει με την απτή διαφορά ότι τα διαθέσιμα εργαλεία έχουν πολλαπλασιαστεί σημαντικά στο πέρασμα του χρόνου, κάτω από το πρίσμα της ζήτησης, λύσεις όπως η εν λόγω εφαρμογή δεν θα είναι μόνο χρήσιμες αλλά και εν δυνάμει απαραίτητες.

Κεφάλαιο 2ο: Τεχνολογίες

2.1 Android

2.1.1 Ιστορικό Android-based εφαρμογών

Το Android είναι ένα λειτουργικό σύστημα και πλατφόρμα προγραμματισμού που αναπτύχθηκε από την Google για κινητά τηλέφωνα και άλλες φορητές συσκευές, όπως τα tablet. Μπορεί να λειτουργήσει σε μία ποικιλία συσκευών από διαφορετικούς κατασκευαστές. Το Android περιλαμβάνει ένα σύνολο εργαλείων ανάπτυξης λογισμικού που βοηθά τους προγραμματιστές να γράψουν πρωτότυπο κώδικα καθώς και δημιουργήσουν modules για τους χρήστες του Android. Επιπλέον παρέχει ένα marketplace για τη διανομή android εφαρμογών. Γενικότερα, μιλάμε για την ύπαρξη ενός οικοσυστήματος εφαρμογών για κινητές και όχι μόνο συσκευές. [18]

Ως η πιο δημοφιλής πλατφόρμα κινητών συσκευών στον κόσμο, το Android χρησιμοποιείται από εκατομμύρια κινητές συσκευές σε περισσότερες από 190 χώρες παγκοσμίως. Διαθέτει τη μεγαλύτερη εγκατεστημένη βάση συσκευών από κάθε άλλη πλατφόρμα και συνεχίζει να αυξάνεται με γρήγορους ρυθμούς. Κάθε μέρα, ένα εκατομμύριο χρήστες ενεργοποιούν τις συσκευές τους που λειτουργούν με Android για πρώτη φορά και αρχίζουν να ψάχνουν για εφαρμογές, παιχνίδια και άλλο ψηφιακό περιεχόμενο. Όπως φαίνεται και στο Σχήμα 2.1, το λειτουργικό σύστημα android κατέχει το μεγαλύτερο κομμάτι της αγοράς λειτουργικών συστημάτων, στις μέρες μας με το λειτουργικό iOS να ακολουθεί στη δεύτερη θέση. Η παγκόσμια κοινότητα συνεχίζει να αποκτά και να δημιουργεί γνώση και εμπειρία στη χρήση των λειτουργικών συστημάτων και εφαρμογών Android.



Σχήμα 2.1: Ποσοστά της αγοράς Λειτουργικών Συστημάτων κινητών συσκευών το 2023

Μεγάλη σημασία την εξάπλωση του android υπήρξε η δημιουργία του Android Market από τη Google, πλέον γνωστό ως Google Play Store, όπου οι χρήστες έχουν τη δυνατότητα να κατεβάζουν και να εγκαθιστούν εφαρμογές για τις συσκευές του Αυτό σημαίνει ότι έγινε ένα σημαντικό βήμα για να φτιάξουμε περισσότερες εφαρμογές για τα Android κινητά. Τα Android λειτουργούν πάνω σε έναν πυρήνα Linux και είναι γνωστά για το ότι είναι ανοιχτά, επιτρέποντας στις εταιρείες και τους προγραμματιστές να προσαρμόζουν το λειτουργικό σύστημα για τις συσκευές τους [19]. Αυτή η "ανοιχτή" προσέγγιση βοήθησε στο να γίνει το Android τόσο δημοφιλές Πρακτικά, οι εφαρμογές που αναπτύσσονται μέσω του Android συστήματος εμφανίζουν κάποια **σχεδιαστικά πρότυπα**. Η Android παρέχει μια οθόνη **αφής διεπαφής χρήστη** (UI) κατά την αλληλεπίδραση με τις εφαρμογές. Το UI του

Κεφάλαιο 2

Android βασίζεται κυρίως στον άμεσο χειρισμό. Οι χρήστες χρησιμοποιούν κινήσεις αφής, όπως σύρσιμο, πάτημα και μεγέθυνση, για τον χειρισμό αντικειμένων στην οθόνη. Εκτός από το πληκτρολόγιο, υπάρχει ένα προσαρμοζόμενο πληκτρολόγιο στην οθόνη για την εισαγωγή κειμένου. Σχεδιασμένο κυρίως για κινητά με οθόνη αφής, το Android εξαπλώθηκε σε διάφορες μορφές όπως το Android Wear για έξυπνα ρολόγια, το Android TV για τηλεοπτικές πλατφόρμες και το Android Auto για συστήματα ψυχαγωγίας στο αυτοκίνητο. Η Google συνεχίζει να βελτιώνει το Android, προσθέτοντας **ασφάλεια και καλύτερη εμπειρία χρήστη**. Η **ευελιξία και ο ανοιχτός χαρακτήρας** του Android, μαζί με την ποικιλία συσκευών από διάφορες εταιρείες, βοήθησαν το Android να γίνει το κυρίαρχο λειτουργικό σύστημα κινητών παγκοσμίως.

Με την πάροδο των χρόνων, το Google Play Store διευρύνθηκε και πρόσθεσε πολλές κατηγορίες εφαρμογών. Αρχικά, περιλάμβανε κυρίως εφαρμογές που βοηθούσαν στην παραγωγικότητα και εφαρμογές εργαλεία, αλλά γρήγορα διέυρυνε το φάσμα των κατηγοριών της. Ορισμένες από τις κατηγορίες αναγράφονται παρακάτω.

Τέχνη και Σχεδιασμός: Μπλοκ ζωγραφικής, εργαλεία ζωγραφικής, εργαλεία σχεδιασμού, βιβλία ζωγραφικής

Αυτοκίνητο και Οχήματα: Αγορά αυτοκινήτου, ασφάλεια αυτοκινήτου, σύγκριση τιμών αυτοκινήτου, οδική ασφάλεια, έλεγχος και νέα αυτοκινήτου

Ομορφιά: Μαθήματα μακιγιάζ, εργαλεία μακιγιάζ, χτένισμα μαλλιών, αγορές ομορφιάς, προσομοιωτής μακιγιάζ

Βιβλία και Παραπομπές: Αναγνώστες βιβλίων, βιβλία παραπομπής, διδακτικά βιβλία, λεξικά, λεξικό συνωνύμων, εγκυκλοπαίδεια

Επιχειρήσεις: Εκδότης/Αναγνώστης εγγράφων, εντοπισμός πακέτων, εργασία εξ αποστάσεως, διαχείριση μηνυμάτων ηλεκτρονικού ταχυδρομείου, αναζήτηση εργασίας

Επικοινωνίες: Μηνύματα, συνομιλίες, βιβλία διευθύνσεων, προγράμματα περιήγησης, διαχείριση κλήσεων

Εκπαίδευση: Προετοιμασία εξετάσεων, σεμινάρια, λεξιλόγιο, εκπαιδευτικά παιχνίδια, εκμάθηση γλωσσών

Διασκέδαση: Μετάδοση βίντεο, ταινίες, τηλεόραση, διαδραστική διασκέδαση

Εκδηλώσεις: Εισιτήρια συναυλιών, εισιτήρια αθλητικών εκδηλώσεων, μεταπώληση εισιτηρίων, εισιτήρια ταινιών

Χρηματοοικονομικά: Τραπεζικά, πληρωμές, εύρεση αυτόματων μηχανημάτων ανάληψης, οικονομικά νέα, ασφάλειες, φορολογία, υπολογισμός φιλοδωρημάτων

Φαγητό και ποτό: Συνταγές, εστιατόρια, οδηγοί φαγητών, γευστιγνωσία κρασιών, συνταγές ποτών

Υγεία και Φυσική κατάσταση: προσωπική φυσική κατάσταση, παρακολούθηση άσκησης, διαίτα και διατροφικές συμβουλές, υγεία και ασφάλεια

Οικία και Σπίτι: Αναζήτηση οικίας, βελτίωση οικίας, εσωτερική διακόσμηση, υποθήκες, ακίνητη περιουσία

Χάρτες και Πλοήγηση: Εργαλεία πλοήγησης, GPS, χαρτογράφηση, μεταφορές, μέσα μαζικής μεταφοράς

Ιατρικά: Κλινικές και φαρμακευτικές αναφορές, αριθμομηχανές, εγχειρίδια για τους παρόχους υγείας, ιατρικές εφημερίδες και νέα

Μουσική και Ήχος: Μουσικές υπηρεσίες, ραδιόφωνο, συσκευές αναπαραγωγής μουσικής

Νέα και Περιοδικά: Εφημερίδες, ιστοσελίδες συγκέντρωση δεδομένων, περιοδικά

Ανατροφή: Εγκυμοσύνη, φροντίδα και παρακολούθηση νεογνού, κηδεμονία

Εξατομίκευση: Ταπετσαρίες, ζωντανές ταπετσαρίες, αρχική οθόνη, οθόνη κλειδώματος, ήχοι κλήσης

Φωτογραφία: Φωτογραφική μηχανή, εργαλεία επεξεργασίας φωτογραφιών, διαχείριση και διαμοιρασμός φωτογραφιών

Παραγωγικότητα: Σημειωματάριο, λίστες, πληκτρολόγιο, εκτύπωση, ημερολόγια, αντίγραφο ασφαλείας, αριθμομηχανή, μετατροπή

Αγορές: Διαδικτυακές αγορές, δημοπρασίες, κουπόνια, σύγκριση τιμών, λίστες σούπερ-μάρκετ, έλεγχος προϊόντων

Κοινωνικά: Μέσα κοινωνικής δικτύωσης

Αθλήματα: Αθλητικά νέα και σχολιασμός, παρακολούθηση σκορ, διαχείριση φανταστικής ομάδας, κάλυψη αγώνα

Εργαλεία: Εργαλεία για συσκευές Android

Ταξίδια και Τοπικά: Εργαλεία κράτησης ταξιδιού, κοινή χρήση οχήματος, οδηγοί πόλεων, πληροφορίες για τοπικές επιχειρήσεις, εργαλεία διαχείρισης ταξιδιού, κράτηση ξενάγησης

Συσκευές αναπαραγωγής βίντεο και Επεξεργασία: Συσκευές αναπαραγωγής βίντεο, επεξεργασία βίντεο, αποθήκευση μέσων

Καιρός: Δελτία καιρού [20]

Με διαρκείς αναβαθμίσεις, βελτιώσεις στην ασφάλεια και την αφοσίωσή της στην καινοτομία, η Google καταφέρνει να διατηρεί το Android στην κορυφή του παγκόσμιου χάρτη των κινητών λειτουργικών συστημάτων. Η συνεχής προσπάθεια για βελτίωση και πρόσβαση σε νέες δυνατότητες καθιστά το

Android όχι απλώς ένα λειτουργικό σύστημα, αλλά έναν οργανισμό που αναπτύσσεται μαζί με τις ανάγκες και τις προσδοκίες των χρηστών του.

2.1.2 Περιβάλλον Android Studio

Το Android SDK (Software Development Kit) παρέχει τα απαραίτητα εργαλεία και βιβλιοθήκες για τη δημιουργία εφαρμογών. Οι εφαρμογές Android διαχειρίζονται τον κύκλο ζωής των Activities, το UI γίνεται με το XML και τα στοιχεία του Material Design συχνά χρησιμοποιούνται για τη δημιουργία μοντέρνων και φιλικών προς τον χρήστη διεπαφών.

Η ανάπτυξη μιας εφαρμογής για Android μπορεί να χρησιμοποιεί την σουίτα βιβλιοθηκών [Jetpack](#), της [Kotlin](#), του Android Studio (του επίσημου περιβάλλοντος ανάπτυξης) και πολλών άλλων βιβλιοθηκών και εργαλείων που καθιστούν τη διαδικασία ανάπτυξης εφαρμογών Android συναρπαστική και προσαρμοσμένη. Η **Kotlin** είναι μια γλώσσα προγραμματισμού που χρησιμοποιείται ευρέως για την ανάπτυξη εφαρμογών Android. Είναι συμβατή με την Java και προσφέρει πολλά χαρακτηριστικά, όπως λιγότερο κώδικα, ασφαλή τύπους και υποστήριξη λειτουργιών όπως οι lambda. Το **Jetpack Compose** είναι ένα σύγχρονο σετ εργαλείων γραφικών χρήστη για τον κατασκευή δυναμικών και σύγχρονων διεπαφών χρήστη σε εφαρμογές Android, χρησιμοποιώντας τη γλώσσα Kotlin. Επιτρέπει τη δημιουργία UI με λιγότερο κώδικα και προσφέρει αναγνώριση των αλλαγών αυτόματα.

Το **Activity Lifecycle** αναφέρεται στη σειρά των καταστάσεων που ένα Activity περνά κατά τη διάρκεια του κύκλου ζωής του σε μια εφαρμογή Android. Αυτό το σύνολο από καταστάσεις είναι κρίσιμο για την κατανόηση της αλληλεπίδρασης μεταξύ του χρήστη και της εφαρμογής. Κάθε Activity ξεκινά από την κατάσταση "Created" όταν δημιουργείται, περνά σε καταστάσεις όπως "Started" όταν εμφανίζεται στην οθόνη, "Resumed" όταν γίνεται ενεργή για τον χρήστη, "Paused" όταν χάνει προσωρινά την εστίαση, "Stopped" όταν δεν είναι πλέον ορατό και, τελικά, "Destroyed" όταν καταστρέφεται. Κατά τη διάρκεια της μετάβασης μεταξύ αυτών των καταστάσεων, το Activity εκτελεί διάφορες μεθόδους όπως onCreate(), onStart(), onResume(), onPause(), onStop() και onDestroy(). Κατανοώντας τον κύκλο ζωής, οι προγραμματιστές μπορούν να διαχειριστούν σωστά τις διάφορες καταστάσεις για να διατηρήσουν τη σταθερότητα και την αποτελεσματικότητα των εφαρμογών τους σε ένα ποικίλο περιβάλλον χρήστη.

Η αρχιτεκτονική μιας Android εφαρμογής συχνά οργανώνεται σε διάφορα **layers (επίπεδα)** για να επιτευχθεί οργανωμένη διαχείριση του κώδικα και ευκολία συντήρησης. Κάθε επίπεδο εκτελεί συγκεκριμένες λειτουργίες και έχει συγκεκριμένες ευθύνες. Εδώ αναφέρονται συνοπτικά τα βασικά επίπεδα αρχιτεκτονικής εφαρμογής:

Presentation Layer (Επίπεδο Παρουσίασης): Αυτό το επίπεδο αναλαμβάνει την ευθύνη για την αλληλεπίδραση με τον χρήστη. Περιλαμβάνει το UI (User Interface), το οποίο μπορεί να αποτελείται

από δραστηριότητες (activities) και fragment, καθώς και τη λογική που αφορά την παρουσίαση των δεδομένων.

Domain Layer (Επίπεδο Πεδίου): Στο επίπεδο αυτό βρίσκεται η επιχειρηματική λογική (business logic) της εφαρμογής. Εδώ ορίζονται οι κανόνες και οι διαδικασίες που σχετίζονται άμεσα με τη λειτουργία της εφαρμογής, ανεξάρτητα από το UI.

Data Layer (Επίπεδο Δεδομένων): Το επίπεδο αυτό αναλαμβάνει την αλληλεπίδραση με τη βάση δεδομένων, τα δίκτυα και άλλες πηγές δεδομένων. Περιλαμβάνει τον κώδικα που χειρίζεται τις επικοινωνίες με τη μόνιμη αποθήκευση δεδομένων, όπως οι SQL εντολές για τις βάσεις δεδομένων.

Η χρήση αυτών των επιπέδων συνήθως ακολουθεί την αρχή του "**Separation of Concerns**" και προωθεί την επαναχρησιμοποίηση και τη συντήρηση του κώδικα. Επιπλέον, συχνά χρησιμοποιούνται αρχιτεκτονικά πρότυπα όπως το MVVM (Model-View-ViewModel) [21] και το Clean Architecture για να επιτύχουν μεγαλύτερη διαχωριστικότητα και οργάνωση του κώδικα.

Η **Dependency Injection** (DI) είναι μια σχεδιαστική αρχή στην ανάπτυξη λογισμικού που αφορά την εισαγωγή (injection) των εξαρτήσεων (dependencies) μιας κλάσης από έναν εξωτερικό προμηθευτή, αντί να τις δημιουργεί η ίδια η κλάση. Σκοπός της Dependency Injection είναι να επιτρέψει τον αποκεντρωμένο έλεγχο των εξαρτήσεων και να καθιστά τον κώδικα πιο ευανάγνωστο, ευκολοσυντηρήσιμο και ευδιάκριτο.

Ο τρόπος με τον οποίο επιτυγχάνεται η Dependency Injection είναι μέσω τριών βασικών τεχνικών:

- **Constructor Injection:** Οι εξαρτήσεις περνιούνται στον κατασκευαστή (constructor) μιας κλάσης κατά τη δημιουργία της. Αυτό εξασφαλίζει ότι οι απαραίτητες εξαρτήσεις είναι διαθέσιμες κατά τη δημιουργία του αντικειμένου.
- **Setter Injection:** Οι εξαρτήσεις περνιούνται μέσω setter μεθόδων της κλάσης, καθιστώντας δυνατή την αλλαγή τους ακόμα και μετά τη δημιουργία του αντικειμένου.
- **Method Injection:** Οι εξαρτήσεις περνιούνται ως παράμετροι σε μεθόδους της κλάσης. Αυτό επιτρέπει την εισαγωγή των εξαρτήσεων κατά την εκτέλεση συγκεκριμένων λειτουργιών.

Η χρήση της Dependency Injection συμβάλλει στον καθαρό και ευανάγνωστο κώδικα, ενισχύει τη διαχωριστικότητα μεταξύ των συστατικών της εφαρμογής και επιτρέπει εύκολο έλεγχο των εξαρτήσεων, κάνοντας τον κώδικα ευέλικτο και εύκολο στη συντήρηση. Συχνά, οι βιβλιοθήκες όπως η Dagger και η Guice χρησιμοποιούνται για την αυτοματοποίηση της διαδικασίας της Dependency Injection στο πλαίσιο εφαρμογών Android. [22]

Στοιχεία-κλειδιά μιας android-based εφαρμογής:

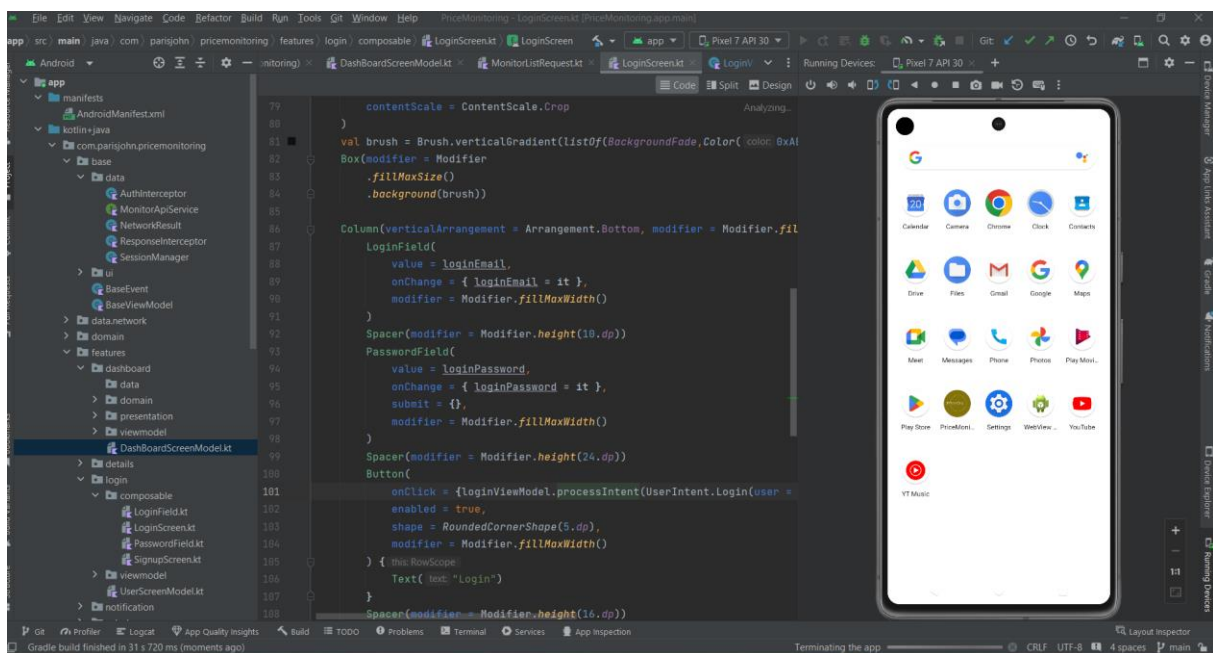
Κεφάλαιο 2

SharedPreferences: Το SharedPreferences είναι ένα μηχανισμός αποθήκευσης δεδομένων σε μορφή ζεύγους "κλειδί-τιμή". Χρησιμοποιείται για την αποθήκευση μικρών ρυθμίσεων ή πληροφοριών που πρέπει να διατηρηθούν ακόμη και αφού η εφαρμογή έχει κλείσει.

ViewModel: Το ViewModel είναι ένα στοιχείο της αρχιτεκτονικής Jetpack που χρησιμοποιείται για τη διατήρηση δεδομένων κατά τη διάρκεια του κύκλου ζωής μιας εφαρμογής Android. Βοηθά στην αποφυγή απώλειας δεδομένων κατά τη διάρκεια αλλαγών κατάστασης του UI, όπως περιστροφή οθόνης.

Intents: Τα Intents είναι μηνύματα που χρησιμοποιούνται για την εκκίνηση διαφόρων συνιστωσών της εφαρμογής, όπως Activities, Services και Broadcast Receivers.

Permissions: Οι Permissions επιτρέπουν ή απαγορεύουν στην εφαρμογή να χρησιμοποιεί συγκεκριμένες λειτουργίες του συστήματος, όπως πρόσβαση σε κάμερα ή το δίκτυο



Σχήμα 2.2: Περιβάλλον ανάπτυξης Android Studio

2.1.3 Kotlin: Γλώσσα Προγραμματισμού

Η **Kotlin** αναδεικνύεται ως μια καινούργια γλώσσα προγραμματισμού ανοιχτού κώδικα, που έχει δημιουργηθεί από την JetBrains, την ομάδα πίσω από το IntelliJ IDEA. Με την επιλογή στατικής πληκτρολόγησης, οι τύποι μεταβλητών δηλώνονται εκ των προτέρων κατά τη διάρκεια της μεταγλώττισης, προσφέροντας έναν σαφή τρόπο προγραμματισμού. Η Kotlin, αν και αντικειμενοστραφής και συμβατή με την Java, ξεχωρίζει για τη δυνατότητά της να παρέχει διατηρητέο κώδικα και να επικεντρώνεται στην ευανάγνωστη σύνταξη και τη σαφήνεια. Σε σύγκριση με την Java,

όπου οι γραμμές κώδικα αυξάνονται λογαριθμικά με το μέγεθος του έργου, **η Kotlin προσφέρει αποτελεσματικότερο προγραμματισμό και εξοικονομεί χρόνο**. Η γλώσσα προγραμματισμού Kotlin έχει ενσωματωθεί ως η κύρια γλώσσα ανάπτυξης των android εφαρμογών, αν και η Java παραμένει υποστηριζόμενη.

2.1.4 Jetpack Compose: UI Library

Το **Jetpack Compose** αποτελεί μια σημαντική καινοτομία στον τομέα της ανάπτυξης εφαρμογών για Android, ως μια ισχυρή βιβλιοθήκη UI που απλοποιεί και ενισχύει τη δημιουργία διεπαφών χρήστη. Αναπτύχθηκε από τη Google και φέρνει μια νέα πνοή στο παραδοσιακό μοντέλο ανάπτυξης UI για Android, προσφέροντας έναν πιο φιλικό, προς τον χρήστη, τρόπο. Με το Jetpack Compose, η διαδικασία σχεδίασης λειτουργικών και δυναμικών διεπαφών επιτρέπουν στους προγραμματιστές να περιγράφουν τα στοιχεία της διεπαφής με έναν πιο φυσικό και συνοπτικό τρόπο. Ένα από τα πιο εντυπωσιακά χαρακτηριστικά του Jetpack Compose είναι η **άμεση ένταξη κώδικα και σχεδίασης**, δίνοντας στους προγραμματιστές τη δυνατότητα να περιγράφουν απευθείας στον κώδικά τους την εμφάνιση και τη συμπεριφορά της διεπαφής. Αυτό όχι μόνο επιταχύνει τη διαδικασία ανάπτυξης, αλλά ενισχύει και τη συνεργασία μεταξύ σχεδιαστών και προγραμματιστών, καλλιεργώντας ένα δημιουργικό περιβάλλον.

2.2 Trello: Οργάνωση Διεργασιών

Ακολουθώντας τις πρακτικές που συναντάμε σε πραγματικά επαγγελματικά περιβάλλοντα ανάπτυξης εφαρμογών χρησιμοποιήθηκε το μοντέλο διαχωρισμού και διαχείρισης, επιμέρους, εργασιών. Αυτή η λειτουργία επιτελέστηκε με την χρήση της ιστοσελίδας Trello.

Το **Trello.com** είναι ένα διαδεδομένο εργαλείο διαχείρισης εργασιών στον χώρο της ανάπτυξης εφαρμογών για κινητά τηλέφωνα και στον χώρο της δημιουργίας εφαρμογών γενικότερα. Η χρησιμότητα του, πέρα από τα πρακτικά εργαλεία, εντοπίζεται και στο γεγονός της εύκολης συνεργασίας πολλαπλών χρηστών για το ίδιο Έργο. Για την οργάνωση των εργασιών χρησιμοποιούνται πίνακες και λίστες. Οι πίνακες αντιπροσωπεύουν ένα “task”, οι λίστες διαιρούν τις φάσεις του και οι κάρτες περιγράφουν συγκεκριμένες εργασίες. Για όλα αυτά τα εργαλεία υπάρχουν οι επιλογές “To Do (προς υλοποίηση)”, “Currently Doing (Υλοποιείται)” και “Done (Υλοποιήθηκε)”. Αυτό το οπτικό σύστημα βελτιώνει τον συντονισμό της ομάδας, την παρακολούθηση της προόδου και την προτεραιότητα των εργασιών καθ' όλη τη διάρκεια του κύκλου ζωής της ανάπτυξης της εφαρμογής. [23]

Στην περίπτωση του PRIMON αξιοποιήθηκε τη δυνατότητα που δίνει το Trello να μορφοποιείς δικούς σου πίνακες με βάσει τις ανάγκες του εκάστοτε χρήστη ή του εκάστοτε project. Όπως φαίνεται και στο Σχήμα 3.1 δημιουργήθηκαν τρεις πίνακες εργασιών κάτω από το project PRIMON. Τον πίνακα To Do, για τα tasks που έπρεπε να πραγματοποιηθούν, τον πίνακα Done, στον οποίο εκχωρήθηκε κάθε task που ολοκληρωνόταν και τέλος τον πίνακα Cannot Be Done, για κάθε task το οποίο δεν μπορεί να

Κεφάλαιο 2

πραγματοποιηθεί στο δεδομένο πλαίσιο, είτε λόγω περιορισμών του Back-End είτε λόγω περιορισμών της δεδομένης υποδομής του Server.

Άλλο ένα εργαλείο του Trello που χρησιμοποιήθηκε κατά την πραγματοποίηση του project είναι το **color-coding**. Δηλαδή, ομαδοποιήθηκε το είδος των εργασιών για κάθε λειτουργία με βάση το χρώμα. Με αυτόν τον τρόπο όλα τα tasks εκχωρήθηκαν σε μία από τις παρακάτω κατηγορίες:

Γενικές εργασίες της Εφαρμογής: Project planning, android studio setup, Git Hub repository creation, acquire app functional requirements, communication with back-end developers

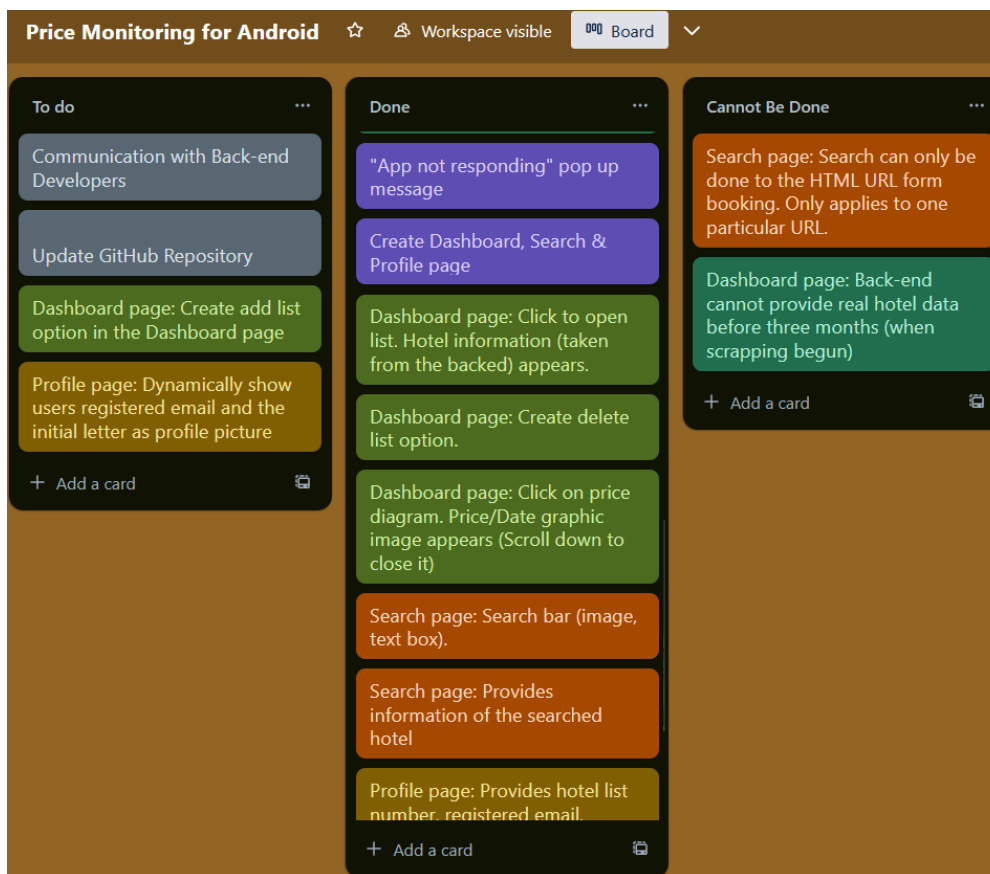
Υποδομή Android: Create UI infrastructure, connect with the backend, app starting page, app "not responding" pop up message, create splash

Σελίδα Sign Up: Run checks for email and password registration in the sign up page, make redirection to one time password page (image, six string boxes), check Login credentials

Σελίδα Dashboard: Click to open monitor list, create, edit, delete list option, click on price diagram. Price/Date graphic image appears

Σελίδα Search: Search bar (image, text box), provides information of the searched hotel on click, Search can only be done to the HTML URL form booking

Σελίδα Profile: Dynamically show users registered email and the initial letter as profile picture, Provides hotel list number, registered email, subscription option, about, share and log out options, on click log out button drives you back to initial login/sign up page, List is dynamically changed according to list number



Σχήμα 2.3: Στιγμιότυπο και τμήμα των πινάκων εργασιών για τη Hotel Price Monitoring android εφαρμογής

2.3 Agile πλάνο της υλοποίησης της εφαρμογής

Ακολουθώντας τι τεχνικές που χρησιμοποιούνται στην αγορά εργασίας για την διαδικασία της ανάπτυξης της εφαρμογής και για τον προγραμματισμό της χρησιμοποιήθηκε η μεθοδολογία της agile και των sprints [24]. Τα **sprints** και η **μεθοδολογία Agile** αντιπροσωπεύουν έναν ευέλικτο και συνεργατικό τρόπο ανάπτυξης λογισμικού που έχει επικεντρωθεί στην παραγωγή αξίας για τον πελάτη. Τα sprints είναι βασικό στοιχείο της μεθοδολογίας Scrum που ανήκει στην οικογένεια του Agile. Κάθε sprint είναι μια περίοδος ανάπτυξης, συνήθως διάρκειας 2-4 εβδομάδων, κατά την οποία οι αναπτυσσόμενες λειτουργίες παρουσιάζονται σε συχνές αναθεωρήσεις, δίνοντας τη δυνατότητα στην ομάδα να προσαρμόζει το έργο της.

Η μεθοδολογία Agile επικεντρώνεται στη συνεχή παράδοση αξίας μέσω της συνεργασίας και της προσαρμογής στις μεταβαλλόμενες απαιτήσεις. Η ουσία του Agile είναι η ικανότητα να ανταποκρίνεται γρήγορα στις αλλαγές και να παρέχετε συνεχώς λειτουργικό λογισμικό. Αντί για μακροχρόνιες σχεδιαστικές φάσεις, η Agile υιοθετεί σύντομες αναπτυξιακές περιόδους, δίνοντας έμφαση στον συχνό προγραμματισμό, την αναθεώρηση και την ανοικτή επικοινωνία εντός της ομάδας ανάπτυξης. Με αυτόν τον τρόπο, η Agile διασφαλίζει την αποτελεσματική παράδοση προϊόντος με βάση τις προτεραιότητες του πελάτη, ενισχύοντας τη συνεργασία και την ευελιξία σε έναν δυναμικό περιβάλλον ανάπτυξης.

Sprints	Features
Phase 1	<ul style="list-style-type: none"> • Login • Signup • Verification Email
Phase 2	<ul style="list-style-type: none"> • Dashboard • Monitor Lists Screen • Search Screen • Profile Screen
Phase 3	<ul style="list-style-type: none"> • Graphs • Monitor List details • Push notifications
Phase 4	<ul style="list-style-type: none"> • Overview the app and check every functionality

Πίνακας 1: Phases of PRIMON development

2.4 GitHub: Οργάνωση κώδικα

Το **GitHub** αποτελεί τον πλέον διαδεδομένο τρόπο για χώρο εναπόθεσης κώδικα (code repository) καθώς και για την επίτευξη ομαδικού προγραμματισμού. Η συγκεκριμένη υπηρεσία είναι ανοιχτή και διαθέσιμη σε όλους, μόνο με την απλή διαδικασία δημιουργίας ενός λογαριασμού. Τα **κύρια εργαλεία αυτής της τεχνολογίας** είναι τα ακόλουθα.

Έλεγχος Εκδόσεων: Το GitHub χρησιμοποιεί το Git, ένα κατακευματισμένο σύστημα ελέγχου εκδόσεων, για να παρακολουθεί τις αλλαγές στον πηγαίο κώδικα κατά τη διάρκεια της ανάπτυξης λογισμικού. Αυτό επιτρέπει τη συνεργασία πολλαπλών προγραμματιστών χωρίς συγκρούσεις και επιτρέπει εύκολη επαναφορά σε προηγούμενες εκδόσεις.

Repositories: Ένα repository περιλαμβάνει όλα τα αρχεία του εκάστοτε project, συμπεριλαμβανομένου του πηγαίου κώδικα. Τα repositories μπορούν να είναι δημόσια ή ιδιωτικά, ανάλογα με τις ρυθμίσεις του χρήστη.

Κλαδιά (Branches): Οι χρήστες δημιουργούν τα κλαδιά ώστε να εργαστούν σε συγκεκριμένα χαρακτηριστικά ή διορθώσεις χωρίς να επηρεάζουν τον κύριο κώδικα. Τα κλαδιά μπορούν να συγχωνευτούν πίσω στον κύριο branch όταν οι αλλαγές έχουν εγκριθεί.

Pull Requests: Όταν ένας προγραμματιστής ολοκληρώνει μια εργασία σε ένα κλαδί, μπορεί να ανοίξει ένα pull request για να προτείνει τις αλλαγές στο κύριο branch. Αυτό εκκινεί μία διαδικασία επιθεώρησης, όπου τα μέλη της ομάδας μπορούν να παρέχουν σχόλια πριν την συγχώνευση.

Συνεργασία: Το GitHub διευκολύνει τη συνεργασία μεταξύ των μελών μιας ομάδας μέσω των issues, τα οποία μπορούν να χρησιμοποιηθούν για την παρακολούθηση και συζήτηση task, και τα wikis, όπου αποθηκεύονται τα δεδομένα του project.

Αναθεώρηση Κώδικα: Το GitHub επιτρέπει την αναθεώρηση κώδικα με σχόλια σε ενσωματωμένες γραμμές κώδικα εντός των pull requests. Αυτό εξασφαλίζει την ποιότητα του κώδικα, τη συμμόρφωση με τα πρότυπα κωδικοποίησης και την κοινοποίηση γνώσεων ανάμεσα στα μέλη της ομάδας.

Συνεχής Ολοκλήρωση/Συνεχής Κατανομή (CI/CD): Το GitHub ενσωματώνεται με εργαλεία CI/CD, αυτοματοποιώντας τη διαδικασία δημιουργίας, δοκιμής και κατανομής των αλλαγών στον κώδικα βοηθώντας στον εντοπισμό προβλημάτων στα πρώιμα στάδια της διαδικασίας ανάπτυξης.

Κοινότητα και Ανοιχτός Κώδικας: Το GitHub αποτελεί ένα κέντρο project ανοιχτού κώδικα, προωθώντας τη συνεργασία και τη συνεισφορά από προγραμματιστές παγκοσμίως. Παρέχει μια πλατφόρμα για συζητήσεις, την παρακολούθηση θεμάτων και τη συνεισφορά σε έργα άλλων. [25]

Ο κώδικας της εφαρμογής PRIMON μπορεί να βρεθεί στο παρακάτω repository <https://github.com/ParisKaragiannopoulos/PRIMON>.

Επιπροσθέτως τα endpoints του backend, απαραίτητα για τη λειτουργία της εν λόγω εφαρμογή, και το απαραίτητο documentation προήλθαν από το repository του αντίστοιχου συμφοιτητή-developer Αρσάλν Καλαιτσίδα . Επίσης τα projects της εφαρμογής σε Web και IOS μορφή, αντίστοιχα, μπορούν να εντοπιστούν σε GitHub repositories.

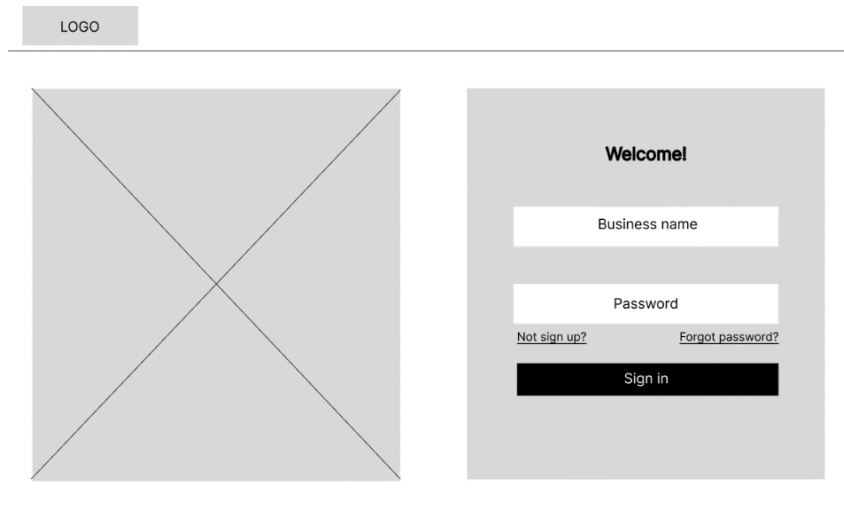
2.5 Figma: Εργαλείο Σχεδίασης και αλληλεπίδρασης

Άλλο ένα πολύ σημαντικό και χρηστικό εργαλείο που χρησιμοποιείται στον σχεδιασμό εφαρμογών είναι το cloud-based σχεδιαστικό εργαλείο **Figma**. Το Figma κατέχει κυρίαρχη θέση στη σχεδίαση διεπαφών με κύριο άξονα τις διαδικτυακές και mobile εφαρμογές. Και αυτό το εργαλείο, όπως και τα προηγούμενα δύο έχει διακριθεί ανάμεσα στους επαγγελματίες του χώρου για την επιλογή της συνεργατικής σχεδίασης που προσφέρει. Η ουσία του Figma βρίσκεται στην απλοποίηση της διαδικασίας της σχεδίασης τόσο σε ατομικό όσο και σε συλλογικό επίπεδο. [26]

Μέσα στο Figma ο χρήστης μπορεί να δημιουργήσει ένα template της εν δυνάμει εφαρμογής, με τη χρήση πρακτικών γραφικών για όλα τα components, όλες τις εικόνες και τα χρώματα που σκοπεύει να χρησιμοποιήσει, καθώς και να δοκιμάσει διαφορετικές επιλογές. Με την ολοκλήρωση του template αυτού μπορεί να μπει ένας άλλος χρήστης, με τα ανάλογα δικαιώματα, και να κάνει και αυτός αλλαγές, ή να χρησιμοποιήσει κάποια υπάρχοντα στοιχεία ή να χρησιμοποιήσει όλα το template όπως είναι. Στο σενάριο της παρούσας εργασίας αυτό μπορεί να βοηθήσει τόσο τον σχεδιαστή mobile εφαρμογών όσο και τον σχεδιαστή διαδικτυακών στο κομμάτι της επικοινωνίας.

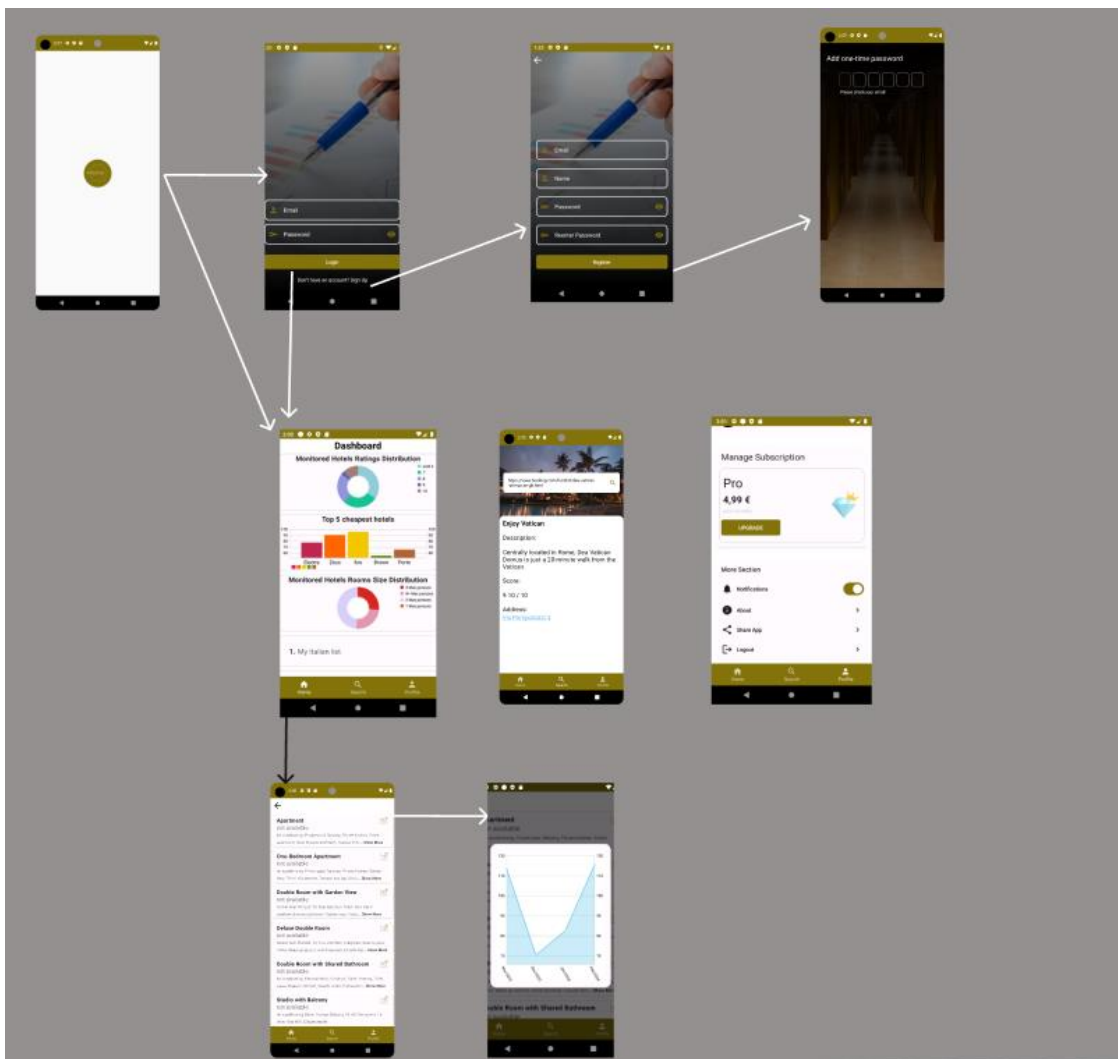
Κεφάλαιο 2

Στη συγκριμένη εργασία προτού ξεκινήσει η ανάπτυξη της android εφαρμογής δόθηκε από τη συμφοιτήτρια Κωνσταντίνα Πατελη το figma σχέδιο της διαδικτυακής Hotel Price Monitoring εφαρμογής. Μέσα από το Σχήμα 3.2 γίνεται κατανοητό ότι πρόκειται για ένα κλασικό template διαδικτυακής εφαρμογής οπότε και ακολούθησα έναν αντίστοιχο σχεδιασμό PRIMON λύση.



Σχήμα 2.4: Γράφημα Figma της Hotel Price Monitoring Web εφαρμογής

Στο Σχήμα 3.3 χρησιμοποιήθηκε το Figma, όχι απλά ως ένα σχεδιαστικό εργαλείο με τα components της εφαρμογής αλλά και ως ένα σχεδιάγραμμα για την πορεία του χρήστη μέσα από τα διάφορα παράθυρα της εφαρμογής, καθώς και τι επιλογές που εμφανίζονται.



Σχήμα 2.5: Γράφημα Figma της Hotel Price Monitoring android εφαρμογής

Μελλοντικά, ο οποιοσδήποτε θέλει να αναπτύξει περαιτέρω την Android εφαρμογή μπορεί πολύ απλά να χρησιμοποιήσει το παραπάνω Figma γράφημα αν θέλει να αλλάξει κάτι στα υπάρχοντα παράθυρα, διαδικασίες ή αν θέλει να δημιουργήσει καινούριες με βάση τα ήδη υπάρχοντα παράθυρα.

Κεφάλαιο 3ο: PRIMON- Hotel Price Monitoring Android App

3.1 Λειτουργία

Όπως αναφέρθηκε και στα προηγούμενα κεφάλαια αυτή η πτυχιακή αφορά την μελέτη και την ανάπτυξη μίας Real-time εφαρμογή για price monitoring ξενοδοχείων με επιπρόσθετες λειτουργίες για hotel analytics, για κινητές συσκευές που διαθέτουν android λειτουργικό, με έκδοση android 8.0 και πάνω. Οι λειτουργίες βασίζονται πάνω σε πραγματικά δεδομένα που συλλέγονται μέσω scrapping από σελίδες booking.

Ένας χρήστης μιας android συσκευής μπορεί μέσω της PRIMON να επιτελέσει τις ακόλουθες ενέργειες, λειτουργίες. Μπορεί να δημιουργήσει έναν λογαριασμό χρήστη μέσα στην εφαρμογή καθώς και να εισέλθει μέσα σε έναν υπάρχον λογαριασμό με τα σωστά credentials. Στη συνέχεια μπορεί να αναζητήσει δωμάτια μέσα από σελίδες booking, και να πληροφορηθεί για τη διαθεσιμότητα, το είδος καθώς και την τιμή των δωματίων που έχει αναζητήσει. Στη συνέχεια μπορεί να εκχωρήσει τα επιθυμητά δωμάτια σε μια λίστα παρακολούθησης. Επίσης του δίνεται η δυνατότητα να ονομάσει τις δικές του λίστες, να τις επεξεργαστεί, να διαγράψει μία λίστα ή να διαγράψει δωμάτιο μέσα από αυτές καθώς και να δει τον μέσο όρο της τιμής του κάθε δωματίου. Επιπρόσθετα μέσα από τις λίστες που έχει δημιουργήσει ο χρήστης μπορεί να δει διάφορα στατιστικά, μέσα από ένα dashboard, όπως τα πέντε φθηνότερα ξενοδοχεία από τις επιλογές του, την κατανομή των κριτικών των ξενοδοχείων και την κατανομή των ειδών των επιλεγμένων δωματίων. Τέλος μπορεί να δει τον αριθμό των λιστών δωματίων που έχει δημιουργήσει, γενικές πληροφορίες για την εφαρμογή καθώς και να κάνει log out από τον λογαριασμό του.

Στη συνέχεια θα αναλυθούν τεχνικά τα χαρακτηριστικά των λειτουργιών αυτών με στιγμιότυπα κώδικα καθώς και στιγμιότυπα από τα παράθυρα διεπαφής.

3.2 Σύνδεση με Backend

Η σύνδεση του backend επιτυγχάνεται μέσω της βιβλιοθήκης retrofit η οποία μας δίνει τη δυνατότητα να συνδεόμαστε στο backend με απλό τρόπο. Δημιουργώντας την κλάση *MonitorApiService* (βλ. Σχήμα 3.1) προσθέτουμε όλα τα endpoints που μας χρειάζονται (βλ. Σχήματα 3.3 & 3.4). Για να επιτύχουμε μία κλήση έχει χρησιμοποιηθεί ένας *interceptor* ο οποίος κάθε φορά που γίνεται μία κλήση από το UI προσθέτει το token του χρήστη ώστε να ταυτοποιηθεί. Μόλις ληφθεί απάντηση από τον server ελέγχεται αντίστοιχα με τη βοήθεια ενός άλλο *interceptor* αν το http code είναι 401, το οποίο έχει δηλωθεί στον server ως λήξη του authentication, τότε η κρυπτογραφημένη μνήμη καθαρίζει το token και ξεκινάει απ' την αρχή ώστε ο χρήστης μετά τη splash screen να πρέπει να δει τη Log In σελίδα για να ξανασυνδεθεί και να πάρει καινούριο authentication token.

Όλες οι κλήσεις προς τον server γίνονται σε ένα background thread ώστε να μην επιβαρύνεται το main thread (UI thread) και να μην κολλάει η οθόνη. Με αυτόν τον τρόπο η απάντηση έρχεται ασύγχρονα και το UI ειδοποιείται μόνο όταν υπάρξει κάποιο αλλαγή στην κατάσταση του response, θετική ή αρνητική. Η ενημέρωση του UI γίνεται μέσω των λεγόμενων flows τα οποία είναι ένα stream δεδομένων με τα οποία μπορούν να σταλούν οι απαντήσεις του server και με ασφάλεια μπορεί να παρακολουθείται το flow από το UI.

Τα objects που χρειαζόμαστε για τη σύνδεση του backend τα δημιουργούμε να τη βοήθεια της hilt ώστε να δημιουργήσουμε μόνο ένα αντίγραφο του καθενός (singleton). Με αυτόν τον τρόπο επιτυγχάνουμε να είναι προσβάσιμα όλα τα object από τα endpoints της εφαρμογής κάνοντας έτσι πιο εύκολη την διαχείριση της αρχιτεκτονικής της εφαρμογή καθώς και των πόρων του κάθε κινητού (performance).

```

19  @Module
20  @InstallIn(SingletonComponent::class)
21  object ApiModule {
22      private const val BASE_URL = "http://195.251.123.174:8085/"
23
24      @Singleton
25      @Provides
26      fun provideAuthorizationInterceptor(@ApplicationContext context: Context): AuthInterceptor {
27          return AuthInterceptor(context)
28      }
29
30      @Singleton
31      @Provides
32      fun provideResponseInterceptor(@ApplicationContext context: Context): ResponseInterceptor {
33          return ResponseInterceptor(context)
34      }
35
36      @Singleton
37      @Provides
38      fun providesOkHttpClient(authInterceptor: AuthInterceptor, responseInterceptor : ResponseInterceptor): OkHttpClient
39          OkHttpClient
40          .Builder()
41          .addInterceptor(authInterceptor)
42          .addInterceptor(responseInterceptor)
43          .addInterceptor(HttpLoggingInterceptor().setLevel(HttpLoggingInterceptor.Level.BODY))
44          .build()

```

Σχήμα 3.1: Κλάση ApiModule

```
23 interface MonitorApiService {
24
25     @GET("/monitor_list/{id}")
26     suspend fun getMonitorListDetails(@Path("id") id: String): MonitorListResponse
27
28     @POST("customer/login")
29     suspend fun login(@Body userLoginRequest: UserLoginRequest): Response<String>
30
31     @POST("customer/register")
32     suspend fun register(@Body userRegisterRequest: UserRegisterRequest): ResponseBody
33
34     @POST("hotel/info")
35     suspend fun getHotelInfo(
36         @Body searchHotelRequest: SearchHotelRequest
37     ): HotelInfoResponse
38
39     @POST("monitor_list")
40     suspend fun postNewMonitorList(@Body monitorListRequest: MonitorListRequest): ResponseBody
41
42     @GET("monitor_list")
43     suspend fun getAllMonitorLists(): MonitorListsResponse
44
45     @DELETE("monitor_list/{id}")
46     suspend fun deleteMonitorList(@Path("id") id: Long): ResponseBody

```

Σχήμα 3.2: Στιγμιότυπο της κλάσης MonitorApiService με τα backend endpoints

GET: /monitor_list/{id}

Description: returns details of requested monitor list.

Authorized: customer.

Response Content-Type:application/json.

Query Parameters:

1. id - the id of the monitor list.

Example Response:

```
{
  "monitorListName": "MyOlympiadaApartments",
  "monitorListID": 1,
  "rooms": [
    {
      "roomID": 1,
      "description": "'not available'",
      "name": "'Apartment'",
      "attributes": [
        "'View'",
        "'Washing machine'",
        "'Free WiFi'",
        "'Balcony'",
        "'Private pool'",
        "'Kitchen'",
        "'Private kitchen'",
        "'Entire apartment'",
        "'Ensuite bathroom'",
        "'Air conditioning'",
        "'Flat-screen TV'",
        "'Terrace'",
        "'160 m²'"
      ]
    }
  ]
}
```

Responses:

1. 200(OK)

2. 401(Unauthorized) - user not authenticated.

3. 403(forbidden) - action not allowed by authenticated user.

Σχήμα 3.3: Monitor List Backend Endpoint

POST: /hotel/info

Description: Returns hotel information including its rooms. Authorized: customer. Request Content-Type: application/json.

Example Request:

```
{
  "link" : "https://test.com?somethingtesat=123"
}
```

Response Content-Type: application/json.

Example Response:

```
{
  "name": "TEST_NAME",
  "description": "TEST_DESCRIPTION",
  "location": {
    "address": "TEST_ADDRESS",
    "city": "TEST_CITY",
    "longitude": 10.5,
    "latitude": 10.5
  },
  "url": "http://test.com/",
  "hotelType": "HOTEL",
  "score": "5.60",
  "hotelID": 229,
  "rooms": [
    {
      "roomID": 455,
      "description": "TEST_DESCRIPTION_1",
      "name": "TEST_NAME_1",
      "attributes": [
        "TEST_ATTR_1",
        "TEST_ATTR_2"
      ]
    },
    {
      "roomID": 456,
      "description": "TEST_DESCRIPTION_2",
      "name": "TEST_NAME_2",
      "attributes": [
        "TEST_ATTR_3",
        "TEST_ATTR_1"
      ]
    }
  ]
}
```

Responses:

1. 200(OK)
2. 400(Bad Request)
3. 401(Unauthorized) - user not authenticated.
4. 403(forbidden) - action not allowed by authenticated user.
5. 503(Not Found) hotel info for the provided url not found and scrapping service is too busy to answer.

Σχήμα 3.4: Hotel Info Backend Endpoint

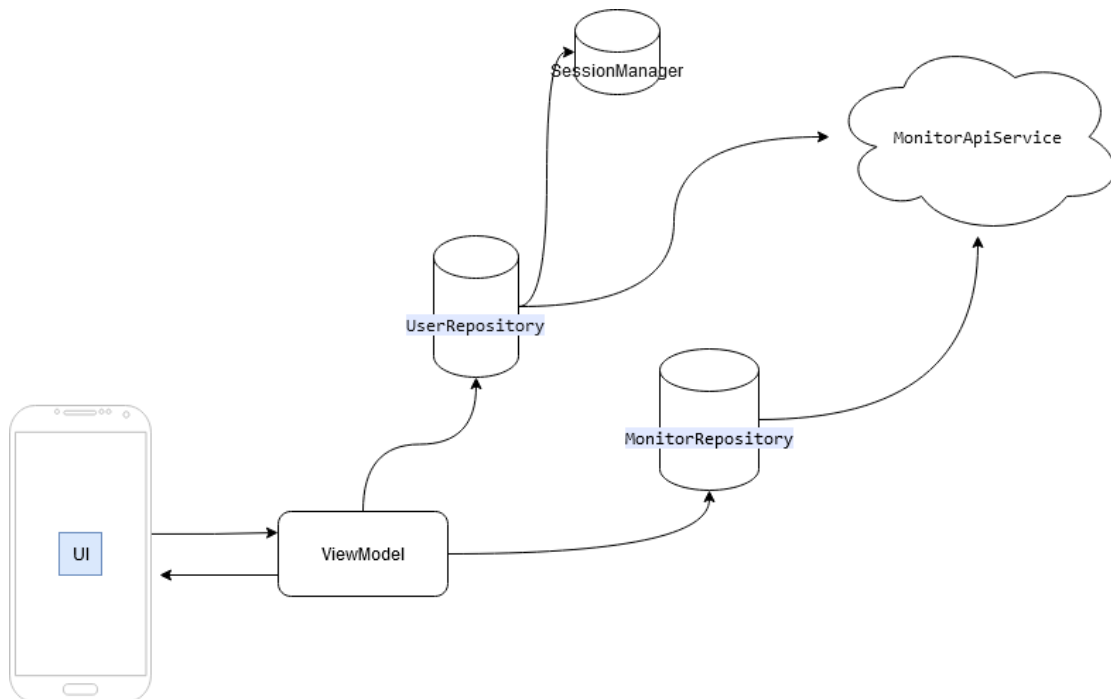
3.3 Ανάπτυξη Front-End

Η αρχιτεκτονική που ακολουθήθηκε και βασίζεται στις προτάσεις της Google αποτελείται από:

1. Το UI το οποίο περιέχει τα γραφικά και τίποτα από business logic, ώστε να είναι agnostic element.
2. Το View Model το οποίο αλληλοεπιδρά με το UI δεχόμενο events και στέλνοντας πίσω σε αυτό ροή δεδομένων (flow of data) για αναπαράσταση με γραφικό τρόπο. Για να βρει τις πληροφορίες που χρειάζεται τις ζητάει από τα repositories.

3. Τα repositories είναι ένα αυτόνομο layer της αρχιτεκτονικής το οποίο δεν έχει καμία σύνδεση με το UI, ώστε το ένα δεν γνωρίζει τι συμβαίνει στο άλλο και ο ρόλος τους είναι να φέρνουν πληροφορίες είτε από server είτε από τοπικές βάσεις δεδομένων ή μνήμες.
4. Τα services τα οποία μας δίνουν τη δυνατότητα αναπαράστασης των διασυνδέσεων με τις online και offline βάσεις δεδομένων.

Αναλυτικότερα, στη παρούσα πτυχιακή έχουμε δημιουργήσει **δύο ξεχωριστά repo** όπου το ένα αφορά αποκλειστικά κλήσεις που αφορούν τον χρήστη – **UserRepository** και το άλλο αφορά αποκλειστικά τις λίστες – **MonitorRepository**. Για την εσωτερική μνήμη έχει χρησιμοποιηθεί τον **SessionManager** ο οποίος κάνει Cache το token του χρήστη.



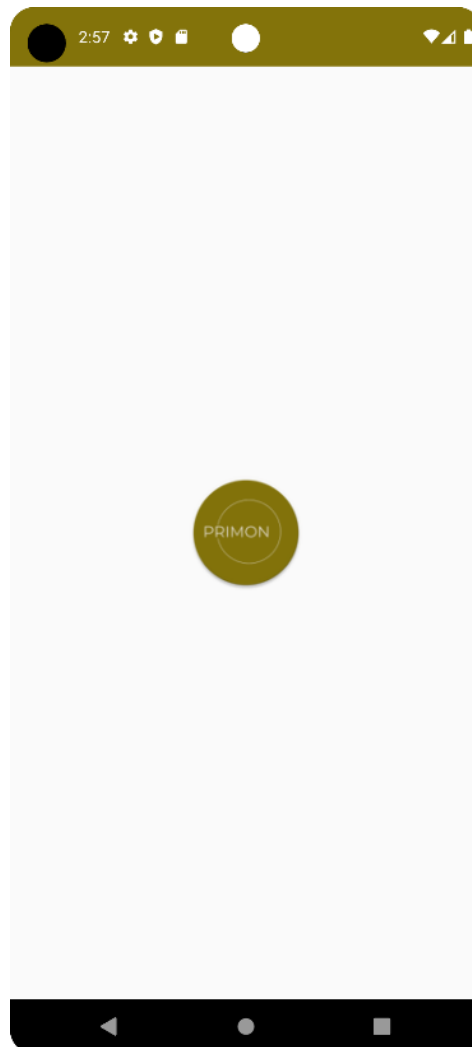
Σχήμα 3.5: Αρχιτεκτονική Εφαρμογής

Η πλοήγηση στις διάφορες οθόνες γίνεται με τη βοήθεια του *Native Navigation Controller* ο οποίος λειτουργεί ως συντονιστής και είναι η μόνη κλάση που μπορεί να δώσει οδηγία για τη μεταφορά από μία οθόνη σε μία άλλη. Με αυτόν τον τρόπο αποδεσμεύουμε το UI από το να γνωρίζει τι θα πρέπει να δείξει στον χρήστη σε περίπτωση κάποιες ενέργειας (action) όπως click σε ένα κουμπί.

3.3.1 Splash Screen

Με το άνοιγμά της εφαρμογής, πρώτη οθόνη που εμφανίζεται στον χρήστη είναι **η οθόνη Splash** (βλ. Σχήμα 3.5). Ο στόχος της οθόνης αυτής είναι ο έλεγχος ύπαρξης ή μη ενός ενεργού authentication token ώστε να κατευθύνουμε τον χρήστη στη συνέχεια της περίπτωση της ύπαρξης στη dashboard ή σε αντίθετη περίπτωσης της οθόνη log in. Για την επίτευξη αυτού του ελέγχου αποθηκεύουμε σε

κρυπτογραφημένη προσωρινή μνήμη το token, στην περίπτωση που ο χρήστης έχει ήδη συνδεθεί στην εφαρμογή. Στην περίπτωση που δεν έχει ξανασυνδεθεί το πεδίο αυτό στη μνήμη είναι κενό.



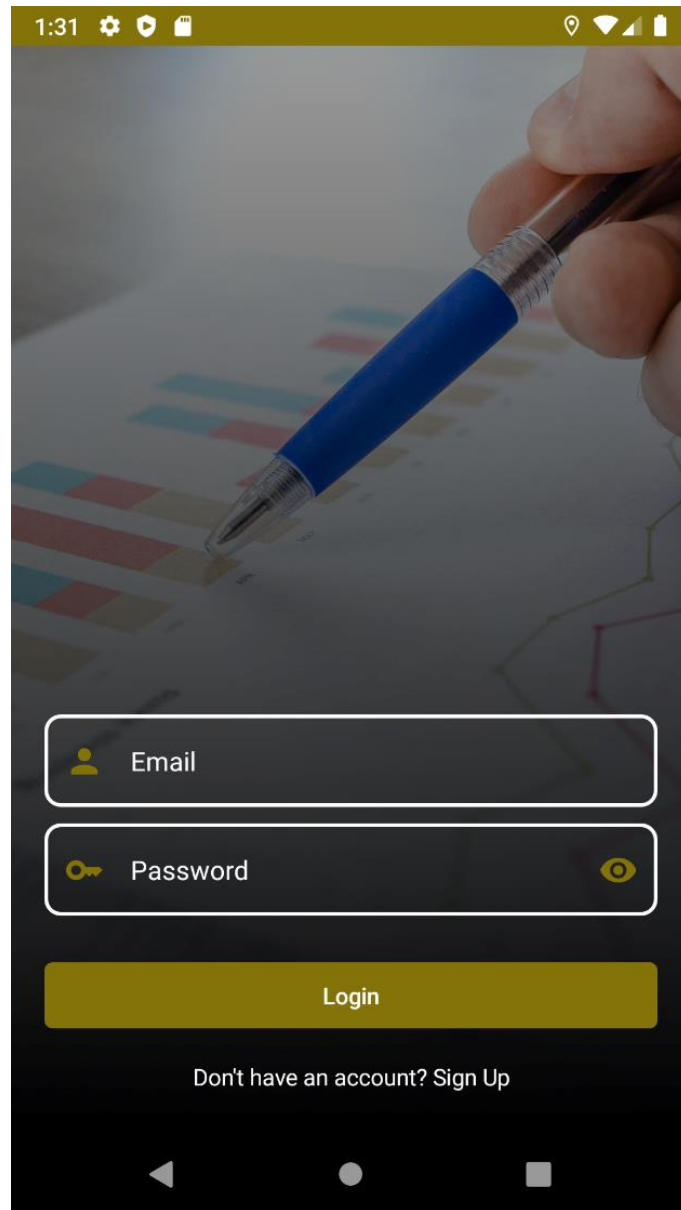
Σχήμα 3.6: Splash Screen

3.3.2 Σελίδα Login

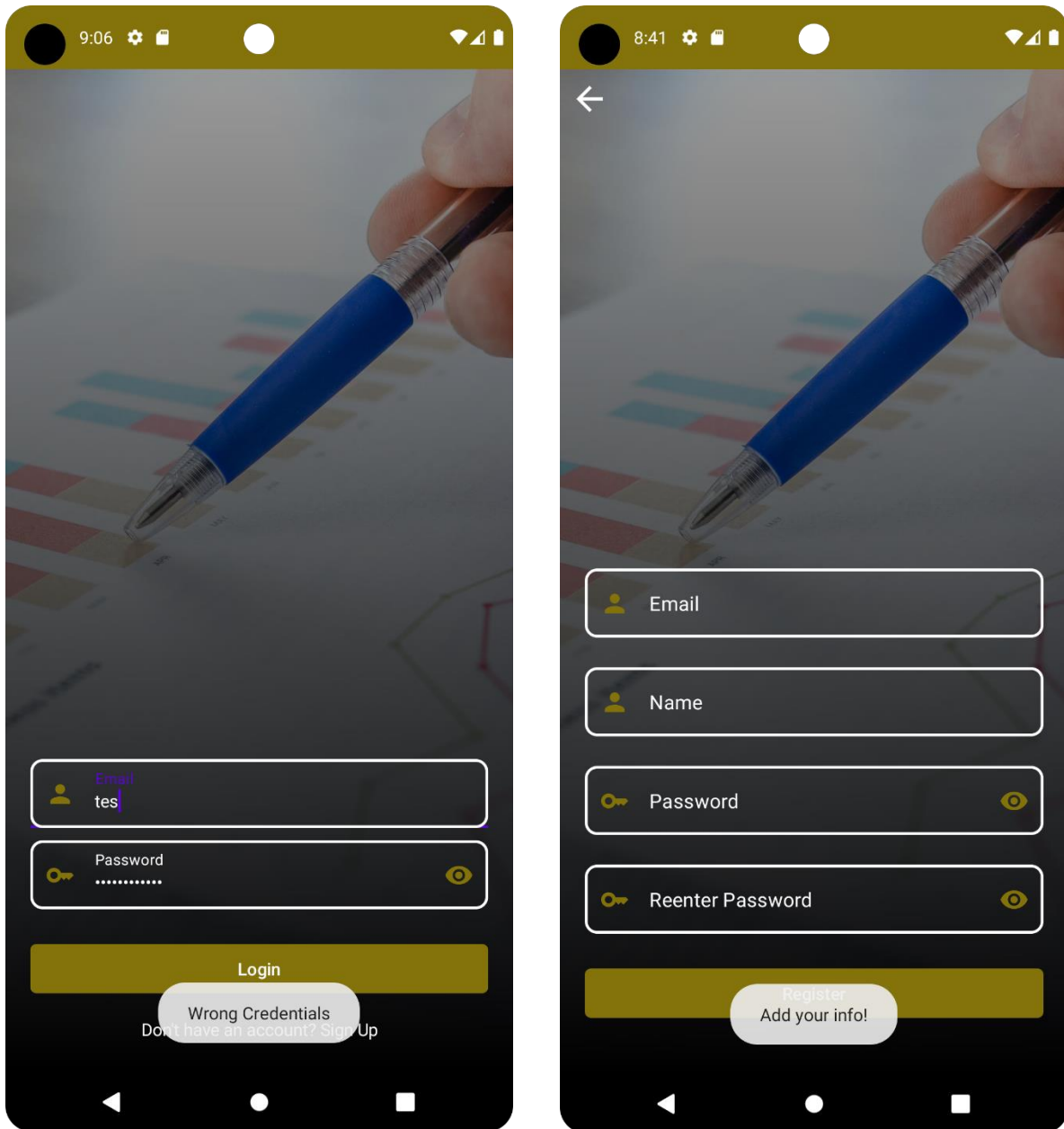
Η οθόνη του **Login** περιέχει δυο πεδία (text fields) ένα για το **email** και ένα για το **password** (βλ Σχήμα 3.6). Τα πεδία αυτά έχουν σχεδιαστεί με custom τρόπο και βρίσκονται στις κλάσεις LoginField & PasswordField αντίστοιχα. Στη περίπτωση που ο χρήστης δεν έχει λογαριασμό υπάρχει ένα clickable text το οποίο μόλις το πατήσει τον μεταφέρει στην οθόνη του Sign Up.

Στην περίπτωση που ο χρήστης έχει credential για να συνδεθεί σε ένα υπάρχον λογαριασμό η εφαρμογή επικοινωνεί με τον server ο οποίος ενημερώνει για τα αν τα credentials είναι ορθά. Στην περίπτωση που ισχύει αυτή η συνθήκη ο server στέλνει πίσω το authentication token το οποίο στη συνέχεια

αποθηκεύεται στη μνήμη, ενώ στη περίπτωση αποτυχίας η http code 401 ενημερώνει τον χρήστη για λανθασμένα credentials όπως φαίνεται και στο σχήμα 3.6.



Σχήμα 3.7: Login Page



Σχήμα 3.8: Wrong Credentials

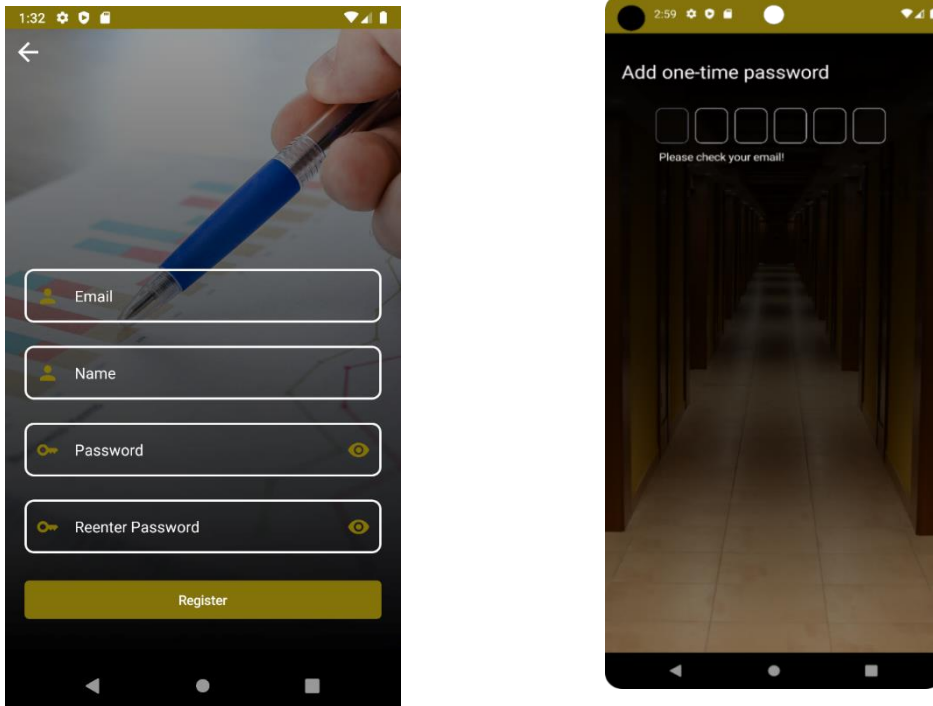
3.3.3 Σελίδα Sign Up

Προκειμένου ο χρήστης να δημιουργήσει λογαριασμό παραπέμπεται στη σελίδα του **sign up** (βλ. Σχήμα 3.7). Στην οθόνη του sign up έχουμε τέσσερά text παιδιά στα οποία καλείται ο χρήστης να συμπληρώσει το **email** του, το **ονοματεπώνυμο** του, το **κωδικό** που επιθυμεί και την **επιβεβαίωση** αυτού του κωδικού. Με βάση τα παραπάνω στοιχεία η εφαρμογή πραγματοποιεί του παρακάτω τοπικούς ελέγχους:

- Αν τα πεδία έχουν συμπληρωθεί όλα
- Αν το email έχει σωστή μορφή (πχ. xxx@xx.xx)
- Αν ο κωδικός είναι ίδιος με τον κωδικό επιβεβαίωσης

Με τους παραπάνω τοπικούς ελέγχους αποφεύγουμε περιττές κλήσεις προς τον server και δίνουμε τη δυνατότητα στον χρήστη μέσω διαφορετικών μηνυμάτων να διορθώσει τα λάθη της φόρμας.

Στον χρήστη μέσω αυτής της οθόνης δίνεται η δυνατότητα πατώντας το βελάκι πάνω αριστερά (βλ. Σχήμα 3.8) να γυρίσει στην οθόνη του Log In είτε να πατήσει το κουμπί register το οποίο θα τον οδηγήσει στον έλεγχο του email αφού έχει περάσει από όλους τους τοπικούς ελέγχους.



Σχήμα 3.9: Sign Up & OTP page

3.3.4 Σελίδα επιβεβαίωσης Email

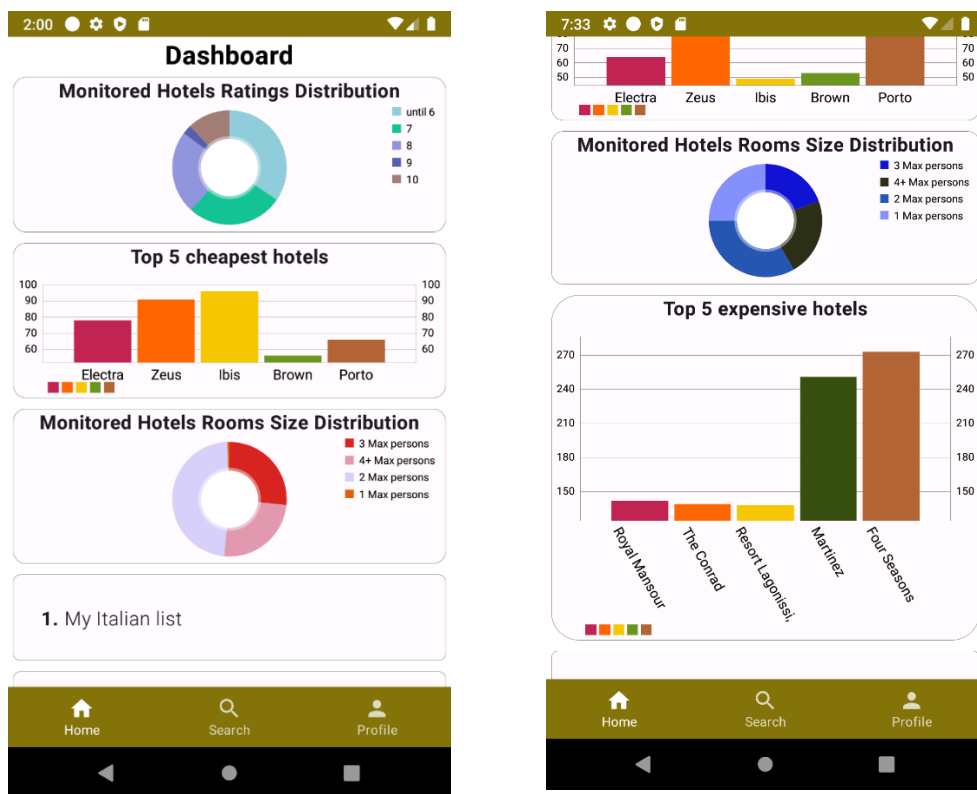
Με την είσοδο στην **OTP σελίδα** (βλ. Σχήμα 3.8) ο χρήστης θα πρέπει να δέχεται στο email του έναν μοναδικό εξαψήφιο αριθμό. Για την υλοποίηση του γραφιστικού δημιουργήθηκε το custom `OtpTextField` το οποίο σπάει το string που πληκτρολογεί ο χρήστης σε χαρακτήρες. Μόλις φτάσει του έξι χαρακτήρες στέλνει το εξαψήφιο στον server για επαλήθευσης. Στη παρούσα υλοποίηση βρίσκεται σε demo στάδιο λόγω αναμονής του feature από τη πλευράς του backend.

3.3.5 Σελίδα Dashboard

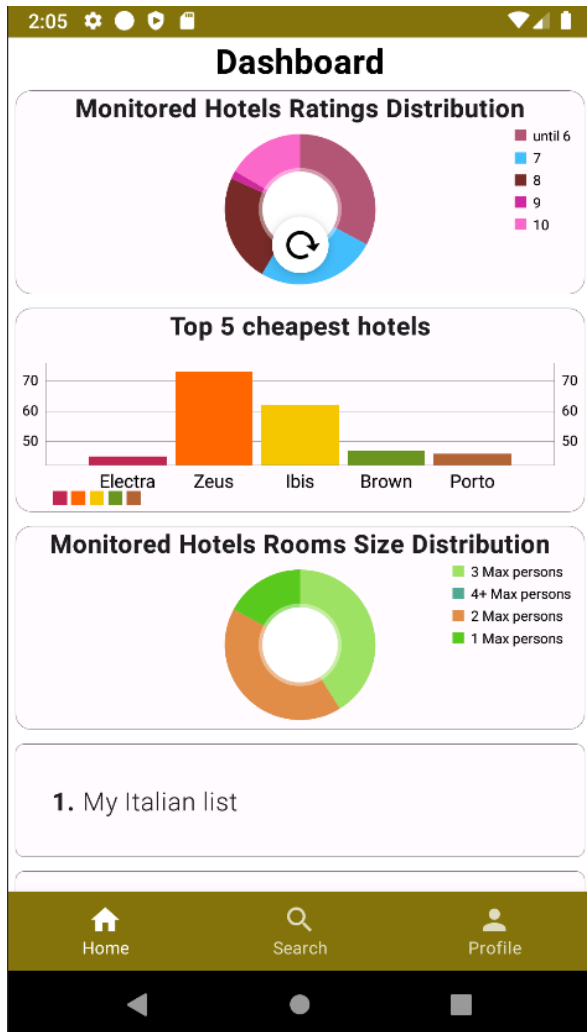
Η σελίδα dashboard περιέχει τρεις υπο-οθόνες στις οποίες μπορεί να περιηγηθεί ο χρήστης από το bottom navigation bar. Οι οθόνες είναι οι:

1. Γενική η οποία περιέχει πληροφορίες για τις προσωποποιημένες λίστες του χρήστη
2. Αναζήτησης νέου ξενοδοχείου
3. Ρυθμίσεων

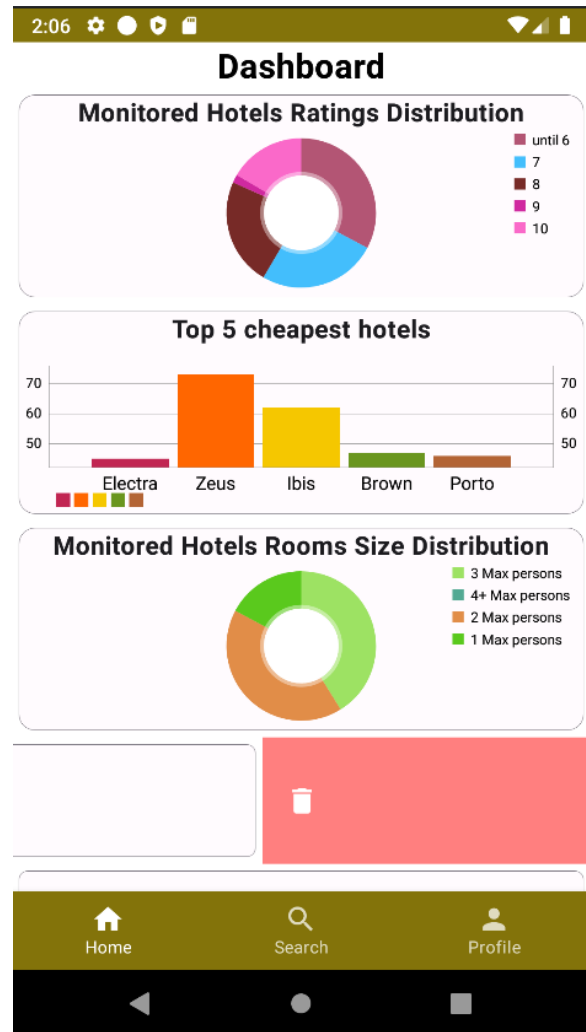
Με τη χρήση της *MPAndroidChart* δημιουργήθηκαν τα τέσσερα στατιστικά γραφήματα σε μορφή *dashboard* και κάτω από τα γραφήματα προβάλλονται οι λίστες που έχει δημιουργήσει ο χρήστης. Ο χρήστης έχει τη δυνατότητα με *swipe left* να διαγράψει μια λίστα όπως φαίνεται στο Σχήμα 3.9 και με *swipe bottom* να κάνει refresh τη σελίδα όπως φαίνεται στο Σχήμα 3.10. Πρόσθετα, μέσω του dashboard, ο χρήστης θα έχει τη δυνατότητα να παρακολουθεί τα πιο ευνοϊκά οικονομικά ξενοδοχεία από τις επιλογές του, τα πιο ακριβά ξενοδοχεία αντίστοιχα σε ένα άλλο γράφημα καθώς και να εξετάζει τη συνολική κατανομή των κριτικών που λαμβάνουν τα ξενοδοχεία (βλ. Σχήμα 3.11). Επιπλέον, θα είναι δυνατόν να αναλύει την ποικιλία των διαθέσιμων δωματίων και να εξάγει πληροφορίες σχετικά με τις προτιμήσεις του κοινού σε σχέση με τους τύπους δωματίων που επιλέγονται περισσότερο. Αυτές οι στατιστικές αναλύσεις θα προσφέρουν στον χρήστη μια πλήρη εικόνα των προτιμήσεών του και του γενικού τοπίου των επιλογών του σχετικά με τις διαμονές του.



Σχήμα 3.10: Dashboard Page



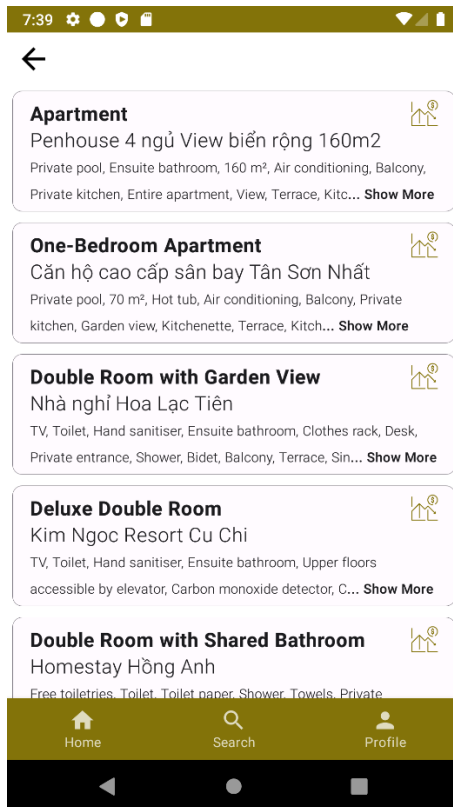
Σχήμα 3.11: Refresh screen gesture actions



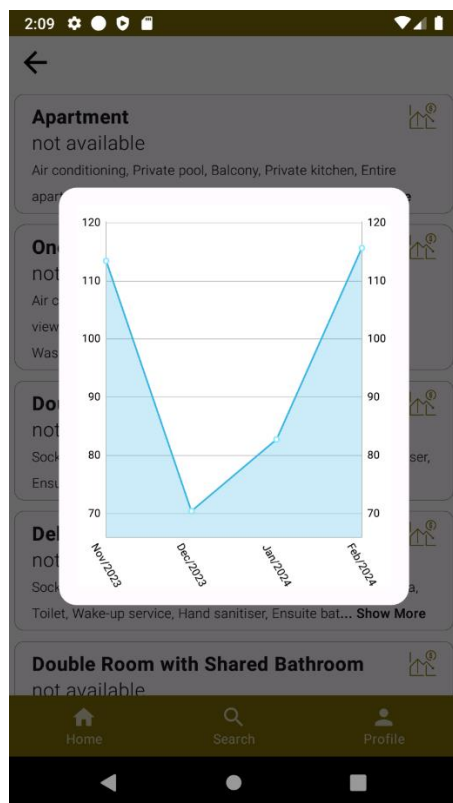
Σχήμα 3.12: Delete list gesture action

Κύρια λειτουργία της εφαρμογής PRIMON είναι η δημιουργία και παρακολούθηση προσωποποιημένων λιστών δωματίων. Στη λίστα φαίνονται διάφορες πληροφορίες για το κάθε δωμάτιο, όπως οι παροχές του, και πατώντας το εικονίδιο του σχεδιαγράμματος πάνω δεξιά (βλ. Σχήμα 3.13) μπορεί ο χρήστης να δει τη μέση τιμή που είχε το συγκεκριμένο δωμάτιο ανά μήνα (Σχήμα 3.14).

Για τη δημιουργία του σχεδιαγράμματος χρειάστηκε μία προσαρμογή της απάντησης του server και η μετατροπή της σε αποδεκτή μορφή γραφήματος με άξονα x, y. Πρώτο στάδιο επεξεργασίας ήταν η μετατροπή του timestamp σε κατανοητή μορφή προγραμματισμού, στη συνέχεια πρόσθεσα τις μέρες απόστασης και ως δεύτερο βήμα χρησιμοποιώντας την Groupby ένωση σε υπο-λίστες όσες τιμές ήταν του ίδιου μήνα και στη συνέχεια δημιούργησα μια Hashmap η οποία είχε ως κλειδιά μήνα/έτος και ως τιμή τον μέσο όρο της κάθε υπο-λίστας. Η σχετική υλοποίηση μπορεί να φανεί και στον κώδικα του σχήματος 3.15.



Σχήμα 3.13: Λίστα δωματίων



Σχήμα 3.14: Σχεδιάγραμμα μέσης τιμής δωματίου ανά μήνα

```

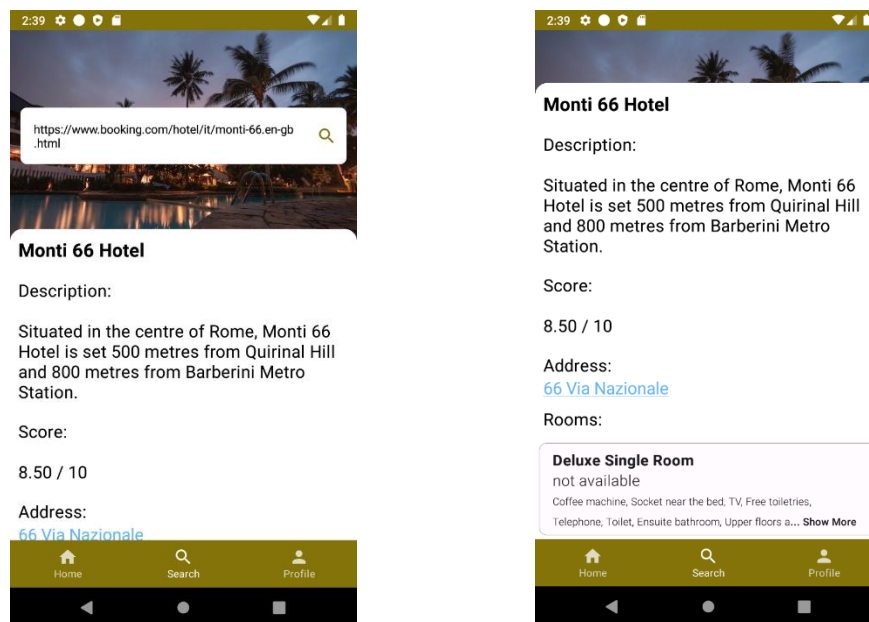
fun PriceRoomResponse.toGraph(): GraphPrice {
    this.embedded.priceInfoList = this.embedded.priceInfoList.map
    ( { it.copy(date = it.timestamp.split(".")[0].toDateime().plusDays(it.distanceDays.toLong()).toStringPattern()) } )
    val dateTimeFormatter: DateTimeFormatter = DateTimeFormatter.ofPattern("dd-MM-yyyy")
    val formatter = DateTimeFormatter.ofPattern("MMM/yyyy")
    val result = this.embedded.priceInfoList.groupBy { LocalDate.parse(it.date, dateTimeFormatter).format(formatter) }
    .mapValues { it.value.map { pair -> pair.price }.average() }
    return GraphPrice(axis_x = result.map { it.key },
        axis_y = result.map { it.value })
}

```

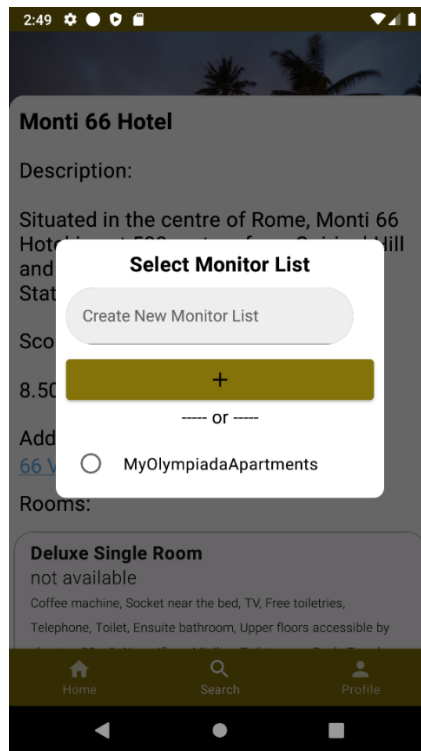
Σχήμα 3.15: Κώδικας μετατροπής της απάντησης του Server σε γράφημα

Ο χρήστης μπορεί να αναζητήσει ξενοδοχεία πληκτρολογώντας το *url* του ξενοδοχείου, της προτίμησης του, από το booking.com και αυτόματα η εφαρμογή μόλις καταλάβει ότι το url καταλήγει σε .html το στέλνει στον server ο οποίος με τη σειρά του στέλνει πίσω πληροφορίες του ξενοδοχείου όπως περιγραφή, αξιολόγηση, διεύθυνση και τα διαθέσιμα δωμάτια (βλ. Σχήμα 3.15). Βάση αυτών των πληροφοριών ο χρήστης μπορεί μέσω της εφαρμογής να πατήσει πάνω στη διεύθυνση και η εφαρμογή τον μεταφέρει στο Google maps στην ακριβή τοποθεσία του ξενοδοχείου. Αυτό επιτυγχάνεται με τη βοήθεια ενός clickable text το οποίο μόλις πατηθεί δημιουργείται μία εντολή προς την εφαρμογή Google Maps στην οποία περνάμε μέσα ως παραμέτρους τις συντεταγμένες (lat long) (βλ τον κώδικα στο Σχήμα 3.16)

Όσον αφορά τα δωμάτια ο χρήστης κάνοντας click επάνω μπορεί να δει το σχεδιάγραμμα των τιμών (βλ. Σχήμα 3.17), ενώ πατώντας παρατεταμένα (long press) μπορεί να το προσθέσει σε μία λίστα παρακολούθησης ή να δημιουργήσει μία καινούργια λίστα εκ νέου (βλ. Σχήμα 3.18)



Σχήμα 3.16: Search Page



Σχήμα 3.17: Εκχώρηση δωματίου σε λίστα παρακολούθησης

```

206 @Composable
207 fun ShareMap(latitude: String, longitude: String, label: String) {
208     Text(
209         modifier = Modifier
210             .padding(start = 10.dp, top = 10.dp),
211         text = "Address:",
212         fontSize = 20.sp,
213     )
214     val context = LocalContext.current
215
216     ClickableText(
217         modifier = Modifier.padding(start = 10.dp),
218         text = AnnotatedString(label),
219         style = TextStyle(
220             color = Color( color: 0xffff64B5F6),
221             fontSize = 20.sp,
222             textDecoration = TextDecoration.Underline
223         ),
224         onClick = {
225             val mapUri = Uri.parse( uriString: "geo:$latitude,$longitude?q=$label")
226             val mapIntent = Intent(Intent.ACTION_VIEW, mapUri)
227             mapIntent.setPackage("com.google.android.apps.maps")
228             context.startActivity(mapIntent)
229         }
230     )
231 }

```

Σχήμα 3.18: Share map component

3.3.6 Σελίδα Προφίλ

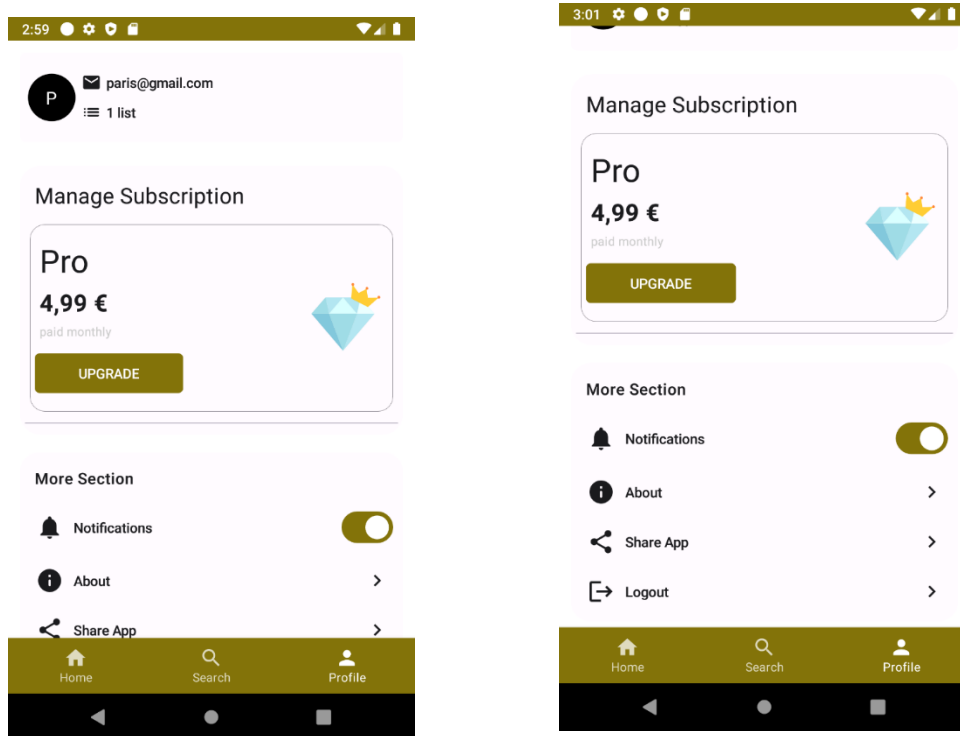
Ο χρήστης μέσω αυτής της οθόνης μπορεί να δει ένα **demo πλάνο συνδρομής της εφαρμογής**. Επιπλέον στοιχεία στην οθόνη είναι η εμφάνιση του email του χρήστη, το αρχικό γράμμα του ονόματός του σαν φωτογραφία προφίλ καθώς και ο αριθμός των αριθμό των λιστών που έχει δημιουργήσει. Όλα αυτά τα στοιχεία προέρχονται από τα endpoints του server και ενημερώνονται δυναμικά.

Στη συνέχεια περνώντας στο τμήμα *More* της σελίδα υπάρχει η επιλογή απενεργοποίησης και ενεργοποίησης των ειδοποιήσεων (notification). Το αρχικό state των ειδοποιήσεων είναι on και πληροφορούν τον χρήστη για τυχών μεταβολές της τιμής των δωματίων των προσωποποιημένων λιστών του (βλ. Σχήματα 3.16-17). Για τη παρουσίαση του συγκεκριμένου feature στην παρούσα πτυχιακή δημιουργήθηκε ένα demo τοπικό *push notification* καθώς η πραγματοποίηση real push notification χρειάζεται τη σύνδεση mobile εφαρμογών και του backend μέσω τρίτης υπηρεσίας όπως το [firebase](#) ή το [onesignal](#). Αμέσως μετά συναντάμε το κουμπί “about” κατά το οποίο όταν γίνεται click εμφανίζεται ένα pop up παράθυρο με τις πληροφορίες της PRIMON εφαρμογής (βλ. Σχήμα 3.18). Στη συνέχεια βλέπουμε την demo επιλογή για κοινοποίηση της εφαρμογής η οποία δίνει τη δυνατότητα να στέλνεις

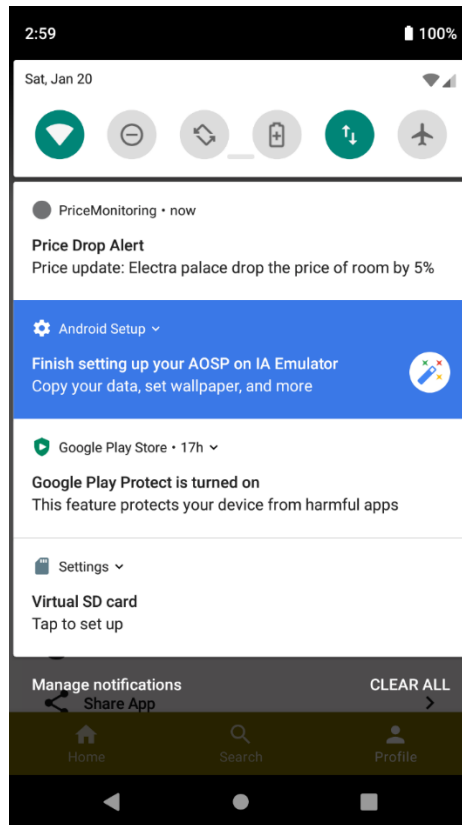
Κεφάλαιο 3

στους φίλους σου μέσω οποιαδήποτε εφαρμογής δίνει δικαίωμα αποστολής μηνύματος ένα ενδεικτικό μήνυμα προτροπής (βλ. Σχήμα 3.19).

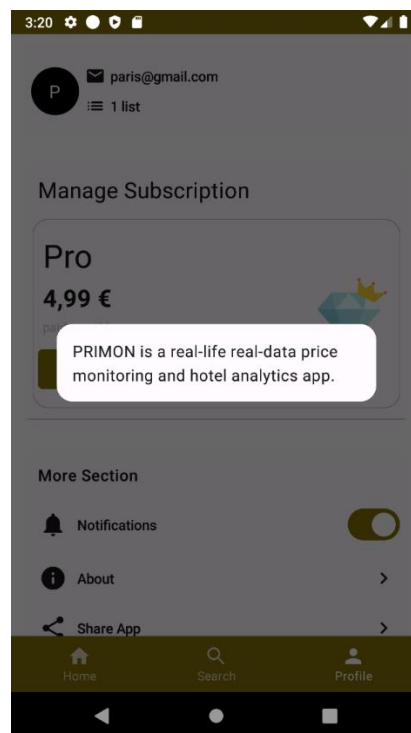
Η τελευταία επιλογή είναι το *Log Out* του χρήστη από την εφαρμογή. Με το πάτημα του Log Out button διαγράφεται το token του χρήστη από τη μνήμη με το κάλεσμα του SessionManager (βλ. Σχήμα 3.18) και τον οδηγεί στη σελίδα του Log In εκ νέου.



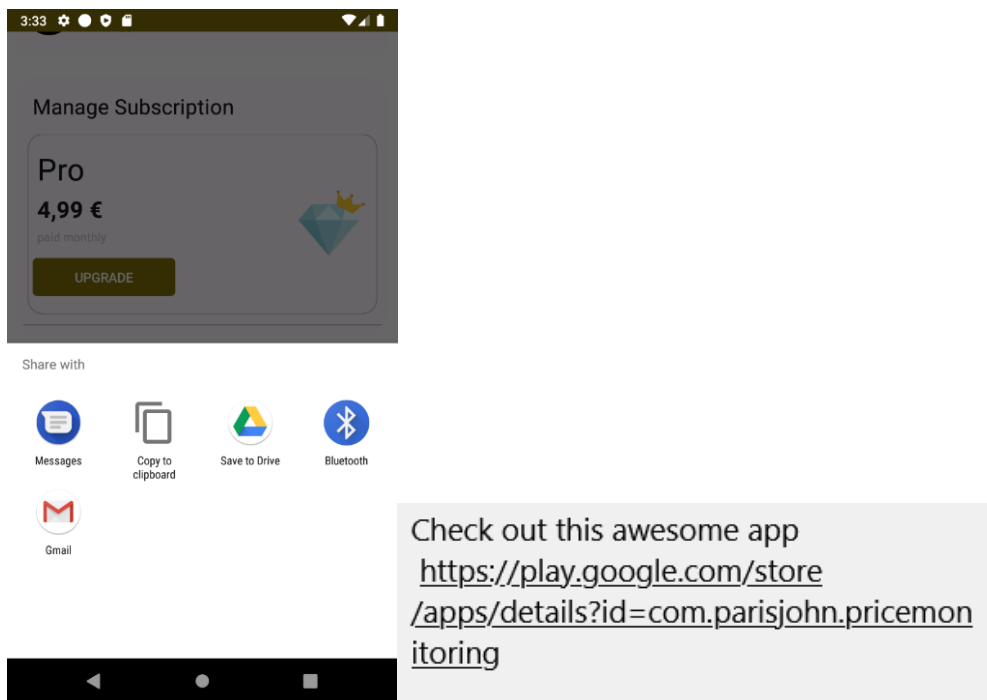
Σχήμα 3.19: Profile page



Σχήμα 3.20: Push Notification



Σχήμα 3.21: About option at Profile Page



Σχήμα 3.22: Share option at profile page & Share Message

```
11 class SessionManager @Inject constructor(  
12     @ApplicationContext private var context: Context  
13 ) {  
14     private val prefs: SharedPreferences = context.getSharedPreferences(context.packageName + "_pr  
15     companion object {  
16         const val USER_TOKEN = "user_token"  
17         const val USER_NOTIFICATIONS = "notifications"  
18     }  
19  
20     /**  
21      * Function to save auth token  
22      */  
23     fun saveAuthToken(token: String?) {  
24         val editor = prefs.edit()  
25         editor.putString(USER_TOKEN, token)  
26         editor.commit()  
27     }  
28  
29     fun clearSession(){  
30         val packageManager: PackageManager = context.packageManager  
31         val intent: Intent = packageManager.getLaunchIntentForPackage(context.packageName)!!  
32         val componentName: ComponentName = intent.component!!  
33         val restartIntent: Intent = Intent.makeRestartActivityTask(componentName)  
34         context.startActivity(restartIntent)  
35         Runtime.getRuntime().exit(status: 0)  
36     }  
}
```

Σχήμα 3.23: SessionManager Class

Κεφάλαιο 4ο: Συμπεράσματα και προτάσεις βελτίωσης

Κύριος στόχος αυτής της εργασίας είναι να εξερευνήσουμε τον ρόλο της πληροφορικής στον τομέα της τουριστικής βιομηχανίας και συγκεκριμένα στον τομέα της τιμολόγησης των ξενοδοχείων-καταλυμάτων. Εντοπίστηκε η ανάγκη για την παρακολούθηση τιμών ανάμεσα στον τεράστιο όγκο πληροφορίας που υπάρχει διαθέσιμος στο διαδίκτυο, καθώς και στο κενό που υπάρχει πάνω σε αυτό το κομμάτι των android εφαρμογών.

Σημαντικό κομμάτι αυτής της εργασίας είναι και ο συνεργατικός χαρακτήρας της που συναντάμε και στην επαγγελματική πραγματικότητα. Φοιτητές του τμήματος, βοήθησαν στην συλλογή πραγματικών δεδομένων (data scraping) , στη δημιουργία του backend της εφαρμογής, και στη δημιουργία endpoints για τη σύνδεσή της. Αντίστοιχα το Web Application κομμάτι του Hotel Price Monitoring υλοποιείται από την συμφοιτήτρια Πατελη Κωνσταντίνα ενώ η IOS εκδοχή της από τον συμφοιτητή Σιαματρα Βασίλη. Σχεδόν σε όλες τις περιπτώσεις η επικοινωνία μεταξύ των επιμέρους ατόμων ήταν αναγκαία. Εν τέλει η συνεργασία ήταν επιτυχής και αρκετά βοηθητική κάτι που θεωρείται ένα εξαιρετικό προσόν στο επαγγελματικό περιβάλλον αυτής της φύσης.

Συνεχίζοντας στο ίδιο πλαίσιο, χρησιμοποιήθηκαν και εργαλεία που εντοπίζονται σε πραγματικά εργασιακά περιβάλλοντα ανάπτυξης εφαρμογών, όπως το Trello για τη δημιουργία Dashboard εργασιών, το Figma για την αρχική σχεδίαση της εφαρμογής βάση του Web Application καθώς και τον server του discord για τη συνεχή επικοινωνία με τους προαναφερθέντες συμφοιτητές.

Σε πρακτικό επίπεδο για την υλοποίηση της android-based εφαρμογής χρησιμοποιήθηκε η γλώσσα προγραμματισμού Kotlin σε συνδυασμό με την νέα UI βιβλιοθήκη που προσφέρει η Google, την Jetpack Compose, τα τελευταία διαθέσιμα εργαλεία του android studio, αποκτώντας πρακτική και χρήσιμη, για το επαγγελματικό μέλλον, γνώση.

Τέλος, αυτή η android εφαρμογή μπορεί να αποτελέσει ένα σοβαρό θεμέλιο για περαιτέρω εξέλιξη. Ένα κύριο χαρακτηριστικό των εφαρμογών αυτής της φύσεως είναι η επεκτασιμότητα. Καθώς, ο ήδη υπάρχων όγκος της πληροφορίας που έρχεται από το διαδίκτυο είναι τεράστιος, αντίστοιχα οι λειτουργίες που καλούνται να τον κατηγοριοποιήσουν και να τον παρουσιάσουν, είναι εξίσου πολλές. Αυτό μπορεί να γίνει είτε με μικρές αλλαγές και προσθήσεις στις υπάρχουσες λειτουργίες, είτε με τη δημιουργία τελείως καινούριων με αναζήτηση βέλτιστων τρόπων υλοποίησης. Επίσης, με μία πιθανή μελλοντική εξέλιξη των πληροφοριών που μαζεύονται μέσω scraping και την περαιτέρω αναβάθμιση της βάσης δεδομένων, η εξέλιξη της ίδιας της εφαρμογής με πρόσθετες λειτουργίες και πληροφορίες όπως και η χρήση του AI με κάποιες λειτουργίες είναι αντιστοίχως πιθανή και συνίσταται ανεπιφύλακτα.

Κεφάλαιο 5ο: Βιβλιογραφία

- [1] «<https://exemplarymarketing.com/impact-of-mobile-apps-on-daily-lives>,» [Ηλεκτρονικό].
- [2] M. A. Camilleri, «The Tourism Industry: An Overview,» σε *Tourism, Hospitality & Event Management*.
- [3] «An investigation of multiple devices and information sources used in the hotel booking process H.C. Murphy, M. Chen, M. Cossutta».
- [4] C. F. D. a. P. R. Rex S. Toh, Travel Planning: Searching for and Booking Hotels on the Internet.
- [5] R. F. Werthner H., «E-commerce and tourism. Communications of the ACM 17 (12): 101-9., 2005.
- [6] «Click to fly,» *The Economist*, pp. 1-3, May 13, 2004.
- [7] «<https://www.dreambigtravelfarblog.com/blog/europe-tourism-travel-statistics>,» [Ηλεκτρονικό].
- [8] A. B. K. N. F. O. Ahmet Bulent Ozturk, «Mobile Hotel Booking Technology in the Hotel Industry,» σε *271THE 3rd INTERNATIONAL ACADEMIC CONFERENCE ON SOCIAL SCIENCES*.
- [9] V. C. Heung, «Internet usage by international travellers: reasons and barriers,» σε *International Journal of Contemporary Hospitality Management*.
- [10] <https://www.dreambigtravelfarblog.com/blog/online-travel-booking-statistics>
- [11] E.-P. H. C. G. C. Lim, «Business Intelligence and Analytics: Research Directions,» *ACM Transactions on Management Information Systems, Volume 3*.
- [12] «<https://www.tableau.com/learn/articles/business-intelligence>,» [Ηλεκτρονικό].
- [13] «<https://atlan.com/data-analytics-in-hospitality-industry/>,» [Ηλεκτρονικό].
- [14] G. K. Nair, «Dynamics of pricing and non-pricing strategies, revenue management performance and competitive advantage in hotel industry,» *International Hospitality Management*.
- [15] <https://www.siteminder.com/r/hotel-pricing>
- [16] <https://www.ezeeabsolute.com/blog/hotel-pricing-strategies>
- [17] H. P. Michael Collins, «Pricing strategies to maximize revenues in the lodging industry,» *International Journal of Hospitality Management*.
- [18] <https://google-developer-training.github.io/android-developer-fundamentals-course-concepts-v2/unit-1-get-started/lesson-1-build-your-first-app/1-0-c-introduction-to-android/1-0-c-introduction-to-android.html>
- [19] M. L. D. McDaniel, «Android market reconstruction and analysis».

- [20] <https://metrikal.io/blog/what-are-google-play-store-tags-and-categories-how-to-use-them>
- [21] C. Anderson, «The Model-View-ViewModel (MVVM) Design Pattern,» σε *Pro Business Applications with Silverlight 5*, 2012, p. pp 461–499.
- [22] D. Prasanna, Dependency Injection: Design patterns using Spring and Guice.
- [23] <https://trello.com/tour>
- [24] <https://www.projectmanager.com/guides/agile-sprints>
- [25] <https://github.com/features>
- [26] <https://www.figma.com/about>