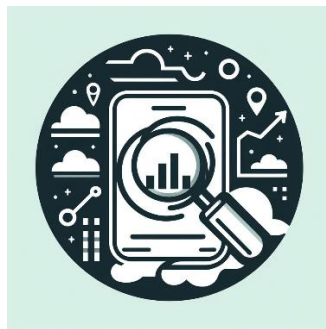


ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ
ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

«Διαδικτυακό σύστημα για διαχείριση τιμών»



Του φοιτητή
Καραβά Μιχαήλ
Αρ. Μητρώου: 175066

Επιβλέπων
Σαλαμπάσης Μιχαήλ
Καθηγητής

Ημερομηνία 29/01/2024

Τίτλος Δ.Ε. «Διαδικτυακό σύστημα για διαχείριση τιμών»

Κωδικός Δ.Ε. 22242

Όνοματεπώνυμο φοιτητή Καραβάς Μιχαήλ

Όνοματεπώνυμο εισηγητή Σαλαμπάσης Μιχαήλ

Ημερομηνία ανάληψης Δ.Ε 01/09/2022

Ημερομηνία περάτωσης Δ.Ε. 29/01/2024

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 Καραβά Μιχαήλ που την εκπόνησε. 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 οποιοδήποτε 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

Βασικό κίνητρο για την πραγματοποίηση της διπλωματικής εργασίας ήταν η επιθυμία για χρήση των νέων τεχνολογιών και μεθοδολογιών προγραμματισμού (Node.js και Docker) με σκοπό την ανάπτυξη μιας διαδικτυακής εφαρμογής για την παρακολούθηση της πορείας τιμών σε ξενοδοχεία. Το περιεχόμενό της κινείται περισσότερο σε πρακτικό πλαίσιο παρά σε θεωρητικό και αυτός ήταν εξ αρχής ο σκοπός. Μετά το πέρας ανάγνωσης του κειμένου της διπλωματικής εργασίας, ο αναγνώστης θα έχει αντιληφθεί βασικά στοιχεία της αρχιτεκτονικής της εφαρμογής, την χρησιμότητά της, αλλά και στοιχεία που μπορούν να εξαχθούν μέσω απλών και κατανοητών παραδειγμάτων εφαρμογής. Η εφαρμογή αυτή θα μπορούσε να αποτελέσει την βάση για μια ολοκληρωμένη πλατφόρμα σύγκρισης παροχών και τιμών στον ξενοδοχειακό κλάδο και εξαγωγής χρήσιμων στατιστικών στοιχείων.

Περίληψη

Βασικό αντικείμενο της συγκεκριμένης διπλωματικής εργασίας ήταν η ανάπτυξη μιας διαδικτυακής εφαρμογής παρακολούθησης τιμών ξενοδοχείων από την ιστοσελίδα Booking.com. Αρχικά, παρουσιάστηκαν τα επιθυμητά χαρακτηριστικά μιας πλατφόρμας σύγκρισης τιμών ξενοδοχείων και πραγματοποίησης κρατήσεων. Στην συνέχεια, παρατέθηκαν δημοφιλείς τέτοιες πλατφόρμες, δίνοντας έμφαση στα χαρακτηριστικά και την λειτουργικότητά τους. Ακολούθησε η παρουσίαση της αρχιτεκτονικής της εφαρμογής, και της λειτουργικότητας βασικών αρχείων κώδικα. Τέλος, μέσω γραφημάτων και πινάκων παρουσιάστηκαν χρήσιμα στοιχεία, τα οποία προέκυψαν από την συλλογή των δεδομένων της εφαρμογής αναφορικά με δύο διαφορετικά ξενοδοχεία.

«Online system for price management»
«Όνομα & Επώνυμο Φοιτητή»
(στην αγγλική γλώσσα)

Abstract

The main object of this thesis was the development of an online application for hotel price monitoring from the website Booking.com. Initially, the desired features of a hotel price comparison and booking platform were presented. Then, popular platforms were listed, emphasizing their features and functionality. This was followed by the presentation of the architecture of the application, and the functionality of basic code files. Finally, through graphs and tables were presented useful data, which emerged from the collection of data of the application regarding two different hotels.

Ευχαριστίες

Στον καθηγητή Μ. Σαλαμπάση για την βοήθεια του, στην οικογένεια μου για την υποστήριξη τους και στους συναδέλφους μου Καλαϊτσίδα Αρσλά και Αβραμίδα Ερωτόκριτο για την συνεργασία τους.

Πίνακας Περιεχομένων

Πρόλογος.....	4
Περίληψη	5
Abstract.....	6
Ευχαριστίες.....	7
ΕΙΣΑΓΩΓΗ	12
Στόχος διπλωματικής εργασίας.....	12
Σχετιζόμενες Θεματικές Περιοχές	12
ΚΕΦΑΛΑΙΟ 1: ΠΛΑΤΦΟΡΜΕΣ ΣΥΓΚΡΙΣΗΣ ΤΙΜΩΝ ΞΕΝΟΔΟΧΕΙΩΝ	15
1.1. Επιθυμητά Χαρακτηριστικά	15
1.2. Προκλήσεις	16
1.3. Δημοφιλείς Πλατφόρμες	17
1.3.1. Booking.com	17
1.3.2. Expedia.....	18
1.3.3. Hotels.com.....	19
1.3.4. Agoda.....	20
1.3.5. Trivago	21
1.3.6. Kayak	22
1.3.7. TripAdvisor.....	24
ΚΕΦΑΛΑΙΟ 2: ΥΛΟΠΟΙΗΣΗ ΕΦΑΡΜΟΓΗΣ ΠΑΡΑΚΟΛΟΥΘΗΣΗΣ ΤΙΜΩΝ ΔΙΑΝΥΚΤΕΡΕΥΣΗΣ ΣΕ ΞΕΝΟΔΟΧΕΙΑ	26
2.1. Βασική Φιλοσοφία Εφαρμογής.....	26
2.2. Node.js.....	28
2.3. Docker	28
2.4. Σύνοψη τεχνολογιών που χρησιμοποιήθηκαν	29
2.5. Βασικά αρχεία εφαρμογής.....	30
2.5.1. Αρχείο server.js	30
2.5.2. Αρχείο incoming-request.js	32
2.5.3. Αρχείο logger.js	34
2.5.4. Αρχείο rabbitmq.js	35
2.5.5. Αρχείο worker.js	38
2.5.6. Αρχείο scrapper.js	39

2.5.7. Αρχείο date.js.....	44
2.5.8. Αρχείο exceptions.js	45
2.5.9. Αρχείο proxy.js	46
2.6. Μορφή αποτελεσμάτων διαδικασίας.....	48
ΚΕΦΑΛΑΙΟ 3 : CASE STUDY	51
3.1. Στοιχεία Δωματίων.....	51
3.2. Εξέλιξη τιμών δωματίων πρώτου ξενοδοχείου.....	52
3.3. Εξέλιξη τιμών δωματίων δεύτερου ξενοδοχείου	54
3.4. Συγκριτικά στοιχεία.....	55
ΚΕΦΑΛΑΙΟ 4: ΣΥΜΠΕΡΑΣΜΑΤΑ	58
4.1. Συνεργατική Ανάπτυξη	58

Κατάλογος Εικόνων

Εικόνα 1: Booking.com.....	17
Εικόνα 2: Expedia	19
Εικόνα 3: Hotels.com	20
Εικόνα 4: Agoda	21
Εικόνα 5: Trivago	22
Εικόνα 6: Kayak	23
Εικόνα 7: TripAdvisor	24
Εικόνα 8: Περιβάλλον πραγματοποίησης πειραματικής διαδικασίας	27
Εικόνα 9: Αρχιτεκτονική Docker	29
Εικόνα 10: Παράδειγμα παραγόμενων αποτελεσμάτων - Μέρος 1.....	48
Εικόνα 11: Παράδειγμα παραγόμενων αποτελεσμάτων - Μέρος 2.....	49
Εικόνα 12: Παράδειγμα αποτελεσμάτων - Μέρος 3.....	49
Εικόνα 13: Εξέλιξη τιμής - Ξενοδοχείο 1 - Δωμάτιο 1 - 7 ημέρες	52
Εικόνα 14: Εξέλιξη τιμής - Ξενοδοχείο 1 - Δωμάτιο 1 - 28 ημέρες	52
Εικόνα 15: Εξέλιξη τιμής - Ξενοδοχείο 1 - Δωμάτιο 2 - 7 ημέρες	53
Εικόνα 16: Εξέλιξη τιμής - Ξενοδοχείο 1 - Δωμάτιο 2 - 28 ημέρες	53
Εικόνα 17: Εξέλιξη τιμής - Ξενοδοχείο 2 - Δωμάτιο 1 - 7 ημέρες	54
Εικόνα 18: Εξέλιξη τιμής - Ξενοδοχείο 2 - Δωμάτιο 1 - 28 ημέρες	54
Εικόνα 19: Συγκριτικά στοιχεία μέσης τιμής - Ξενοδοχείο 1 - Δωμάτιο 1	56
Εικόνα 20: Συγκριτικά στοιχεία μέσης τιμής - Ξενοδοχείο 1 - Δωμάτιο 2	57
Εικόνα 21: Συγκριτικά στοιχεία μέσης τιμής - Ξενοδοχείο 2 - Δωμάτιο 1	57

Κατάλογος Πινάκων

Πίνακας 1: Πληροφορίες πρώτου ξενοδοχείου	26
Πίνακας 2: Πληροφορίες δεύτερου ξενοδοχείου	27
Πίνακας 3: Πληροφορίες Δωματίου 1 - Ξενοδοχείου 1	51
Πίνακας 4: Πληροφορίες Δωματίου 2 - Ξενοδοχείου 1	51
Πίνακας 5: Πληροφορίες Δωματίου 1 - Ξενοδοχείου 2	51
Πίνακας 6: Μέση τιμή δωματίων - 7 ημέρες	55
Πίνακας 7: Μέση τιμή δωματίων - 28 ημέρες	55
Πίνακας 8: Μέγιστη τιμή δωματίων - 7 ημέρες	55
Πίνακας 9: Μέγιστη τιμή δωματίων - 28 ημέρες	55
Πίνακας 10: Ελάχιστη τιμή δωματίων - 7 ημέρες	56
Πίνακας 11: Ελάχιστη τιμή δωματίων - 28 ημέρες	56

ΕΙΣΑΓΩΓΗ

Στα πλαίσια της συγκεκριμένης ενότητας, θα παρατεθούν ο στόχος της διπλωματικής εργασίας, καθώς και μια σύντομη παρουσίαση όσο αφορά τις θεματικές περιοχές τις οποίες πραγματεύεται.

Στόχος διπλωματικής εργασίας

Το πρόβλημα της παρακολούθησης της εξέλιξης των τιμών ενός προϊόντος ή μιας υπηρεσίας αποτελεί μια πρόκληση τόσο για τους καταναλωτές – πελάτες, όσο και για τις ίδιες της επιχειρήσεις. Λύση στο συγκεκριμένο πρόβλημα μπορεί να δώσει η επιστήμη της πληροφορικής μέσω της ανάπτυξης συστημάτων, τα οποία κινούνται προς την συγκεκριμένη κατεύθυνση. Η καταγραφή της πορείας των τιμών δεν χρησιμοποιείται μόνο για σκοπούς αποθήκευσης, αλλά και για την πραγματοποίηση περαιτέρω ανάλυσης και εξαγωγής χρήσιμων συμπερασμάτων.

Προς την συγκεκριμένη κατεύθυνση κινήθηκε και η συγκεκριμένη διπλωματική εργασία, της οποίας το βασικό αντικείμενο είναι η ανάπτυξη μιας εφαρμογής, η οποία ακολουθεί την λογική Web Scraper, με σκοπό την συλλογή δεδομένων ξενοδοχείων από την ιστοσελίδα Booking.com. Πέρα όμως από την συλλογή των δεδομένων, παρουσιάζονται και παραδείγματα στα οποία πραγματοποιείται ανάλυση των δεδομένων και δημιουργία συγκριτικών γραφημάτων με σκοπό την εξαγωγή χρήσιμων συμπερασμάτων. Η ανάλυση των δεδομένων των ξενοδοχείων (hotel analytics) μπορεί να συμβάλλει με θετικό πρόσημο στην ενίσχυση της επιχειρησιακής ευφυΐας (business intelligence), οδηγώντας με τον τρόπο αυτό σε στρατηγικές αποφάσεις σχετικά με τον καθορισμό των τιμών εκείνων οι οποίες θα οδηγήσουν στην προσέλκυση πελατών και αύξηση των κερδών. Για παράδειγμα, αν ένας υπεύθυνος ανάλυσης δεδομένων παρατηρήσει την εξέλιξη των τιμών των δωματίων ενός ξενοδοχείου και την εξέλιξη των κρατήσεων, μπορεί να εξάγει χρήσιμα συμπεράσματα και να παρέχει χρήσιμες συμβουλευτικές υπηρεσίες.

Οι τρεις λοιπόν βασικοί πυλώνες της διπλωματικής εργασίας είναι:

- Παρακολούθηση τιμών μέσω Scraper
- Ανάλυση δεδομένων ξενοδοχείων (Hotel Analytics)
- Επιχειρηματική Ευφυΐα (Business Intelligence)

Σχετιζόμενες Θεματικές Περιοχές

Η **επιχειρηματική ευφυΐα (BI)** [11] [12] [13] αναφέρεται στη χρήση τεχνολογιών, διαδικασιών και εργαλείων για την ανάλυση και την παρουσίαση επιχειρηματικών πληροφοριών. Ο πρωταρχικός στόχος της επιχειρηματικής ευφυΐας είναι να υποστηρίξει την καλύτερη λήψη αποφάσεων μέσα σε έναν οργανισμό παρέχοντας πληροφορίες για τα δεδομένα του. Επιπλέον, περιλαμβάνει ένα ευρύ φάσμα δραστηριοτήτων,

συμπεριλαμβανομένης της ανάλυσης δεδομένων, της υποβολής εκθέσεων και της αναζήτησης. Στην συνέχεια παρουσιάζονται ορισμένες βασικές πτυχές της επιχειρηματικής ευφυΐας:

- **Ενσωμάτωση δεδομένων:** Τα συστήματα επιχειρηματικής ευφυΐας περιλαμβάνουν την ενσωμάτωση δεδομένων από διάφορες πηγές εντός ενός οργανισμού, όπως βάσεις δεδομένων, υπολογιστικά φύλλα και εξωτερικές πηγές. Αυτή η ενσωμάτωση εξασφαλίζει μια ενοποιημένη προβολή των δεδομένων.
- **Αποθήκευση δεδομένων:** Πολλά συστήματα τέτοιου τύπου χρησιμοποιούν αποθήκες δεδομένων για την αποθήκευση και τη διαχείριση μεγάλων όγκων δεδομένων. Οι αποθήκες δεδομένων έχουν σχεδιαστεί για να υποστηρίζουν αποτελεσματικές ερωτήσεις και αναφορές, διευκολύνοντας τους χρήστες να εξάγουν πολύτιμες πληροφορίες.
- **Ανάλυση δεδομένων:** Τα εργαλεία επιχειρηματικής ευφυΐας επιτρέπουν στους χρήστες να αναλύουν δεδομένα για να ανακαλύψουν τάσεις, να εντοπίσουν μοτίβα και να αποκτήσουν βαθύτερη κατανόηση της επιχειρηματικής απόδοσης. Η ανάλυση αυτή μπορεί να περιλαμβάνει την αναζήτηση βάσεων δεδομένων, τη δημιουργία απεικονίσεων και την εκτέλεση στατιστικών αναλύσεων.
- **Δημιουργία αναφορών:** Η δημιουργία αναφορών περιλαμβάνει την παρουσίαση δεδομένων σε ουσιαστική και προσβάσιμη μορφή. Οι εκθέσεις μπορούν να λάβουν τη μορφή παραδοσιακών έντυπων εγγράφων ή διαδραστικών πινάκων ελέγχου, παρέχοντας στους ενδιαφερόμενους μια σαφή και συνοπτική εικόνα των βασικών δεικτών απόδοσης και άλλων σχετικών μετρήσεων.

Όσο αφορά την **ανάλυση δεδομένων ξενοδοχείων** (hotel analytics) [14] [15] περιλαμβάνει τη χρήση τεχνικών ανάλυσης δεδομένων και επιχειρηματικής ευφυΐας προκειμένου να αποκτήσει γνώσεις σχετικά με διάφορες πτυχές των λειτουργιών του ξενοδοχείου, της απόδοσης και της συμπεριφοράς των πελατών. Αξιοποιώντας τα δεδομένα, τα ξενοδοχεία μπορούν να λαμβάνουν αποτελεσματικές αποφάσεις για να βελτιώσουν τις εμπειρίες των επισκεπτών, να βελτιστοποιήσουν τις λειτουργίες και να βελτιώσουν τη συνολική κερδοφορία τους. Ενδεικτικοί τομείς, οι οποίοι σχετίζονται με την ανάλυση δεδομένων ξενοδοχείων είναι οι παρακάτω:

- **Διαχείριση εσόδων:** Η ανάλυση ξενοδοχείων βοηθά στη βελτιστοποίηση της τιμολόγησης δωματίων με βάση παράγοντες όπως η ζήτηση, οι εποχιακές διακυμάνσεις, οι τοπικές εκδηλώσεις και η τιμολόγηση των ανταγωνιστών. Αυτό επιτρέπει στα ξενοδοχεία να μεγιστοποιήσουν τα έσοδα προσαρμόζοντας δυναμικά τις τιμές των δωματίων. Ο συγκεκριμένος τομέας, σχετίζεται άμεσα με το αντικείμενο της διπλωματικής εργασίας.
- **Πρόβλεψη πληρότητας και ζήτησης:** Οι προγνωστικές αναλύσεις μπορούν να χρησιμοποιηθούν για την πρόβλεψη των ποσοστών πληρότητας και της ζήτησης για διαφορετικές περιόδους. Αυτό βοηθά τους ξενοδόχους να σχεδιάζουν τους διαθέσιμους πόρους πιο αποτελεσματικά, διασφαλίζοντας ότι διαθέτουν το κατάλληλο προσωπικό και παροχές.
- **Τμηματοποίηση πελατών:** Η ανάλυση δεδομένων επισκεπτών επιτρέπει στα ξενοδοχεία να εντοπίζουν διαφορετικά τμήματα πελατών με βάση τα δημογραφικά στοιχεία, τις προτιμήσεις και τις συμπεριφορές όσο αφορά τις κρατήσεις τους. Αυτές οι πληροφορίες είναι πολύτιμες για στοχευμένες καμπάνιες μάρκετινγκ, και εξατομικευμένες υπηρεσίες.

Τέλος, δεν θα πρέπει να παραλείψουμε ότι στα πλαίσια της διπλωματικής εργασίας ενός **Web Scraper** [16] [17] [18] με σκοπό την συγκέντρωση πληροφοριών για δωμάτια ξενοδοχείων. Ένας Web Scraper είναι ένα εργαλείο που έχει σχεδιαστεί για την εξαγωγή πληροφοριών από ιστότοπους. Η διαδικασία web scraping είναι η διαδικασία αυτόματης συλλογής δεδομένων από ιστότοπους και περιλαμβάνει την αποστολή αιτημάτων HTTP σε ιστοσελίδες, την ανάλυση του περιεχομένου HTML και την εξαγωγή των επιθυμητών πληροφοριών. Η διαδικασία web scraping μπορεί να χρησιμοποιηθεί για διάφορους σκοπούς, συμπεριλαμβανομένης της εξόρυξης δεδομένων, της έρευνας αγοράς (όπως στην περίπτωση της διπλωματικής εργασίας) και της ανάλυσης ανταγωνιστών.

Η διαδικασία web scraping περιλαμβάνει μια σειρά βημάτων με σκοπό τη συστηματική ανάκτηση και εξαγωγή δεδομένων από ιστότοπους. Η διαδικασία συνήθως ξεκινά με την αποστολή αιτημάτων HTTP στον ιστότοπο προορισμού, ζητώντας το περιεχόμενο HTML συγκεκριμένων ιστοσελίδων. Αυτό γίνεται συχνά χρησιμοποιώντας βοηθητικές βιβλιοθήκες. Μόλις ληφθεί το περιεχόμενο HTML, το επόμενο βήμα περιλαμβάνει την ανάλυση του HTML περιεχομένου για την εξαγωγή των επιθυμητών πληροφοριών. Αυτή η διαδικασία ανάλυσης είναι κρίσιμη και συχνά επιτυγχάνεται με βιβλιοθήκες ανάλυσης HTML.

Σε περιπτώσεις όπου οι ιστότοποι χρησιμοποιούν δυναμική φόρτωση περιεχομένου μέσω JavaScript, ένα κρίσιμο βήμα είναι ο χειρισμός αυτής της δυναμικής συμπεριφοράς. Στην περίπτωση αυτή χρησιμοποιούνται άλλα εργαλεία, επιτρέποντας την αυτοματοποίηση των αλληλεπιδράσεων με δυναμικά στοιχεία. Για να αποφευχθεί η ανίχνευση και τα πιθανά μπλοκ από ιστότοπους, οι Web Scrapers συχνά ορίζουν έναν κατάλληλο πράκτορα στα αιτήματά τους HTTP, κάνοντας τον Scraper να μοιάζει περισσότερο με ένα κανονικό πρόγραμμα περιήγησης ιστού.

Μόλις τα δεδομένα εξαχθούν με επιτυχία, οι Web Scrapers συνήθως τα αποθηκεύουν για περαιτέρω χρήση ή ανάλυση. Αυτό μπορεί να περιλαμβάνει την αποθήκευση δεδομένων σε τοπικά αρχεία, βάσεις δεδομένων ή αποθήκευση σε περιβάλλον cloud. Επιπλέον, η καταγραφή (logging) χρησιμοποιείται για την καταγραφή πληροφοριών σχετικά με τη διαδικασία scraping, συμπεριλαμβανομένων των επιτυχημένων διαδικασιών και τυχόν σφαλμάτων που συναντώνται. Οι κατάλληλοι μηχανισμοί χειρισμού σφαλμάτων είναι κρίσιμοι για τη διαχείριση απροσδόκητων προβλημάτων κατά τη διαδικασία απόξεσης.

ΚΕΦΑΛΑΙΟ 1: ΠΛΑΤΦΟΡΜΕΣ ΣΥΓΚΡΙΣΗΣ ΤΙΜΩΝ ΞΕΝΟΔΟΧΕΙΩΝ

1.1. Επιθυμητά Χαρακτηριστικά

Στα πλαίσια της συγκεκριμένης ενότητας, παρατίθεται τα επιθυμητά χαρακτηριστικά μιας πλατφόρμας σύγκρισης τιμών ξενοδοχείων [1]:

- **Φίλτρα αναζήτησης:** Οι χρήστες θα πρέπει να μπορούν να εφαρμόζουν διάφορα φίλτρα προκειμένου να περιορίσουν την αναζήτησή τους, όπως τοποθεσία, εύρος τιμών, ανέσεις, και αξιολογήσεις επισκεπτών.
- **Κριτικές χρηστών και αξιολογήσεις:** Η πρόσβαση σε λεπτομερείς κριτικές χρηστών και αξιολογήσεις βοηθά τους ταξιδιώτες να λαμβάνουν τεκμηριωμένες αποφάσεις σχετικά με την ποιότητα ενός ξενοδοχείου.
- **Χρήση χαρτών:** Ένας διαδραστικός χάρτης που εμφανίζει την τοποθεσία των ξενοδοχείων μπορεί να είναι χρήσιμος για τους χρήστες προκειμένου να κατανοήσουν την εγγύτητα με αξιοθέατα, μέσα μαζικής μεταφοράς και άλλα σημεία ενδιαφέροντος.
- **Ειδοποιήσεις τιμών:** Η επιλογή ορισμού ειδοποιήσεων τιμών για συγκεκριμένα ξενοδοχεία ή προορισμούς επιτρέπει στους χρήστες να ειδοποιούνται όταν μειώνονται οι τιμές, βοηθώντας τους να εξασφαλίσουν καλύτερες προσφορές και να ελαχιστοποιήσουν το κόστος διαμονής τους.
- **Εγγυήσεις καλύτερων τιμών:** Οι πλατφόρμες που προσφέρουν τις καλύτερες εγγυήσεις τιμών ή αποκλειστικές προσφορές μπορούν να προσελκύσουν χρήστες που θέλουν να διασφαλίσουν ότι λαμβάνουν τις πιο ανταγωνιστικές τιμές.
- **Ευέλικτη αναζήτηση ημερομηνιών:** Η δυνατότητα αναζήτησης ξενοδοχείων με ευέλικτες επιλογές ημερομηνίας βοηθά τους χρήστες να βρουν τις καλύτερες τιμές σε διαφορετικές ημερομηνίες, διευκολύνοντας τον πιο οικονομικό προγραμματισμό των ταξιδιών τους.
- **Υποστήριξη εφαρμογής για κινητές συσκευές:** Πολλοί χρήστες προτιμούν πλατφόρμες που διαθέτουν εφαρμογή για κινητές συσκευές, προκειμένου να πραγματοποιούν τις κρατήσεις τους με τρόπο ευέλικτο και αποτελεσματικό.
- **Πραγματοποίηση ασφαλών συναλλαγών:** Η διασφάλιση των συναλλαγών, μαζί με διαφανείς πολιτικές κράτησης, είναι απαραίτητες για την ενίσχυση της εμπιστοσύνης των χρηστών.
- **Υποστήριξη πελατών:** Η αξιόπιστη υποστήριξη πελατών, συμπεριλαμβανομένης της διαδικτυακής συνομιλίας, του ηλεκτρονικού ταχυδρομείου ή της τηλεφωνικής βοήθειας, μπορεί να είναι ζωτικής σημασίας σε περίπτωση που οι χρήστες αντιμετωπίσουν προβλήματα με κρατήσεις ή έχουν ερωτήσεις.
- **Πολύγλωσση υποστήριξη:** Οι πλατφόρμες που υποστηρίζουν πολλές γλώσσες εξυπηρετούν ένα ευρύτερο κοινό και παρέχουν μια βελτιστοποιημένη εμπειρία χρήστη.
- **Πολιτικές ακύρωσης:** Οι διαφανείς και ευέλικτες πολιτικές ακύρωσης εκτιμώνται από τους ταξιδιώτες, ειδικά λόγω απροσδόκητων αλλαγών στα σχέδια τους.

- **Προσφορές και εκπτώσεις:** Η παροχή αποκλειστικών προσφορών, εκπτώσεων ή επιλογών πακέτων μπορεί να καταστήσει μια πλατφόρμα πιο ελκυστική.
- **Συνεργασία με άλλες ταξιδιωτικές υπηρεσίες:** Οι πλατφόρμες που ενσωματώνουν τις κρατήσεις ξενοδοχείων με άλλες ταξιδιωτικές υπηρεσίες, όπως κρατήσεις πτήσεων και ενοικιάσεις αυτοκινήτων, παρέχουν μια πιο ολοκληρωμένη εμπειρία σχεδιασμού ταξιδιού.

1.2. Προκλήσεις

Όσο αφορά τις προκλήσεις τέτοιου τύπου πλατφορμών, παρατίθενται συνοπτικά στην συνέχεια:

- **Ακρίβεια δεδομένων:** Η διατήρηση ακριβών και ενημερωμένων πληροφοριών σχετικά με τις τιμές, τη διαθεσιμότητα και τις ανέσεις του ξενοδοχείου μπορεί να είναι δύσκολη. Οι αποκλίσεις στα δεδομένα μπορεί να οδηγήσουν σε δυσαρέσκεια των χρηστών και να υπονομεύσουν την εμπιστοσύνη στην πλατφόρμα.
- **Δυσκολία στην λήψη αποφάσεων:** Η αφθονία των επιλογών σε αυτές τις πλατφόρμες μπορεί μερικές φορές να κατακλύσει τους χρήστες, οδηγώντας τους σε δυσκολία λήψης αποφάσεων. Η ταξινόμηση πολλών επιλογών μπορεί να είναι χρονοβόρα και οι χρήστες ενδέχεται να δυσκολεύονται να λάβουν μια απόφαση.
- **Δυναμική τιμολόγηση:** Τα ξενοδοχεία χρησιμοποιούν συχνά δυναμικές στρατηγικές τιμολόγησης, προσαρμόζοντας τα ποσοστά με βάση τη ζήτηση και άλλους παράγοντες. Επιπλέον, οι προσφορές περιορισμένου χρόνου μπορούν να δυσκολέψουν τις πλατφόρμες να παρέχουν σε πραγματικό χρόνο, συνεπείς πληροφορίες τιμολόγησης.
- **Αξιοπιστία κριτικών χρηστών:** Ενώ οι κριτικές χρηστών είναι πολύτιμες, μπορεί να υπάρχουν ανησυχίες σχετικά με την αυθεντικότητα των κριτικών. Οι ψεύτικες ή προκατειλημμένες κριτικές ενδέχεται να παραπλανήσουν τους χρήστες, επηρεάζοντας τη διαδικασία λήψης αποφάσεων.
- **Ζητήματα ασφάλειας και απορρήτου:** Οι χρήστες μοιράζονται ευαίσθητες πληροφορίες, συμπεριλαμβανομένων προσωπικών και οικονομικών στοιχείων, κατά τη διαδικασία κράτησης. Η διασφάλιση της ασφάλειας και του απορρήτου αυτών των πληροφοριών είναι ζωτικής σημασίας για τη διατήρηση της φήμης της πλατφόρμας.
- **Τεχνικές δυσλειτουργίες:** Τεχνικά ζητήματα όπως σφάλματα ιστότοπου, αργοί χρόνοι φόρτωσης ή σφάλματα κατά τη διαδικασία κράτησης μπορούν να απογοητεύσουν τους χρήστες και ενδεχομένως να οδηγήσουν σε απώλεια κρατήσεων.
- **Προκλήσεις ενσωμάτωσης:** Η ενσωμάτωση με διάφορα συστήματα κρατήσεων ξενοδοχείων και η διατήρηση της συμβατότητας με διαφορετικές τεχνολογίες μπορεί να αποτελέσει τεχνική πρόκληση για αυτές τις πλατφόρμες, ειδικά όταν πρόκειται για μεγάλο αριθμό ξενοδοχείων και διαφορετικών συστημάτων κρατήσεων.
- **Μεταβλητότητα:** Τα ξενοδοχεία και τα καταλύματα σε όλο τον κόσμο ενδέχεται να έχουν διαφορετικά συστήματα κρατήσεων, τρόπους πληρωμής και πολιτικές ακύρωσης. Η διαχείριση αυτής της μεταβλητότητας αποτελεί πρόκληση για τις πλατφόρμες που στοχεύουν στην παροχή μιας συνεπούς και φιλικής προς το χρήστη εμπειρίας.

1.3. Δημοφιλείς Πλατφόρμες

Στα πλαίσια της συγκεκριμένης ενότητας, θα παρουσιαστούν συνοπτικά και περιεκτικά οι δημοφιλείς πλατφόρμες σύγκρισης τιμών ξενοδοχείων και πραγματοποίησης κρατήσεων [19]:

- Booking.com
- Expedia
- Hotels.com
- Agoda
- Trivago
- Kayak
- TripAdvisor

1.3.1. Booking.com

Η πλατφόρμα Booking.com [2] αποτελεί ένα παγκοσμίως γνωστό διαδικτυακό ταξιδιωτικό γραφείο που έχει γίνει συνώνυμο με εύκολες και αποτελεσματικές κρατήσεις ξενοδοχείων. Ξεκινώντας από το 1996 στο Άμστερνταμ της Ολλανδίας, η πλατφόρμα έχει εξελιχθεί σε έναν από τους μεγαλύτερους και πιο δημοφιλείς ιστότοπους κρατήσεων καταλυμάτων παγκοσμίως. Γνωστή για τη φιλική προς το χρήστη διεπαφή, τις εκτεταμένες καταχωρίσεις καταλυμάτων και την πολύγλωσση υποστήριξη, η πλατφόρμα Booking.com προσφέρει στους ταξιδιώτες ένα ολοκληρωμένο εργαλείο προκειμένου να συγκρίνουν τις τιμές των ξενοδοχείων, να διαβάζουν κριτικές και να πραγματοποιούν κρατήσεις.

The screenshot displays the Booking.com search results for Venice. On the left, there are filter sections: 'Φιλτράρισμα κατά:' with 'Τα προηγούμενα φίλτρα σας' (listing filters like 'Πολύ καλό: 8+', 'Ξενοδοχεία', etc.), 'Ο προϋπολογισμός σας (ανά βράδυ)' (a price range slider from €40 to €400+), and 'Δημοφιλή φίλτρα' (listing filters like 'Ξενοδοχεία', 'San Marco', etc.). The main content area shows three accommodation cards:

- Ada home**: 9.8 rating, 17 reviews. Location: Βενετία, 9.6 χλμ. από το κέντρο. Price: €519 (discounted from €600). Includes 2 bedrooms, 1 bathroom, 1 sofa, 1 bed, and 3 beds.
- Venezia centoperceto Rooms**: 8.1 rating, 660 reviews. Location: Castello, Βενετία, 1.2 χλμ. από το κέντρο. Price: €284 (discounted from €305). Includes 1 double or 2 single beds.
- Serenissima**: 8.2 rating, 16 reviews. Location: Cannaregio, Βενετία, 1.4 χλμ. από το κέντρο. Price: €549 (discounted from €752). Includes 1 bedroom, 1 bathroom, 1 sofa, 1 bed, and 1 large double bed.

Εικόνα 1: Booking.com

Ένα αξιοσημείωτο χαρακτηριστικό του Booking.com είναι η τεράστια βάση δεδομένων του, που καλύπτει ένα ευρύ φάσμα καταλυμάτων, συμπεριλαμβανομένων ξενοδοχείων, ξενώνων, και διαμερισμάτων. Τα ισχυρά φίλτρα αναζήτησης της πλατφόρμας επιτρέπουν στους χρήστες να προσαρμόζουν την αναζήτησή τους με βάση προτιμήσεις όπως τοποθεσία, εύρος τιμών, ανέσεις και αξιολογήσεις επισκεπτών. Επιπλέον, η δέσμευση της εταιρείας να παρέχει διαφανείς πληροφορίες, συμπεριλαμβανομένων λεπτομερών περιγραφών καταλυμάτων και επαληθευμένων κριτικών επισκεπτών, συμβάλλει στην οικοδόμηση εμπιστοσύνης μεταξύ των χρηστών.

Η επιτυχία της μπορεί να αποδοθεί εν μέρει στην καινοτόμο προσέγγισή της και στη συνεχή προσαρμογή της στις τάσεις της τουριστικής βιομηχανίας. Η ενσωμάτωση ευέλικτων αναζητήσεων ημερομηνίας από την πλατφόρμα, οι ενημερώσεις διαθεσιμότητας σε πραγματικό χρόνο, και η εισαγωγή λειτουργιών όπως οι ειδοποιήσεις τιμών αποδεικνύουν τη δέσμευσή της να καλύψει τις εξελισσόμενες ανάγκες των σύγχρονων ταξιδιωτών. Με παγκόσμια παρουσία και συνεργασίες με εκατομμύρια καταλύματα, η πλατφόρμα Booking.com παραμένει ένας προορισμός για άτομα που αναζητούν μια απρόσκοπτη και αξιόπιστη εμπειρία στην κράτηση των καταλυμάτων τους .

1.3.2. Expedia

Η Expedia [3], που ιδρύθηκε το 1996, αποτελεί ένα από τα πρωτοποριακά online ταξιδιωτικά γραφεία που έφεραν επανάσταση στον τρόπο με τον οποίο οι άνθρωποι σχεδιάζουν και κλείνουν τα ταξίδια τους. Με έδρα το Σιάτλ της Ουάσινγκτον, η Expedia έχει εξελιχθεί σε γίγαντα, παρέχοντας μια ολοκληρωμένη πλατφόρμα για τους χρήστες να πραγματοποιούν κρατήσεις πτήσεων, ξενοδοχείων, ενοικιάσεων αυτοκινήτων και πακέτων διακοπών. Ο σχεδιασμός της πλατφόρμας και η φιλική προς το χρήστη διεπαφή συνέβαλαν στην ευρεία δημοτικότητά της στους ταξιδιώτες παγκοσμίως.

Ένα από τα βασικά πλεονεκτήματα της Expedia έγκειται στην ικανότητά της να προσφέρει ένα ευρύ φάσμα ταξιδιωτικών υπηρεσιών σε μια ενιαία πλατφόρμα. Πέρα από τις κρατήσεις ξενοδοχείων, οι χρήστες μπορούν να προγραμματίσουν απρόσκοπτα ολόκληρο το ταξίδι τους, συμπεριλαμβανομένης της εύρεσης ανταγωνιστικών προσφορών πτήσεων και ενοικίασης αυτοκινήτων. Η επιλογή "Bundle and Save" της Expedia επιτρέπει στους χρήστες να συνδυάζουν διάφορα στοιχεία ταξιδιού, προσφέροντας πιθανές εξοικονομήσεις και απλοποιώντας τη διαδικασία κράτησης. Αυτή η ολιστική προσέγγιση στον ταξιδιωτικό σχεδιασμό ξεχωρίζει την Expedia ως μια ευέλικτη και ολοκληρωμένη λύση για διαφορετικές ταξιδιωτικές ανάγκες.

Η έμφαση της Expedia στην αφοσίωση και τις ανταμοιβές των πελατών έχει σταθεροποιήσει περαιτέρω τη θέση της στον ταξιδιωτικό κλάδο. Μέσω του προγράμματος Expedia Rewards, οι χρήστες μπορούν να κερδίσουν πόντους στις κρατήσεις, οι οποίοι μπορούν να εξαργυρωθούν για εκπτώσεις σε μελλοντικά ταξίδια. Αυτό το κίνητρο, μαζί με αποκλειστικές προσφορές μόνο για μέλη, ενθαρρύνει την επαναλαμβανόμενη χρήση της πλατφόρμας. Η δέσμευση της Expedia για καινοτομία, σε συνδυασμό με το εκτεταμένο δίκτυο συνεργασιών και την παγκόσμια εμβέλεια, συνεχίζει να την καθιστά μια επιλογή για ταξιδιώτες που αναζητούν ευκολία και ολοκληρωμένες ταξιδιωτικές λύσεις.

The screenshot shows the Expedia search interface for hotels in London. At the top, there are search filters for destination ("London (and vicinity), England, U..."), dates ("Jan 19 - Jan 25"), and travelers ("2 travelers, 1 room"). Below the filters, there is a map of London and a "View in a map" button. A dark banner promotes signing in for better prices. The main content area displays two hotel listings:

- Sloane Place**: Located in Chelsea, described as a "Boutique Hotel in the Heart of Chelsea". It features a "VIP Access" badge and a "Sign in for extra savings" button. The price is shown as \$228 crossed out and \$205, with a total of \$1,479 including taxes and fees. It has a 9.2 rating from 487 reviews.
- Kingsland Locke**: Located in London, described as a "Design-led aparthotel near Shoreditch". It features a "Kitchen" icon and a "We have 4 left at" badge. The price is shown as \$144 crossed out and \$137, with a total of \$987 including taxes and fees. It has a 9.0 rating from 167 reviews.

On the left side, there are additional filters: "Compare properties" (checked), "Search by property name" (with a search box containing "e.g. Marriott"), and "Filter by" (with popular filters: London, Breakfast included, Hotel, Reserve now, pay later).

Εικόνα 2: Expedia

1.3.3. Hotels.com

Η πλατφόρμα Hotels.com [4], που ιδρύθηκε το 1991, έχει καθιερωθεί ως μια εξέχουσα διαδικτυακή πλατφόρμα κρατήσεων καταλυμάτων, που εξυπηρετεί ταξιδιώτες που αναζητούν ένα ευρύ φάσμα επιλογών διαμονής σε όλο τον κόσμο. Με έδρα το Ντάλας του Τέξας, η Hotels.com έχει εξελιχθεί σε έναν από τους κορυφαίους παίκτες στη διαδικτυακή ταξιδιωτική βιομηχανία, γνωστή για τον φιλικό προς τον χρήστη ιστότοπο και την εφαρμογή για κινητά, διευκολύνοντας τους χρήστες να εξερευνήσουν και να πραγματοποιούν κράτηση ξενοδοχείων και εναλλακτικών καταλυμάτων.

Ένα χαρακτηριστικό γνώρισμα του Hotels.com είναι η δέσμευσή της για απλότητα και διαφάνεια. Η διεπαφή χρήστη της πλατφόρμας έχει σχεδιαστεί για να διευκολύνει τις απλές αναζητήσεις ξενοδοχείων, επιτρέποντας στους χρήστες να φιλτράρουν τα αποτελέσματα με βάση παράγοντες όπως η τοποθεσία, το εύρος τιμών, η βαθμολογία με αστέρια και οι κριτικές πελατών. Η Hotels.com προσφέρει επίσης ένα πρόγραμμα επιβράβευσης γνωστό ως "Hotels.com Rewards," όπου οι χρήστες μπορούν να κερδίσουν μια δωρεάν διανυκτέρευση μετά τη συλλογή ενός συγκεκριμένου αριθμού διανυκτερεύσεων μέσω επιλέξιμων κρατήσεων. Αυτό το πρόγραμμα ανταμοιβών συνέβαλε στη διατήρηση και την ενίσχυση της ικανοποίησης των χρηστών.

The screenshot displays the Hotels.com search interface. At the top, there are three main filters: 'Μετάβαση σε' (Destination) set to 'Λονδίνο, Αγγλία, Ηνωμένο Βασίλειο...', 'Ημερομηνίες' (Dates) set to '25 Ιαν - 26 Ιαν', and 'Ταξιδιώτες' (Travelers) set to '2 ταξιδιώτες, 1 δωμάτιο'. A search icon is visible on the right. Below these filters, a map of London is shown with a 'Προβολή σε χάρτη' button. To the right of the map, it says '300+ καταλύματα' and 'Τα τέλη που πληρωνόμαστε επηρεάζουν τη σειρά κατάταξης'. A dropdown menu for 'Ταξινόμηση κατά' is set to 'Προτεινόμενα'. A dark blue banner offers a 'Σύνδεση' button and a promotion: 'Εξοικονομήστε κατά μέσο όρο 15% σε χιλιάδες ξενοδοχεία όταν συνδεθείτε'. Below this, there are three hotel listings:

- citizenM London Victoria Station** (Λονδίνο): Price 214€ → 182€ (15% discount). Rating 9,6. 'Εξαιρετικό' (622 reviews). Includes a 'Συλλέξτε σφραγίδες' badge.
- The Chilworth London Paddington** (Πάντινγκτον): Price 199€ → 170€ (11% discount). Rating 8,4. 'Πολύ καλό' (1,013 reviews). Includes a 'Συλλέξτε σφραγίδες' badge and a 'Συνδεθείτε για πρόσθετες εκπτώσεις' button.
- citizenM London Victoria Station** (Λονδίνο): Price 214€ → 182€ (15% discount). Rating 9,6. 'Εξαιρετικό' (622 reviews). Includes a 'Συλλέξτε σφραγίδες' badge.

On the left side, there are additional filters: 'Σύγκριση καταλυμάτων' (toggle off), 'Αναζήτηση με βάση το όνομα καταλύματος' (search bar with 'π.χ. Marriott'), and 'Φιλτράρισμα κατά' (Demφιλή φίλτρα) with options for 'Κέντρο του Λονδίνου', 'Περιλαμβάνεται πρωινό', 'Πισίνα', 'Κάντε κράτηση τώρα, πληρώστε αργότερα', and 'Ξενοδοχείο'.

Εικόνα 3: Hotels.com

Η στρατηγική μάρκετινγκ της εταιρείας περιλαμβάνει συχνά χιουμοριστικές και αξέχαστες διαφημιστικές καμπάνιες, ενισχύοντας την ταυτότητά της. Συνολικά, η πλατφόρμα Hotels.com συνεχίζει να είναι μια δημοφιλής επιλογή για ταξιδιώτες που αναζητούν μια απρόσκοπτη και διαφανή εμπειρία κατά την κράτηση καταλυμάτων.

1.3.4. Agoda

Η Agoda [5], που ιδρύθηκε το 2005, έχει μεταβληθεί σε έναν εξέχων παίκτη στον κλάδο των online ταξιδιών, ειδικευμένη στις κρατήσεις ξενοδοχείων και καταλυμάτων. Αρχικά ιδρύθηκε στη Σιγκαπούρη, και έχει επεκτείνει την εμβέλειά της παγκοσμίως, παρέχοντας μια φιλική προς το χρήστη πλατφόρμα που εξυπηρετεί ένα ευρύ φάσμα ταξιδιωτών. Εξαγοράστηκε από τον Όμιλο Booking Holdings το 2007, και λειτουργεί ως ανεξάρτητο διαδικτυακό ταξιδιωτικό γραφείο, προσφέροντας μια μεγάλη ποικιλία καταλυμάτων, από ξενοδοχεία και θέρετρα έως ξενώνες.

Ένα από τα διακριτικά χαρακτηριστικά της Agoda είναι η ισχυρή παρουσία της στην ασιατική ταξιδιωτική αγορά. Η πλατφόρμα έχει δημιουργήσει ένα ισχυρό δίκτυο ξενοδοχειακών συνεργασιών, ιδιαίτερα στην περιοχή Ασίας-Ειρηνικού, παρέχοντας στους ταξιδιώτες μια ολοκληρωμένη επιλογή καταλυμάτων σε προορισμούς που ενδέχεται να καλύπτονται λιγότερο από άλλα διαδικτυακά ταξιδιωτικά γραφεία. Η

εστίαση της Agoda στην κάλυψη των μοναδικών προτιμήσεων και των ταξιδιωτικών προτύπων των Ασιατών ταξιδιωτών συνέβαλε στη δημοτικότητά της στην περιοχή.

The screenshot shows the Agoda search interface for London. The search bar at the top indicates 4,291 choices for London, with dates for 20 Jan 2024 (Saturday) and 21 Jan 2024 (Sunday), and 2 adults in 1 room. The search results are sorted by 'Our picks'. Two hotels are displayed:

- Britannia International Hotel Canary Wharf**: 4.5 stars, 15,893 reviews, Good (6.7). Price: €92 after tax & fees. Includes breakfast and free cancellation.
- Maitrise Hotel Maida Vale London**: 4.5 stars, 2,466 reviews, Very good (7.1). Price: €106 after tax & fees. Includes AGODASPONSORED applied - €16 off!

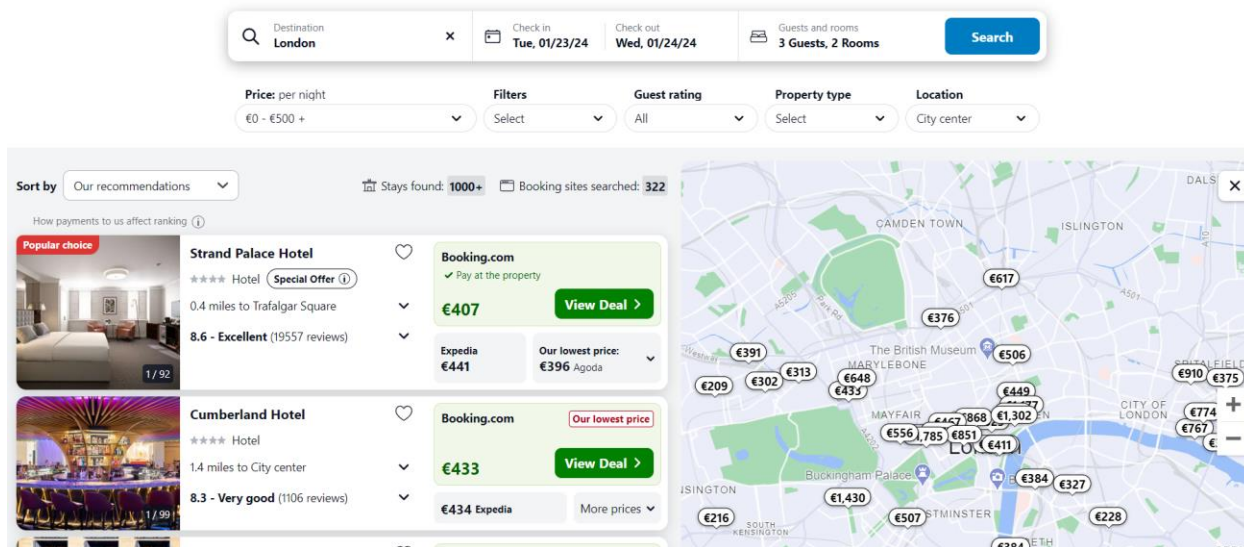
Εικόνα 4: Agoda

Επιπλέον, ο ιστότοπος και η εφαρμογή για κινητά της Agoda έχουν σχεδιαστεί για την διευκόλυνση των χρηστών, προσφέροντας μια απλή διαδικασία κράτησης και παρέχοντας λεπτομερείς πληροφορίες για κάθε κατάλυμα, συμπεριλαμβανομένων κριτικών χρηστών, αξιολογήσεων και φωτογραφιών. Το πρόγραμμα επιβράβευσης της πλατφόρμας, Agoda Rewards, επιτρέπει στους χρήστες να κερδίζουν πόντους με κάθε κράτηση, οι οποίοι μπορούν να εξαργυρωθούν για μελλοντικές εκπτώσεις. Με τη δέσμευση να προσφέρει μια απρόσκοπτη και ικανοποιητική εμπειρία, η Agoda συνεχίζει να προσελκύει μια παγκόσμια βάση χρηστών που αναζητά μια αξιόπιστη πλατφόρμα για τις ανάγκες διαμονής τους.

1.3.5. Trivago

Η πλατφόρμα Trivago [6], που ιδρύθηκε το 2005 και εδρεύει στο Ντίσελντορφ της Γερμανίας, έχει κερδίσει την αναγνώριση ως μια εξέχουσα μηχανή μετα-αναζήτησης ξενοδοχείων, επιτρέποντας στους χρήστες να συγκρίνουν τις τιμές των ξενοδοχείων και να βρουν τις καλύτερες προσφορές σε διάφορα διαδικτυακά ταξιδιωτικά γραφεία. Η ξεχωριστή προσέγγιση της Trivago επικεντρώνεται στην ολοκληρωμένη λειτουργικότητα αναζήτησης, συγκεντρώνοντας τις τιμές των ξενοδοχείων από πολλούς

ιστότοπους κρατήσεων. Η πλατφόρμα ξεχωρίζει με τις εμβληματικές καμπάνιες μάρκετινγκ, δίνοντας έμφαση στην ιδέα της ενδυνάμωσης των χρηστών να πραγματοποιούν ενημερωμένες επιλογές προβάλλοντας ένα ευρύ φάσμα επιλογών και τιμών ξενοδοχείων.



Εικόνα 5: Trivago

Η βασική δύναμη του Trivago έγκειται στη φιλική προς το χρήστη διεπαφή του, προσφέροντας μια οπτικά ελκυστική πλατφόρμα για τους ταξιδιώτες ώστε να αναζητούν και να συγκρίνουν τις επιλογές ξενοδοχείων. Με την εμφάνιση τιμών από πολλούς ιστότοπους κρατήσεων δίπλα-δίπλα, οι χρήστες μπορούν εύκολα να εντοπίσουν τις καλύτερες προσφορές και να λάβουν τεκμηριωμένες αποφάσεις με βάση τις προτιμήσεις τους. Η έμφαση της Trivago στη διαφάνεια όσο αφορά την τιμολόγηση των ξενοδοχείων ευθυγραμμίζεται με τη δέσμευσή της να παρέχει μια απλοποιημένη και αποτελεσματική εμπειρία χρήστη.

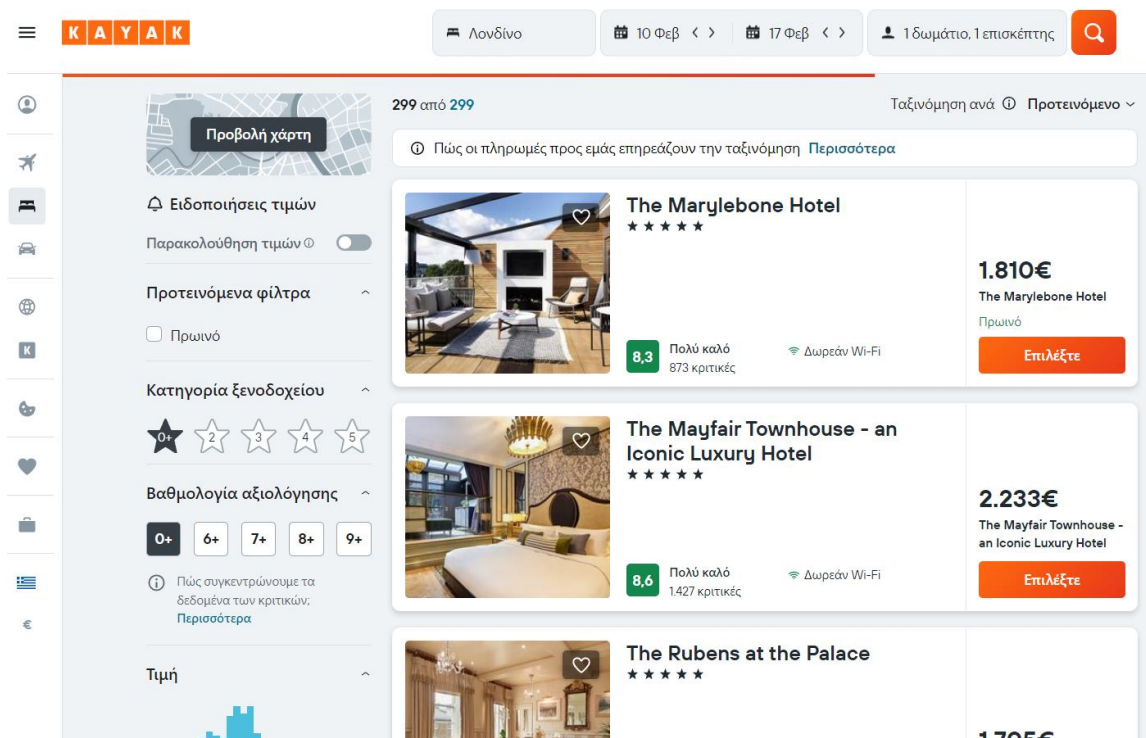
Παρά τον ανταγωνισμό στην διαδικτυακή ταξιδιωτική βιομηχανία, η εστίαση της Trivago στη σαφή και άμεση επικοινωνία, σε συνδυασμό με την καινοτόμο προσέγγιση μετα-αναζήτησης, συνέβαλε στην ιδιότητά της ως δημοφιλούς επιλογής για ταξιδιώτες που αναζητούν τις καλύτερες προσφορές ξενοδοχείων.

1.3.6. Kayak

Η Kayak [7], που ιδρύθηκε το 2004, έχει εξελιχθεί σε μια δημοφιλή μηχανή αναζήτησης ταξιδιών που ξεπερνά τις συγκρίσεις ξενοδοχείων προκειμένου να συμπεριλάβει πτήσεις, ενοικιάσεις αυτοκινήτων και πακέτα διακοπών. Με έδρα το Στάμφορντ του Κονέκτικατ, το Kayak παρέχει στους χρήστες μια ολοκληρωμένη πλατφόρμα προκειμένου να σχεδιάσουν και να υλοποιήσουν διάφορες πτυχές των ταξιδιών τους. Αυτό που ξεχωρίζει το Kayak είναι η εστίασή του στην ενδυνάμωση των χρηστών με μια σειρά

ταξιδιωτικών πληροφοριών και επιλογών, επιτρέποντάς τους να λαμβάνουν καλά ενημερωμένες αποφάσεις σχετικά με τα ταξίδια τους.

Ένα από τα χαρακτηριστικά του Kayak είναι η ισχυρή μηχανή αναζήτησης, η οποία συγκεντρώνει πληροφορίες από πολυάριθμους ιστότοπους κρατήσεων ταξιδιών και ξενοδοχείων. Αυτό επιτρέπει στους χρήστες να συγκρίνουν τιμές και επιλογές, βοηθώντας τους να βρουν τις πιο ανταγωνιστικές τιμές για ξενοδοχεία και άλλες ταξιδιωτικές υπηρεσίες. Η φιλική προς το χρήστη διεπαφή του Kayak, εξοπλισμένη με προηγμένα φίλτρα και επιλογές ταξινόμησης, διασφαλίζει ότι οι ταξιδιώτες μπορούν να προσαρμόσουν τις παραμέτρους αναζήτησης με βάση τις προτιμήσεις τους.



Εικόνα 6: Kayak

Η δέσμευση της Kayak για καινοτομία είναι εμφανής σε χαρακτηριστικά όπως το εργαλείο εξερεύνησης, το οποίο επιτρέπει στους χρήστες να ανακαλύψουν νέους προορισμούς και να βρουν την καλύτερη στιγμή για επίσκεψη. Επιπλέον, η πλατφόρμα προσφέρει μια λειτουργία ειδοποιήσεων τιμών, ειδοποιώντας τους χρήστες όταν υπάρχουν διακυμάνσεις στις τιμές για συγκεκριμένα ξενοδοχεία ή πτήσεις. Με το ευρύ φάσμα λειτουργιών και την έμφαση στην ενδυνάμωση των χρηστών, η πλατφόρμα Kayak παραμένει μια δημοφιλής επιλογή για ταξιδιώτες που αναζητούν ένα ευέλικτο και ολοκληρωμένο εργαλείο για τον προγραμματισμό και την κράτηση των ταξιδιών τους.

1.3.7. TripAdvisor

Το TripAdvisor [8] είναι μια ευρέως αναγνωρισμένη διαδικτυακή πλατφόρμα που χρησιμεύει ως προορισμός για ταξιδιώτες που αναζητούν κριτικές, προτάσεις και πληροφορίες για ξενοδοχεία, εστιατόρια, αξιοθέατα και διάφορες ταξιδιωτικές υπηρεσίες. Ξεκίνησε το 2000, και έχει εξελιχθεί σε μία από τις μεγαλύτερες και πιο σημαντικές ταξιδιωτικές κοινότητες παγκοσμίως. Το μοντέλο περιεχομένου που δημιουργείται από χρήστες επιτρέπει σε εκατομμύρια ταξιδιώτες να μοιράζονται τις εμπειρίες, τις αξιολογήσεις και τις φωτογραφίες τους, δίνοντας τη δυνατότητα σε άλλους να λαμβάνουν τεκμηριωμένες αποφάσεις κατά τον προγραμματισμό των ταξιδιών τους. Η τεράστια βάση δεδομένων της πλατφόρμας περιλαμβάνει ένα ευρύ φάσμα προορισμών και καταλυμάτων, καθιστώντας την πολύτιμη πηγή τόσο για όλες τις κατηγορίες ταξιδιωτών.

Ένα από τα βασικά χαρακτηριστικά του TripAdvisor είναι το σύστημα αξιολογήσεων που διαθέτει, όπου οι χρήστες μπορούν να αξιολογήσουν και να παρέχουν λεπτομερή σχόλια σχετικά με τις εμπειρίες τους. Αυτές οι πληροφορίες βοηθούν τους πιθανούς ταξιδιώτες να λάβουν αποφάσεις σχετικά με τα μελλοντικά τους ταξίδια. Επιπλέον, Το TripAdvisor προσφέρει μια πλατφόρμα κρατήσεων, επιτρέποντας στους χρήστες να πραγματοποιούν κρατήσεις απευθείας μέσω του ιστότοπου για ξενοδοχεία, εστιατόρια και άλλες ταξιδιωτικές υπηρεσίες. Αυτή η ενσωμάτωση των κριτικών και της λειτουργικότητας κρατήσεων δημιουργεί μια απρόσκοπτη εμπειρία χρήστη, καθιστώντας το ένα one-stop-shop για τον προγραμματισμό ταξιδιών.

The screenshot displays the TripAdvisor search results for hotels in London. The interface is organized into several sections:

- Filters:** On the left, there are filters for 'Deals' (Fully refundable, No prepayment needed, Properties with Special Offers), 'Price' (a price range slider from \$0 to \$889+ and a dropdown for 'Price per night'), and 'Popular' (checkboxes for star ratings and amenities like breakfast included).
- Search Results:** The main area shows 4,451 properties in London, sorted by 'Best Value'. Two hotel listings are highlighted:
 - The Suites - St Pancras Hotel Group:** A sponsored listing with an 'Official Hotel Site' badge. The price is \$192. It features a photo of the hotel entrance and a 'View deal' button. The description mentions a cozy atmosphere and convenient location.
 - 1. Park Grand London Hyde Park:** A listing with an 'Expedia' badge. The price is \$155. It features a photo of the hotel entrance and a 'View deal' button. The description mentions a well-located hotel near Paddington Station and Hyde Park.

Εικόνα 7: TripAdvisor

Ενώ το TripAdvisor έχει επηρεάσει σημαντικά την ταξιδιωτική βιομηχανία, έχει επίσης αντιμετωπίσει κάποια κριτική. Ανησυχίες έχουν εγερθεί σχετικά με την αυθεντικότητα των κριτικών, με περιστασιακές περιπτώσεις ψεύτικων ή προκατειλημμένων ανατροφοδοτήσεων. Παρ ' όλα αυτά, η πλατφόρμα εξακολουθεί να αποτελεί μια δύναμη στον ταξιδιωτικό χώρο, διαμορφώνοντας τον τρόπο με τον οποίο οι άνθρωποι εξερευνούν και βιώνουν τον κόσμο παρέχοντας μια δυναμική και καθοδηγούμενη από τον χρήστη πλατφόρμα σχεδιασμού ταξιδιών.

ΚΕΦΑΛΑΙΟ 2: ΥΛΟΠΟΙΗΣΗ ΕΦΑΡΜΟΓΗΣ ΠΑΡΑΚΟΛΟΥΘΗΣΗΣ ΤΙΜΩΝ ΔΙΑΝΥΚΤΕΡΕΥΣΗΣ ΣΕ ΞΕΝΟΔΟΧΕΙΑ

2.1. Βασική Φιλοσοφία Εφαρμογής

Βασικός στόχος του πρακτικού μέρους της διπλωματικής εργασίας, που ουσιαστικά αποτελεί και το κύριο μέρος αυτής είναι η υλοποίηση μιας εφαρμογής, που ακολουθεί την λογική ενός Web Scraper και συλλέγει πληροφορίες σχετικά με την τιμή δωματίων ξενοδοχείο από την ιστοσελίδα Booking.com. Πέρα από την απλή καταγραφή των τιμών, στόχος υλοποίηση της εφαρμογής είναι η παρακολούθηση της εξέλιξης των τιμών, ώστε να εξαχθούν χρήσιμες γραφικές παραστάσεις και στατιστικά στοιχεία, που θα παρατεθούν στο επόμενο κεφάλαιο.

Οι πληροφορίες των ξενοδοχείων, για τα οποία συνελέγησαν πληροφορίες μέσω του Scraper από την ιστοσελίδα Booking.com συνοψίζονται στους παρακάτω πίνακες. Αρχικά παρατίθενται οι πληροφορίες για το πρώτο ξενοδοχείο:

Όνομα	Elisa Boutique Saigon Hotel
Διεύθυνση	172 ĐƯỜNG SỐ 7, KDC TRUNG SƠN BÌNH HƯNG BÌNH CHÁNH
Πόλη	Ho Chi Minh Municipality
URL	https://www.booking.com/hotel/vn/elisa-tp-ho-chi-minh.html
Βαθμολογία	700
Περιγραφή	Situated 4.9 km from Saigon Exhibition and Convention Center, Elisa Boutique Saigon Hotel provides rooms with air conditioning in Ho Chi Minh City.

Πίνακας 1: Πληροφορίες πρώτου ξενοδοχείου

Στην συνέχεια, παρατίθενται οι πληροφορίες του δεύτερου ξενοδοχείου:

Όνομα	Chilli and Chum Apartment
Διεύθυνση	42/1A Hồ Hảo Hớn
Πόλη	Ho Chi Minh Municipality
URL	https://www.booking.com/hotel/vn/chilli-and-chum-apartment.html
Βαθμολογία	800
Περιγραφή	Located within the District 1 district in Ho Chi Minh City, Chilli and Chum Apartment has air conditioning, a balcony, and garden views.

Πίνακας 2: Πληροφορίες δεύτερου ξενοδοχείου

Επιπλέον, αξίζει να αναφερθεί ότι η συλλογή των δεδομένων πραγματοποιήθηκε σε διαδοχικές ημερομηνίες εντός ενός συγκεκριμένου εύρους κα εξετάστηκαν τα δεδομένα με βάση τις δύο παρακάτω παραλλαγές όσο αφορά την απόσταση της ημερομηνίας άφιξης από την εκάστοτε ημερομηνία:

- Απόσταση 7 ημερών
- Απόσταση 28 ημερών

Όσο αφορά τις τεχνικές προδιαγραφές της εφαρμογής, πρόκειται για μια Node.JS εφαρμογή, η οποία υλοποιήθηκε σε περιβάλλον Docker. Συνοπτικές πληροφορίες για τα δύο παραπάνω, παρατίθενται στις δύο επόμενες ενότητες αντίστοιχα. Επιπλέον ακολουθεί ενδεικτικό screenshot από το περιβάλλον πραγματοποίησης της πειραματικής διαδικασίας:

```
> price-monitoring@1.0.0 start
> node ./src/index.js

[info] Server is running at http://localhost:4000!
[message] Server is running at http://localhost:4000!
[info] Incoming request for /api/get-room-info/
[info] {"requestMethod":"POST","requestUrl":"/api/get-room-info/","userAgent":"PostmanRuntime/7.36.0","hashedIp":"::1","level":"info","message":"Incoming request for /api/get-room-info/","logDate":"2024-01-16T17:45:29.794Z"}
[info] Received task
[reqBody]:{"link":"https://www.booking.com/hotel/gr/a-amp-j-apartments-rhodes.e1.html","dayDistance":7,"stayingDays":1,"message":"Received task","level":"info","logDate":"2024-01-16T17:45:29.797Z"}
[debug] Navigated to page
[reqBody]:{"link":"https://www.booking.com/hotel/gr/a-amp-j-apartments-rhodes.e1.html","dayDistance":7,"stayingDays":1,"url":"https://www.booking.com/hotel/gr/a-amp-j-apartments-rhodes.e1.html?checkin=2024-01-23&checkout=2024-01-24&group_adults=1&req_adults=1&no_rooms=1&group_children=0&req_children=0&hpos=1&hapos=1","useProxy":false,"userAgent":"Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/114.0.0.0 Safari/537.36","level":"debug","message":"Navigated to page","logDate":"2024-01-16T17:45:52.418Z"}
[debug] Extracted data from page
[reqBody]:{"link":"https://www.booking.com/hotel/gr/a-amp-j-apartments-rhodes.e1.html","dayDistance":7,"stayingDays":1,"message":"Extracted data from page","level":"debug","logDate":"2024-01-16T17:45:52.919Z"}
[info] Completed task
[reqBody]:{"link":"https://www.booking.com/hotel/gr/a-amp-j-apartments-rhodes.e1.html","dayDistance":7,"stayingDays":1,"message":"Completed task","level":"info","logDate":"2024-01-16T17:45:52.920Z"}
[info] Incoming request for /api/get-room-info/
[info] {"requestMethod":"POST","requestUrl":"/api/get-room-info/","userAgent":"PostmanRuntime/7.36.0","hashedIp":"::1","level":"info","message":"Incoming request for /api/get-room-info/","logDate":"2024-01-16T17:50:59.849Z"}
[info] Received task
[reqBody]:{"link":"https://www.booking.com/hotel/gr/a-amp-j-apartments-rhodes.e1.html","dayDistance":7,"stayingDays":3,"message":"Received task","level":"info","logDate":"2024-01-16T17:50:59.850Z"}
[debug] Navigated to page
[reqBody]:{"link":"https://www.booking.com/hotel/gr/a-amp-j-apartments-rhodes.e1.html","dayDistance":7,"stayingDays":3,"url":"https://www.booking.com/hotel/gr/a-amp-j-apartments-rhodes.e1.html?checkin=2024-01-23&checkout=2024-01-26&group_adults=1&req_adults=1&no_rooms=1&group_children=0&req_children=0&hpos=1&hapos=1","useProxy":true,"userAgent":"Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/114.0.0.0 Safari/537.36 Edg/114.0.1823.51","level":"debug","message":"Navigated to page","logDate":"2024-01-16T17:51:04.426Z"}
[debug] Extracted data from page
[reqBody]:{"link":"https://www.booking.com/hotel/gr/a-amp-j-apartments-rhodes.e1.html","dayDistance":7,"stayingDays":3,"message":"Extracted data from page","level":"debug","logDate":"2024-01-16T17:51:05.062Z"}
[info] Completed task
[reqBody]:{"link":"https://www.booking.com/hotel/gr/a-amp-j-apartments-rhodes.e1.html","dayDistance":7,"stayingDays":3,"message":"Completed task","level":"info","logDate":"2024-01-16T17:51:05.063Z"}
```

Εικόνα 8: Περιβάλλον πραγματοποίησης πειραματικής διαδικασίας

2.2. Node.js

Η τεχνολογία Node.js [9] [20], αποτελεί ένα περιβάλλον εκτέλεσης το οποίο επιτρέπει τη δημιουργία πολλών και διαφορετικών εργαλείων και εφαρμογών στην πλευρά του εξυπηρετητή με τη χρήση της JavaScript. Δεν αποτελεί μια ακόμα γλώσσα προγραμματισμού, η οποία χρησιμοποιείται στην πλευρά του εξυπηρετητή, αλλά μια πλατφόρμα/περιβάλλον ανάπτυξης εφαρμογών, οι οποίες έχουν δημιουργηθεί σε γλώσσα JavaScript. Η JavaScript η οποία χρησιμοποιείται στη Node.js είναι σχεδιασμένη ώστε να λειτουργεί εκτός του περιβάλλοντος ενός φυλλομετρητή. Για το λόγο αυτό, έχουν αποκλειστεί συναρτήσεις και δυνατότητές της, οι οποίες ήταν αυστηρά συνδεδεμένες με το περιβάλλον του φυλλομετρητή. Αντίθετα, έχουν προστεθεί άλλες, οι οποίες βοηθούν στη συνεργασία με το λειτουργικό σύστημα του εξυπηρετητή, π.χ. τη διαχείριση φακέλων και αρχείων ή ενέργειες του πρωτοκόλλου HTTP.

Η Node.js δημιουργήθηκε από τον Ryan Dahl το 2009, είναι ανοικτού κώδικα και σχεδιασμένη να συνεργάζεται με πολλαπλές πλατφόρμες λειτουργικών συστημάτων, όπως με: Linux, macOS, Solaris, FreeBSD, OpenBSD και Microsoft Windows (σε όλες τις νεότερες εκδόσεις). Τα αρχεία που παράγει έχουν επέκταση .js, όπως όλα τα αρχεία κώδικα JavaScript.

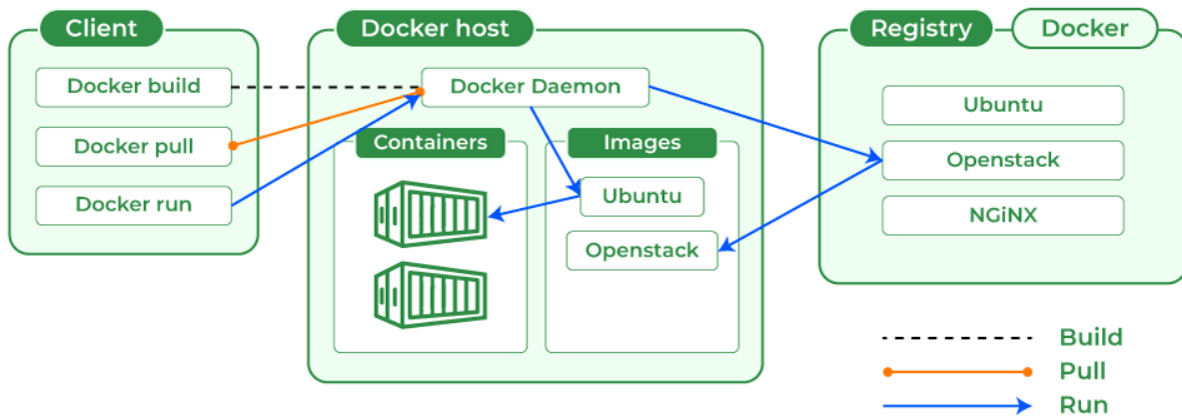
Πέντε βασικά χαρακτηριστικά της Node.js είναι τα παρακάτω:

- Ασύγχρονος Προγραμματισμός
- Επεκτασιμότητα
- Καθολική χρήση γλώσσας JavaScript
- Επικοινωνία σε πραγματικό χρόνο
- Χρήση βιβλιοθηκών μέσω του NPM (Node Package Manager)

2.3. Docker

Το περιβάλλον Docker [10] [21] είναι μια πλατφόρμα που απλοποιεί τη διαδικασία ανάπτυξης και εκτέλεσης εφαρμογών χρησιμοποιώντας τεχνολογία containers. Τα containers είναι ελαφριές, φορητές και αυτόνομες μονάδες (modules) που ενσωματώνουν μια εφαρμογή μαζί με τις εξαρτήσεις, τις βιβλιοθήκες και το περιβάλλον χρόνου εκτέλεσης. Η πλατφόρμα Docker επιτρέπει στους προγραμματιστές να ενσωματώνουν τις εφαρμογές τους σε containers και να εξασφαλίζουν συνέπεια σε διαφορετικά περιβάλλοντα, διευκολύνοντας την απρόσκοπτη μετακίνηση εφαρμογών μεταξύ περιβαλλόντων ανάπτυξης, δοκιμών και παραγωγής.

Ένα από τα βασικά οφέλη του Docker είναι η ικανότητά του να παρέχει συνέπεια στην ανάπτυξη εφαρμογών σε διάφορα περιβάλλοντα. Αυτή η συνέπεια εξαλείφει το πρόβλημα "λειτουργεί στο μηχάνημά μου" που συναντάται συχνά στην ανάπτυξη λογισμικού, μειώνοντας πιθανά προβλήματα και εξορθολογίζοντας τη διαδικασία ανάπτυξης.



Εικόνα 9: Αρχιτεκτονική Docker

Το Docker προωθεί επίσης την επεκτασιμότητα και την αποδοτικότητα των πόρων. Δεδομένου ότι τα containers μοιράζονται τον πυρήνα του λειτουργικού συστήματος του κεντρικού υπολογιστή και εκτελούνται ως απομονωμένες διαδικασίες, είναι πιο ελαφριά σε σύγκριση με τις παραδοσιακές εικονικές μηχανές. Αυτή η δυνατότητα βελτιστοποιεί τη χρήση των πόρων και διευκολύνει την κλιμάκωση των εφαρμογών. Η ευελιξία του Docker και η ικανότητα να ενορχηστρώνει εφαρμογές σε containers χρησιμοποιώντας εργαλεία όπως το Docker Compose και το Kubernetes το έχουν καταστήσει θεμελιώδη τεχνολογία όσο αφορά την ανάπτυξη λογισμικού. Αυτός είναι και ο λόγος για τον οποίο προτιμήθηκε για την ανάπτυξη της εφαρμογής.

2.4. Σύνοψη τεχνολογιών που χρησιμοποιήθηκαν

Στα πλαίσια της συγκεκριμένης ενότητας, παρατίθενται συνοπτικά οι τεχνολογίες, οι οποίες χρησιμοποιήθηκαν:

- Node.js: Περιβάλλον χρόνου εκτέλεσης για την εκτέλεση κώδικα JavaScript στην πλευρά του διακομιστή.
- Express.js: Ένα framework για Node.js, που χρησιμοποιείται για την κατασκευή της εφαρμογής διακομιστή.
- RabbitMQ: Ένας διαμεσολαβητής μηνυμάτων που χρησιμοποιείται για το χειρισμό κατανεμημένης επεξεργασίας εργασιών.
- Cheerio: Μα γρήγορη, εύελκτη και λιτή εφαρμογή του jQuery core για την ανάλυση HTML και τον χειρισμό του DOM. Χρησιμοποιείται για την εξαγωγή δεδομένων από σελίδες HTML.
- AMQP (Advanced Message Queuing Protocol): Ένα πρωτόκολλο ανταλλαγής μηνυμάτων που χρησιμοποιείται από το RabbitMQ για επικοινωνία μεταξύ διαφορετικών τμημάτων του συστήματος.

- Vault: Ένα εργαλείο για τη διαχείριση συνθηματικών και την προστασία ευαίσθητων δεδομένων. Χρησιμοποιείται για την ασφαλή αποθήκευση και πρόσβαση σε μεταβλητές περιβάλλοντος.
- dotenv: Μια μονάδα μηδενικής εξάρτησης που φορτώνει μεταβλητές περιβάλλοντος από ένα `.env` αρχείο σε μία διαδικασία `.env`.
- Winston: μια ευέλικτη βιβλιοθήκη καταγραφής για Node.js
- Loki: Μια βιβλιοθήκη καταγραφής για Node.js, η οποία συχνά χρησιμοποιείται συνδυαστικά με την βιβλιοθήκη Winston.
- httpagent: Ένα βοηθητικό πρόγραμμα για υποστήριξη διακομιστή μεσολάβησης HTTP(S) σε Node.js. Χρησιμοποιείται για τη δημιουργία ενός πράκτορα διακομιστή μεσολάβησης HTTP.
- got: Μια φιλική προς τον άνθρωπο και ισχυρή βιβλιοθήκη αιτημάτων HTTP για Node.js.
- user-agents: Μια βιβλιοθήκη για τη δημιουργία τυχαίων συμβολοσειρών User-Agent.
- Joi: Μία γλώσσα περιγραφής σχήματος και επικύρωσης δεδομένων για αντικείμενα JavaScript.

2.5. Βασικά αρχεία εφαρμογής

Στα πλαίσια της συγκεκριμένης ενότητας, και για λόγους κατανόησης της λειτουργικότητας της εφαρμογής παρατίθενται πληροφορίες σχετικά με τα βασικά αρχεία της εφαρμογής, που αφορούν:

- την λειτουργικότητά τους
- τον κώδικα υλοποίησής τους

2.5.1. Αρχείο `server.js`

Το συγκεκριμένο module εγκαθιστά έναν Express.js server, προκειμένου να χειριστεί τα εισερχόμενα HTTP αιτήματα. Λειτουργεί σαν σημείο εισόδου για την πραγματοποίηση επικοινωνίας με το Web Scraping σύστημα.

Όσο αφορά την λειτουργικότητά του, συνοψίζεται στα παρακάτω σημεία:

- **Ρύθμιση Server:**
 - Δημιουργεί έναν Express.js server.
 - Καθορίζει τη θύρα του σέρβερ (διαβάζοντας την τιμή από το αρχείο `.env`, αλλιώς χρησιμοποιεί την προκαθορισμένη τιμή 3000).

```

const express = require('express');
const app = express();
const PORT = process.env.PORT || 3000;
...
app.listen(PORT, () => {
  console.log( ⚡ Server is running at http://localhost:${PORT}! );
});

```

- **Παρακολούθηση Κατάστασης Server:**

- Αναποκρίνεται σε αιτήματα '/health-check' με ένα JSON αντικείμενο που δηλώνει την κατάσταση του σέρβερ.

```

app.get("/health-check", (_req, res) =>
  res.end(JSON.stringify({ status: "ok" })))
);

```

- **Επεξεργασία Εργασιών Scraping:**

- Αναλαμβάνει να λάβει εργασίες scraping μέσω αιτημάτων POST προς τη διαδρομή '/api/get-room-info'.
- Καταγράφει τα εισερχόμενα αιτήματα.
- Εκτελεί την εργασία scraping χρησιμοποιώντας το module 'scraper.js'.

```

const { executeTask: scraperTask } = require("../services/tasks/scraper");
...
app.post("/api/get-room-info", async (req, res) => {
  const payload = req.body;
  const log = logger.child({ reqBody: payload });
  try {
    log.info(" 📧 Received task")
    const result = await scraperTask(payload, log);
    log.info(" 🟢 Completed task");
    res.json(result);
  } catch (error) {

```

```

if (error instanceof ValidationErrorScrapperPayloadError) {
  res.status(400).json({ error: error.message });
} else {
  res.sendStatus(500);
  log.error("● Failed task", { error });
}
}
});

```

- **Καταγραφή Εργασιών:**
 - Καταγράφει τα εισερχόμενα αιτήματα, χρησιμοποιώντας ένα middleware για logging.
- **Διαχείριση Σφαλμάτων:**
 - Χειρίζεται σφάλματα κατά την εκτέλεση των εργασιών scraping.
 - Επιστρέφει κατάλληλους κωδικούς κατάστασης HTTP και μηνύματα σφάλματος.

2.5.2. Αρχείο incoming-request.js

Το αρχείο αυτό καταγράφει τα εισερχόμενα HTTP αιτήματα και παρέχει μεταδεδομένα, ενισχύοντας τις δυνατότητες παρακολούθησης και αποσφαλμάτωσης της υλοποίησης. Η λειτουργικότητά του συνοψίζεται στα παρακάτω σημεία:

- **Εξαγωγή Πληροφοριών:**
 - Εξάγει σχετικές πληροφορίες από τα εισερχόμενα αιτήματα.
 - Χρησιμοποιεί το αρχείο logger.js για την καταγραφή αυτών των πληροφοριών.

```

const { logger } = require("../services/logger");
const Browser = require("browser");

const getUserAgent = (req) => {
  if (!req.headers["user-agent"] || req.headers["user-agent"].length < 1) return "";
  const userAgent = Browser.parse(req.headers["user-agent"]);
  if (userAgent.browser.name === "") {
    userAgent.browser = { name: req.headers["user-agent"] };
  }
}

```

```
return req.headers["user-agent"];
};
```

- **Ανάκτηση URI και IP:**

- Καθορίζει το URI του αιτήματος, αγνοώντας τυχόν παραμέτρους στο URL.
- Ανακτά τη διεύθυνση IP του αιτούντος, είτε από το header x-forwarded-for είτε από το request IP.

```
const incomingRequestLogger = async (req, _res, next) => {
  // Καθορισμός URI
  const requestUri = (req.originalUrl || req.url).split("?")[0];

  // Ανάκτηση IP
  let ip;
  if (req.headers["x-forwarded-for"] && req.headers["x-forwarded-for"].length > 0) {
    const forwardedIps = (req.headers["x-forwarded-for"]).split(",");
    ip = forwardedIps[0];
  } else {
    ip = req.ip;
  }

  ...
};
```

- **Καταγραφή Πληροφοριών Αιτήματος:**

- Καταγράφει τα στοιχεία του εισερχομένου αιτήματος, όπως η μέθοδος, το URI, το User Agent, και το hashed IP.

```
const incomingRequestLogger = async (req, _res, next) => {
  ...

  logger.info(`Incoming request for ${requestUri}`, {
    info: {
      requestMethod: req.method,
      requestUri,
      userAgent: getUserAgent(req),
    }
  });
};
```

```

    hashedIp: ip
  },
});

return next();
};

module.exports = {
  incomingRequestLogger
};

```

2.5.3. Αρχείο logger.js

Το αρχείο αυτό παραμετροποιεί και διαχειρίζεται την καταγραφή πληροφοριών για ολόκληρη την υλοποίηση, καθιστώντας ευκολότερη την καταγραφή γεγονότων, σφαλμάτων και πληροφοριών. Η λειτουργικότητά του συνοψίζεται στα παρακάτω σημεία:

- Χρησιμοποιεί την βιβλιοθήκη Winston, προκειμένου να δημιουργηθεί ένας ευέλικτος και παραμετροποιήσιμος καταγραφέας.

```

const winston = require('winston');
const LokiTransport = require('winston-loki');
const chalk = require('chalk');
const { inspect } = require('util');
const stripAnsi = (() => {
  ...
})();

```

- Το επίπεδο καταγραφής ορίζεται στο 'silly'. Τα επίπεδα καταγραφής καθορίζουν το επίπεδο σημαντικότητας των καταγραφών, με το 'silly' να είναι το πιο λεπτομερές.

```

const logger = winston.createLogger({
  level: 'silly',
  ...
});

```

- Δημιουργία ευανάγνωστων καταγραφών

```
format: winston.format.combine(  
  errorObjectFormat(),  
  winston.format.json()  
),
```

- Καταγραφή στο Loki

```
new LokiTransport({  
  host: process.env.LOKI_HOST,  
  ...  
})  
...  
labels: { service: process.env.SERVICE_TYPE },  
...  
// Ενημέρωση σε Περίπτωση Σφάλματος Σύνδεσης  
onConnectionError: (err) => console.error(err)
```

Το συγκεκριμένο αρχείο έχει χρησιμοποιηθεί σε διάφορα modules της εφαρμογής προκειμένου να καταγράψει σημαντικές πληροφορίες γεγονότων, σφαλμάτων και αποσφαλμάτωσης.

2.5.4. Αρχείο rabbitmq.js

Το συγκεκριμένο αρχείο διαχειρίζεται την επικοινωνία με το RabbitMQ, ένα σύστημα αναμονής μηνυμάτων, προκειμένου να κατανέμει και να διαχειρίζεται τις εργασίες scraping. Η λειτουργικότητα του αρχείου αυτού συνοψίζεται στα παρακάτω σημεία:

- Αποστολή μίας εργασίας στην ουρά (sendTaskToQueue):
 - Χρησιμοποιείται για την αποστολή μιας εργασίας (task) σε μια συγκεκριμένη ουρά.
 - Δεν είναι υπεύθυνη για τη διαχείριση της εργασίας ή την παρακολούθηση των αποτελεσμάτων.

- Η λειτουργία της είναι συγκεκριμένη: συνδέεται, δημιουργεί ένα κανάλι, επιβεβαιώνει την ύπαρξη της ουράς, και αποστέλλει την εργασία. Στη συνέχεια, κλείνει τη σύνδεση.

```
async function sendTaskToQueue(queue, payload, log) {
  const connection = await amqp.connect(process.env.RABBIT_MQ_HOST);
  const channel = await connection.createChannel();
  await channel.assertQueue(queue);
  await channel.sendToQueue(queue, Buffer.from(JSON.stringify(payload)));
  log.info("📦 Sent task to queue");
  await channel.close();
  await connection.close();
}
```

- Εγγραφή μιας εργασίας στην ουρά (registerTaskToQueue)
 - Χρησιμοποιείται για τη δημιουργία ενός καταναμημένου συστήματος εργασιών.
 - Διαχειρίζεται την επεξεργασία των εργασιών από μια ουρά, εκτελεί τη συγκεκριμένη εργασία (taskFunction), και ανταποκρίνεται στα αποτελέσματα.
 - Είναι υπεύθυνη για τη διαχείριση των επαναλήψεων, των αποτυχιών, και της ουράς αποτυχημένων εργασιών.

```
async function registerTaskToQueue(queue, taskFunction, options) {
  const connection = await amqp.connect(process.env.RABBIT_MQ_HOST);
  const channel = await connection.createChannel();
  const dlqName = `${queue}-dlq`;
  const resultsQueueName = `${queue}-results`;

  await channel.assertQueue(queue);
  await channel.assertQueue(dlqName);
  await channel.assertQueue(resultsQueueName);

  // Αυξάνει το πλήθος των μηνυμάτων που μπορούν να επεξεργαστούν ταυτόχρονα
  channel.prefetch(options.concurrency);
  channel.consume(
    queue,
    async (message) => {
      const payload = JSON.parse(message.content.toString());
```

```

const taskId = uuid();
const taskTimestamp = new Date();
const log = logger.child({ payload, taskId, taskTimestamp });
log.info(" 📁 Received task");
try {
  const data = await taskFunction(payload, log);
  log.info(" 🟢 Completed task");
  // Αποστολή αποτελέσματος εργασίας σε διαφορετική ουρά
  channel.sendToQueue(resultsQueueName, Buffer.from(JSON.stringify({
    taskId,
    payload,
    data,
    timestamp: taskTimestamp
  })));
  // Επιβεβαίωση επεξεργασίας του μηνύματος
  channel.ack(message);
} catch (error) {
  log.error(" 🚫 Failed task", { error });
  if (!payload.retryCount) {
    payload.retryCount = 0;
  }
  if (payload.retryCount < options.maxRetries) {
    // Επανάληψη εργασίας μετά από αποτυχία
    payload.retryCount = payload.retryCount + 1;
    channel.nack(message, false, false);
    channel.sendToQueue(queue, Buffer.from(JSON.stringify(payload)));
    log.info(" 🔄 Sent task to queue for another retry, retries: ${payload.retryCount}`, {
      retryCount: payload.retryCount
    });
  } else {
    // Αποτυχία μετά από το μέγιστο αριθμό επαναλήψεων
    const enumeratedErrorObject = enumerateErrorObject(error);
    channel.nack(message, false, false);
  }
}

```

```

channel.sendToQueue(dlqName, Buffer.from(JSON.stringify({
  taskId,
  payload,
  error: enumeratedErrorObject,
  timestamp: taskTimestamp
})));
log.info(' 📦 Sent task to DLQ after failing ${options.maxRetries} retries', {
  retryCount: payload.retryCount,
  maxRetries: options.maxRetries
});
}
}
},
{
  autoAck: false,
  noAck: false
}
);

logger.info(" 🔗 Linked process to scrapper queue ... waiting for tasks", {
  queue,
  options
});
}

```

2.5.5. Αρχείο worker.js

Το συγκεκριμένο αρχείο διαχειρίζεται την εκτέλεση των εργασιών scraping με καταναμημένο τρόπο, χρησιμοποιώντας το RabbitMQ για διαχείριση και επεξεργασία της ουράς εργασιών. Όσο αφορά την λειτουργικότητά του, συνοψίζεται στα παρακάτω σημεία:

- Εισάγει λογική επεξεργασίας στις RabbitMQ ουρές, προσθέτοντας ρυθμίσεις που σχετίζονται με τον συγχρονισμό και το μέγιστο πλήθος προσπαθειών.
- Λαμβάνει εργασίες από την ουρά, τις επεξεργάζεται και χειρίζεται περιπτώσεις αποτυχιών.

- Υποστηρίζει επαναλήψεις εργασιών σύμφωνα με ένα προκαθορισμένο όριο.
- Μεταφέρει τις αποτυχημένες εργασίες σε μια συγκεκριμένη ουρά (DLQ) όταν φτάσουν το όριο προσπαθειών
- Αρχικοποιεί την διαδικασία scraping.

Ακολουθεί ο κώδικας του αρχείου:

```
const { registerTaskToQueue } = require("../services/rabbitmq");
const { executeTask: scrapperTask } = require("../services/tasks/scrapper");

async function main() {
  await registerTaskToQueue("scrapper-tasks", scrapperTask, {
    concurrency: process.env.CONCURRENCY ? +process.env.CONCURRENCY : 3,
    maxRetries: process.env.MAX_RETRIES ? +process.env.MAX_RETRIES : 5,
  });
}

module.exports = {
  main
}
```

2.5.6. Αρχείο scrapper.js

Πρόκειται για το κύριο αρχείο της εφαρμογής, το οποίο είναι υπεύθυνο για την εξαγωγή δεδομένων που σχετίζονται με ξενοδοχεία και δωμάτια από την ιστοσελίδα Booking.com. Η λειτουργικότητα του συγκεκριμένου αρχείου συνοψίζεται στην συνέχεια:

- Συνάρτηση executeTask, η οποία υποβάλλει το ερώτημα, προκειμένου να ξεκινήσει η διαδικασία scraping:
 - Επικυρώνει το payload, εξάγει πληροφορίες και δημιουργεί ένα URL για το scraping των ξενοδοχείων.
 - Χρησιμοποιεί το ProxyService για τη λήψη του περιεχομένου της σελίδας (HTML) με προαιρετική χρήση proxy.
 - Χρησιμοποιεί το Cheerio για την ανάλυση HTML και εξάγει δεδομένα δωματίων. Στο σημείο αυτό αξίζει να αναφερθεί ότι αρχικά είχε χρησιμοποιηθεί η βιβλιοθήκη Puppeteer,

αλλά αποφασίστηκε να αντικατασταθεί για λόγους χρόνου εκτέλεσης της πειραματικής διαδικασίας (σε συνεννόηση με τον DevOps).

- Επιστρέφει τα εξαγόμενα δεδομένα.

```
async function executeTask(payload, log) {
  const { link, dayDistance, stayingDays } = validateScrapperTaskPayload(payload);
  const url = new URL(link);
  url.search = "";
  url.hash = "";
  const checkInDate = DateUtil.addDaysFromCurrentTimestamp(dayDistance);
  let checkOutDate;
  if (stayingDays)
  {
    checkOutDate = DateUtil.addDaysFromGivenISO(stayingDays, checkInDate);
  } else
  {
    checkOutDate = DateUtil.addDaysFromGivenISO(1, checkInDate);
  }
  url.searchParams.append("checkin", checkInDate);
  url.searchParams.append("checkout", checkOutDate);
  url.searchParams.append("group_adults", "1");
  url.searchParams.append("req_adults", "1");
  url.searchParams.append("no_rooms", "1");
  url.searchParams.append("group_children", "0");
  url.searchParams.append("req_children", "0");
  url.searchParams.append("hpos", "1");
  url.searchParams.append("hapos", "1");
  const proxyService = new ProxyService({ useProbability: PROXY_USE_PROBABILITY });
  const {
    response: page,
    useProxy,
    userAgent
  } = await proxyService.get(url.href);
  log.debug("Navigated to page", { url: url.href, useProxy, userAgent });
  const $ = cheerio.load(page.body)
```

```
const data = fetchRoomDataFromPage($, url.href);
log.debug("Extracted data from page");
return data;
}
```

- Ακολουθεί ο κώδικας της συνάρτησης `fetchRoomDataFromPage`, η οποία προσκομίζει τα αποτελέσματα σχετικά με τις πληροφορίες των δωματίων
 - Επιλογή Στοιχείων Δωματίων

```
const roomElements = chr("table#hprt-table > tbody > tr");
```

- Τύπος Δωματίου και Παροχές:

```
roomType = chr(roomFields[0]).find("a.hprt-roomtype-link").text().trim();
facilities = [...chr(roomFields[0]).find(".hprt-facilities-facility")].map((element) => {
  return chr(element).text().trim().replace(/\n/g, "");
}).filter((el) => el.length !== 0);
```

- Διαθέσιμος αριθμός επισκεπτών

```
const sleepsEl = chr(roomFields[startingIndex + 1])
.find("div.c-occupancy-icons,div.wholesalers_table__occupancy__icons");
let sleeps = 0;

if (sleepsEl.find("span.c-occupancy-icons__multiplier-number").length > 0) {
  sleeps = +sleepsEl.find("span.c-occupancy-icons__multiplier-number").text()
} else {
  sleeps = chr(sleepsEl).find("i").length;
}
```

- Εξαγωγή τιμής

```
const price = +chr(roomFields[startingIndex + 2])
.find("div.hprt-price-block div.bui-price-display__value > span")
.text().match(/[0-9\.,]+/)[0]
.replace(/\./, ",");
```

- Χαρακτηριστικά όπως δυνατότητα ακύρωσης, πρωινό κλπ.

```
const conditionDescriptions = [...chr(roomFields[startingIndex + 3]).find("ul.hprt-conditions-bui > li")].map((element) => {
  return chr(element).text().replace(/\n|•/g, "").trim();
}).filter((el) => el.length !== 0);
```

- validateScrapperResult, η οποία πραγματοποιεί επικύρωση στα αποτελέσματα με τη βοήθεια της Joi

```
function validateScrapperResult(obj)
{
  const schema = Joi.object({
    name: Joi.string().required(),
    description: Joi.string().required(),
    location: Joi.object({
      address: Joi.string().required(),
      city: Joi.string().required(),
      longitude: Joi.number().required(),
      latitude: Joi.number().required(),
    }).required(),
    url: Joi.string().uri().required(),
    hotelType: Joi.string().valid('HOTEL', 'MOTEL').required(),
    score: Joi.number().required(),
    rooms: Joi.array()
      .items(
        Joi.object({
          description: Joi.string().required(),
          name: Joi.string().required(),
          roomViews: Joi.array().items(
            Joi.object({
              sleeps: Joi.number().required(),
              price: Joi.number().required(),
              quantity: Joi.number().required(),
              distanceDays: Joi.number().required(),
              attributes: Joi.array().items(Joi.string()).required(),
            })
          )
        })
      )
  });
}
```

```

        ).required(),
        attributes: Joi.array().items(Joi.string()).required(),
    )))
    .required(),
});
const { error, value } = schema.validate(obj);
if (error) {
    throw new ValidationErrorScrapperResultError(`Validation error: ${error.details[0].message}`);
}
return obj;
}

```

- Τέλος, ο κώδικας της συνάρτησης `validateScrapperTaskPayload`, η οποία πραγματοποιεί επικύρωση στα στοιχεία του αιτήματος, το οποίο υποβάλλεται με σκοπό την εκκίνηση της διαδικασίας `scraping`:

```

function validateScrapperTaskPayload(obj) {
    const schema = Joi.object({
        link: Joi.string().required(),
        dayDistance: Joi.number().required(),
        stayingDays: Joi.number().optional(),
        retryCount: Joi.number().optional(),
    });
    const { error, value } = schema.validate(obj);
    if (error) {
        throw new ValidationErrorScrapperPayloadError(`Validation error: ${error.details[0].message}`);
    }
    return value;
}

module.exports = {
    executeTask
}

```

2.5.7. Αρχείο date.js

Το συγκεκριμένο αρχείο παρέχει λειτουργίες για διαχείριση ημερομηνιών, όπως υπολογισμός διαφοράς μεταξύ ημερομηνιών και μορφοποίηση ημερομηνιών. Η λειτουργικότητά του συνοψίζεται στα παρακάτω σημεία:

- Διασφαλίζει ότι τα αλφαριθμητικά που αντιστοιχούν στις ημερομηνίες είναι της μορφής «YYYY-MM-DD».
- Υπολογίζει την διαφορά σε ημέρες ανάμεσα σε δύο ημερομηνίες ή ανάμεσα σε μια ημερομηνία και μια χρονοσφραγίδα (timestamp)
- Προσθέτει ένα συγκεκριμένο πλήθος ημερών σε μια δοσμένη ημερομηνία.
- Υποστηρίζει διαχείριση ημερομηνιών όσο αφορά την δημιουργία ημερομηνιών άφιξης και αναχώρησης στις εργασίες scraping.

Ακολουθεί ο κώδικας του αρχείου:

```
const dateFns = require("date-fns");
/** Converts X to 0X */
function formatDigitToDateString(digit) {
  return digit.split("-").map((digit) => {

    if (+digit < 10 && digit.length === 1) {

      return "0" + digit;
    }
    return digit;
  }).join("-");
}
/** Date format: YYYY-MM-DD */
function getDifferenceInDaysFromNow(to) {
  to = formatDigitToDateString(to);
  return dateFns.differenceInCalendarDays(dateFns.parseISO(to), new Date());
}
function getDifferenceInDays(from, to) {
  to = formatDigitToDateString(to);
  from = formatDigitToDateString(from);
```

```

return dateFns.differenceInCalendarDays(dateFns.parseISO(to), dateFns.parseISO(from));
}
/** Date format: YYYY-MM-DD */
function addDaysFromCurrentTimestamp(days) {
  const date = dateFns.addDays(new Date(), days);
  return dateFns.format(date, "yyyy-MM-dd");
}
function addDaysFromGivenISO(days, timestamp) {
  const date = dateFns.addDays(dateFns.parseISO(timestamp), days);
  return dateFns.format(date, "yyyy-MM-dd");
}

module.exports = {
  getDifferenceInDaysFromNow,
  getDifferenceInDays,
  addDaysFromCurrentTimestamp,
  addDaysFromGivenISO
}

```

2.5.8. Αρχείο exceptions.js

Στην συγκεκριμένη περίπτωση ορίζονται κλάσεις εξαιρέσεων, προκειμένου να μπορούν υποστούν διαχείριση σφάλματα που σχετίζονται με ελέγχους εγκυρότητας. Η λειτουργικότητα του συνοψίζεται στα παρακάτω σημεία:

- Οι κλάσεις `ValidationErrorScrapperResultError` και `ValidationErrorScrapperPayloadError` επεκτείνουν την κλάση `Error`, προκειμένου να αναπαριστούν πιο συγκεκριμένα σφάλματα.
- Οι εξαιρέσεις αυτές εμφανίζονται όταν τα δεδομένα εισόδου ή τα αποτελέσματα αποτυγχάνουν στο να καλύψουν τα κριτήρια εγκυρότητας.

Ακολουθεί ο κώδικας του αρχείου:

```

class ValidationErrorScrapperResultError extends Error {
  constructor(message) {
    super(message);
  }
}

```

```

    }
  }
  class ValidationErrorScrapperPayloadError extends Error {
    constructor(message) {
      super(message);
    }
  }
}
module.exports = {
  ValidationErrorScrapperResultError,
  ValidationErrorScrapperPayloadError
}

```

2.5.9. Αρχείο proxy.js

Το συγκεκριμένο αρχείο επιτρέπει στην εφαρμογή να υποβάλλει αιτήματα μέσω ενός proxy server, ενισχύοντας την ασφάλεια και την ανωνυμοποίηση των δεδομένων. Η λειτουργικότητά του συνοψίζεται στα παρακάτω σημεία:

- Χρησιμοποιεί την βιβλιοθήκη hparent προκειμένου να δημιουργηθεί ένας proxy agent με συγκεκριμένα στοιχεία πρόσβασης.
- Δημιουργεί HTTP αιτήματα για την πρόσβαση στις σελίδες του booking.com.

Ακολουθεί ο κώδικας:

```

const { HttpsProxyAgent } = require('hpagent');
const got = require('got');
const UserAgent = require("user-agents")
const { probabilityOfTrue } = require("../utils/random")
class ProxyService {
  constructor({ useProbability = 1 }) {
    this.useProbability = useProbability;
    this._agent = new HttpsProxyAgent({

```

```

    proxy:
    `https://${process.env.PROXY_USERNAME}:${process.env.PROXY_PASSWORD}@${process.env.
PROXY_HOST}:${process.env.PROXY_PORT}`
    });
  }
  async get(link) {
    const userAgent = new UserAgent();
    const useProxy = probabilityOfTrue(this.useProbability);
    const response = await got.get(link, {
      ...(useProxy && {
        agent: {
          https: this._agent
        },
      }),
      headers: {
        "User-Agent": userAgent.data.userAgent
      }
    });
    return { response, userAgent: userAgent.data.userAgent, useProxy };
  }
}
module.exports = ProxyService;

```


Ακολουθεί screenshot με δεύτερο μέρος από τα αποτελέσματα:

```
"hotelType": "HOTEL",
"rooms": [
  {
    "name": "Κατοικία 2 Υπνοδωματίων",
    "description": "not available",
    "attributes": [
      "Ολόκληρη Παραθεριστική κατοικία",
      "60 m2",
      "Ιδιωτική κουζίνα",
      "Ιδιωτικό μπάνιο",
      "Μπαλκόνι",
      "Θέα στη θάλασσα",
      "Θέα στο βουνό",
      "Θέα στην Πόλη",
      "Κλιματισμός",
      "Τηλεόραση επίπεδης οθόνης",
```

Εικόνα 11: Παράδειγμα παραγόμενων αποτελεσμάτων - Μέρος 2

Στο μέρος αυτό εμφανίζονται πληροφορίες σχετικά με τις παροχές του δωματίου (εμβαδόν, θέα, κλιματισμός, τηλεόραση κτλ.). Ακολουθεί και ένα τρίτο μέρος, το οποίο παρουσιάζεται στην συνέχεια:

```
],
"roomViews": [
  {
    "attributes": [
      "Δωρεάν ακύρωση Πριν από 18 Ιανουαρίου 2024",
      "Πληρωμή εκ των Προτέρων",
      "Πλήρως επιστρέψιμη χρέωση μέσα στην Περίοδο δωρεάν ακύρωσης"
    ],
    "price": 242,
    "sleeps": 1,
    "distanceInDays": 7,
    "quantity": 1
  },
  {
    "attributes": [
      "Δωρεάν ακύρωση Πριν από 18 Ιανουαρίου 2024",
```

Εικόνα 12: Παράδειγμα αποτελεσμάτων - Μέρος 3

Στο τρίτο αυτό μέρος παρουσιάζονται πληροφορίες όπως:

- Ημερομηνία μέχρι την οποία επιτρέπεται δωρεάν ακύρωση
- Τρόπος πραγματοποίησης πληρωμής
- Τρόπος επιστροφής χρημάτων μέχρι την ημερομηνία δωρεάν ακύρωσης
- Τιμή
- Απόσταση σε ημέρες μεταξύ της ημερομηνίας καταγραφής και της ημερομηνίας άφιξης
- Ποσότητα

ΚΕΦΑΛΑΙΟ 3: CASE STUDY

3.1. Στοιχεία Δωματίων

Στους δύο παρακάτω πίνακες παρουσιάζονται τα στοιχεία των δωματίων των δύο ξενοδοχείων, που αποτελούν το σημείο ενδιαφέροντος. Στο σημείο αυτό αξίζει να αναφερθεί ότι έχουν ληφθεί δεδομένα από δύο δωμάτια (με κωδικό 229 και 233) όσο αφορά το πρώτο ξενοδοχείο και δεδομένα από ένα δωμάτιο (με κωδικό 231) από το δεύτερο ξενοδοχείο, οι τιμές είναι σε Ευρώ.

Οι πληροφορίες του πρώτου δωματίου του πρώτου ξενοδοχείου:

Κωδικός Δωματίου	229
Είδος Δωματίου	Deluxe Double Room
Εύρος Καταγραφών	23/12/2023 – 05/01/2024
Πολιτική Ακύρωσης	Δωρεάν
Χρονικό Διάστημα για άφιξη	7 και 28 ημέρες
Χρονικό Διάστημα για δωρεάν ακύρωση	5 και 26 ημέρες

Πίνακας 3: Πληροφορίες Δωματίου 1 - Ξενοδοχείου 1

Οι πληροφορίες του δεύτερου δωματίου του πρώτου ξενοδοχείου:

Κωδικός Δωματίου	233
Είδος Δωματίου	King Room
Εύρος Καταγραφών	23/12/2023 – 05/01/2024
Πολιτική Ακύρωσης	Δωρεάν
Χρονικό Διάστημα για άφιξη	7 και 28 ημέρες
Χρονικό Διάστημα για δωρεάν ακύρωση	5 και 26 ημέρες

Πίνακας 4: Πληροφορίες Δωματίου 2 - Ξενοδοχείου 1

Τέλος, οι πληροφορίες του πρώτου δωματίου του δεύτερου ξενοδοχείου:

Κωδικός Δωματίου	231
Είδος Δωματίου	Deluxe King Studio
Εύρος Καταγραφών	1/11/2023 – 05/01/2024
Πολιτική Ακύρωσης	Δωρεάν
Χρονικό Διάστημα για άφιξη	7 και 28 ημέρες
Χρονικό Διάστημα για δωρεάν ακύρωση	5 και 26 ημέρες

Πίνακας 5: Πληροφορίες Δωματίου 1 - Ξενοδοχείου 2

3.2. Εξέλιξη τιμών δωματίων πρώτου ξενοδοχείου

Αρχικά παρατίθεται το γράφημα, το οποίο παρουσιάζει τα αποτελέσματα που προέκυψαν όσο αφορά την εξέλιξη της τιμής του πρώτου δωματίου στο πρώτο ξενοδοχείο έχοντας ως χρονικό ορίζοντα 7 ημέρες μεταξύ της ημερομηνίας λήψης των δεδομένων και της ημερομηνίας άφιξης:



Εικόνα 13: Εξέλιξη τιμής - Ξενοδοχείο 1 - Δωμάτιο 1 - 7 ημέρες

Στην συνέχεια, παρατίθεται το γράφημα, το οποίο παρουσιάζει τα αποτελέσματα που προέκυψαν όσο αφορά την εξέλιξη της τιμής του πρώτου δωματίου στο πρώτο ξενοδοχείο έχοντας ως χρονικό ορίζοντα 28 ημέρες μεταξύ της ημερομηνίας λήψης των δεδομένων και της ημερομηνίας άφιξης:



Στην συνέχεια, παρατίθεται το γράφημα, το οποίο παρουσιάζει τα αποτελέσματα που προέκυψαν όσο αφορά την εξέλιξη της τιμής του δεύτερου δωματίου στο πρώτο ξενοδοχείο έχοντας ως χρονικό ορίζοντα 7 ημέρες μεταξύ της ημερομηνίας λήψης των δεδομένων και της ημερομηνίας άφιξης:



Εικόνα 15: Εξέλιξη τιμής - Ξενοδοχείο 1 - Δωμάτιο 2 - 7 ημέρες

Στην συνέχεια, παρατίθεται το γράφημα, το οποίο παρουσιάζει τα αποτελέσματα που προέκυψαν όσο αφορά την εξέλιξη της τιμής του δεύτερου δωματίου στο πρώτο ξενοδοχείο έχοντας ως χρονικό ορίζοντα 28 ημέρες μεταξύ της ημερομηνίας λήψης των δεδομένων και της ημερομηνίας άφιξης:



Εικόνα 16: Εξέλιξη τιμής - Ξενοδοχείο 1 - Δωμάτιο 2 - 28 ημέρες

3.3. Εξέλιξη τιμών δωματίων δεύτερου ξενοδοχείου

Αρχικά, παρατίθεται το γράφημα, το οποίο παρουσιάζει τα αποτελέσματα που προέκυψαν όσο αφορά την εξέλιξη της τιμής του πρώτου δωματίου στο δεύτερο ξενοδοχείο έχοντας ως χρονικό ορίζοντα 7 ημέρες μεταξύ της ημερομηνίας λήψης των δεδομένων και της ημερομηνίας άφιξης:



Εικόνα 17: Εξέλιξη τιμής - Ξενοδοχείο 2 - Δωμάτιο 1 - 7 ημέρες

Στην συνέχεια, παρατίθεται το γράφημα, το οποίο παρουσιάζει τα αποτελέσματα που προέκυψαν όσο αφορά την εξέλιξη της τιμής του πρώτου δωματίου στο δεύτερο ξενοδοχείο έχοντας ως χρονικό ορίζοντα 28 ημέρες μεταξύ της ημερομηνίας λήψης των δεδομένων και της ημερομηνίας άφιξης:



Εικόνα 18: Εξέλιξη τιμής - Ξενοδοχείο 2 - Δωμάτιο 1 - 28 ημέρες

3.4. Συγκριτικά στοιχεία

Στον παρακάτω συγκριτικό πίνακα, παρουσιάζεται η μέση τιμή των δωματίων των ξενοδοχείων για την περίπτωση των 7 ημερών μέχρι την άφιξη:

Ξενοδοχείο	Δωμάτιο	Μέση Τιμή
Ξενοδοχείο 1	Δωμάτιο 1	27.73
Ξενοδοχείο 1	Δωμάτιο 2	32.39
Ξενοδοχείο 2	Δωμάτιο 1	33.30

Πίνακας 6: Μέση τιμή δωματίων - 7 ημέρες

Επιπλέον, στον παρακάτω συγκριτικό πίνακα, παρουσιάζεται η μέση τιμή των δωματίων των ξενοδοχείων για την περίπτωση των 28 ημερών μέχρι την άφιξη:

Ξενοδοχείο	Δωμάτιο	Μέση Τιμή
Ξενοδοχείο 1	Δωμάτιο 1	30.29
Ξενοδοχείο 1	Δωμάτιο 2	33.71
Ξενοδοχείο 2	Δωμάτιο 1	33.95

Πίνακας 7: Μέση τιμή δωματίων - 28 ημέρες

Στον ακόλουθο συγκριτικό πίνακα, παρουσιάζεται η μέγιστη τιμή των δωματίων των ξενοδοχείων για την περίπτωση των 7 ημερών μέχρι την άφιξη:

Ξενοδοχείο	Δωμάτιο	Μέγιστη Τιμή
Ξενοδοχείο 1	Δωμάτιο 1	41
Ξενοδοχείο 1	Δωμάτιο 2	56
Ξενοδοχείο 2	Δωμάτιο 1	38

Πίνακας 8: Μέγιστη τιμή δωματίων - 7 ημέρες

Στον επόμενο συγκριτικό πίνακα, παρουσιάζεται η μέγιστη τιμή των δωματίων των ξενοδοχείων για την περίπτωση των 28 ημερών μέχρι την άφιξη:

Ξενοδοχείο	Δωμάτιο	Μέγιστη Τιμή
Ξενοδοχείο 1	Δωμάτιο 1	38
Ξενοδοχείο 1	Δωμάτιο 2	51
Ξενοδοχείο 2	Δωμάτιο 1	38

Πίνακας 9: Μέγιστη τιμή δωματίων - 28 ημέρες

Επιπροσθέτως, στον παρακάτω συγκριτικό πίνακα, παρουσιάζεται η ελάχιστη τιμή των δωματίων των ξενοδοχείων για την περίπτωση των 7 ημερών μέχρι την άφιξη:

Ξενοδοχείο	Δωμάτιο	Ελάχιστη Τιμή
Ξενοδοχείο 1	Δωμάτιο 1	15
Ξενοδοχείο 1	Δωμάτιο 2	13
Ξενοδοχείο 2	Δωμάτιο 1	32

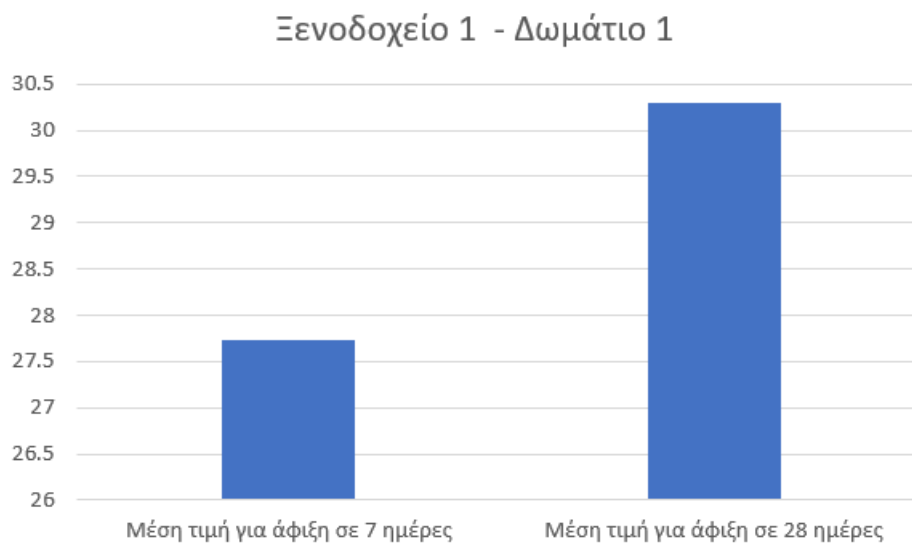
Πίνακας 10: Ελάχιστη τιμή δωματίων - 7 ημέρες

Τέλος, στον παρακάτω συγκριτικό πίνακα, παρουσιάζεται η ελάχιστη τιμή των δωματίων των ξενοδοχείων για την περίπτωση των 28 ημερών μέχρι την άφιξη:

Ξενοδοχείο	Δωμάτιο	Ελάχιστη Τιμή
Ξενοδοχείο 1	Δωμάτιο 1	25
Ξενοδοχείο 1	Δωμάτιο 2	22
Ξενοδοχείο 2	Δωμάτιο 1	32

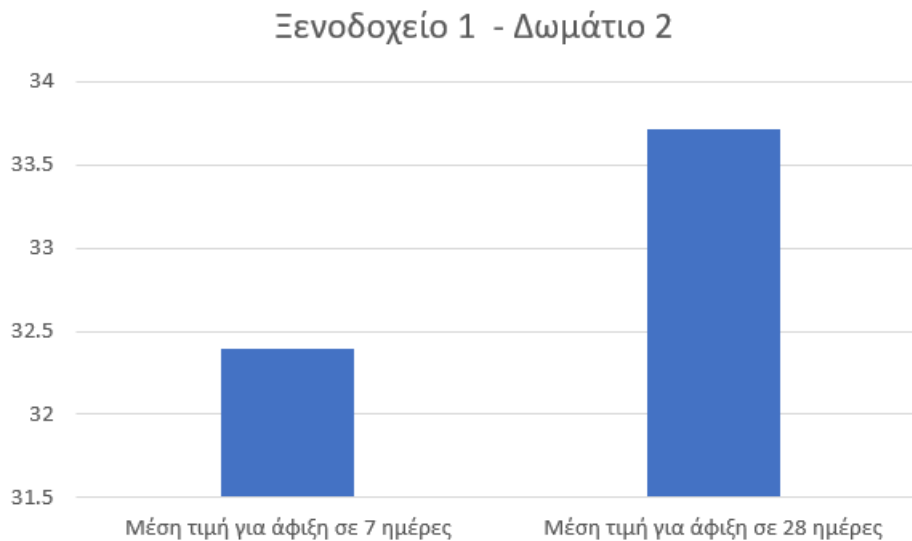
Πίνακας 11: Ελάχιστη τιμή δωματίων - 28 ημέρες

Στο επόμενο γράφημα, και για καλύτερη οπτικοποίηση των αποτελεσμάτων, παρουσιάζονται συγκριτικά στοιχεία όσο αφορά την μέση τιμή για το πρώτο δωμάτιο του πρώτου ξενοδοχείου :



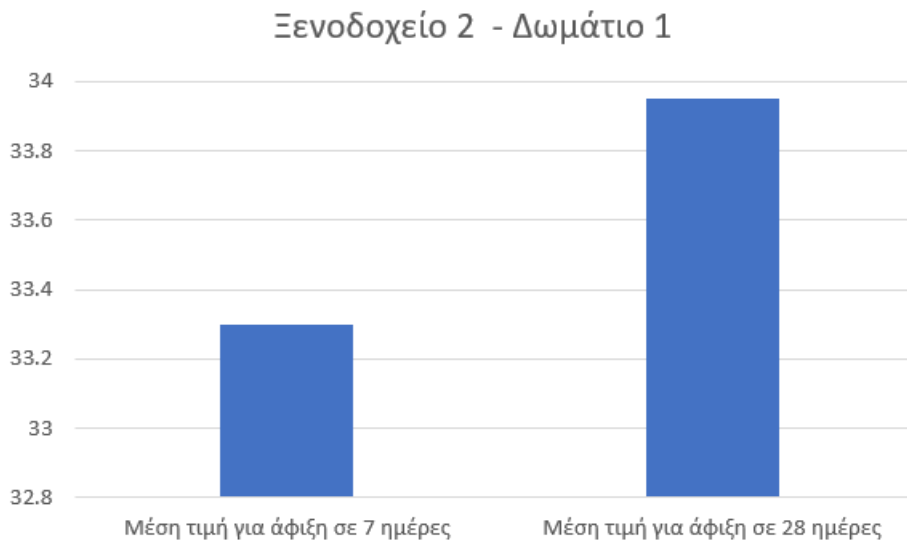
Εικόνα 19: Συγκριτικά στοιχεία μέσης τιμής - Ξενοδοχείο 1 - Δωμάτιο 1

Το αντίστοιχο γράφημα για το δεύτερο δωμάτιο του πρώτου ξενοδοχείου:



Εικόνα 20: Συγκριτικά στοιχεία μέσης τιμής - Ξενοδοχείο 1 - Δωμάτιο 2

Τέλος, το αντίστοιχο διάγραμμα για το πρώτο δωμάτιο του δεύτερου ξενοδοχείου:



Εικόνα 21: Συγκριτικά στοιχεία μέσης τιμής - Ξενοδοχείο 2 - Δωμάτιο 1

ΚΕΦΑΛΑΙΟ 4: ΣΥΜΠΕΡΑΣΜΑΤΑ

Στα πλαίσια της συγκεκριμένης διπλωματικής εργασίας, αρχικά πραγματοποιήθηκε η κατάλληλη βιβλιογραφική ανασκόπηση όσο αφορά τις πλατφόρμες σύγκρισης τιμών μεταξύ ξενοδοχείων και πραγματοποίησης κρατήσεων. Το κύριο όμως τμήμα της διπλωματικής εργασίας αφορούσε την ανάπτυξη διαδικτυακής εφαρμογής στηριζόμενης στις τεχνολογίες Node.js και Docker για την παρακολούθηση των τιμών σε ξενοδοχεία.

Η λειτουργικότητα και σημασία της εφαρμογής έγινε εμφανής στο τελευταίο κεφάλαιο, μέσω των αποτελεσμάτων τα οποία παρατηρήθηκαν. Ένα πρώτο συμπέρασμα το οποίο προκύπτει είναι ότι η πειραματική προσπάθεια στέφθηκε από επιτυχία, καθώς το υλοποιητικό μέρος της διπλωματικής εργασίας πέτυχε τον σκοπό του. Η εφαρμογή συγκεντρώνει δεδομένα σχετικά με δωμάτια ξενοδοχείων και παράγει έξοδο, η οποία είναι εύκολα επεξεργάσιμη και ερμηνεύσιμη.

Επομένως, η συγκεκριμένη εφαρμογή θα μπορούσε να αποτελέσει βασικό συστατικό για μια εφαρμογή μεγαλύτερης κλίμακας, που θα μπορούσε να συγκεντρώνει δεδομένα μεγαλύτερου όγκου και να τα επεξεργάζεται και παρουσιάζει με φιλικό τρόπο στον χρήστη. Σημαντικό ρόλο προς την συγκεκριμένη κατεύθυνση διαδραμάτισαν οι δύο βασικοί πυλώνες της εφαρμογής: Node.js και Docker, που ενίσχυσαν σημαντικά την απόδοσή της.

Στο σημείο αυτό αξίζει να αναφερθεί ότι τα δεδομένα καταγράφονται σε φιλική ως προς την χρήση μορφή. Ωστόσο η μετέπειτα επεξεργασία τους για την δημιουργία των γραφημάτων πραγματοποιήθηκε με μη αυτόματο τρόπο. Μια μελλοντική επέκταση της εργασίας θα μπορούσε να είναι η συγγραφή κώδικα με σκοπό την αυτόματη επεξεργασία και οπτικοποίηση των δεδομένων.

Τέλος, τα δεδομένα τα οποία συλλέγονται από την εφαρμογή θα μπορούσαν να αποτελέσουν είσοδο για αλγόριθμους μηχανικής μάθησης με στόχο την ανάπτυξη προβλεπτικών μοντέλων όσο αφορά την πορεία των τιμών. Πρόκειται για μια επέκταση η οποία θα μπορούσε να καταδείξει ακόμη περισσότερο την σημασία της εφαρμογής που υλοποιήθηκε.

4.1. Συνεργατική Ανάπτυξη

Στο σημείο αυτό αξίζει να αναφερθεί ότι το πρακτικό μέρος της συγκεκριμένης διπλωματικής εργασίας αποτελούσε τμήμα μιας μεγαλύτερης εφαρμογής, η οποία είχε διαιρεθεί σε τρία τμήματα:

- Scraper
- Backend (Καλαϊτσίδης Αρσλάν)
- DevOps/Scraper (Αβραμίδης Ερωτόκριτος)

Ως το μέλος της ομάδας που είχε αναλάβει το κομμάτι του scraping θα ήθελα αρχικά να ευχαριστήσω από καρδιάς τόσο τα υπόλοιπα μέλη της ομάδας όσο και τον επιβλέποντα καθηγητή για την άποψη μεταξύ μας συνεργασία.

Όσο αφορά την συχνότητα πραγματοποίησης συναντήσεων της ομάδας με τον επιβλέποντα, ήταν 2 φορές το μήνα (ανά δύο εβδομάδες), αρκετά ικανό διάστημα προκειμένου να διαπιστώνεται η πρόοδος της εργασίας και να γίνονται οι αντίστοιχες επισημάνσεις. Πέρα όμως από την πραγματοποίηση συναντήσεων, υπήρχε αρκετά συχνή επικοινωνία μεταξύ μας. Αρχικά, η επικοινωνία πραγματοποιείτο μέσω emails, μια διαδικασία η οποία ήταν αρκετά χρονοβόρα, και επομένως αποφασίστηκε αλλαγή του τρόπου επικοινωνίας και αντικατάστασή του με χρήση της πλατφόρμας discord, μέσω της οποίας πραγματοποιούνταν συνεχείς κλήσεις (ειδικά μεταξύ εμού και του μέλους που είχε αναλάβει το κομμάτι devops).

Στα πρώτα meetings τέθηκαν τα θεμέλια και οι προδιαγραφές του project:

- Δημιουργία ενός scraper που δεν θα επιστρέφει εσφαλμένα δεδομένα
- Δημιουργία μιας αξιόπιστης εφαρμογής
- Συμπερίληψη μεγάλης λίστας ξενοδοχείων
- Πραγματοποίηση ανάλυσης δεδομένων
- Δημιουργία μιας δυναμικά εμπορικής εφαρμογής

Σε επόμενα meetings τέθηκαν τα παρακάτω ζητήματα:

- Ανάπτυξη βάσης δεδομένων (MySQL)
- Τροποποίηση πινάκων
- Χρήση διαφορετικών βιβλιοθηκών (όπως η περίπτωση της βιβλιοθήκης Cheerio που αναφέρθηκε προηγουμένως)
- Τεχνικά ζητήματα σχετικά με την φιλοξενία της εφαρμογής (χρήση μηχανήματος της σχολής μετά από την άδεια και βοήθεια του καθηγητή)
- Εξασφάλιση προσβασιμότητας

Σε επόμενα meetings πραγματοποιήθηκαν:

- Δοκιμές
- Προτάσεις για βελτιστοποίηση

Στα τελευταία meeting, αφού η εφαρμογή είχε φτάσει σε ικανοποιητικό επίπεδο, αποφασίστηκε να "τρέξει" για 2 μήνες για την συλλογή δεδομένων και να γίνει έπειτα ανάλυση των δεδομένων.

Συμπερασματικά, ήταν μια γόνιμη και άκρως εκπαιδευτική διαδικασία, στην οποία υπήρχε συνεχής συνεργασία και κατανόηση, με στόχο την επίτευξη του βέλτιστου δυνατού αποτελέσματος.

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1]. Hu, X. S., & Yang, Y. (2020). Determinants of consumers' choices in hotel online searches: A comparison of consideration and booking stages. *International Journal of Hospitality Management*, 86, 102370.
- [2]. <https://www.booking.com/index.el.html>
- [3]. <https://www.expedia.com/>
- [4]. <https://www.hotels.com/>
- [5]. <https://www.agoda.com/el-gr/>
- [6]. <https://www.trivago.gr/>
- [7]. <https://www.gr.kayak.com/>
- [8]. <https://www.tripadvisor.com/>
- [9]. <https://nodejs.org/>
- [10]. <https://www.docker.com/>
- [11]. Turban, E., Sharda, R., Aronson, J. E., & King, D. (2008). *Business intelligence: A managerial approach* (pp. 58-59). Upper Saddle River, NJ: Pearson Prentice Hall.
- [12]. Grossmann, W., & Rinderle-Ma, S. (2015). *Fundamentals of business intelligence*.
- [13]. Kristoffersen, E., Mikalef, P., Blomsma, F., & Li, J. (2021). The effects of business analytics capability on circular economy implementation, resource orchestration capability, and firm performance. *International Journal of Production Economics*, 239, 108205.
- [14]. Xiang, Z., Schwartz, Z., Gerdes Jr, J. H., & Uysal, M. (2015). What can big data and text analytics tell us about hotel guest experience and satisfaction?. *International journal of hospitality management*, 44, 120-130.
- [15]. Moro, S., Rita, P., & Oliveira, C. (2018). Factors influencing hotels' online prices. *Journal of Hospitality Marketing & Management*, 27(4), 443-464.
- [16]. Zhao, B. (2017). Web scraping. *Encyclopedia of big data*, 1.
- [17]. Upadhyay, S., Pant, V., Bhasin, S., & Pattanshetti, M. K. (2017, February). Articulating the construction of a web scraper for massive data extraction. In *2017 Second International Conference on Electrical, Computer and Communication Technologies (ICECCT)* (pp. 1-4). IEEE.
- [18]. Gheorghe, M., Mihai, F. C., & Dârdală, M. (2018). Modern techniques of web scraping for data scientists. *International Journal of User-System Interaction*, 11(1), 63-75.
- [19]. Tsai, C. H. (2023). The study of e-business models in hospitality management: Comparing hotel reservation systems of Booking. com, Expedia and Agoda.
- [20]. Canelon, M., Harter, M., Holowaychuk, T. J., & Rajlich, N. (2014). *Node.js in Action* (pp. 17-20). Greenwich: Manning.
- [21]. Wan, X., Guan, X., Wang, T., Bai, G., & Choi, B. Y. (2018). Application deployment using Microservice and Docker containers: Framework and optimization. *Journal of Network and Computer Applications*, 119, 97-109.