**Structure**

Οι περισσότεροι φάκελοι στήθηκαν σύμφωνα με τα suggestion που δόθηκαν στις διαλέξεις με πολύ μικρές αποκλείσεις. Σημαντική διαφοροποίηση όμως έγινε στον φάκελο Public (στον οποίο πρέπει να δείχνει και το VM του server). Ακολουθεί μια μικρή εξήγηση για κάθε φάκελο και τα αρχεία που περιέχουν για να ύπαρξη καλύτερη κατανόηση από τους εξεταστές.

**public/components**

Κομμάτια Html κώδικα που γίνονται Include στις βασικές σελίδες τις εφαρμογής είτε για να βοηθήσουν στο να μην χρειάζεται να διατηρείτε ο ίδιος κώδικας σε 2 σημεία είτε για χρήση στην ενημέρωση της σελίδας μέσο ασύγχρονων μεθόδων. Ποιο συγκεκριμένα.

Το αρχείο components/**navbar.php** περιέχει το navigation bar για τις σελίδες room.php και list\_page.php που είναι ακριβός ο ίδιος κώδικας.

Το αρχείο components**/earch\_results.php** περιέχει τον κώδικα για την αναζήτηση δωματίου και την εμφάνιση όλης της λίστας αποτελεσμάτων δωματίου ( η αναζήτηση γίνετε στο component για αυτό και δεν υπάρχει σχετικό action παρακάτω για search). Αυτό το component γίνετε append στην html του αρχείου list\_page.php μέσο js ( ανατρέξτε στο αρχείο assets/js/list\_page.js .

To αρχείο components/**review.php** περιέχει τον κώδικα εμφάνισης ενός μοναδικού review. To components καλείτε από την room.php μέσο include για να εμφανίσει όλα τα reviews αλλά και μέσο του action που διαχειρίζεται το ajax request για review (action/ajax/review.php) ώστε να το γεμίσει με δεδομένα και να το επιστρέψει στη JS για να γίνει append.

**public/actions**

Πραγματοποιούν τις λειτουργίες της εφαρμογής, όπως και στις διαλέξεις, γίνεται έλεγχος (αν υπάρχει χρήστης, room id, etc) καλούν το ανάλογο Hotel service και κάνουν redirect στη σελίδα.

**public/actions/ajax**

Σε αυτά τα αρχεία στέλνουμε τα ajax requests, αφού εκτελέσουν την απαιτούμενη επικοινωνία με την βάση αντί να κάνουν redirect επιστρέφουν json ή html. (Επειδή έχουν προγραμματιστεί λειτουργίες για *favorite* και *review* και ασύγχρονα και σύγχρονα θα βρούμε αρχεία με το ίδιο όνομα και εδώ αλλά και ένα επίπεδο ποιο πανω).Ποιο συγκεκριμένα.

Το αρχείο favorite.php καλείτε όταν χρήστης πατάει το εικονίδιο favorite, δέχεται ένα roomId, ελέγχει αν είναι favorite για τον συνδεδεμένο χρήστη, αλλάζει την κατάσταση στην αντίθετη βάση και επιστρέφει το αποτέλεσμα.

Το αρχείο checkRoomAvailability.php καλείτε μέσο ajax όταν ο χρήστης αλλάζει τις τιμές check in και check out date στη σελίδα room. Δέχεται ένα roomId και 2 ημερομηνίες και επιστρέφει αν το δωμάτιο είναι available αυτές τις ημερομηνίες.

Το αρχείο review.php καλείτε όταν ο χρήστης κάνει Post ένα review. Δέχεται τα δεδομένα του review τα οποία τα περνάει στη βάση στη συνέχεια καλεί το component review.php to οποίο κάνει populate με τα δεδομένα που περάσει ο χρήστης και το επιστρέφει στην JS για να το κάνει append στη σελίδα.

**public/assets/css**

Περιέχει τα αρχεία css κάθε σελίδας το όνομα κάθε αρχείου είναι είναι ίδιο με την σελίδα που αντιστοιχεί (πχ Index.css -> index.php).

**public/assets/images**

Oi φωτογραφίες τις εφαρμογής (obviously).

**public/assets/js**

Τα αρχεία javascript των σελίδων της εφαρμογής όπως και στα αρχεία css υπάρχει αντιστοιχία ονομάτων. Ειδική εξαίρεση είναι η σελίδα room.php που λόγο των πολλών διαφορετικών ajax call που απαιτούνταν επέλεξα να κρατήσω τα αρχεία που πραγματοποιούν τις λειτουργίες favorite, review και ελέγχου διαθεσιμότητας δωμάτιου, (μαζί με το αρχείο test\_async\_await\_review.js που είναι μια αρχική προσπάθεια υλοποίησης του review ασύγχρονα χωρίς ajax), ξεχωριστά από την υπόλοιπη js του room. Όλα τα αρχεία Js που υλοποιούν λειτουργίες του room.php στο φάκελο public/assets/js/room.

**Κοινές λειτουργίες**

Κάποιες κοινές λειτουργίες που μοιράζονται σελίδες της εφαρμογής εξηγούνται παρακάτω για να αποφύγουμε επανάληψη περιγραφής της λειτουργίας σε κάθε σελίδα.

* Το Navigation bar σε όλες της σελίδες (πλην του Register και Login), δίνει την δυνατότητα στο χρήστη να μετακινηθεί μεταξύ των σελίδων *register.php*, *login.php* και *index.php* αν ο χρήστης ΔΕΝ είναι συνδεμένος. Αν ο χρήστης συνδεθεί, υπάρχει έλεγχος που αλλάζεις τις επιλογές ανακατεύθυνσης σε *index.php, profile.php* καθώς και την επιλογή να τρέξει την λειτουργεία Logout.   
  Οι σελίδες register και login έχουν την δυνατότητα ανακατεύθυνσης μεταξύ τους και του index.php.  
  H σελίδα profile.php δεν έχει την δυνατότητα ανακατεύθυνσης στην ίδια.
* Στις σελίδες όπου δίνεται η δυνατότητα επιλογής ημερομηνίας (index.php, list\_page.php και room.php) έχουν εφαρμοστεί οι παρακάτω περιορισμοί σε επίπεδο JS.  
  O χρήστης δε μπορεί να διαλέξει Check In ή/και Check Out ημερομηνία πιο παλιά από τη σημερινή και αυριανή αντίστοιχα.  
  Αν υπάρχει επιλεγμένη Check In ημερομηνία ο χρήστης δε μπορεί να επιλέξει Check Out παλιότερη ημερομηνία από την επιλεγμένη Check In.  
  Αν υπάρχει επιλεγμένη Check Out ημερομηνία ο χρήστης δε μπορεί να επιλέξει Check In μεταγενέστερη ημερομηνία από την επιλεγμένη Check Out.  
  Για να γίνει reset ένας περιορισμός (πλην αυτού της σημερινής ως ελάχιστης) ο χρήστης πρέπει να διαγράψει την ημερομηνία από το input field.
* Στις σελίδες list\_page, room.php και profile.php υπάρχει λειτουργία “go to top” αν ο χρήστης κάνει scroll down αρκετά.

**public/index.php**

Το σημείο εισόδου της εφαρμογής. Το VM θα πρέπει να βλέπει τον φάκελο public για να δουλεύουν σωστά τα redirect.

H σελίδα δουλεύει όπως ζητάνε οι απαιτήσεις.

**public/list\_page.php**

Όπως ζητείται στις απαιτήσεις, η σελίδα περιέχει τα αποτελέσματα της αναζήτησης καθώς και μια φόρμα για ποιο advanced αναζήτηση.  
  
Αρχικά γίνεται έλεγχος σε όλες τις δυνατές παραμέτρους αν έχουν δοθεί από τον χρήστη, σε όσες δεν έχουν δοθεί γίνεται assign empty string (εκτός από τις ημερομηνίες check In και Checkout) αυτό γίνεται για να αποφύγουμε warnings ότι κάποια τιμή είναι Null. Κανονικά ο χρήστης δε μπορεί να έρθει σε αυτή σελίδα (αν δεν να αλλάξει το Url χειροκίνητα) χωρίς να έχει δώσει τουλάχιστον τις παραμέτρους city ,check\_in\_date και check\_out\_date   
  
Στην φόρμα με τα επιμέρους φίλτρα στο Input “price range” αρχικοποιουμε τις Min και Μax τιμές με τιμές του ποιο φτηνού και ακριβού ξενοδοχείου που υπάρχουν στη database μας αντίστοιχα.  
  
Κάνουμε include το component navbar.php που περιέχει τον κώδικα για το navigation bar.  
  
Στο section με id=”search\_results” είναι το σημείο εμφανίζονται τα αποτελέσματα τις αναζήτησης. Το section γίνεται populate μέσο ajax call όταν κάνουμε submit την φόρμα αναζήτησης δωμάτιου searchForm αλλά ΚΑΙ όταν φορτώνεται την σελίδα με τον ίδιο τρόπο έτσι δεν χρειάζεται να συντηρούμε τον ίδιο κώδικα σε δύο σημεία. Ο κώδικάς που εμφανίζει τα αποτελέσματα υλοποιείται στο component search\_results.php αλλά τον έχω αφήσει και commented στη σελίδα για την βοήθεια του εξεταστή.   
  
  
**public/room.php**

H σελίδα του κάθε δωμάτιου. Εμφανίζει διαφορετική συμπεριφορά αν ο χρήστης είναι συνδεμένος η όχι. Ποιο συγκεκριμένα αν ο χρήστης δεν είναι συνδεμένος:

Δεν εμφανίζεται η επιλογή για favorite.  
Δε μπορεί να κάνει book το δωμάτιο, μπορεί να αλλάξει ημερομηνίες για να δει πότε είναι ελεύθερο αλλά το κουμπί για book είναι disabled.  
Δε μπορεί να αφήσει review το κουμπί και φόρμα είναι disabled.  
  
Οι λειτουργίες favorite και comment γίνονται ασύγχρονα (μπλοκάρετε το redirect από js), επίσης ασύγχρονα ενημερώνεται και σελίδα με κάθε αλλαγή ημερομηνίας για το αν το δωμάτιο είναι available. H λειτουργία book γίνεται κανονικά με redirect στο αρχείο **actions/book.php**.

Για την υλοποίηση της λειτουργείας favorite ανατρέξτε στα αρχεία **actions/favorite.php** για την σύγχρονη υλοποίηση της και **assets/js/room/favorite.js** και **actions/ajax/favorite.php** για την ασύγχρονη. \*  
\*θα ήθελα να αναφέρω ότι υλοποίησα την λειτουργεία favorite πρώτου συζητηθεί στη διάλεξη οπότε χρησιμοποιήθηκε διαφορετική μεθοδολογία (await fetch για ενημέρωση της βάσης και DOM manipulation )

Για την υλοποίηση του ελέγχου διαθεσιμότητας δωμάτιου ανατρέξτε στα αρχεία **assets/js/room/** **booking\_date\_change.js** και **actions/ajax/checkRoomAvailability.php.**

Για την υλοποίηση του ελέγχου διαθεσιμότητας δωμάτιου ανατρέξτε στα αρχεία **actions/book.php.**

Για λειτουργεία review ανατρέξτε στα αρχεία **assets/js/room/review.js και actions/ajax/checkRoomAvailability.php.**

**ΣΗΜΕΙΩΣΗ**. Επειδή κάνουμε Include το κομμάτι κώδικα που περιέχει την html του review άλλο αρχείο (το αρχείο **components/review.php** ) θα **πρέπει**  στο foreach να χρησιμοποιηθούν τα ονόματα μεταβλητών $**counter** και **$roomReview** αλλιώς δε θα γνωρίζει το component τι μεταβλητές να χρησιμοποιήσει στα πεδία. Αυτή η μεθοδολογία επιλεχτείτε για να μπορούμε να κάνουμε append ένα review στην αρχή της λίστας χωρίς να χρειάζεται να διατηρούμε τον ίδιο κώδικα σε δυο σημεία.

Τέλος η λειτουργία βαθμολόγησης μέσο αστερίων έγινε με την χρήση css και js την υλοποίηση μπορείτε να την βρείτε αντίστοιχα στα αρχεία assets/css/room.css και assets/js/room/room.js.

**public/profile.php**

Εκτός από την δική του Navbar (για να μην περιέχει redirect στην ιδιά σελίδα) και gototo arrow η σελίδα είναι ίδια με τις απαιτήσεις.

**public/login.php** και **public/register.php**

Αυτές οι σελίδες δεν είναι προσβάσιμες αν ο χρήστης είναι logged in και πρέπει να γίνεται redirect στην Index.php αν κάποιος προσπαθήσει να αποκτήσει πρόσβαση μέσο URL.

**\*\* ΣΧΟΛΙΑ ΠΑΝΩ ΣΤΗ ΥΛΟΠΟΙΗΣΗ ΤΗΣ ΚΛΑΣΗΣ BaseService.\*\***

Κατά την διάλεξη, όταν υλοποιήσαμε τα functions fetch, fetchΑll και execute χρησιμοποιήθηκε η μέθοδος του pdo “bindParam()” μέσα σε ένα foreach loop για να κάνουμε bind έναν άγνωστο αριθμό παραμέτρων. Αυτό έχει ως αποτέλεσμα να γίνεται bind μόνο η τελευταία παράμετρος κάτι που δυστυχώς δεν ανακαλύφθηκε κατά την διάρκεια εκεινού το μαθήματος.

Για να πέτυχω το επιθυμητό αποτέλεσμα χρησιμοποίησα την μέθοδο bindValue().