**Περίληψη Escaping**

* ***Εισαγωγή***

Στα πλαίσια της πτυχιακής εργασίας θα αναπτύξουμε ένα ολοκληρωμένο συστημα κρατήσεων (**Booking System**) που θα απευθύνεται στην κάλυψη των αναγκών για ψυχαγωγικές εταιρίες τύπου **Escape Room**. Πιο συγκεκριμένα προτίθεμαι να υλοποιήσουμε μια πλατφόμα στην οποία θα μπορούν δυναμικά να στεγάζονται υπηρεσίες τέτοιου τύπου ώστε να προσφέρουν τα Rooms τους στη δίαθεση του κοινου για κράτηση(**Reservation**). Αυτή η πλάτφόρμα πέραν από την δυνατότητα προφανώς να παρουσιαζεί όλα τα Room που συμμετέχουν σε αυτή, θα ενημερώνεi το **Booking Calendar** του εκάστοτε Room κάθε φορά που υπάρχει κάποια αλλαγή σε αυτό, είτε αυτή η αλλάγή-κράτηση-ακύρωση γίνεται από την πλατφόρμα είτε από την προσωπική σελίδα κάθε εταιρίας που συμμετέχει στην πλατφόρμα. Δηλαδή οι Users θα μπορούν να κάνουν κράτηση για το δωμάτιο της επιλογής τους από 2 sites. Επίσης θα υπάρχει η δυνατότητα αλλαγής των κρατήσεων(ενδεχομένως και του περιεχομένου (images,description,κτλπ )) του κάθε Room από τις ίδιες τις εταιρίες μέσω ενός **Admin Page** το οποίο θα υλοποιηθέι για να συμπληρώσει αυτήν την υπηρεσία μέσω ενός διαχειριστηκού Page για την πλευρά της κάθε εταιρίας. Με αυτόν τον τρόπο θα γίνεται πιο εύχρηστη η διαχείρηση των Rooms της κάθε εταιρίας τόσο στο κομμάτι των κρατήσεων αλλά και σε επιμέρους τομείς.

Στόχος μας είναι η υλοποίηση ενός σύγχρονου **WebApp** με νέες τεχνολογίες τόσο στο front όσο και στο back(**API**) κομμάτι της εργασίας που θα ανταποκρίνεται στις σύγχρονες ανάγκες για την κάλυψη των Booking System υπηρεσιών.

* Παρακάτω περιγράφονται οι 2 διακριτές οντότητες της πτυχιακής εργασίας:

α) ***Booking Page:***

Το πρώτο κομμάτι θα αποτελείται από την υλοποίηση ενός web application, στo οποίo θα εισάγονται εταιρίες απόδρασης(Escape Rooms) με τα δωμάτια τους και η υπηρεσία που θα προσφέρει η εφαρμογή θα είναι το σύστημα κρατήσεων. Η εφαρμογή θα υλοποιηθεί σε ReactJS, ένα σύγχρονο framework της Javascript με πολλά πλεονεκτήματα στην εμπειρία χρήστη(UX) .

Επίσης θα υλοποιήσουμε και ένα Booking calendar template το οποίο θα μπορεί η εταιρία να χρησιμοποιήσει στην προσωπική της σελίδα για τον συγχρονισμό της με το δικό μας σύστημα. Το Template θα υλοποιηθεί σε JS/HTML για να μπορεί να ενσωματωθεί σε οποιαδήποτε ιστοσελίδα χωρίς πρόβλημα συμβατότητας. Επίσης θα παρέχεται και React component για να ενσωματώνεται σε τέτοιου είδους σελίδες το Booking calendar.

β) ***Admin Page:***

Στο δεύτερο κομμάτι της εργασίας θα υλοποιηθεί μία σελίδα διαχείρισης με την οποία οι εταιρίες θα διαχειρίζονται τα δωμάτια τους και τις κρατήσεις τους. Και αυτό το κομμάτι θα υλοποιηθεί σε ReactJS.

* ***Back-end:***

Στο τρίτο αλλά αφανές κομμάτι της εργασίας θα υλοποιηθεί το RESTful **API** του project που θα χρησιμοποιούν οι 2 παραπάνω εφαρμογές ιστού ώστε να ανταλλάσσουν δεδομένα. Τα δεδομένα θα είναι σε μορφή JSON καθώς χρησιμοποιεί μικρή σήμανση και είναι εύκολα προσπελάσιμη. Σε αυτό το τμήμα της εργασίας θα χρησιμοποιηθεί nodeJS, καθώς αποτελεί ένα εργαλείο υψηλής απόδοσης και προσφέρει πολλές δυνατότητες για την υλοποίηση εφαρμογών και υπηρεσιών ιστού.

Ένα ερώτημα που προκύπτει είναι το πώς θα γίνεται η συνεργασία του συστήματος κρατήσεων της εφαρμογή μας με την προσωπική σελίδα πρωτίστως της κάθε εταιρίας ή/και τυχόν διαφορετικά συστήματα κρατήσεων που μπορεί να χρησιμοποιούν. Επομένως το βασικό “πρόβλημα” που θα πρέπει να λύσουμε είναι η επικοινωνία-συγχρονισμός των booking systems. Μετά από έρευνα καταλήξαμε στο γεγονός ότι είναι βέλτιστο να χρησιμοποιήσουμε WebSockets για τον συγχρονισμό όλων τον εφαρμογών που χρησιμοποιούν την βάση δεδομένων μας σε πραγματικό χρόνο. Τα WebSockets παρέχουν μια συνεχή σύνδεση μεταξύ των εφαρμογών μας και του API server που μπορούν να χρησιμοποιήσουν και τα δύο μέρη για να ξεκινήσουν την αποστολή δεδομένων ανά πάσα στιγμή. Οι εφαρμογή δημιουργεί μια σύνδεση με τον server και από εκεί θα λαμβάνει μηνύματα για τυχόν αλλαγές που έγιναν από άλλο σημείο του συστήματος μας.

* Αναλυτική περιγραφή συγχρονισμού των booking calendars

Μετά από έρευνα καταλήξαμε στη χρήση **Socket.IO**, ένα JS library το οποίο επιτρέπει αμφίδρομη επικοινωνία σε πραγματικό χρόνο. Για την χρήση αυτού πρέπει να εγκαταστήσουμε 2 packages:

* **socket.io**, το οποίο είναι το websocket server που θα ενημερώνει τους clients για τα νέα δεδομένα.
* **socket.io-client**, με το οποίο οι clients θα “ακούνε”(listening) για data update events.

Πιο συγκεκριμένα, κάθε φορά που θα γίνεται κάποιο επιτυχημένο **booking** (είτε από την εφαρμογή κρατήσεων είτε από την προσωπική σελίδα της εταιρίας είτε από την εφαρμογή διαχειριστή ) θα ενεργοποιείτε ο Socket.IO Server το οποίο θα εκπέμπει(broadcast) προς όλους τους Socket.IO Clients ένα γεγονός αλλαγής δεδομένων. Όταν ένας Socket.IO Client λάβει σήμα ότι έγιναν αλλαγές θα ενεργοποιεί τις ανάλογες συναρτήσεις να ενημερώσουν την εφαρμογή με τα νέα δεδομένα.

* Παρακάτω φαίνεται σχηματικά πώς θα γίνεται η αμφίδρομή real-time επικοινωνία των εφαρμογών μας με το API και θα επιτυγχάνεται η ενημέρωση αυτών.

