



**ΠΑΝΕΠΙΣΤΗΜΙΟ
ΠΕΛΟΠΟΝΝΗΣΟΥ**
University of the Peloponnese

Π.Μ.Σ. στην Επιστήμη Υπολογιστών

Μεταπτυχιακή Διπλωματική Εργασία

**Διαδικτυακή εφαρμογή κεντρικής διαχείρισης
εκδηλώσεων**

"EventHub"

Παρασκευόπουλος Μάριος

ΑΜ: 2022202402015

Επιβλέπων:

Κώστας Βασιλάκης

<Τρίπολη, Μάιος 2026>

ΠΕΡΙΛΗΨΗ

Στο πλαίσιο της παρούσας διπλωματικής εργασίας σχεδιάστηκε και αναπτύχθηκε η πλατφόρμα **EventHub**. Η πλατφόρμα **EventHub** αποτελεί μια σύγχρονη και ολοκληρωμένη διαδικτυακή πλατφόρμα διαχείρισης και ανακάλυψης εκδηλώσεων, η οποία δημιουργήθηκε με στόχο να διευκολύνει τόσο τους διοργανωτές όσο και τους συμμετέχοντες. Η εφαρμογή λειτουργεί ως ένας ψηφιακός χώρος αλληλεπίδρασης και προβολής πολιτιστικών, κοινωνικών και ψυχαγωγικών δράσεων, προσφέροντας ένα εύχρηστο και δυναμικό περιβάλλον για την ενημέρωση και τη συμμετοχή των χρηστών.

Μέσω της πλατφόρμας, οι χρήστες έχουν τη δυνατότητα να αναζητούν εκδηλώσεις με λειτουργικό και αποδοτικό τρόπο. Η αναζήτηση δεν περιορίζεται μόνο στο όνομα ή την κατηγορία μιας εκδήλωσης, αλλά αξιοποιεί και δεδομένα τοποθεσίας, επιτρέποντας την εμφάνιση εκδηλώσεων που πραγματοποιούνται κοντά στην περιοχή ενδιαφέροντος του χρήστη. Με αυτόν τον τρόπο, το EventHub ενισχύει την προσωποποιημένη εμπειρία πλοήγησης και διευκολύνει την ανακάλυψη τοπικών δραστηριοτήτων.

Κάθε εκδήλωση διαθέτει αναλυτική σελίδα παρουσίασης, όπου παρέχονται πλήρεις πληροφορίες σχετικά με την περιγραφή, την ημερομηνία και τον χώρο διεξαγωγής της. Παράλληλα, ενσωματώνεται διαδραστικός χάρτης για εύκολη πλοήγηση προς την τοποθεσία της εκδήλωσης, πληροφορίες σχετικά με το κόστος συμμετοχής ή των εισιτηρίων, καθώς και άμεσοι σύνδεσμοι για κρατήσεις ή αγορά εισιτηρίων.

Η πλατφόρμα δίνει ιδιαίτερη έμφαση στη διαδραστικότητα και στη συμμετοχή της κοινότητας. Οι χρήστες μπορούν να δηλώνουν τη συμμετοχή τους σε εκδηλώσεις, να κοινοποιούν φωτογραφικό υλικό από την εμπειρία τους και να αξιολογούν τις εκδηλώσεις μέσω σχολίων και βαθμολογιών. Η δυνατότητα αυτή συμβάλλει στη δημιουργία ενός αξιόπιστου συστήματος ανατροφοδότησης, το οποίο βοηθά τόσο τους διοργανωτές όσο και τους μελλοντικούς συμμετέχοντες.

Επιπλέον, κάθε διοργανωτής διαθέτει το δικό του δημόσιο προφίλ, στο οποίο παρουσιάζονται στοιχεία επικοινωνίας, το ιστορικό των εκδηλώσεων που έχει δημιουργήσει, καθώς και η συνολική του αξιολόγηση, βάσει της εμπειρίας των

χρηστών. Η λειτουργία αυτή ενισχύει τη διαφάνεια και την αξιοπιστία της πλατφόρμας, επιτρέποντας στους χρήστες να επιλέγουν εκδηλώσεις με μεγαλύτερο επίπεδο εμπιστοσύνης.

Βασικός στόχος του EventHub είναι η ενίσχυση της τοπικής πολιτιστικής και κοινωνικής δραστηριότητας μέσω της αξιοποίησης σύγχρονων τεχνολογιών πληροφορικής. Η πλατφόρμα φιλοδοξεί να καταστήσει τη διαδικασία αναζήτησης, οργάνωσης και συμμετοχής σε εκδηλώσεις πιο άμεση, εύκολη και αποτελεσματική, προσφέροντας μια ολοκληρωμένη ψηφιακή εμπειρία στους χρήστες της.

Λέξεις Κλειδιά: EventHub, Εφαρμογή Ιστού, Έξυπνη Αναζήτηση, Διαδραστικότητα, Προφίλ Διοργανωτών

ABSTRACT

In the context of this thesis, the EvenHub platform was designed and developed. The **EventHub** system is a modern and comprehensive web-based platform for event management and discovery, aiming to facilitate interaction between event organizers and participants. The application serves as a digital hub for promoting cultural, social, and entertainment activities, offering users an intuitive and dynamic environment for discovering and engaging with events.

Using the platform, users are able to search for events in an facile and efficient manner. The search functionality is not limited to the event name or category, but also incorporates location-based data, allowing the system to display events that take place near the user's area of interest. In this way, EventHub enhances the personalized browsing experience and simplifies the discovery of local activities.

Each event is presented through a dedicated page that provides detailed information regarding its description, date, and venue. In addition, the platform integrates an interactive map for easier navigation to the event location, information about ticket prices or participation costs, as well as direct links for reservations or ticket purchases.

The platform places significant emphasis on interactivity and community engagement. Users can confirm their attendance at events, share photos from their experience, and evaluate events through ratings and reviews. This functionality contributes to the creation of a reliable feedback system that benefits both organizers and future participants.

Furthermore, each organizer maintains a public profile containing contact information, a history of previously organized events, and an overall reputation score based on user evaluations. This feature enhances transparency and trustworthiness, enabling users to make more informed decisions when selecting events.

The primary objective of EventHub is to strengthen local cultural and social activity through the use of modern information technologies. The platform aims to simplify and improve the process of discovering, organizing, and participating in events, while providing users with a complete and engaging digital experience.

Keywords: EventHub, Web Application, Smart Search, Interactivity, Host Profiles

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

ΠΕΡΙΛΗΨΗ	<i>i</i>
ABSTRACT	<i>iii</i>
ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ	<i>v</i>
ΕΥΡΕΤΗΡΙΟ ΕΙΚΟΝΩΝ	<i>vii</i>
1 ΕΙΣΑΓΩΓΗ	1
1.1 Περιγραφή του Προβλήματος	1
1.2 Σκοπός και Αντικείμενο της Εργασίας	2
1.3 Βασικά Χαρακτηριστικά και Καινοτομίες	3
2 ΑΝΑΛΥΣΗ ΑΠΑΙΤΗΣΕΩΝ	5
2.1 Κατηγορίες Χρηστών	6
2.1.1 Επισκέπτης (Guest)	6
2.1.2 Μέλος / Συμμετέχων (Member / Attendee)	6
2.1.3 Διοργανωτής (Organizer)	7
2.2 Λειτουργικές Απαιτήσεις ανά Κατηγορία	8
2.2.1 Επισκέπτης (Guest)	8
2.2.2 Μέλος / Συμμετέχων (Member)	10
2.2.3 Διοργανωτής (Organizer)	13
3 ΣΧΕΔΙΑΣΗ ΣΥΣΤΗΜΑΤΟΣ	17
3.1 Αρχιτεκτονική Συστήματος	17
3.1.1 Frontend – Περιβάλλον Χρήστη (React.js)	18
3.1.2 Backend – Εξυπηρετητής Εφαρμογών (Spring Boot)	18
3.1.3 Βάση Δεδομένων – PostgreSQL (NeonDB)	19
3.2 Σχήμα Βάσης Δεδομένων	19
3.3 Σχεδιασμός χωρικής αναζήτησης	21
3.4 Διεπαφές Επικοινωνίας (API Endpoints)	22
3.4.1 Εκδηλώσεις (Events)	23
3.4.2 Χρήστες & Προφίλ (Users & Profiles)	28
3.4.3 Κριτικές & Υλικό (Reviews & Media)	30
4 ΕΡΓΑΛΕΙΑ ΚΑΙ ΤΕΧΝΟΛΟΓΙΕΣ ΑΝΑΠΤΥΞΗΣ	32
4.1 Backend (Διακομιστής)	32

4.2	Frontend (Περιβάλλον Χρήστη)	33
4.3	Βάση Δεδομένων και Cloud Υπηρεσίες	34
4.4	Εργαλεία Ανάπτυξης και Docker	34
5	ΟΔΗΓΙΕΣ ΕΓΚΑΤΑΣΤΑΣΗΣ	36
5.1	Προαπαιτούμενα	36
5.2	Ρύθμιση Backend	37
5.3	Ρύθμιση Frontend	38
6	ΟΔΗΓΙΕΣ ΧΡΗΣΗΣ	40
6.1	Λειτουργίες για το Μέλος (Attendee)	40
6.1.1	Αναζήτηση Εκδήλωσης	40
6.1.2	Συμμετοχή	41
6.1.3	Υποβολή Αξιολόγησης και Κριτικής	42
6.2	Λειτουργίες για τον Διοργανωτή (Organizer)	43
6.2.1	Δημιουργία Εκδήλωσης	43
6.2.2	Διαχείριση Δημόσιου Προφίλ	44
6.2.3	Επεξεργασία Εκδήλωσης	46
7	ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ	48
7.1	Συμπεράσματα	48
7.2	Μελλοντικές Επεκτάσεις	49
7.2.1	Σύστημα Ειδοποιήσεων (Notifications)	49
7.2.2	Ανάπτυξη Native Mobile Εφαρμογής	49
7.2.3	Ενσωμάτωση Ηλεκτρονικών Πληρωμών	49
7.2.4	Προηγμένα Στατιστικά και Analytics	49
7.2.5	Ολοκλήρωση με τρίτα συστήματα	50
7.2.6	Διαχείριση Κατηγοριών από Διαχειριστή	50
7.2.7	Διαχείριση Εισιτηρίων μέσω ξεχωριστού API	50
	ΑΝΑΦΟΡΕΣ	52

ΕΥΡΕΤΗΡΙΟ ΕΙΚΟΝΩΝ

<i>Εικόνα 1. Αρχιτεκτονική της Εφαρμογής</i>	<i>17</i>
<i>Εικόνα 2. Διάγραμμα του σχήματος της βάσης δεδομένων</i>	<i>21</i>
<i>Εικόνα 3. Αλγόριθμος Χωρικής Αναζήτησης</i>	<i>22</i>
<i>Εικόνα 4. Ενδεικτική απάντηση στην κλήση API για αναζήτηση εκδηλώσεων</i>	<i>24</i>
<i>Εικόνα 5. Παράδειγμα αιτήματος δημιουργίας εκδήλωσης</i>	<i>25</i>
<i>Εικόνα 6. Παράδειγμα αιτήματος ενημέρωσης εκδήλωσης</i>	<i>27</i>
<i>Εικόνα 7. Ενδεικτική απάντηση σε αίτημα δημιουργίας λογαριασμού</i>	<i>29</i>
<i>Εικόνα 8. Ενδεικτική απάντηση προφίλ ενός χρήστη</i>	<i>30</i>
<i>Εικόνα 9. Ενδεικτική απάντηση σε κλήση του API ανάκτησης των κατηγοριών</i>	<i>31</i>
<i>Εικόνα 10. Αναζήτηση εκδηλώσεων</i>	<i>41</i>
<i>Εικόνα 11. Δήλωση συμμετοχής</i>	<i>42</i>
<i>Εικόνα 12. Υποβολή αξιολόγησης</i>	<i>43</i>
<i>Εικόνα 13. Δημιουργία εκδήλωσης</i>	<i>44</i>
<i>Εικόνα 14. Διαχείριση δημόσιου προφίλ</i>	<i>45</i>
<i>Εικόνα 15. Επεξεργασία εκδήλωσης</i>	<i>47</i>

1 ΕΙΣΑΓΩΓΗ

1.1 Περιγραφή του Προβλήματος

Στη σύγχρονη ψηφιακή εποχή, η ανακάλυψη και η οργάνωση τοπικών εκδηλώσεων, πολιτιστικών δράσεων και κοινωνικών γεγονότων αποτελεί μια διαδικασία που συνοδεύεται από σημαντικές δυσκολίες, τόσο για τους διοργανωτές όσο και για το κοινό. Παρόλο που υπάρχει μεγάλος όγκος διαθέσιμης πληροφορίας στο διαδίκτυο, η ενημέρωση σχετικά με τις εκδηλώσεις μιας περιοχής παραμένει συχνά αποσπασματική και χωρίς επαρκή οργάνωση.

Οι ενδιαφερόμενοι χρήστες καλούνται να αναζητήσουν πληροφορίες μέσα από πολλαπλές και ετερογενείς πηγές, όπως μέσα κοινωνικής δικτύωσης, ιστοσελίδες, τοπικά ιστολόγια ή ακόμη και φυσικούς πίνακες ανακοινώσεων. Η κατακερματισμένη αυτή μορφή πληροφόρησης δυσκολεύει τον έγκαιρο εντοπισμό εκδηλώσεων που ανταποκρίνονται στα ενδιαφέροντα και την τοποθεσία του χρήστη. Επιπλέον, η έλλειψη προηγμένων μηχανισμών γεωγραφικής αναζήτησης περιορίζει τη δυνατότητα εύρεσης δράσεων που πραγματοποιούνται σε κοντινή απόσταση, γεγονός που επηρεάζει αρνητικά τη συμμετοχή του κοινού σε τοπικές δραστηριότητες.

Παράλληλα, οι διοργανωτές εκδηλώσεων - όπως πολιτιστικοί σύλλογοι, τοπικές επιχειρήσεις, καλλιτεχνικές ομάδες και ανεξάρτητοι δημιουργοί - συχνά δεν διαθέτουν ένα ενιαίο, εύχρηστο και οικονομικά προσιτό σύστημα για την προβολή και διαχείριση των εκδηλώσεών τους. Η προώθηση δράσεων μέσω πολλαπλών καναλιών απαιτεί χρόνο, κόστος και τεχνικές γνώσεις, ενώ σε πολλές περιπτώσεις δεν παρέχονται λειτουργίες όπως διαχείριση κρατήσεων, ηλεκτρονική έκδοση εισιτηρίων ή συγκέντρωση αξιολογήσεων από το κοινό.

Επιπλέον, η απουσία ενός οργανωμένου ψηφιακού προφίλ περιορίζει τη δυνατότητα των διοργανωτών να αναπτύξουν αξιοπιστία και σταθερή σχέση εμπιστοσύνης με το κοινό τους. Ως αποτέλεσμα, τόσο η προβολή των εκδηλώσεων όσο και η συνολική εμπειρία των χρηστών παραμένουν περιορισμένες και λιγότερο αποτελεσματικές.

Η ανάγκη για μια ολοκληρωμένη ψηφιακή πλατφόρμα που θα συγκεντρώνει, θα οργανώνει και θα προβάλλει αποτελεσματικά τις εκδηλώσεις σε τοπικό επίπεδο

καθίσταται πλέον ιδιαίτερα σημαντική. Ένα τέτοιο σύστημα μπορεί να συμβάλλει ουσιαστικά στη βελτίωση της επικοινωνίας μεταξύ διοργανωτών και κοινού, καθώς και στην ενίσχυση της πολιτιστικής και κοινωνικής δραστηριότητας μιας περιοχής.

1.2 Σκοπός και Αντικείμενο της Εργασίας

Αντικείμενο της παρούσας εργασίας αποτελεί η σχεδίαση και ανάπτυξη μιας ολοκληρωμένης διαδικτυακής πλατφόρμας διαχείρισης και ανακάλυψης εκδηλώσεων (Web Application), με την ονομασία **EventHub**. Η πλατφόρμα φιλοδοξεί να παρέχει ένα σύγχρονο ψηφιακό περιβάλλον που θα διευκολύνει την επικοινωνία και την αλληλεπίδραση μεταξύ διοργανωτών εκδηλώσεων και ενδιαφερόμενων χρηστών.

Κύριος στόχος του συστήματος είναι η αντιμετώπιση των προβλημάτων που σχετίζονται με την αποσπασματική ενημέρωση και την περιορισμένη προβολή τοπικών δράσεων, παρέχοντας ένα ενιαίο και εύχρηστο σύστημα αναζήτησης, διαχείρισης και συμμετοχής σε εκδηλώσεις. Μέσω της αξιοποίησης σύγχρονων τεχνολογιών διαδικτύου και γεωεντοπισμού, η πλατφόρμα EventHub επιδιώκει να ενισχύσει την πολιτιστική και κοινωνική δραστηριότητα σε τοπικό επίπεδο.

Πιο συγκεκριμένα, η εφαρμογή στοχεύει στην κάλυψη των αναγκών δύο βασικών κατηγοριών χρηστών: του ευρύτερου κοινού και των διοργανωτών εκδηλώσεων.

- *Για το ευρύ κοινό*, η πλατφόρμα παρέχει ένα φιλικό και εύχρηστο περιβάλλον αναζήτησης εκδηλώσεων, με δυνατότητα χρήσης προηγμένων φίλτρων, όπως κατηγορίες, ημερομηνίες και τύποι δραστηριοτήτων. Παράλληλα, ενσωματώνεται μηχανισμός «έξυπνης» γεωγραφικής αναζήτησης, ο οποίος επιτρέπει στους χρήστες να εντοπίζουν εκδηλώσεις που πραγματοποιούνται εντός συγκεκριμένης ακτίνας από την τοποθεσία τους ή από σημείο που επιλέγουν στον χάρτη. Επιπλέον, οι χρήστες μπορούν να δηλώνουν συμμετοχή σε εκδηλώσεις, να αξιολογούν την εμπειρία τους μέσω κριτικών και βαθμολογιών, καθώς και να κοινοποιούν φωτογραφικό υλικό από τη συμμετοχή τους.
- *Για τους διοργανωτές*, το EventHub προσφέρει ένα ολοκληρωμένο εργαλείο δημιουργίας και διαχείρισης εκδηλώσεων. Η εφαρμογή υποστηρίζει λειτουργίες γεωεντοπισμού μέσω διαδραστικού χάρτη, διευκολύνοντας την ακριβή καταχώριση της τοποθεσίας διεξαγωγής κάθε εκδήλωσης. Παράλληλα,

παρέχεται δυνατότητα δυναμικής διαχείρισης διαφορετικών τύπων εισιτηρίων, όπως κανονικά, φοιτητικά ή δωρεάν εισιτήρια, καλύπτοντας διαφορετικές ανάγκες συμμετοχής. Επιπλέον, κάθε διοργανωτής διαθέτει δημόσιο προφίλ (portfolio), στο οποίο συγκεντρώνονται οι εκδηλώσεις που έχει διοργανώσει, καθώς και ο μέσος όρος αξιολογήσεων που έχει λάβει από τους χρήστες, ενισχύοντας έτσι την αξιοπιστία και την ψηφιακή του παρουσία.

Συνολικά, η παρούσα εργασία στοχεύει στην ανάπτυξη μιας σύγχρονης και λειτουργικής πλατφόρμας, η οποία θα βελτιώνει την εμπειρία αναζήτησης και συμμετοχής σε εκδηλώσεις, ενώ ταυτόχρονα θα παρέχει στους διοργανωτές ένα αποτελεσματικό εργαλείο προβολής και διαχείρισης των δράσεών τους.

1.3 Βασικά Χαρακτηριστικά και Καινοτομίες

Η πλατφόρμα EventHub σχεδιάστηκε και αναπτύχθηκε ακολουθώντας σύγχρονες αρχές και πρακτικές Μηχανικής Λογισμικού (Software Engineering), με στόχο τη δημιουργία ενός ασφαλούς, επεκτάσιμου και αποδοτικού πληροφοριακού συστήματος. Η εφαρμογή ενσωματώνει τεχνολογίες και λειτουργίες που βελτιώνουν σημαντικά τόσο την εμπειρία χρήσης όσο και τη συνολική λειτουργικότητα της πλατφόρμας.

1. Ένα από τα βασικότερα χαρακτηριστικά του συστήματος είναι η υποστήριξη υπηρεσιών βασισμένων στην τοποθεσία (Location-Based Services) (Huang et al., 2018). Για την υλοποίηση της γεωγραφικής αναζήτησης αξιοποιείται το API του OpenStreetMap (Haklay & Weber, 2008), σε συνδυασμό με τον μαθηματικό τύπο Haversine (Winarno et al., 2017) στο επίπεδο του backend. Η προσέγγιση αυτή επιτρέπει τον υπολογισμό της χιλιομετρικής απόστασης μεταξύ δύο γεωγραφικών σημείων, παρέχοντας στους χρήστες τη δυνατότητα έξυπνης και αποδοτικής αναζήτησης εκδηλώσεων με βάση την τοποθεσία και την ακτίνα ενδιαφέροντος.
2. Παράλληλα, το EventHub υποστηρίζει δυναμική διαχείριση περιεχομένου μέσω της χρήσης Rich Text Editors¹, επιτρέποντας στους διοργανωτές να δημιουργούν αναλυτικές και μορφοποιημένες περιγραφές εκδηλώσεων. Για την

¹ Οι **Rich Text Editors** (Επεξεργαστές Εμπλουτισμένου Κειμένου) στα ελληνικά είναι εφαρμογές που επιτρέπουν τη συγγραφή και επεξεργασία κειμένου, εφαρμόζοντας μορφοποίηση (έντονα γράμματα, πλάγια, υπογράμμιση, εικόνες, πίνακες κ.λπ.).

προστασία του συστήματος από επιθέσεις τύπου Cross-Site Scripting (XSS) (Gurta & Gurta, 2015), εφαρμόζονται μηχανισμοί καθαρισμού και απολύμανσης περιεχομένου με χρήση της βιβλιοθήκης DOMPurify². Με τον τρόπο αυτό διασφαλίζεται η ασφαλής προβολή περιεχομένου χωρίς να περιορίζεται η λειτουργικότητα ή η ποιότητα παρουσίασης των πληροφοριών.

3. Επιπλέον, η πλατφόρμα διαθέτει ένα ευέλικτο και επεκτάσιμο σύστημα κρατήσεων και διαχείρισης εισιτηρίων. Οι διοργανωτές μπορούν να δημιουργούν και να διαχειρίζονται πολλαπλούς τύπους εισιτηρίων, όπως κανονικά, φοιτητικά ή δωρεάν, μέσω δυναμικών σχεσιακών δομών στη βάση δεδομένων. Παράλληλα, υποστηρίζεται η ενσωμάτωση εξωτερικών συνδέσμων κρατήσεων ή αγοράς εισιτηρίων, διευκολύνοντας τη διαδικασία συμμετοχής των χρηστών.
4. Ιδιαίτερη έμφαση δίνεται επίσης στη διαδραστικότητα και στα χαρακτηριστικά κοινωνικής δικτύωσης της εφαρμογής. Το σύστημα εφαρμόζει μηχανισμούς επιβεβαίωσης πραγματικής συμμετοχής (Verified Attendance), επιτρέποντας την υποβολή αξιολογήσεων και κριτικών μόνο από χρήστες που έχουν δηλώσει συμμετοχή στην εκδήλωση. Με αυτόν τον τρόπο ενισχύεται η αξιοπιστία του συστήματος αξιολογήσεων και περιορίζεται η πιθανότητα παραπλανητικών ή ψευδών κριτικών. Επιπλέον, πραγματοποιείται υπολογισμός συγκεντρωτικών στατιστικών στοιχείων σε πραγματικό χρόνο, όπως ο μέσος όρος βαθμολογίας κάθε διοργανωτή, προσφέροντας μια αντικειμενική εικόνα της συνολικής του δραστηριότητας και φήμης.

Συνολικά, τα τεχνολογικά χαρακτηριστικά του EventHub συμβάλλουν στη δημιουργία μιας σύγχρονης, ασφαλούς και διαδραστικής πλατφόρμας, η οποία ανταποκρίνεται στις απαιτήσεις τόσο των χρηστών όσο και των διοργανωτών εκδηλώσεων.

² <https://dompurify.com/>

2 ΑΝΑΛΥΣΗ ΑΠΑΙΤΗΣΕΩΝ

Η εφαρμογή **EventHub** σχεδιάστηκε με σκοπό να καλύψει τις βασικές ανάγκες που σχετίζονται με την **αναζήτηση, οργάνωση και διαχείριση εκδηλώσεων** μέσα από ένα ενιαίο και εύχρηστο ψηφιακό περιβάλλον. Η πλατφόρμα επιδιώκει να προσφέρει στους χρήστες έναν απλό και αποτελεσματικό τρόπο πρόσβασης σε εκδηλώσεις, επιτρέποντάς τους να ενημερώνονται για διαθέσιμες δραστηριότητες, να αναζητούν εκδηλώσεις που τους ενδιαφέρουν και να συμμετέχουν σε αυτές με οργανωμένο τρόπο.

Για την αποτελεσματική λειτουργία του συστήματος, είναι απαραίτητος ο καθορισμός συγκεκριμένων **λειτουργικών απαιτήσεων**, οι οποίες περιγράφουν τις βασικές δυνατότητες που πρέπει να παρέχει η εφαρμογή. Οι απαιτήσεις αυτές αφορούν λειτουργίες όπως η εγγραφή και σύνδεση χρηστών, η προβολή εκδηλώσεων, η αναζήτηση με βάση συγκεκριμένα κριτήρια, η δημιουργία και διαχείριση εκδηλώσεων από τους διοργανωτές, καθώς και η δυνατότητα συμμετοχής των χρηστών σε εκδηλώσεις. Με αυτόν τον τρόπο, η εφαρμογή μπορεί να ανταποκρίνεται στις ανάγκες διαφορετικών χρηστών και να υποστηρίζει με συνέπεια τις κύριες υπηρεσίες της.

Παράλληλα, το EventHub διαχωρίζει τους χρήστες σε διαφορετικές **κατηγορίες**, ανάλογα με τον ρόλο και τις αρμοδιότητές τους μέσα στο σύστημα. Κάθε κατηγορία χρηστών διαθέτει συγκεκριμένα δικαιώματα πρόσβασης και λειτουργικότητας, ώστε να διασφαλίζεται η σωστή χρήση της πλατφόρμας. Για παράδειγμα, οι απλοί χρήστες μπορούν να αναζητούν και να παρακολουθούν εκδηλώσεις, ενώ οι διοργανωτές έχουν τη δυνατότητα να δημιουργούν, να επεξεργάζονται και να διαχειρίζονται τις εκδηλώσεις τους.

Η συγκεκριμένη προσέγγιση συμβάλλει στην καλύτερη οργάνωση των υπηρεσιών της εφαρμογής και στη σαφή κατανομή των λειτουργιών ανάμεσα στους χρήστες. Επιπλέον, ενισχύει την ασφάλεια και την αξιοπιστία του συστήματος, καθώς κάθε χρήστης έχει πρόσβαση μόνο στις λειτουργίες που αντιστοιχούν στον ρόλο του. Ως αποτέλεσμα, βελτιώνεται η συνολική εμπειρία χρήσης και δημιουργείται ένα πιο δομημένο, λειτουργικό και φιλικό περιβάλλον τόσο για τους απλούς συμμετέχοντες όσο και για τους διοργανωτές εκδηλώσεων.

2.1 Κατηγορίες Χρηστών

Η πλατφόρμα **EventHub** υποστηρίζει τρεις βασικές κατηγορίες χρηστών, οι οποίες χωρίζονται με βάση τον ρόλο τους μέσα στο σύστημα και τις λειτουργίες στις οποίες έχουν πρόσβαση. Ο διαχωρισμός αυτός είναι σημαντικός, καθώς επιτρέπει την καλύτερη οργάνωση της εφαρμογής, την ασφαλή διαχείριση των δεδομένων και την παροχή διαφορετικών δυνατοτήτων ανάλογα με τις ανάγκες κάθε χρήστη. Οι βασικές κατηγορίες χρηστών είναι ο **Επισκέπτης**, το **Μέλος / Συμμετέχων** και ο **Διοργανωτής**.

2.1.1 Επισκέπτης (Guest)

Ο **επισκέπτης** αποτελεί τον μη εγγεγραμμένο χρήστη της πλατφόρμας. Πρόκειται για χρήστη που μπορεί να περιηγηθεί στην εφαρμογή χωρίς να απαιτείται η δημιουργία λογαριασμού ή η διαδικασία σύνδεσης. Η δυνατότητα αυτή διευκολύνει την άμεση πρόσβαση στο περιεχόμενο της πλατφόρμας και επιτρέπει σε κάθε ενδιαφερόμενο να γνωρίσει τις διαθέσιμες εκδηλώσεις πριν αποφασίσει να εγγραφεί.

Οι επισκέπτες έχουν τη δυνατότητα να πραγματοποιούν βασικές αναζητήσεις εκδηλώσεων και να προβάλλουν γενικές πληροφορίες για αυτές. Ενδεικτικά, μπορούν να δουν τον τίτλο μιας εκδήλωσης, την περιγραφή της, την ημερομηνία διεξαγωγής, την τοποθεσία, καθώς και βασικά στοιχεία που σχετίζονται με το περιεχόμενό της. Με αυτόν τον τρόπο, ο χρήστης μπορεί να ενημερωθεί γρήγορα για εκδηλώσεις που τον ενδιαφέρουν.

Ωστόσο, η πρόσβαση του επισκέπτη σε πιο διαδραστικές ή προσωποποιημένες λειτουργίες είναι περιορισμένη. Για παράδειγμα, δεν έχει τη δυνατότητα να δηλώσει συμμετοχή σε κάποια εκδήλωση, να καταθέσει αξιολόγηση, να αποθηκεύσει προτιμήσεις ή να δημιουργήσει και να διαχειριστεί προσωπικό προφίλ. Οι περιορισμοί αυτοί είναι απαραίτητοι, ώστε να προστατεύονται τα δεδομένα της πλατφόρμας και να διασφαλίζεται ότι οι ενέργειες που δημιουργούν περιεχόμενο που καταχωρείται στην πλατφόρμα πραγματοποιούνται μόνο από εγγεγραμμένους χρήστες.

2.1.2 Μέλος / Συμμετέχων (Member / Attendee)

Το **μέλος** ή **συμμετέχων** είναι εγγεγραμμένος χρήστης της πλατφόρμας, ο οποίος διαθέτει προσωπικό λογαριασμό και έχει πρόσβαση σε περισσότερες λειτουργίες σε

σχέση με τον επισκέπτη. Η εγγραφή επιτρέπει στο σύστημα να αναγνωρίζει τον χρήστη και να του παρέχει μια πιο ολοκληρωμένη και προσωποποιημένη εμπειρία χρήσης.

Οι χρήστες αυτής της κατηγορίας μπορούν να δηλώνουν συμμετοχή σε εκδηλώσεις που τους ενδιαφέρουν, να παρακολουθούν το ιστορικό των συμμετοχών τους και να διαχειρίζονται τις προσωπικές τους πληροφορίες μέσω του προφίλ τους. Επιπλέον, μπορούν να αποθηκεύουν προτιμήσεις, ώστε η πλατφόρμα να υποστηρίζει καλύτερα τις ανάγκες και τα ενδιαφέροντά τους.

Μία ακόμη σημαντική δυνατότητα των μελών είναι η ενεργός αλληλεπίδραση με το περιεχόμενο της εφαρμογής. Συγκεκριμένα, οι συμμετέχοντες μπορούν να αξιολογούν εκδηλώσεις μέσω βαθμολογιών και κριτικών, εισφέροντας έτσι στο περιεχόμενο της πλατφόρμας και συμβάλλοντας στην επαύξηση της ποιότητας των παρεχόμενων υπηρεσιών. Παράλληλα, έχουν τη δυνατότητα να κοινοποιούν φωτογραφικό υλικό από εκδηλώσεις στις οποίες συμμετείχαν, ενισχύοντας τον κοινωνικό και διαδραστικό χαρακτήρα της πλατφόρμας και εμπλουτίζοντας το πολυμεσικό της περιεχόμενο.

Με αυτόν τον τρόπο, η λειτουργικότητα που έχει διαθέσιμη το μέλος επεκτείνεται πέρα από την αναζήτηση πληροφοριών, περιλαμβάνοντας την ενεργό συμμετοχή στη λειτουργία της εφαρμογής. Η κατηγορία αυτή αποτελεί βασικό κομμάτι του EventHub, καθώς οι συμμετέχοντες είναι οι κύριοι αποδέκτες των εκδηλώσεων και συμβάλλουν στη δημιουργία μιας δραστήριας και «ζωντανής» κοινότητας χρηστών.

2.1.3 Διοργανωτής (Organizer)

Ο **διοργανωτής** αποτελεί ειδική κατηγορία εγγεγραμμένου χρήστη, η οποία διαθέτει αυξημένα δικαιώματα πρόσβασης και διαχείρισης. Εκτός από τις λειτουργίες που παρέχονται στα απλά μέλη, ο διοργανωτής έχει τη δυνατότητα να δημιουργεί και να διαχειρίζεται εκδηλώσεις μέσα από την πλατφόρμα. Ο ρόλος του είναι ιδιαίτερα σημαντικός, καθώς αποτελεί τον βασικό πάροχο περιεχομένου του συστήματος.

Οι διοργανωτές μπορούν να δημιουργούν νέες εκδηλώσεις, εισάγοντας όλες τις απαραίτητες πληροφορίες, όπως τίτλο, περιγραφή, ημερομηνία, ώρα, τοποθεσία και κατηγορία εκδήλωσης. Επιπλέον, έχουν τη δυνατότητα να επεξεργάζονται ή να ενημερώνουν τα στοιχεία μιας εκδήλωσης, σε περίπτωση αλλαγών ή νέων

πληροφοριών. Η δυνατότητα αυτή εξασφαλίζει ότι το περιεχόμενο της πλατφόρμας παραμένει ενημερωμένο και αξιόπιστο.

Παράλληλα, οι διοργανωτές μπορούν να διαχειρίζονται πληροφορίες που σχετίζονται με εισιτήρια, κρατήσεις και διαθέσιμες θέσεις, όπου αυτό απαιτείται. Μέσω αυτών των λειτουργιών, η πλατφόρμα διευκολύνει την οργάνωση των εκδηλώσεων και προσφέρει στους διοργανωτές ένα πιο ολοκληρωμένο εργαλείο διαχείρισης.

Επιπλέον, κάθε διοργανωτής μπορεί να διαθέτει δημόσιο προφίλ, μέσα από το οποίο παρουσιάζονται οι δράσεις και οι εκδηλώσεις που έχει δημιουργήσει. Το προφίλ αυτό λειτουργεί ως μέσο προβολής και ενίσχυσης της αξιοπιστίας του διοργανωτή, καθώς οι χρήστες μπορούν να ενημερώνονται για προηγούμενες ή μελλοντικές εκδηλώσεις του.

Συνολικά, ο διοργανωτής διαδραματίζει κεντρικό ρόλο στη λειτουργία του EventHub, καθώς είναι υπεύθυνος για τη δημιουργία, ενημέρωση και οργάνωση του βασικού περιεχομένου της πλατφόρμας. Μέσα από τα αυξημένα δικαιώματα που διαθέτει, συμβάλλει ουσιαστικά στη συνεχή λειτουργία και ανάπτυξη του συστήματος.

2.2 Λειτουργικές Απαιτήσεις ανά Κατηγορία

Οι **λειτουργικές απαιτήσεις** του συστήματος καθορίζουν τις βασικές δυνατότητες και τις ενέργειες που υποστηρίζονται από την πλατφόρμα **EventHub**. Μέσα από αυτές περιγράφεται ο τρόπος με τον οποίο κάθε κατηγορία χρήστη αλληλεπιδρά με την εφαρμογή, καθώς και οι υπηρεσίες που μπορεί να αξιοποιήσει ανάλογα με τον ρόλο της.

Οι απαιτήσεις αυτές διαμορφώνονται με βάση τις ανάγκες των χρηστών και έχουν ως στόχο να εξασφαλίσουν μια ολοκληρωμένη, εύχρηστη και αποδοτική εμπειρία χρήσης. Παράλληλα, συμβάλλουν στη σωστή οργάνωση της πλατφόρμας, καθώς κάθε λειτουργία συνδέεται με συγκεκριμένα δικαιώματα πρόσβασης και επίπεδα αλληλεπίδρασης.

2.2.1 Επισκέπτης (Guest)

Ο **επισκέπτης** αποτελεί τον μη εγγεγραμμένο χρήστη της πλατφόρμας και έχει πρόσβαση στις βασικές λειτουργίες του συστήματος. Η κατηγορία αυτή μπορεί να

χρησιμοποιήσει την εφαρμογή χωρίς να απαιτείται η δημιουργία λογαριασμού ή η σύνδεση στο σύστημα. Με αυτόν τον τρόπο, κάθε χρήστης έχει τη δυνατότητα να περιηγηθεί ελεύθερα στο περιεχόμενο της πλατφόρμας και να ενημερωθεί για τις διαθέσιμες εκδηλώσεις.

Η πρόσβαση του επισκέπτη επικεντρώνεται κυρίως στην αναζήτηση και προβολή πληροφοριών. Παρ' όλο που δεν μπορεί να χρησιμοποιήσει διαδραστικές λειτουργίες, όπως δήλωση συμμετοχής ή αξιολόγηση εκδηλώσεων, μπορεί να αποκτήσει μια σαφή εικόνα για το περιεχόμενο και τις δυνατότητες της εφαρμογής. Η λειτουργία αυτή είναι σημαντική, καθώς επιτρέπει στον χρήστη να γνωρίσει την πλατφόρμα πριν αποφασίσει να δημιουργήσει προσωπικό λογαριασμό.

2.2.1.1 Αναζήτηση και Φιλτράρισμα Εκδηλώσεων

Ο επισκέπτης μπορεί να πραγματοποιεί αναζήτηση εκδηλώσεων με βάση διάφορα κριτήρια, όπως λέξεις-κλειδιά, κατηγορίες, ημερομηνίες ή τοποθεσίες. Η δυνατότητα αυτή διευκολύνει τον χρήστη να εντοπίζει γρήγορα εκδηλώσεις που ανταποκρίνονται στα ενδιαφέροντά του, χωρίς να χρειάζεται να περιηγείται σε μεγάλο όγκο πληροφοριών.

Παράλληλα, η πλατφόρμα παρέχει δυνατότητες φιλτραρίσματος των αποτελεσμάτων. Μέσω των φίλτρων, ο επισκέπτης μπορεί να περιορίσει τα αποτελέσματα αναζήτησης και να εμφανίσει μόνο τις εκδηλώσεις που πληρούν συγκεκριμένα χαρακτηριστικά. Για παράδειγμα, μπορεί να αναζητήσει εκδηλώσεις που πραγματοποιούνται σε συγκεκριμένη περιοχή, σε συγκεκριμένη χρονική περίοδο ή ανήκουν σε συγκεκριμένη θεματική κατηγορία.

Η λειτουργία αναζήτησης και φιλτραρίσματος είναι θεμελιώδης για την ευχρηστία της εφαρμογής, καθώς καθιστά πρακτικά εφικτή την εύρεση εκδηλώσεων που ταιριάζουν με τα ενδιαφέροντα των χρηστών, καθιστώντας έτσι τη χρήση της πλατφόρμας αποδοτική.

2.2.1.2 Προβολή Λεπτομερειών Εκδήλωσης

Η πλατφόρμα επιτρέπει στον επισκέπτη να προβάλλει αναλυτικές πληροφορίες για κάθε διαθέσιμη εκδήλωση. Οι πληροφορίες αυτές μπορεί να περιλαμβάνουν τον τίτλο

της εκδήλωσης, την περιγραφή της, την ημερομηνία και ώρα διεξαγωγής, την τοποθεσία, καθώς και στοιχεία σχετικά με τη συμμετοχή ή τη διαθεσιμότητα εισιτηρίων.

Επιπλέον, η τοποθεσία της εκδήλωσης μπορεί να παρουσιάζεται μέσω διαδραστικού χάρτη, ώστε ο χρήστης να μπορεί να εντοπίσει εύκολα το σημείο διεξαγωγής. Η δυνατότητα αυτή καθιστά την πληροφόρηση πιο ολοκληρωμένη και βοηθά τον επισκέπτη να αξιολογήσει καλύτερα αν μια εκδήλωση τον ενδιαφέρει.

Η προβολή λεπτομερειών αποτελεί βασική λειτουργία της πλατφόρμας, καθώς παρέχει στον χρήστη όλες τις απαραίτητες πληροφορίες πριν προχωρήσει σε εγγραφή ή συμμετοχή. Έτσι, ο επισκέπτης μπορεί να λαμβάνει πιο ενημερωμένες αποφάσεις σχετικά με τις εκδηλώσεις που επιθυμεί να παρακολουθήσει.

2.2.1.3 Εγγραφή και Σύνδεση

Ο επισκέπτης έχει τη δυνατότητα να δημιουργήσει προσωπικό λογαριασμό στην πλατφόρμα, εφόσον επιθυμεί να αποκτήσει πρόσβαση σε περισσότερες λειτουργίες. Η διαδικασία εγγραφής επιτρέπει τη μετάβαση από τον ρόλο του απλού επισκέπτη στον ρόλο του μέλους ή συμμετέχοντα.

Με τη σύνδεση στο σύστημα, ο χρήστης μπορεί να αξιοποιήσει προχωρημένες διαδραστικές λειτουργίες, όπως δήλωση συμμετοχής σε εκδηλώσεις, αποθήκευση προτιμήσεων, αξιολόγηση εκδηλώσεων και διαχείριση προσωπικού προφίλ. Με αυτόν τον τρόπο, η πλατφόρμα προσφέρει μια πιο προσωποποιημένη εμπειρία χρήσης και ενισχύει την αλληλεπίδραση του χρήστη με το περιεχόμενο.

Η δυνατότητα εγγραφής και σύνδεσης αποτελεί το κεντρικό εργαλείο για τον χρήστη προκειμένου να εμπλουτίσει τη λειτουργικότητα που του είναι διαθέσιμη, καθώς του επιτρέπει να αξιοποιήσει πλήρως τις υπηρεσίες του EventHub και να συμμετέχει ενεργά στο οικοσύστημα της πλατφόρμας.

2.2.2 Μέλος / Συμμετέχων (Member)

Το μέλος αποτελεί εγγεγραμμένο χρήστη της εφαρμογής και διαθέτει πρόσβαση σε επιπλέον λειτουργίες αλληλεπίδρασης, συμμετοχής και προσωποποιημένης χρήσης. Σε αντίθεση με τον επισκέπτη, το μέλος έχει δημιουργήσει προσωπικό λογαριασμό και

μπορεί να αξιοποιήσει περισσότερες δυνατότητες της πλατφόρμας, οι οποίες σχετίζονται με τη συμμετοχή του σε εκδηλώσεις και τη διαχείριση των προσωπικών του ενεργειών.

Η κατηγορία αυτή είναι ιδιαίτερα σημαντική για τη λειτουργία του **EventHub**, καθώς οι εγγεγραμμένοι χρήστες αποτελούν το βασικό κοινό των εκδηλώσεων που έχει τη δυνατότητα να εισφέρει περιεχόμενο στην πλατφόρμα. Μέσα από τις διαθέσιμες λειτουργίες, το μέλος δεν περιορίζεται μόνο στην αναζήτηση πληροφοριών, αλλά μπορεί να αλληλεπιδρά ενεργά με την πλατφόρμα, να καταγράψει τη συμμετοχή του σε εκδηλώσεις, να αξιολογεί την εμπειρία του και να συμβάλλει στη δημιουργία μιας πιο ενεργής κοινότητας χρηστών.

2.2.2.1 Δήλωση Συμμετοχής (Join Event)

Οι χρήστες έχουν τη δυνατότητα να δηλώνουν συμμετοχή ή ενδιαφέρον για συγκεκριμένες εκδηλώσεις που είναι διαθέσιμες στην πλατφόρμα. Η λειτουργία αυτή επιτρέπει στο μέλος να οργανώνει καλύτερα τις εκδηλώσεις που επιθυμεί να παρακολουθήσει και να διατηρεί μια σαφή εικόνα των συμμετοχών του.

Με τη δήλωση συμμετοχής, η εκδήλωση συσχετίζεται με το προσωπικό προφίλ του χρήστη, καθιστώντας έτσι εύκολη τη μελλοντική αναζήτηση και παρακολούθησή της. Παράλληλα, η λειτουργία αυτή είναι χρήσιμη και για τους διοργανωτές, καθώς τους επιτρέπει να γνωρίζουν τον αριθμό των χρηστών που έχουν δηλώσει ενδιαφέρον ή συμμετοχή, και να εκτιμήσουν έτσι το ενδιαφέρον για την εκδήλωση ή το πλήθος που θα συμμετέχει τελικά σε αυτή. Η δυνατότητα αυτή συμβάλλει έτσι στην καλύτερη οργάνωση των εκδηλώσεων, καθώς διευκολύνει τόσο τους συμμετέχοντες όσο και τους διοργανωτές στη διαχείριση της συμμετοχής και της προσέλευσης.

2.2.2.2 Διαχείριση Προσωπικού Προφίλ

Κάθε μέλος διαθέτει προσωπικό προφίλ, μέσω του οποίου μπορεί να διαχειρίζεται τις πληροφορίες και τις δραστηριότητές του μέσα στην πλατφόρμα. Το προφίλ λειτουργεί ως κεντρικό σημείο πρόσβασης στις προσωπικές ενέργειες του χρήστη και ενισχύει την προσωποποιημένη εμπειρία χρήσης.

Μέσα από το προσωπικό του προφίλ, το μέλος μπορεί να προβάλλει το ιστορικό συμμετοχών του, καθώς και τις εκδηλώσεις στις οποίες έχει λάβει μέρος ή έχει αλληλεπιδράσει. Επιπλέον, μπορεί να ενημερώνει βασικά στοιχεία του λογαριασμού του και να παρακολουθεί τις ενέργειες που σχετίζονται με τη χρήση της πλατφόρμας.

Η διαχείριση προσωπικού προφίλ προσφέρει στον χρήστη καλύτερο έλεγχο των δεδομένων του και διευκολύνει την οργάνωση των εκδηλώσεων που τον ενδιαφέρουν. Με αυτόν τον τρόπο, η εφαρμογή γίνεται πιο φιλική και προσαρμοσμένη στις ανάγκες κάθε μέλους.

2.2.2.3 Υποβολή Κριτικής και Αξιολόγησης

Μετά την ολοκλήρωση μιας εκδήλωσης, οι χρήστες που έχουν δηλώσει συμμετοχή μπορούν να υποβάλλουν βαθμολογίες και σχόλια σχετικά με την εμπειρία τους. Η λειτουργία αυτή επιτρέπει στα μέλη να εκφράζουν την άποψή τους και να παρέχουν χρήσιμη ανατροφοδότηση τόσο προς τους διοργανωτές όσο και προς άλλους χρήστες της πλατφόρμας.

Οι αξιολογήσεις μπορούν να αφορούν διάφορες πτυχές μιας εκδήλωσης, όπως την οργάνωση, το περιεχόμενο, την τοποθεσία ή τη συνολική εμπειρία συμμετοχής. Με αυτόν τον τρόπο δημιουργείται ένα πιο αξιόπιστο σύστημα πληροφόρησης, το οποίο βοηθά τους μελλοντικούς συμμετέχοντες να σχηματίσουν καλύτερη εικόνα για μια εκδήλωση ή έναν διοργανωτή.

Παράλληλα, η υποβολή κριτικών συμβάλλει στη συνεχή βελτίωση της ποιότητας των εκδηλώσεων. Οι διοργανωτές μπορούν να αξιοποιούν τα σχόλια των χρηστών, ώστε να εντοπίζουν δυνατά σημεία αλλά και περιθώρια βελτίωσης στις μελλοντικές τους δράσεις.

2.2.2.4 Μεταφόρτωση Φωτογραφικού Υλικού

Οι συμμετέχοντες έχουν τη δυνατότητα να προσθέτουν φωτογραφίες στη συλλογή μιας εκδήλωσης μετά την ολοκλήρωσή της. Η λειτουργία αυτή ενισχύει τον διαδραστικό χαρακτήρα της πλατφόρμας και επιτρέπει στους χρήστες να μοιράζονται στιγμές και εμπειρίες από τις εκδηλώσεις στις οποίες συμμετείχαν.

Το φωτογραφικό υλικό μπορεί να λειτουργήσει ως μέσο προβολής της εκδήλωσης, αλλά και ως τρόπος ενίσχυσης της κοινότητας γύρω από την πλατφόρμα. Μέσα από τη συλλογή φωτογραφιών, οι χρήστες μπορούν να ανατρέχουν σε προηγούμενες εκδηλώσεις, να βλέπουν στιγμιότυπα από τη συμμετοχή άλλων μελών και να αποκτούν μια πιο «ζωντανή» εικόνα της εμπειρίας που προσφέρει κάθε εκδήλωση.

Η δυνατότητα μεταφόρτωσης φωτογραφιών συμβάλλει επίσης στην κοινωνική διάσταση του EventHub, καθώς ενθαρρύνει τη συμμετοχή, την αλληλεπίδραση και τη δημιουργία περιεχομένου από τους ίδιους τους χρήστες. Έτσι, η πλατφόρμα δεν λειτουργεί μόνο ως εργαλείο αναζήτησης εκδηλώσεων, αλλά και ως χώρος ανταλλαγής εμπειριών μεταξύ των μελών της κοινότητας.

2.2.3 Διοργανωτής (Organizer)

Ο **διοργανωτής** αποτελεί ειδική κατηγορία εγγεγραμμένου χρήστη της πλατφόρμας και διαθέτει διευρυμένα δικαιώματα διαχείρισης περιεχομένου και εκδηλώσεων. Ο ρόλος του είναι ιδιαίτερα σημαντικός για τη λειτουργία του **EventHub**, καθώς οι διοργανωτές είναι υπεύθυνοι για τη δημιουργία, την ενημέρωση και τη διαχείριση των εκδηλώσεων που προβάλλονται στην εφαρμογή.

Σε αντίθεση με τα απλά μέλη, οι διοργανωτές έχουν πρόσβαση σε πιο σύνθετες λειτουργίες, οι οποίες τους επιτρέπουν να οργανώνουν ολοκληρωμένα τις δράσεις τους. Μέσα από την πλατφόρμα μπορούν να καταχωρίζουν νέες εκδηλώσεις, να διαχειρίζονται πληροφορίες σχετικά με εισιτήρια και συμμετοχές, καθώς και να διατηρούν δημόσιο επαγγελματικό προφίλ. Με αυτόν τον τρόπο, το EventHub λειτουργεί ως εργαλείο προβολής, οργάνωσης και επικοινωνίας για κάθε διοργανωτή.

2.2.3.1 Δημιουργία Εκδήλωσης

Ο διοργανωτής έχει τη δυνατότητα να δημιουργεί νέες εκδηλώσεις μέσα από ειδική φόρμα καταχώρησης. Κατά τη διαδικασία αυτή μπορεί να εισάγει όλες τις απαραίτητες πληροφορίες που χρειάζεται ένας χρήστης για να ενημερωθεί σωστά σχετικά με την εκδήλωση.

Συγκεκριμένα, ο διοργανωτής μπορεί να καταχωρίζει τον τίτλο της εκδήλωσης, την αναλυτική περιγραφή της, την κατηγορία στην οποία ανήκει, τις ημερομηνίες και ώρες

διεξαγωγής, καθώς και την ακριβή τοποθεσία της. Η περιγραφή μπορεί να υποστηρίζεται από **Rich Text Editor**, ώστε ο διοργανωτής να έχει τη δυνατότητα μορφοποίησης του περιεχομένου, όπως χρήση παραγράφων, έντονης γραφής, λιστών ή άλλων στοιχείων που κάνουν την παρουσίαση πιο ευανάγνωστη και ελκυστική.

Επιπλέον, η ακριβής τοποθεσία της εκδήλωσης μπορεί να ορίζεται μέσω διαδραστικού χάρτη. Η λειτουργία αυτή διευκολύνει τόσο τον διοργανωτή, κατά την καταχώρηση της εκδήλωσης, όσο και τους χρήστες, οι οποίοι μπορούν να εντοπίσουν εύκολα το σημείο διεξαγωγής. Έτσι, η δημιουργία εκδήλωσης γίνεται με οργανωμένο και ολοκληρωμένο τρόπο.

2.2.3.2 Διαχείριση Εισιτηρίων

Η πλατφόρμα υποστηρίζει τη δημιουργία και διαχείριση πολλαπλών τύπων εισιτηρίων, προσφέροντας ευελιξία στον τρόπο οργάνωσης κάθε εκδήλωσης. Ο διοργανωτής μπορεί να καταχωρίζει διαφορετικές κατηγορίες εισιτηρίων, ανάλογα με τις ανάγκες της εκδήλωσης και το κοινό στο οποίο απευθύνεται.

Ενδεικτικά, μπορούν να υπάρχουν κανονικά εισιτήρια, φοιτητικά εισιτήρια, δωρεάν εισιτήρια ή άλλες ειδικές κατηγορίες, ανάλογα με την πολιτική του διοργανωτή. Κάθε τύπος εισιτηρίου μπορεί να συνοδεύεται από σχετικές πληροφορίες, όπως περιγραφή, τιμή ή διαθεσιμότητα, ώστε οι χρήστες να γνωρίζουν τις επιλογές που έχουν πριν δηλώσουν συμμετοχή σε κράτηση.

Παράλληλα, παρέχεται η δυνατότητα προσθήκης εξωτερικών συνδέσμων για κρατήσεις ή ηλεκτρονική αγορά εισιτηρίων. Η λειτουργία αυτή είναι χρήσιμη όταν η διαδικασία αγοράς πραγματοποιείται μέσω τρίτης υπηρεσίας ή εξωτερικής πλατφόρμας. Με αυτόν τον τρόπο, το EventHub μπορεί να λειτουργεί ως κεντρικό σημείο πληροφόρησης, ενώ παράλληλα υποστηρίζει και εξωτερικές διαδικασίες κράτησης ή πληρωμής.

2.2.3.3 Επεξεργασία και Διαγραφή Εκδηλώσεων

Οι διοργανωτές έχουν πλήρη δυνατότητα επεξεργασίας των εκδηλώσεων που έχουν δημιουργήσει. Η λειτουργία αυτή είναι απαραίτητη, καθώς οι πληροφορίες μιας εκδήλωσης μπορεί να χρειαστεί να τροποποιηθούν μετά την αρχική καταχώρηση. Για

παράδειγμα, μπορεί να αλλάξει η ημερομηνία διεξαγωγής, η τοποθεσία, η περιγραφή, η διαθεσιμότητα εισιτηρίων ή άλλες σημαντικές λεπτομέρειες.

Μέσω της δυνατότητας επεξεργασίας, ο διοργανωτής μπορεί να διατηρεί το περιεχόμενο της εκδήλωσης ενημερωμένο και αξιόπιστο. Αυτό συμβάλλει στη σωστή ενημέρωση των χρηστών και μειώνει την πιθανότητα σύγχυσης ή λανθασμένης πληροφόρησης.

Επιπλέον, οι διοργανωτές μπορούν να διαγράφουν εκδηλώσεις που έχουν δημιουργήσει, όταν αυτές δεν είναι πλέον ενεργές, έχουν ακυρωθεί ή δεν πρέπει να εμφανίζονται στην πλατφόρμα. Η δυνατότητα διαγραφής ενισχύει τον έλεγχο του διοργανωτή πάνω στο περιεχόμενό του και βοηθά στη διατήρηση μιας καθαρής και οργανωμένης πλατφόρμας.

2.2.3.4 Δημόσιο Επαγγελματικό Προφίλ

Κάθε διοργανωτής διαθέτει δημόσια σελίδα προβολής, η οποία λειτουργεί ως επαγγελματικό προφίλ ή **portfolio** μέσα στην πλατφόρμα. Η σελίδα αυτή παρουσιάζει βασικές πληροφορίες σχετικά με τον διοργανωτή και τη δραστηριότητά του, προσφέροντας στους χρήστες τη δυνατότητα να γνωρίσουν καλύτερα το άτομο, την ομάδα ή τον οργανισμό που βρίσκεται πίσω από μια εκδήλωση.

Το δημόσιο προφίλ μπορεί να περιλαμβάνει στοιχεία όπως βιογραφικό/επιχειρηματικό προφίλ ή σύντομη περιγραφή, στοιχεία επικοινωνίας, διεύθυνση, τηλέφωνο και ηλεκτρονική διεύθυνση επικοινωνίας. Επιπλέον, μπορεί να παρουσιάζει τις εκδηλώσεις που έχει δημιουργήσει ο διοργανωτής, ενισχύοντας την προβολή και την αξιοπιστία του.

Η ύπαρξη δημόσιου επαγγελματικού προφίλ είναι σημαντική, καθώς προσφέρει μεγαλύτερη διαφάνεια και ενισχύει την εμπιστοσύνη των χρηστών. Οι συμμετέχοντες μπορούν να ενημερωθούν για την εμπειρία, τη δραστηριότητα και την αξιοπιστία του διοργανωτή πριν αποφασίσουν να συμμετάσχουν σε κάποια εκδήλωση.

2.2.3.5 Προβολή Λίστας Συμμετεχόντων

Η πλατφόρμα παρέχει στους διοργανωτές τη δυνατότητα προβολής της λίστας των χρηστών που έχουν δηλώσει συμμετοχή στις εκδηλώσεις τους. Η λειτουργία αυτή

διευκολύνει σημαντικά την οργάνωση και την παρακολούθηση της συμμετοχής του κοινού.

Μέσα από τη λίστα συμμετεχόντων, ο διοργανωτής μπορεί να γνωρίζει τον αριθμό των ατόμων που έχουν εκδηλώσει ενδιαφέρον ή έχουν δηλώσει συμμετοχή σε μια εκδήλωση. Η πληροφορία αυτή είναι ιδιαίτερα χρήσιμη για τον προγραμματισμό της εκδήλωσης, καθώς βοηθά στην καλύτερη εκτίμηση της προσέλευσης, της χωρητικότητας του χώρου και των απαραίτητων οργανωτικών ενεργειών.

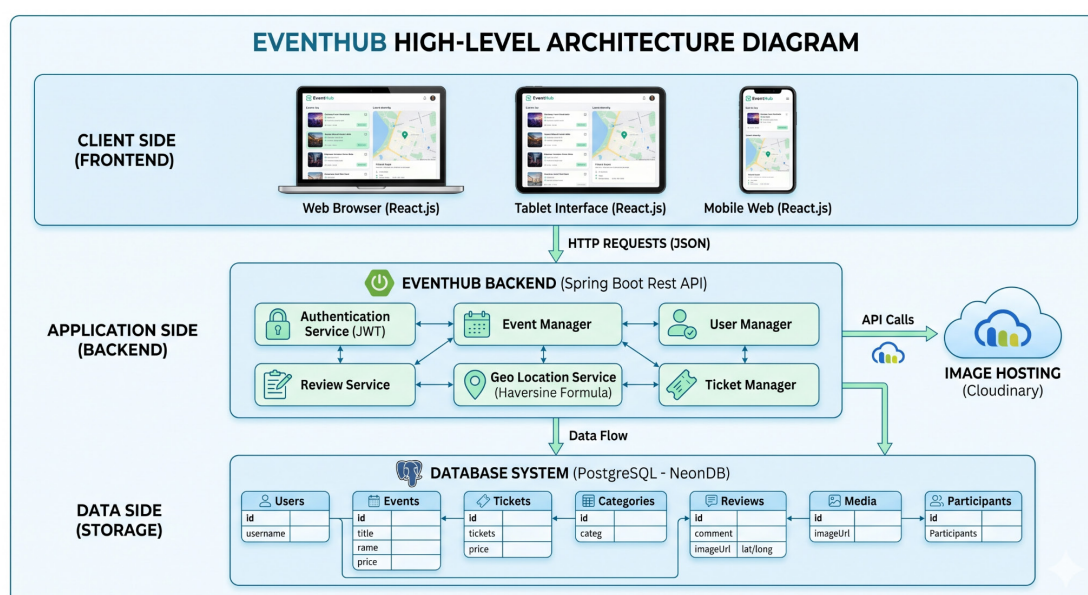
Παράλληλα, η λειτουργία αυτή μπορεί να υποστηρίξει καλύτερη επικοινωνία και διαχείριση πριν από την πραγματοποίηση της εκδήλωσης. Ο διοργανωτής έχει στη διάθεσή του μια πιο ολοκληρωμένη εικόνα του κοινού του, γεγονός που συμβάλλει στην αποτελεσματικότερη οργάνωση και στην παροχή καλύτερης εμπειρίας στους συμμετέχοντες.

Συνολικά, οι λειτουργίες που παρέχονται στον διοργανωτή καθιστούν το EventHub ένα ολοκληρωμένο εργαλείο διαχείρισης εκδηλώσεων. Μέσα από τη δημιουργία περιεχομένου, τη διαχείριση εισιτηρίων, την ενημέρωση εκδηλώσεων, την προβολή επαγγελματικού προφίλ και την παρακολούθηση συμμετοχών, ο διοργανωτής μπορεί να οργανώνει και να προβάλλει αποτελεσματικά τις δράσεις του μέσα από ένα ενιαίο ψηφιακό περιβάλλον.

3 ΣΧΕΔΙΑΣΗ ΣΥΣΤΗΜΑΤΟΣ

Η εφαρμογή **EventHub** βασίζεται σε σύγχρονες αρχιτεκτονικές προσεγγίσεις ανάπτυξης διαδικτυακών εφαρμογών, με στόχο τη δημιουργία ενός ασφαλούς και αποδοτικού συστήματος. Η σχεδίαση της πλατφόρμας ακολουθεί τον διαχωρισμό Frontend και Backend, επιτρέποντας την ανεξάρτητη ανάπτυξη και διαχείριση των επιμέρους τμημάτων της εφαρμογής.

Παράλληλα, ιδιαίτερη έμφαση δόθηκε στη σωστή οργάνωση της επικοινωνίας μεταξύ πελάτη και εξυπηρετητή μέσω RESTful APIs (Masse, 2011), καθώς και στη διαχείριση των δεδομένων μέσα από σχεσιακή βάση δεδομένων. Η αρχιτεκτονική αυτή συμβάλλει στη βελτίωση της συντηρησιμότητας, της απόδοσης, της αντικαταστασιμότητας των συνιστωσών και της δυνατότητας μελλοντικής επέκτασης του συστήματος.



Εικόνα 1. Αρχιτεκτονική της Εφαρμογής

3.1 Αρχιτεκτονική Συστήματος

Η εφαρμογή EventHub υλοποιείται με βάση την αρχιτεκτονική προσέγγιση **Client-Server**, η οποία διαχωρίζει πλήρως το επίπεδο παρουσίασης (Presentation Layer) από τη λογική της εφαρμογής και τη διαχείριση των δεδομένων. Ο διαχωρισμός αυτός συμβάλλει στη βελτίωση της επεκτασιμότητας, της συντηρησιμότητας και της ασφάλειας του συστήματος, επιτρέποντας την ανεξάρτητη ανάπτυξη και διαχείριση των επιμέρους τμημάτων της εφαρμογής.

Η συνολική αρχιτεκτονική του συστήματος αποτελείται από τρία βασικά επίπεδα, (α) το Frontend – Περιβάλλον Χρήστη (user interface layer), (β) το Backend – Εξυπηρετητή Εφαρμογών (application logic layer) και (γ) το επίπεδο διαχείρισης δεδομένων (data layer). Τα τρία αυτά επίπεδα περιγράφονται συνοπτικά στις ακόλουθες παραγράφους.

3.1.1 Frontend – Περιβάλλον Χρήστη (React.js)

Το επίπεδο διεπαφής χρήστη υλοποιείται με τη χρήση της βιβλιοθήκης **React.js**, ακολουθώντας το πρότυπο των Single Page Applications (SPA) (Fink & Flatow, 2014). Η προσέγγιση αυτή επιτρέπει τη δυναμική φόρτωση περιεχομένου χωρίς πλήρη ανανέωση της σελίδας, προσφέροντας ταχύτερη και πιο διαδραστική εμπειρία χρήσης.

Το Frontend είναι υπεύθυνο για:

- τη διαχείριση της κατάστασης (state) της εφαρμογής,
- την αλληλεπίδραση του χρήστη με το σύστημα,
- την προβολή εκδηλώσεων και δεδομένων,
- την εμφάνιση διαδραστικών χαρτών και γεωγραφικών πληροφοριών,
- καθώς και την επικοινωνία με το Backend μέσω RESTful API endpoints.

Η χρήση της React.js³ συμβάλλει στη δημιουργία ενός σύγχρονου και επεκτάσιμου περιβάλλοντος διεπαφής, το οποίο μπορεί να προσαρμοστεί εύκολα σε μελλοντικές λειτουργικές απαιτήσεις.

3.1.2 Backend – Εξυπηρετητής Εφαρμογών (Spring Boot)

Το Backend της εφαρμογής αναπτύχθηκε με το framework **Spring Boot** (VMware, 2026), το οποίο παρέχει ένα ισχυρό και αξιόπιστο περιβάλλον ανάπτυξης εφαρμογών βασισμένων στη γλώσσα προγραμματισμού Java.

Ο εξυπηρετητής εφαρμογών είναι υπεύθυνος για:

- την υλοποίηση της επιχειρησιακής λογικής (Business Logic),
- τη διαχείριση των χρηστών και των εκδηλώσεων,
- την επεξεργασία και επικύρωση δεδομένων,

³ <https://react.dev/>

- την εφαρμογή μηχανισμών ασφάλειας και πιστοποίησης χρηστών, οι οποίοι αξιοποιούν την προσέγγιση JWT Authentication (Peyrott, 2024),
- καθώς και την εκτέλεση γεωγραφικών υπολογισμών για την υποστήριξη λειτουργιών αναζήτησης βάσει τοποθεσίας.

Η αρχιτεκτονική REST API που ακολουθείται επιτρέπει την αποδοτική και ασφαλή ανταλλαγή δεδομένων μεταξύ Frontend και Backend, εγκαθιδρύοντας ένα συμβόλαιο (contract) μεταξύ των δύο επιπέδων και επιτρέποντας την ανεξάρτητη ανάπτυξη εξέλιξη και προσαρμογή του κάθε επιπέδου, υπό την προϋπόθεση ότι το συμβόλαιο παραμένει σεβαστό (Jakob et al., 2008).

3.1.3 Βάση Δεδομένων – PostgreSQL (NeonDB)

Για την αποθήκευση και διαχείριση των δεδομένων της εφαρμογής χρησιμοποιείται η σχεσιακή βάση δεδομένων **PostgreSQL** (Ferrari & Pirozzi, 2023), μέσω της cloud πλατφόρμας **NeonDB**⁴. Η επιλογή σχεσιακής βάσης δεδομένων εξασφαλίζει αξιοπιστία, ακεραιότητα δεδομένων και αποδοτική διαχείριση σύνθετων σχέσεων μεταξύ των οντοτήτων του συστήματος.

Η βάση δεδομένων αποθηκεύει πληροφορίες που σχετίζονται με:

- τους χρήστες και τα προφίλ τους,
- τις εκδηλώσεις,
- τις συμμετοχές,
- τα εισιτήρια,
- τις αξιολογήσεις και τα σχόλια,
- καθώς και το πολυμεσικό περιεχόμενο της πλατφόρμας.

Η χρήση του NeonDB παρέχει επιπλέον δυνατότητες cloud υποδομής, όπως αυξημένη διαθεσιμότητα, ικανότητα κλιμάκωσης και ασφαλή αποθήκευση δεδομένων.

3.2 Σχήμα Βάσης Δεδομένων

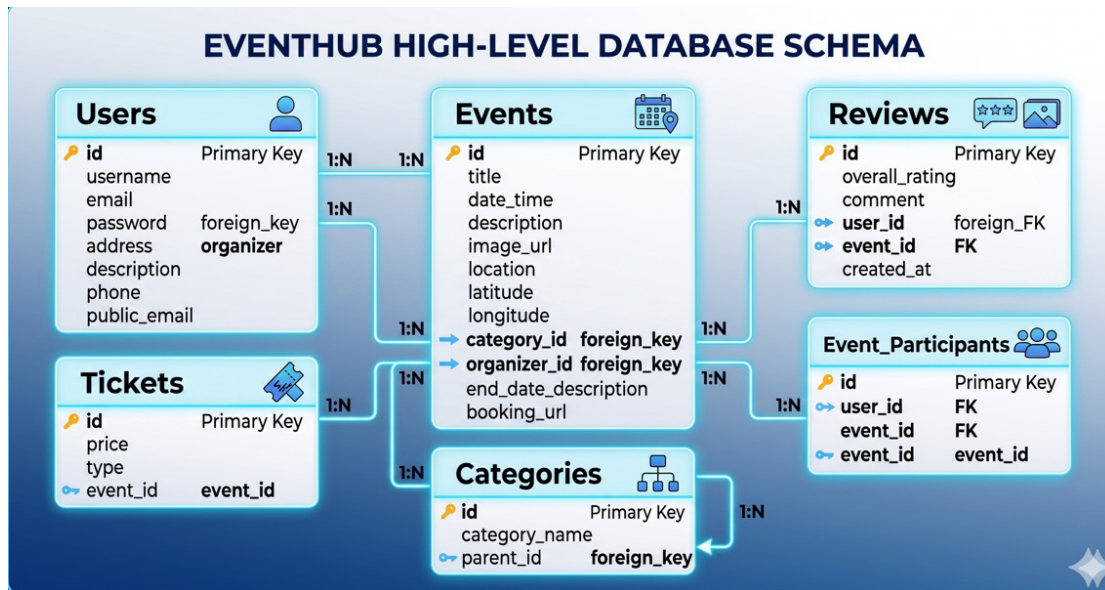
Το σχήμα της βάσης αποτελείται από τους εξής βασικούς πίνακες:

⁴ <https://neon.com/>

Πίνακας 1. Κατάλογος πινάκων του σχήματος της βάσης δεδομένων

Πίνακας	Περιγραφή
Users	Αποθηκεύει στοιχεία λογαριασμού (username, password, email) και πληροφορίες προφίλ (description, phone, address).
Events	Η κεντρική οντότητα. Περιλαμβάνει τίτλο, περιγραφή, ημερομηνίες, τοποθεσία (κείμενο) και συντεταγμένες (latitude, longitude).
Tickets	Συνδέεται με τα Events. Αποθηκεύει τύπους εισιτηρίων (π.χ. VIP) και τις αντίστοιχες τιμές (BigDecimal).
Categories	Ιεραρχικός πίνακας για την ταξινόμηση (π.χ. Μουσική → Rock).
Reviews	Αποθηκεύει σχόλια και βαθμολογίες (1-5) που συνδέονται με έναν χρήστη και μια εκδήλωση.
Media/Images	Αποθηκεύει τα URLs των φωτογραφιών που ανεβαίνουν στο Cloudinary ⁵ , συνδεδεμένα με το αντίστοιχο Event.
Participants	Πίνακας συσχέτισης (Many-to-Many) μεταξύ Users και Events για τις δηλώσεις συμμετοχής.

⁵ <https://cloudinary.com/>



Εικόνα 2. Διάγραμμα του σχήματος της βάσης δεδομένων

3.3 Σχεδιασμός χωρικής αναζήτησης

Η διαδικασία της χωρικής αναζήτησης ξεκινά από τη διεπαφή του χρήστη, όπου ο χρήστης επιλέγει την αναζήτηση εκδηλώσεων με βάση μια συγκεκριμένη τοποθεσία, για παράδειγμα μια πόλη ή μια περιοχή ενδιαφέροντος. Το αίτημα αναζήτησης αποστέλλεται στο backend της εφαρμογής, όπου ενεργοποιείται η υπηρεσία γεωεντοπισμού.

Αρχικά, η εφαρμογή λαμβάνει το ερώτημα του χρήστη και το αποστέλλει σε εξωτερική υπηρεσία γεωκωδικοποίησης, όπως το OpenStreetMap Nominatim API⁶. Η υπηρεσία αυτή μετατρέπει την τοποθεσία που έχει εισάγει ο χρήστης σε γεωγραφικές συντεταγμένες, δηλαδή σε γεωγραφικό πλάτος και μήκος. Με αυτόν τον τρόπο, η εφαρμογή μπορεί να προσδιορίσει με ακρίβεια το σημείο γύρω από το οποίο θα πραγματοποιηθεί η αναζήτηση.

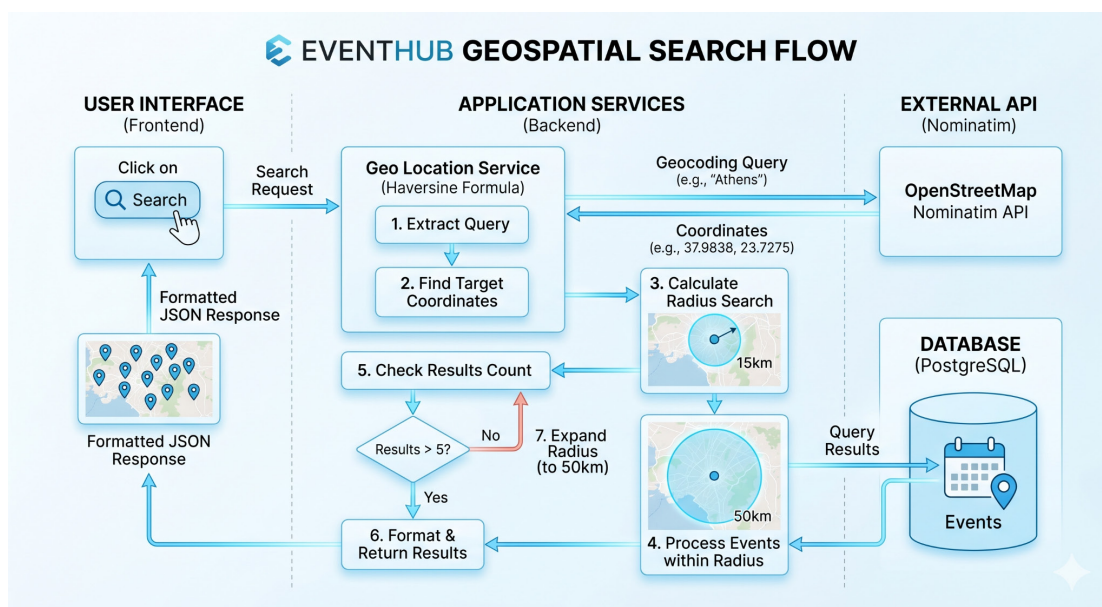
Στη συνέχεια, το σύστημα εκτελεί χωρικό ερώτημα στη βάση δεδομένων PostgreSQL, αναζητώντας εκδηλώσεις που βρίσκονται εντός μιας προκαθορισμένης ακτίνας από το σημείο αναφοράς. Αρχικά χρησιμοποιείται μια μικρότερη ακτίνα αναζήτησης, για παράδειγμα 15 χιλιόμετρα, ώστε να εμφανιστούν εκδηλώσεις που βρίσκονται κοντά

⁶ <https://nominatim.org/>

στον χρήστη. Τα αποτελέσματα που επιστρέφονται από τη βάση δεδομένων επεξεργάζονται από την εφαρμογή και ελέγχεται ο αριθμός τους.

Αν τα αποτελέσματα είναι επαρκή σε πλήθος, η εφαρμογή τα μορφοποιεί σε κατάλληλη μορφή JSON και τα επιστρέφει στο frontend, ώστε να εμφανιστούν στον χρήστη με οργανωμένο τρόπο. Αντίθετα, αν ο αριθμός των αποτελεσμάτων είναι μικρός (κάτω από ορισμένο όριο), η εφαρμογή επεκτείνει αυτόματα την ακτίνα αναζήτησης, για παράδειγμα στα 50 χιλιόμετρα, ώστε να εντοπιστούν περισσότερες σχετικές εκδηλώσεις. Με αυτόν τον τρόπο, η αναζήτηση γίνεται πιο ευέλικτη και αυξάνεται η πιθανότητα ο χρήστης να βρει εκδηλώσεις που τον ενδιαφέρουν, ακόμη και αν δεν υπάρχουν αρκετές πολύ κοντά στην αρχική τοποθεσία.

Τέλος, τα τελικά αποτελέσματα επιστρέφονται στη διεπαφή του χρήστη ως μορφοποιημένη απάντηση JSON και προβάλλονται στην εφαρμογή. Η συγκεκριμένη διαδικασία βελτιώνει την εμπειρία χρήσης, καθώς συνδυάζει την τοποθεσία του χρήστη, τη γεωκωδικοποίηση και τα χωρικά ερωτήματα στη βάση δεδομένων, προσφέροντας πιο στοχευμένη και πρακτική αναζήτηση εκδηλώσεων.



Εικόνα 3. Αλγόριθμος Χωρικής Αναζήτησης

3.4 Διεπαφές Επικοινωνίας (API Endpoints)

Η επικοινωνία μεταξύ του Frontend και του Backend της εφαρμογής πραγματοποιείται μέσω αρχιτεκτονικής **REST API**. Τα API endpoints αποτελούν τα σημεία πρόσβασης

μέσω των οποίων ανταλλάσσονται δεδομένα μεταξύ πελάτη (client) και εξυπηρετητή (server), επιτρέποντας την υλοποίηση των λειτουργιών της πλατφόρμας.

Η χρήση RESTful υπηρεσιών προσφέρει ευελιξία, επεκτασιμότητα και διαχωρισμό των επιπέδων της εφαρμογής, διευκολύνοντας τόσο τη συντήρηση όσο και τη μελλοντική επέκταση του συστήματος.

Παρακάτω παρουσιάζονται τα βασικότερα API endpoints του συστήματος, οργανωμένα ανά λειτουργική κατηγορία.

3.4.1 Εκδηλώσεις (Events)

Οι διεπαφές που σχετίζονται με τις εκδηλώσεις χρησιμοποιούνται για τη διαχείριση, αναζήτηση και συμμετοχή σε δράσεις της πλατφόρμας.

> GET /api/events

Χρησιμοποιείται για την ανάκτηση λίστας εκδηλώσεων από τη βάση δεδομένων. Η λειτουργία αυτή αποτελεί τον βασικό μηχανισμό αναζήτησης της πλατφόρμας. Οι παράμετροι που μεταβιβάζονται στη διαδικασία έχουν ως ακολούθως:

Παράμετρος	Θέση	Τύπος	Περιγραφή
title	Query string	String	Τμήμα της συμβολοσειράς του τίτλου της εκδήλωσης
location	Query string	String	Όνομα τοποθεσίας. Η αναζήτηση με βάση την τοποθεσία, όταν αυτή ορίζεται, πραγματοποιείται όπως περιγράφεται στην ενότητα 3.3.
date	Query string	LocalDate	Ημερομηνία διοργάνωσης της εκδήλωσης
categoryId	Query string	Integer	Χρησιμοποιείται για το φιλτράρισμα των εκδηλώσεων με βάση την κατηγορία στην οποία ανήκουν. Η τιμή αντιστοιχεί στο μοναδικό αναγνωριστικό της κατηγορίας.

```
▼ [{"bookingDescription": null, bookingUrl: null, category: null, dateTime: "2026-03-31T22:00:00",...}]
  ▼ 0: {bookingDescription: null, bookingUrl: null, category: null, dateTime: "2026-03-31T22:00:00",...}
    bookingDescription: null
    bookingUrl: null
    category: null
    dateTime: "2026-03-31T22:00:00"
    description: "21η ΑΓΩΝΗΣΤΙΚΗ - ΑΝΕΞΑΡΤΗΤΟ ΠΡΩΤΑΘΛΗΜΑ \n\nPREMIER NIGHTS"
    endDateTime: null
    gallery: []
    id: 6
    imageUrl: "https://res.cloudinary.com/dimcgbmqd/image/upload/v1774813806/bg09vb81obrpufojde9z.png"
    latitude: null
    location: "ΓΗΠΕΔΟ ΠΟΔΟΣΦΑΙΡΟΥ -ΣΟΥΡΜΕΝΑ"
    longitude: null
  ▼ organizer: {address: null, description: null, email: "makisparas4@gmail.com", id: 3,...}
    address: null
    description: null
    email: "makisparas4@gmail.com"
    id: 3
    password: "$2a$10$Qjg1GyWXXbQVg8cQY6JxquU4go6ioVXIDc2Qk1LozmkZlIEZcFVC6"
    phone: null
    publicEmail: null
    username: "makis"
  ▼ participants: [{address: null, description: null, email: "nikpar1996@yahoo.com", id: 4,...},...]
    ▶ 0: {address: null, description: null, email: "nikpar1996@yahoo.com", id: 4,...}
    ▼ 1: {address: null, description: null, email: "costas@uop.gr", id: 5,...}
      address: null
      description: null
      email: "costas@uop.gr"
      id: 5
      password: "$2a$10$gQ1W56ujhtFjjgdHnG720WpM5N2ayVsLMpkuZ3ZmqoeHjaCG/xXa"
      phone: null
      publicEmail: null
      username: "Costas"
    ▶ 2: {address: null, description: null, email: "lastking@gmail.com", id: 2,...}
```

Εικόνα 4. Ενδεικτική απάντηση στην κλήση API για αναζήτηση εκδηλώσεων

Κάθε αντικείμενο αντιπροσωπεύει μία εκδήλωση που πληροί τα κριτήρια αναζήτησης. Αν δεν δοθούν φίλτρα, επιστρέφεται η συνολική λίστα των εκδηλώσεων που υπάρχουν στο σύστημα.

Κάθε εκδήλωση περιλαμβάνει βασικές πληροφορίες παρουσίασης, στοιχεία τοποθεσίας, πληροφορίες διοργανωτή, στοιχεία εισιτηρίων, κατηγορία, εικόνες και πληροφορίες συμμετοχών.

➤ POST /api/events

Το συγκεκριμένο API χρησιμοποιείται για τη δημιουργία νέας εκδήλωσης στην πλατφόρμα EventHub. Η λειτουργία αυτή απευθύνεται κυρίως στους διοργανωτές, οι οποίοι έχουν τη δυνατότητα να καταχωρίζουν νέες εκδηλώσεις.

Η κλήση δέχεται δεδομένα σε μορφή **multipart/form-data**, καθώς αποστέλλονται ταυτόχρονα δύο διαφορετικά είδη πληροφορίας: τα στοιχεία της εκδήλωσης σε μορφή JSON και ένα αρχείο εικόνας. Το έγγραφο JSON μεταβιβάζεται ως συμβολοσειρά, ενώ το αρχείο αποθηκεύεται μέσω της υπηρεσίας διαχείρισης πολυμέσων.

Μετά την αποθήκευση της εικόνας, το όνομα του αρχείου συνδέεται με την εκδήλωση και αποθηκεύεται στο πεδίο **imageUrl**. Στη συνέχεια, η εκδήλωση αποθηκεύεται στη βάση δεδομένων και επιστρέφεται ως αποτέλεσμα της κλήσης. Οι παράμετροι που μεταβιβάζονται στη διαδικασία έχουν ως ακολούθως:

Παράμετρος	Θέση	Τύπος	Περιγραφή
event	Payload multipart/form-data	String	Περιλαμβάνει τα στοιχεία της εκδήλωσης σε μορφή JSON
file	Payload multipart/form-data	MultipartFile	Περιλαμβάνει το αρχείο εικόνας που θα χρησιμοποιηθεί ως βασική εικόνα προβολής της εκδήλωσης. Το αρχείο αποθηκεύεται στο σύστημα και το όνομά του συνδέεται με την εκδήλωση.

```
{
  "title": "Technology Conference 2026",
  "description": "Συνέδριο τεχνολογίας με παρουσιάσεις και εργαστήρια.",
  "startDate": "2026-06-10T10:00:00",
  "endDate": "2026-06-10T18:00:00",
  "locationName": "Athens Conference Center",
  "latitude": 37.9838,
  "longitude": 23.7275,
  "category": {
    "id": 3
  },
  "tickets": [
    {
      "type": "Regular",
      "description": "Κανονικό εισιτήριο",
      "price": 20.00
    }
  ],
  "ticketUrl": "https://example.com/tickets",
  "organizer": {
    "id": 5
  }
}
```

Εικόνα 5. Παράδειγμα αιτήματος δημιουργίας εκδήλωσης

➤ **PUT /api/events/{id}**

Χρησιμοποιείται για την ενημέρωση υπαρχουσών εκδηλώσεων. Ο διοργανωτής μπορεί να τροποποιήσει στοιχεία όπως:

- περιγραφή
- κατηγορία
- γεωγραφική τοποθεσία μέσω χάρτη
- διαθέσιμα εισιτήρια και πληροφορίες συμμετοχής

Οι παράμετροι που μεταβιβάζονται στη διαδικασία έχουν ως ακολούθως:

Παράμετρος	Θέση	Τύπος	Περιγραφή
id	Path variable	Long integer	Το μοναδικό αναγνωριστικό της εκδήλωσης που πρόκειται να ενημερωθεί.
eventDetails	Payload / Request body	Event	Περιλαμβάνει τα νέα ή τροποποιημένα στοιχεία της εκδήλωσης σε μορφή JSON. Τα δεδομένα αυτά χρησιμοποιούνται για την ενημέρωση της υπάρχουσας εγγραφής στη βάση δεδομένων.
authentication	Security context / Headers	Authentication	Αντιπροσωπεύει τα στοιχεία ταυτοποίησης του συνδεδεμένου χρήστη.

```
{
  "title": "Technology Conference 2026 - Updated",
  "description": "Ενημερωμένη περιγραφή του συνεδρίου τεχνολογίας.",
  "startDate": "2026-06-10T11:00:00",
  "endDate": "2026-06-10T19:00:00",
  "locationName": "Athens Conference Center",
  "latitude": 37.9838,
  "longitude": 23.7275,
  "category": {
    "id": 3
  },
  "tickets": [
    {
      "type": "Regular",
      "description": "Κανονικό εισιτήριο",
      "price": 25.00
    }
  ],
  "ticketUrl": "https://example.com/updated-tickets"
}
```

Εικόνα 6. Παράδειγμα αιτήματος ενημέρωσης εκδήλωσης

➤ **POST /api/events/{eventId}/join**

Το συγκεκριμένο API χρησιμοποιείται για τη δήλωση συμμετοχής ενός εγγεγραμμένου χρήστη σε μια εκδήλωση. Μέσω αυτής της λειτουργίας, το μέλος της πλατφόρμας μπορεί να δηλώσει συμμετοχή ή ενδιαφέρον για μια συγκεκριμένη εκδήλωση.

Η κλήση απαιτεί το αναγνωριστικό της εκδήλωσης στο URL και χρησιμοποιεί τα στοιχεία ταυτοποίησης του συνδεδεμένου χρήστη μέσω του αντικειμένου Authentication. Με αυτόν τον τρόπο, το σύστημα γνωρίζει ποιος χρήστης πραγματοποιεί την ενέργεια και μπορεί να τον συσχετίσει με την αντίστοιχη εκδήλωση.

Οι παράμετροι που μεταβιβάζονται στη διαδικασία έχουν ως ακολούθως:

Παράμετρος	Θέση	Τύπος	Περιγραφή
eventId	Path variable	Long	Το μοναδικό αναγνωριστικό της εκδήλωσης στην οποία ο χρήστης θέλει να δηλώσει συμμετοχή.
authentication	Security context / Headers	Authentication	Αντιπροσωπεύει τα στοιχεία ταυτοποίησης του συνδεδεμένου χρήστη.

➤ **DELETE /api/events/{eventId}**

Το συγκεκριμένο API χρησιμοποιείται για τη διαγραφή μιας υπάρχουσας εκδήλωσης από την πλατφόρμα EventHub. Η λειτουργία αυτή αφορά κυρίως τους διοργανωτές και επιτρέπει την αφαίρεση εκδηλώσεων που έχουν δημιουργήσει, όταν αυτές δεν είναι πλέον ενεργές, έχουν ακυρωθεί ή δεν χρειάζεται να εμφανίζονται στην πλατφόρμα.

Οι παράμετροι που μεταβιβάζονται στη διαδικασία έχουν ως ακολούθως:

Παράμετρος	Θέση	Τύπος	Περιγραφή
eventId	Path variable	Long	Το μοναδικό αναγνωριστικό της εκδήλωσης που πρόκειται να διαγραφεί.
authentication	Security context / Headers	Authentication	Αντιπροσωπεύει τα στοιχεία ταυτοποίησης του συνδεδεμένου χρήστη.
currentUsername	Security context	String	Το username του συνδεδεμένου χρήστη

3.4.2 Χρήστες & Προφίλ (Users & Profiles)

Τα συγκεκριμένα endpoints αφορούν τη διαχείριση λογαριασμών χρηστών και δημόσιων προφίλ διοργανωτών.

➤ **POST /api/auth/register**

Χρησιμοποιείται για τη δημιουργία νέου λογαριασμού χρήστη στην πλατφόρμα. Κατά τη διαδικασία εγγραφής πραγματοποιείται έλεγχος

εγκυρότητας των στοιχείων και ασφαλής αποθήκευση των δεδομένων πιστοποίησης.

Οι παράμετροι που μεταβιβάζονται στη διαδικασία έχουν ως ακολούθως:

Παράμετρος	Θέση	Τύπος	Περιγραφή
id	Path variable	Long	Το μοναδικό αναγνωριστικό της εκδήλωσης στην οποία ο χρήστης θέλει να δηλώσει συμμετοχή.
username	Query string	String	Το username του χρήστη
password	Query string	String	Το password του χρήστη

```
{
  "address": null,
  "description": null,
  "email": "alex@gmail.com",
  "id": 6,
  "password": "$2a$10$dZTd0dumkPh3SM.HwecNV.pPJMKsc0xUS3diE77e8TQgMELloVXm",
  "phone": null,
  "publicEmail": null,
  "username": "Alex"
}
```

Εικόνα 7. Ενδεικτική απάντηση σε αίτημα δημιουργίας λογαριασμού

➤ **GET /api/users/{id}/public**

Παρέχει πρόσβαση στα δημόσια στοιχεία ενός διοργανωτή, όπως:

- όνομα και στοιχεία επικοινωνίας
- περιγραφή ή βιογραφικό
- ιστορικό εκδηλώσεων
- μέσος όρος αξιολογήσεων από τους χρήστες

Η λειτουργία αυτή συμβάλλει στη διαφάνεια και αξιοπιστία της πλατφόρμας. Δεν παρέχεται πρόσβαση σε ευαίσθητα στοιχεία όπως το email (έχει την δυνατότητα να προσθέσει διαφορετικό email) ή το password που έχει βάλει ο χρήστης κατά την εγγραφή του στην πλατφόρμα.

```
{
  "address": "Ερμού 15",
  "phone": "210225566",
  "averageRating": 0.0,
  "description": "<p>Μικρή&nbsp;ομάδα&nbsp;που&nbsp;διοργανώνει&nbsp;events.</p>",
  "id": 1,
  "publicEmail": "marios@gmail.com",
  "events": [...],
  "username": "Marios13"
}
```

Εικόνα 8. Ενδεικτική απάντηση προφίλ ενός χρήστη

➤ **PUT /api/users/{id}**

Επιτρέπει την ενημέρωση στοιχείων προφίλ χρήστη, όπως:

- βιογραφικό
- email επικοινωνίας
- τηλέφωνο
- διεύθυνση ή άλλα προσωπικά στοιχεία προβολής

3.4.3 Κριτικές & Υλικό (Reviews & Media)

Οι διεπαφές αυτής της κατηγορίας υποστηρίζουν τη διαδραστικότητα και τη συμμετοχή των χρηστών μέσω αξιολογήσεων και πολυμεσικού περιεχομένου.

➤ **POST /api/reviews**

Χρησιμοποιείται για την υποβολή κριτικής και βαθμολογίας σε μια εκδήλωση. Πριν την αποθήκευση της αξιολόγησης, το σύστημα ελέγχει:

- αν η εκδήλωση έχει ολοκληρωθεί
- καθώς και αν ο χρήστης είχε δηλώσει συμμετοχή σε αυτή.

Αν ο χρήστης είχε δηλώσει συμμετοχή πριν λήξει η εκδήλωση τότε στη κριτική που θα κάνει θα υπάρχει το λεκτικό «**Επιβεβαιωμένη Συμμετοχή**». Ο μηχανισμός αυτός διασφαλίζει την εγκυρότητα και αξιοπιστία των αξιολογήσεων.

➤ **POST /api/events/images/upload**

Επιτρέπει τη μεταφόρτωση φωτογραφιών στη συλλογή (gallery) μιας εκδήλωσης. Η λειτουργία χρησιμοποιείται κυρίως μετά την ολοκλήρωση μιας

εκδήλωσης, ώστε οι συμμετέχοντες να μπορούν να μοιράζονται φωτογραφικό υλικό από την εμπειρία τους.

Η διαχείριση των πολυμέσων πραγματοποιείται με τρόπο που εξασφαλίζει ασφαλή αποθήκευση και σωστή συσχέτιση των αρχείων με την αντίστοιχη εκδήλωση.

➤ **GET /api/categories**

Αυτό το API endpoint χρησιμοποιείται για την ανάκτηση μιας λίστας με όλες τις διαθέσιμες κατηγορίες εκδηλώσεων από τη βάση δεδομένων. Η λειτουργία αυτή είναι θεμελιώδης, καθώς παρέχει στο Frontend τις επιλογές που είναι απαραίτητες για τη λειτουργία του φιλτραρίσματος των εκδηλώσεων. Οι κατηγορίες οργανώνονται σε ιεραρχικό πίνακα για την καλύτερη οργάνωση των τύπων εκδηλώσεων (π.χ. Cultural and Entertainment Events → Music Events).

```
{
  "name": "Cultural and Entertainment Events",
  "parent": null,
  "id": 1
},
{
  "name": "Music Events",
  "parent": {
    "name": "Cultural and Entertainment Events",
    "parent": null,
    "id": 1
  },
  "id": 2
}
```

Εικόνα 9. Ενδεικτική απάντηση σε κλήση του API ανάκτησης των κατηγοριών

4 ΕΡΓΑΛΕΙΑ ΚΑΙ ΤΕΧΝΟΛΟΓΙΕΣ ΑΝΑΠΤΥΞΗΣ

Η ανάπτυξη της πλατφόρμας **EventHub** πραγματοποιήθηκε με τη χρήση σύγχρονων τεχνολογιών και εργαλείων λογισμικού, τα οποία επιλέχθηκαν με βασικά κριτήρια την αξιοπιστία, την ασφάλεια, την επεκτασιμότητα και την ευκολία ανάπτυξης. Ο συνδυασμός τεχνολογιών Frontend, Backend και cloud υπηρεσιών επέτρεψε τη δημιουργία μιας ολοκληρωμένης και αποδοτικής διαδικτυακής πλατφόρμας.

Παράλληλα, αξιοποιήθηκαν εργαλεία διαχείρισης εξαρτήσεων, version control και deployment, ώστε να εξασφαλιστεί η ομαλή ανάπτυξη, η εύκολη συντήρηση και η σταθερή λειτουργία της εφαρμογής σε περιβάλλον παραγωγής.

4.1 Backend (Διακομιστής)

Για την ανάπτυξη του Backend χρησιμοποιήθηκε η γλώσσα προγραμματισμού **Java** σε συνδυασμό με το framework **Spring Boot**. Η επιλογή αυτή βασίστηκε στην αξιοπιστία, τη σταθερότητα και τις εκτεταμένες δυνατότητες που προσφέρει το οικοσύστημα του Spring Framework για την ανάπτυξη διαδικτυακών εφαρμογών και RESTful υπηρεσιών.

Το Spring Boot παρέχει:

- εύκολη δημιουργία και διαχείριση REST APIs,
- ενσωματωμένους μηχανισμούς ασφαλείας μέσω του Spring Security⁷,
- διαχείριση αυθεντικοποίησης χρηστών με JWT Authentication,
- καθώς και αποτελεσματική διασύνδεση με σχεσιακές βάσεις δεδομένων μέσω του Spring Data JPA⁸.

Η χρήση της Java και του Spring Boot συμβάλλει στη δημιουργία ενός ασφαλούς, επεκτάσιμου και υψηλής απόδοσης συστήματος Backend.

⁷ <https://spring.io/projects/spring-security>

⁸ <https://spring.io/projects/spring-data-jpa>

Για τη διαχείριση των εξαρτήσεων (dependencies) και την αυτοματοποίηση της διαδικασίας build χρησιμοποιήθηκε το εργαλείο **Maven**. Μέσω του Maven επιτυγχάνεται:

- εύκολη διαχείριση βιβλιοθηκών και frameworks,
- αυτοματοποίηση διαδικασιών μεταγλώττισης και packaging,
- καθώς και καλύτερη οργάνωση της δομής του έργου.

4.2 Frontend (Περιβάλλον Χρήστη)

Το περιβάλλον διεπαφής χρήστη αναπτύχθηκε με τη βιβλιοθήκη **React.js**, η οποία χρησιμοποιείται ευρέως για την ανάπτυξη δυναμικών και διαδραστικών Single Page Applications (SPA).

Η React.js επιτρέπει:

- τη δημιουργία επαναχρησιμοποιήσιμων components,
- τη δυναμική ενημέρωση του περιβάλλοντος χρήστη,
- καθώς και τη βελτίωση της απόδοσης μέσω αποδοτικής διαχείρισης του Virtual DOM (Chéc & Nowak, 2018).

Η βασιζόμενη σε συνιστώσες (component-based) αρχιτεκτονική της React συμβάλλει στη μείωση του χρόνου ανάπτυξης και στη βελτίωση της συντηρησιμότητας του κώδικα.

Για την επικοινωνία του Frontend με το Backend API χρησιμοποιήθηκε η βιβλιοθήκη **Axios**⁹. Η βιβλιοθήκη αυτή επιτρέπει την εκτέλεση αιτημάτων HTTP (GET, POST, PUT, DELETE) με εύκολο και αποδοτικό τρόπο, διευκολύνοντας την ανταλλαγή δεδομένων μεταξύ πελάτη και εξυπηρετητή.

Η ενσωμάτωση χαρτών πραγματοποιήθηκε με τη χρήση της βιβλιοθήκης **React Leaflet**¹⁰, η οποία βασίζεται στην υπηρεσία OpenStreetMap. Μέσω αυτής παρέχεται:

- προβολή διαδραστικών χαρτών,
- επιλογή τοποθεσίας εκδηλώσεων,

⁹ <https://axios.rest/>

¹⁰ <https://react-leaflet.js.org/>

- καθώς και λειτουργίες γεωεντοπισμού και αναζήτησης βάσει τοποθεσίας.

Για τη δημιουργία και μορφοποίηση μορφοποιημένου κειμένου (rich text) στις περιγραφές εκδηλώσεων χρησιμοποιήθηκε ο editor **React Quill**¹¹. Το εργαλείο αυτό παρέχει δυνατότητες μορφοποίησης περιεχομένου, όπως:

- έντονη και πλάγια γραφή,
- λίστες,
- σύνδεσμοι,
- καθώς και πολυεπίπεδη μορφοποίηση κειμένου.

4.3 Βάση Δεδομένων και Cloud Υπηρεσίες

Για την αποθήκευση και διαχείριση των δεδομένων της εφαρμογής χρησιμοποιήθηκε η σχεσιακή βάση δεδομένων **PostgreSQL**. Η βάση δεδομένων PostgreSQL επιλέχθηκε λόγω της αξιοπιστίας, της υποστήριξης σύνθετων σχεσιακών δομών, καθώς και της υψηλής απόδοσης σε διαδικτυακές εφαρμογές.

Η φιλοξενία της βάσης δεδομένων πραγματοποιήθηκε μέσω της υπηρεσίας νέφους (cloud) **NeonDB**, η οποία παρέχει απομακρυσμένη διαχείριση, δυνατότητα κλιμάκωσης, καθώς και αυξημένη διαθεσιμότητα και ασφάλεια δεδομένων.

Για την αποθήκευση και διαχείριση των εικόνων της πλατφόρμας χρησιμοποιήθηκε η υπηρεσία **Cloudinary**. Η υπηρεσία αυτή επιτρέπει την ασφαλή αποθήκευση πολυμεσικού υλικού, την αυτόματη βελτιστοποίηση εικόνων, καθώς και ταχεία φόρτωση και διανομή περιεχομένου μέσω cloud υποδομής

4.4 Εργαλεία Ανάπτυξης και Docker

Για τη δημιουργία ομοιόμορφου περιβάλλοντος ανάπτυξης και εκτέλεσης χρησιμοποιήθηκε η τεχνολογία **Docker** (Oeggli & Kofler, 2023). Με τη χρήση του Docker οι εφαρμογές «πακετάρονται» με τη μορφή *εικόνων* (images), όπου κάθε εικόνα περιλαμβάνει το σύνολο του απαραίτητου κώδικα και των εξαρτήσεων που απαιτούνται για την εκτέλεση της εφαρμογής. Ακολούθως, οι εικόνες μπορούν να εκτελεστούν σε οποιοδήποτε μηχάνημα υποστηρίζει την τεχνολογία Docker,

¹¹ <https://www.npmjs.com/package/react-quill>

σχηματίζοντας *περιέκτες* (containers). Μέσω της χρήσης containers επιτυγχάνεται η απομόνωση των εξαρτήσεων, η σταθερότητα του περιβάλλοντος ανάπτυξης, καθώς και δυνατότητα εύκολης μεταφοράς και εκτέλεσης της εφαρμογής σε διαφορετικά συστήματα. Η χρήση Docker συμβάλλει σημαντικά στη μείωση προβλημάτων συμβατότητας μεταξύ διαφορετικών υπολογιστικών περιβαλλόντων.

Για τον έλεγχο εκδόσεων (Version Control) και τη διαχείριση του πηγαίου κώδικα χρησιμοποιήθηκαν τα εργαλεία **Git**¹² και **GitHub**¹³. Τα εργαλεία αυτά παρέχουν:

- δυνατότητα παρακολούθησης αλλαγών στον κώδικα,
- ασφαλή αποθήκευση εκδόσεων,
- καθώς και δυνατότητα συνεργασίας μεταξύ προγραμματιστών.

Για τη δημοσίευση (deployment) της εφαρμογής χρησιμοποιήθηκαν οι πλατφόρμες **Vercel**¹⁴ και **Render**¹⁵. Ειδικότερα:

- Η πλατφόρμα Vercel χρησιμοποιήθηκε για τη φιλοξενία του Frontend της εφαρμογής.
- Η πλατφόρμα Render χρησιμοποιήθηκε για τη φιλοξενία του Backend και των υπηρεσιών REST API.

Η αξιοποίηση υπηρεσιών cloud deployment επιτρέπει την εύκολη ανάπτυξη, συντήρηση και κλιμάκωση της εφαρμογής στο διαδίκτυο.

¹² <https://git-scm.com/tools>

¹³ <https://github.com/>

¹⁴ <https://vercel.com/>

¹⁵ <https://render.com/>

5 ΟΔΗΓΙΕΣ ΕΓΚΑΤΑΣΤΑΣΗΣ

Η σωστή εγκατάσταση και ρύθμιση της εφαρμογής **EventHub** αποτελεί απαραίτητη προϋπόθεση για την ομαλή λειτουργία της σε τοπικό περιβάλλον ανάπτυξης. Για τον σκοπό αυτό απαιτείται η εγκατάσταση συγκεκριμένων τεχνολογιών και εργαλείων, καθώς και η κατάλληλη παραμετροποίηση των συνιστωσών Backend και Frontend.

Η διαδικασία εγκατάστασης περιλαμβάνει την εγκατάσταση και ρύθμιση της βάσης δεδομένων, του ορισμού και ρύθμισης των μεταβλητών περιβάλλοντος και των υπηρεσιών νέφους που χρησιμοποιεί η εφαρμογή, καθώς και την εκτέλεση των απαραίτητων εντολών για την εκκίνηση όλων των επιμέρους υποσυστημάτων.

5.1 Προαπαιτούμενα

Πριν από την εγκατάσταση της εφαρμογής, απαιτείται η ύπαρξη των παρακάτω εργαλείων και τεχνολογιών στο σύστημα του χρήστη:

- **Java JDK¹⁶ 17** ή νεότερη έκδοση, για την εκτέλεση του Backend που βασίζεται στο Spring Boot.
- **Node.js έκδοση 18+**, για την ανάπτυξη και εκτέλεση του Frontend περιβάλλοντος React.js.
- **Docker Desktop** (προαιρετικά), σε περίπτωση που η εφαρμογή εκτελεστεί μέσω containers για καλύτερη διαχείριση του περιβάλλοντος ανάπτυξης.
- Πρόσβαση σε βάση δεδομένων PostgreSQL ή σε cloud υπηρεσία NeonDB.
- Λογαριασμός και API credentials για την υπηρεσία Cloudinary, εφόσον χρησιμοποιείται αποθήκευση εικόνων στο cloud.

Η σωστή εγκατάσταση των παραπάνω εργαλείων αποτελεί απαραίτητη προϋπόθεση για την ορθή λειτουργία της εφαρμογής. Σημειώνεται ότι για την εκτέλεση των συνιστωσών δεν είναι απαραίτητο να υπάρχουν οι εξαρτήσεις ανάπτυξης (Java JDK και Node.js), καθώς μέσω της διαδικασίας παραγωγής εικόνων Docker, οι απαραίτητες εξαρτήσεις ενσωματώνονται στις εικόνες Docker.

¹⁶ <https://www.oracle.com/java/technologies/downloads/>

5.2 Ρύθμιση Backend

Για την εκτέλεση του Backend της εφαρμογής ακολουθούνται τα παρακάτω βήματα:

- *Βήμα 1: Μετάβαση στον κατάλογο του Backend*

Ο χρήστης πρέπει αρχικά να μεταβεί στον φάκελο που περιέχει τον πηγαίο κώδικα του Backend:

```
cd backend
```

- *Βήμα 2: Ρύθμιση αρχείου παραμέτρων*

Στη συνέχεια απαιτείται η δημιουργία αρχείου ρυθμίσεων **application.properties** ή η χρήση μεταβλητών περιβάλλοντος (environment variables), όπου θα δηλωθούν:

- τα στοιχεία σύνδεσης της βάσης δεδομένων PostgreSQL,
- τα credentials της υπηρεσίας Cloudinary,
- καθώς και τυχόν μυστικά κλειδιά (JWT secrets) που χρησιμοποιούνται για την αυθεντικοποίηση χρηστών.

Παράδειγμα βασικών ρυθμίσεων:

```
spring.datasource.url=jdbc:postgresql://localhost:5432/eventhub
```

```
spring.datasource.username=postgres
```

```
spring.datasource.password=your_password
```

```
cloudinary.cloud-name=your_cloud_name
```

```
cloudinary.api-key=your_api_key
```

```
cloudinary.api-secret=your_api_secret
```

- *Βήμα 3: Εκτέλεση της εφαρμογής*

Η εκκίνηση του Backend πραγματοποιείται μέσω του Maven Wrapper με την παρακάτω εντολή:

```
./mvnw spring-boot:run
```

Μετά την επιτυχή εκτέλεση, το Backend API θα είναι διαθέσιμο συνήθως στη διεύθυνση:

```
http://localhost:8080
```

5.3 Ρύθμιση Frontend

Για την εγκατάσταση και εκτέλεση του Frontend ακολουθούνται τα παρακάτω βήματα:

- *Βήμα 1: Μετάβαση στον κατάλογο του Frontend*

Ο χρήστης μεταβαίνει στον φάκελο του Frontend:

```
cd frontend
```

- *Βήμα 2: Εγκατάσταση εξαρτήσεων*

Οι απαραίτητες βιβλιοθήκες και εξαρτήσεις της εφαρμογής εγκαθίστανται μέσω της εντολής

```
npm install
```

Η διαδικασία αυτή εγκαθιστά όλα τα packages που ορίζονται στο αρχείο **package.json**.

- *Βήμα 3: Δημιουργία αρχείου περιβάλλοντος (.env)*

Απαιτείται η δημιουργία αρχείου **.env**, στο οποίο ορίζεται η διεύθυνση του Backend API

```
VITE_API_URL=http://localhost:8080
```

Η μεταβλητή αυτή χρησιμοποιείται από το Frontend για την αποστολή αιτημάτων HTTP προς το Backend σύστημα.

- *Βήμα 4: Εκκίνηση της εφαρμογής*

Η εφαρμογή εκκινείται με την ακόλουθη εντολή:

```
npm run dev
```

Μετά την επιτυχή εκκίνηση, το Frontend θα είναι διαθέσιμο συνήθως στη διεύθυνση:

`http://localhost:5173`

Με την ολοκλήρωση των παραπάνω βημάτων, η εφαρμογή EventHub είναι έτοιμη προς χρήση σε τοπικό περιβάλλον ανάπτυξης, επιτρέποντας την πλήρη λειτουργία και δοκιμή όλων των δυνατοτήτων της πλατφόρμας.

6 ΟΔΗΓΙΕΣ ΧΡΗΣΗΣ

Η πλατφόρμα **EventHub** παρέχει ένα σύνολο λειτουργιών που επιτρέπουν την αλληλεπίδραση των χρηστών με τις διαθέσιμες εκδηλώσεις και τις υπηρεσίες της εφαρμογής. Οι λειτουργικές ροές του συστήματος περιγράφουν τα βασικά βήματα που ακολουθούν οι χρήστες κατά τη χρήση της πλατφόρμας, τόσο από την πλευρά των συμμετεχόντων όσο και από την πλευρά των διοργανωτών.

Μέσα από τις λειτουργίες αυτές υποστηρίζονται διαδικασίες όπως η αναζήτηση εκδηλώσεων, η δήλωση συμμετοχής, η αξιολόγηση δράσεων, καθώς και η δημιουργία και διαχείριση εκδηλώσεων από τους διοργανωτές.

6.1 Λειτουργίες για το Μέλος (Attendee)

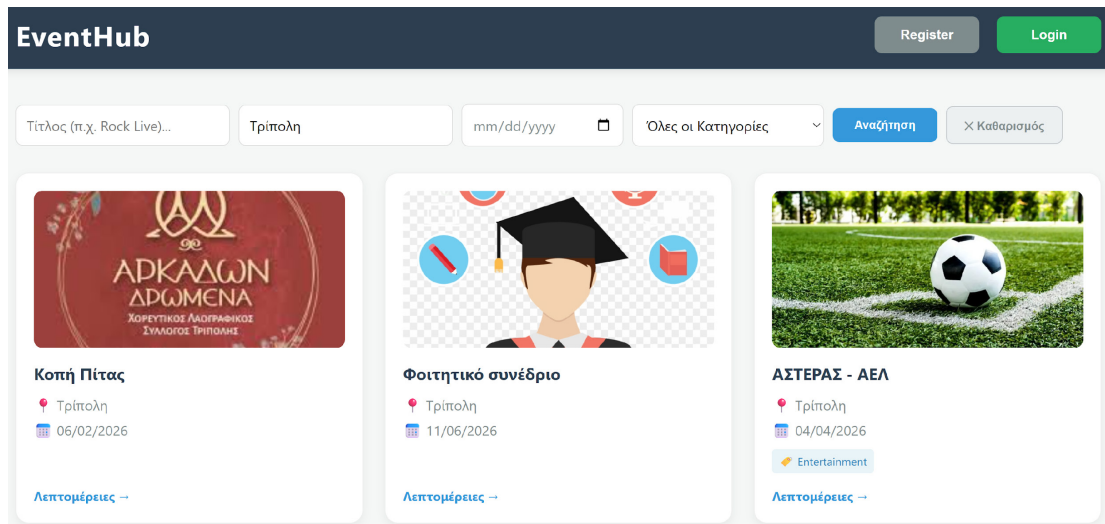
6.1.1 Αναζήτηση Εκδήλωσης

Η διαδικασία αναζήτησης εκδηλώσεων αποτελεί μία από τις βασικότερες λειτουργίες της πλατφόρμας. Ο χρήστης μπορεί να εισάγει λέξεις-κλειδιά, κατηγορία ή τοποθεσία στη γραμμή αναζήτησης της εφαρμογής.

Το σύστημα αποστέλλει το αίτημα στο Backend API, όπου πραγματοποιείται επεξεργασία των δεδομένων αναζήτησης. Για τις λειτουργίες γεωγραφικής αναζήτησης χρησιμοποιείται ο μαθηματικός αλγόριθμος Haversine (Winarno et al., 2017), ο οποίος υπολογίζει τη χιλιομετρική απόσταση μεταξύ του χρήστη και των διαθέσιμων εκδηλώσεων. Με βάση τα αποτελέσματα του υπολογισμού, επιστρέφονται οι κοντινότερες και πιο σχετικές εκδηλώσεις.

Τα αποτελέσματα εμφανίζονται δυναμικά στο περιβάλλον του χρήστη, συνοδευόμενα από βασικές πληροφορίες, όπως τίτλος, ημερομηνία, τοποθεσία και εικόνα εκδήλωσης.

Στην Εικόνα 4 εμφανίζεται ένα παράδειγμα χρήσης της λειτουργίας αναζήτησης εκδηλώσεων.



Εικόνα 10. Αναζήτηση εκδηλώσεων

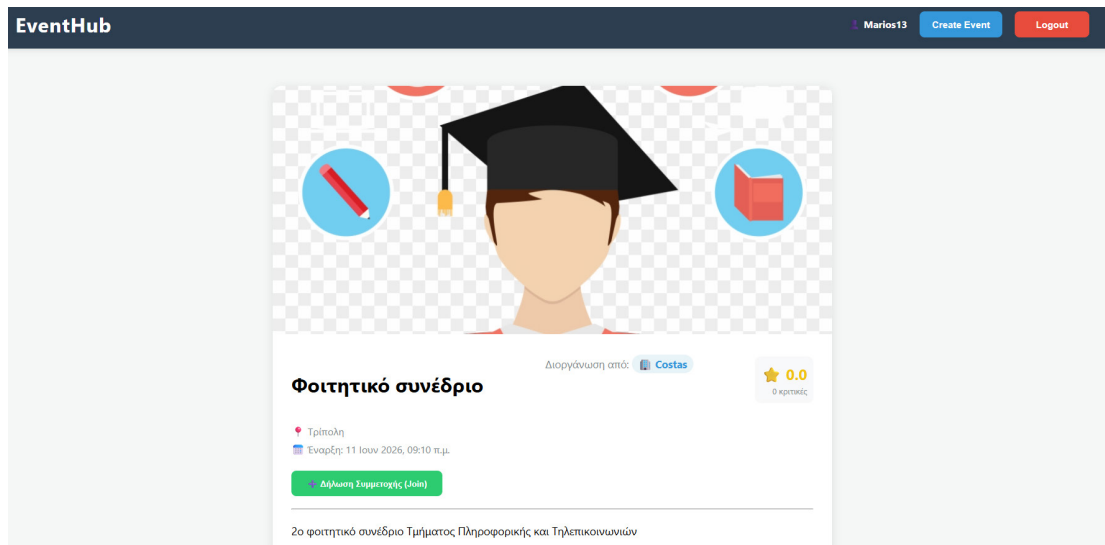
6.1.2 Συμμετοχή

Αφού ο χρήστης επιλέξει μια εκδήλωση και μεταβεί στη σελίδα λεπτομερειών της, μπορεί να δηλώσει συμμετοχή μέσω του κουμπιού «Join».

Με την ενέργεια αυτή:

- αποστέλλεται αίτημα στο Backend,
- καταγράφεται η συμμετοχή στη βάση δεδομένων,
- και ενημερώνεται αυτόματα το περιβάλλον χρήστη χωρίς ανανέωση της σελίδας

Η λειτουργία αυτή επιτρέπει στους χρήστες να οργανώνουν τις συμμετοχές τους, ενώ παράλληλα δίνει στους διοργανωτές τη δυνατότητα παρακολούθησης του ενδιαφέροντος για κάθε εκδήλωση.



Εικόνα 11. Δήλωση συμμετοχής

6.1.3 Υποβολή Αξιολόγησης και Κριτικής

Μετά την ολοκλήρωση μιας εκδήλωσης, οι συμμετέχοντες έχουν τη δυνατότητα να αξιολογήσουν την εμπειρία τους μέσω ειδικής φόρμας κριτικής.

Ο χρήστης μπορεί να επιλέξει βαθμολογία, να προσθέσει γραπτό σχόλιο, καθώς και να καταχωρήσει συνολική αξιολόγηση της εκδήλωσης.

Πριν από την αποθήκευση της κριτικής, το σύστημα ελέγχει αν:

- η εκδήλωση έχει ολοκληρωθεί,
- και αν ο χρήστης είχε δηλώσει συμμετοχή.

Ο μηχανισμός αυτός διασφαλίζει την εγκυρότητα και αξιοπιστία των αξιολογήσεων που εμφανίζονται στην πλατφόρμα.

The screenshot shows a sports event page for "PIASTON VOULA FC- 3KTR". At the top, there is a blue header with a soccer ball. Below the header, the event title "PIASTON VOULA FC- 3KTR" is displayed in bold. To the right of the title, it says "Διοργάνωση από: makis" and a rating of "4.5" with "2 κριτικές". Below the title, there is a location pin icon and the text "ΓΗΠΕΔΟ ΠΟΔΟΣΦΑΙΡΟΥ - ΣΟΥΡΜΕΝΑ", followed by a calendar icon and "Έναρξη: 31 Μαρ 2026, 10:00 μ.μ.". A grey button with the text "Η εκδήλωση έχει λήξει" is visible. Below this, the text "21η ΑΓΩΝΗΣΤΙΚΗ - ΑΝΕΞΑΡΤΗΤΟ ΠΡΩΤΑΘΛΗΜΑ PREMIER NIGHTS" is shown. A section titled "Κριτικές Χρηστών" contains two reviews. The first review is by "Costas" with a "4.0" rating and the text "Ήθελα να πάω αλλά τελικά το έχασα...". The second review is by "Marios13" with a "5.0" rating and the text "Ωραίο ματς με πολλά γκολ!".

Εικόνα 12. Υποβολή αξιολόγησης

6.2 Λειτουργίες για τον Διοργανωτή (Organizer)

6.2.1 Δημιουργία Εκδήλωσης

Ο διοργανωτής έχει τη δυνατότητα δημιουργίας νέας εκδήλωσης μέσω ειδικής φόρμας καταχώρησης. Κατά τη διαδικασία δημιουργίας ο χρήστης (α) συμπληρώνει τίτλο και περιγραφή εκδήλωσης, (β) επιλέγει κατηγορία και ημερομηνίες διεξαγωγής, (γ) ορίζει τοποθεσία μέσω διαδραστικού χάρτη βασισμένου στη βιβλιοθήκη Leaflet και (δ) δημιουργεί διαφορετικούς τύπους εισιτηρίων μέσω δυναμικού συστήματος επιλογών.

Η χρήση διαδραστικού χάρτη επιτρέπει την ακριβή επιλογή γεωγραφικού στίγματος, βελτιώνοντας τη λειτουργία γεωγραφικής αναζήτησης της πλατφόρμας.

Δημιουργία Νέου Event

Περιγραφή

Normal • B I U

■ Εισιτήρια & Κρατήσεις

Επιλέξτε Τύπο...
▼

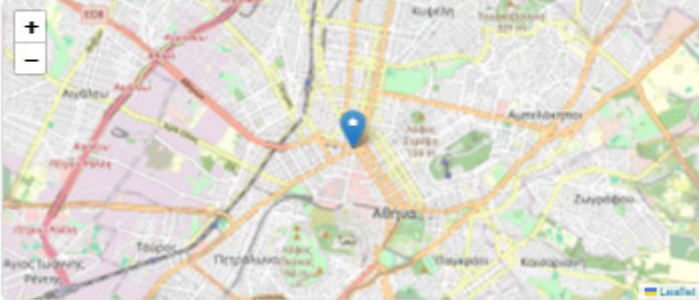
Τιμή (€) - Αφήστε κενό για δωρεάν

+ Προσθήκη Τύπου Εισιτηρίου

URL Κράτησης/Αγοράς (π.χ. <https://viva.gr/events/>...)

Οδηγίες κράτησης (π.χ. 'Η αγορά γίνεται στην είσοδο' ή 'Τηλέφωνο κρατήσεων: 210...')

Στίγμα στον Χάρτη:



Ονομασία Τοποθεσίας (π.χ. Πλατεία Συντάγματος)

Έναρξη
mm/dd/yyyy --:-- --

Λήξη
mm/dd/yyyy --:-- --

Επιλέξτε Κατηγορία ▼

Choose File
No file chosen

Δημοσίευση Εκδήλωσης

Εικόνα 13. Δημιουργία εκδήλωσης

6.2.2 Διαχείριση Δημόσιου Προφίλ

Μέσω της λειτουργίας «**Edit Profile**», ο διοργανωτής μπορεί να επεξεργάζεται τα στοιχεία του δημόσιου επαγγελματικού του προφίλ.

Η λειτουργία αυτή επιτρέπει την ενημέρωση του βιογραφικού, την προσθήκη/τροποποίηση στοιχείων επικοινωνίας, την ενημέρωση των επαγγελματικών πληροφοριών, καθώς και τη διαχείριση της συνολικής ψηφιακής παρουσίας του διοργανωτή στην πλατφόρμα.

Οι αλλαγές που πραγματοποιούνται εμφανίζονται άμεσα στη δημόσια σελίδα του προφίλ του.

Επεξεργασία Προφίλ Διοργανωτή

Δημόσιο Email Επικοινωνίας

events.house@gmail.com

Αυτό το email θα είναι ορατό σε όλους. (Δεν αλλάζει το email σύνδεσής σας).

Τηλέφωνο Επικοινωνίας

210225566

Ταχυδρομική Διεύθυνση / Έδρα

Ερμού 15

Περιγραφή / Βιογραφικό

Normal ↕ **B** *I* U

Μικρή ομάδα που διοργανώνει events.

Αποθήκευση Αλλαγών

Ακύρωση

Εικόνα 14. Διαχείριση δημόσιου προφίλ

6.2.3 Επεξεργασία Εκδήλωσης

Ο διοργανωτής έχει τη δυνατότητα να επεξεργάζεται οποιαδήποτε εκδήλωση έχει δημιουργήσει. Συγκεκριμένα, μπορεί να τροποποιήσει:

- τις πληροφορίες της εκδήλωσης,
- την περιγραφή,
- τις ημερομηνίες,
- την τοποθεσία στον χάρτη,
- καθώς και τους διαθέσιμους τύπους εισιτηρίων

Η δυνατότητα αυτή παρέχει ευελιξία στη διαχείριση των εκδηλώσεων και επιτρέπει την άμεση ενημέρωση του περιεχομένου σε περίπτωση αλλαγών ή διορθώσεων.

Επεξεργασία Εκδήλωσης

Τίτλος

ΑΣΤΕΡΑΣ - ΑΕΛ

Περιγραφή

Normal **B** **I** **U**   

Ο αγώνας μεταξύ Αστέρη και Λάρισα

■ Εισιτήρια & Κρατήσεις

Κανονικό



30



Μειωμένο / Ανέργων



15



Φοιτηκό



15

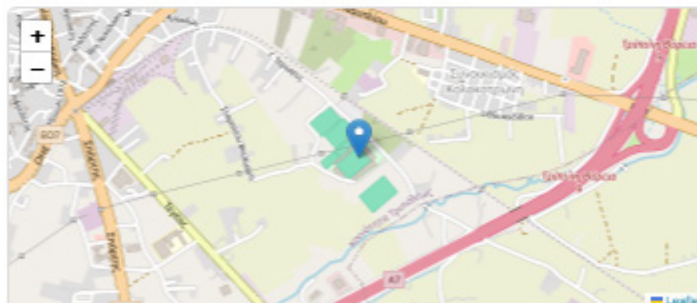


+ Προσθήκη Τύπου Εισιτηρίου

URL Κράτησης/Αγοράς

θα λειτουργήσουν και τα εκδοτήρια την ημέρα του αγώνα από τις 18:00

Αλλαγή Στίγματος στον Χάρτη:



Τοποθεσία (Κείμενο)

Τρίπολη

Κατηγορία

Entertainment



Ενημέρωση Εκδήλωσης

Ακύρωση

Εικόνα 15. Επεξεργασία εκδήλωσης

7 ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ

7.1 Συμπεράσματα

Η ανάπτυξη της εφαρμογής **EventHub** ανέδειξε τη σημασία της σύγχρονης αρχιτεκτονικής διαχωρισμού Frontend και Backend (Decoupled Architecture) στη δημιουργία αποδοτικών και επεκτάσιμων διαδικτυακών εφαρμογών. Ο διαχωρισμός αυτός επέτρεψε την ανεξάρτητη ανάπτυξη και διαχείριση των επιμέρους τμημάτων του συστήματος, βελτιώνοντας τη συντηρησιμότητα, την επεκτασιμότητα και τη συνολική απόδοση της εφαρμογής.

Η χρήση της γλώσσας προγραμματισμού Java σε συνδυασμό με το πλαίσιο ανάπτυξης (framework) Spring Boot παρέχει ένα σταθερό και ασφαλές περιβάλλον για την υλοποίηση του Backend συστήματος. Μέσω του Spring Boot κατέστη δυνατή η ανάπτυξη RESTful υπηρεσιών υψηλής αξιοπιστίας, η εφαρμογή μηχανισμών ασφαλείας μέσω JWT Authentication, καθώς και η αποδοτική διαχείριση των επιχειρησιακών λειτουργιών της εφαρμογής.

Παράλληλα, η αξιοποίηση της βιβλιοθήκης React.js συνέβαλε ουσιαστικά στη δημιουργία μιας δυναμικής και διαδραστικής διεπαφής χρήστη. Η component-based αρχιτεκτονική της React διευκόλυνε την ανάπτυξη επαναχρησιμοποιήσιμων στοιχείων διεπαφής, ενώ επέτρεψε την ενσωμάτωση προηγμένων λειτουργιών, όπως διαδραστικοί χάρτες, δυναμικές φόρμες και rich text editors.

Ένα από τα σημαντικότερα τεχνικά ζητήματα που αντιμετωπίστηκαν κατά την ανάπτυξη της εφαρμογής ήταν η διαχείριση και αξιοποίηση γεωγραφικών δεδομένων. Η πρόκληση αυτή επιλύθηκε μέσω του συνδυασμού του OpenStreetMap API με τον μαθηματικό τύπο Haversine στο επίπεδο του Backend, επιτρέποντας την υλοποίηση αποδοτικής γεωγραφικής αναζήτησης εκδηλώσεων βάσει χιλιομετρικής απόστασης.

Συνολικά, η υλοποίηση του EventHub απέδειξε ότι η αξιοποίηση σύγχρονων τεχνολογιών διαδικτύου και cloud υπηρεσιών μπορεί να οδηγήσει στη δημιουργία μιας ασφαλούς, λειτουργικής και εύχρηστης πλατφόρμας διαχείρισης εκδηλώσεων, ικανής να καλύψει τόσο τις ανάγκες των χρηστών όσο και των διοργανωτών.

7.2 Μελλοντικές Επεκτάσεις

Παρόλο που η εφαρμογή καλύπτει ένα μεγάλο εύρος λειτουργικών απαιτήσεων, υπάρχουν σημαντικές δυνατότητες περαιτέρω επέκτασης και βελτίωσης του συστήματος.

7.2.1 Σύστημα Ειδοποιήσεων (Notifications)

Μία σημαντική μελλοντική επέκταση αφορά την υλοποίηση μηχανισμού ειδοποιήσεων για τους χρήστες της πλατφόρμας. Η λειτουργία αυτή μπορεί να βασιστεί σε τεχνολογίες όπως WebSockets (Sharma & Agarwal, 2023) ή Email Services, επιτρέποντας ενημερώσεις για αλλαγές σε εκδηλώσεις, υπενθυμίσεις συμμετοχής, καθώς και ειδοποιήσεις σχετικά με νέες δραστηριότητες ή ανακοινώσεις

7.2.2 Ανάπτυξη Native Mobile Εφαρμογής

Μια ακόμη σημαντική επέκταση είναι η ανάπτυξη εφαρμογής για κινητές συσκευές με χρήση React Native. Η mobile εφαρμογή θα μπορεί να αξιοποιεί το ίδιο Spring Boot API που χρησιμοποιείται από το Web Frontend, επιτρέποντας κοινή επιχειρησιακή λογική, ευκολότερη συντήρηση, καθώς και καλύτερη εμπειρία χρήσης σε κινητές συσκευές

7.2.3 Ενσωμάτωση Ηλεκτρονικών Πληρωμών

Η προσθήκη συστήματος ηλεκτρονικών πληρωμών αποτελεί μία ιδιαίτερα σημαντική λειτουργική αναβάθμιση. Μέσω της ενσωμάτωσης υπηρεσιών όπως το Stripe API, οι χρήστες θα μπορούν να πραγματοποιούν αγορές εισιτηρίων απευθείας μέσα από την πλατφόρμα, χωρίς ανάγκη μετάβασης σε εξωτερικούς συνδέσμους.

Η λειτουργία αυτή θα βελτιώσει σημαντικά την εμπειρία χρήσης, την αυτοματοποίηση των κρατήσεων, καθώς και τη συνολική λειτουργικότητα της πλατφόρμας

7.2.4 Προηγμένα Στατιστικά και Analytics

Μία επιπλέον προτεινόμενη επέκταση είναι η δημιουργία προηγμένου dashboard αναλυτικών στοιχείων (**analytics dashboard**) για τους διοργανωτές. Το dashboard θα μπορεί να παρέχει (ενδεικτικά) γραφήματα σχετικά με τη δημοτικότητα εκδηλώσεων,

στατιστικά συμμετοχών, ανάλυση αξιολογήσεων, καθώς και δεδομένα επισκεψιμότητας και αλληλεπίδρασης χρηστών.

Η αξιοποίηση τέτοιων αναλυτικών εργαλείων θα βοηθήσει τους διοργανωτές να λαμβάνουν καλύτερες αποφάσεις σχετικά με τη διοργάνωση και προώθηση των εκδηλώσεών τους.

7.2.5 Ολοκλήρωση με τρίτα συστήματα

Η ολοκλήρωση με τρίτα συστήματα με δυνατότητα ανταλλαγής δεδομένων με αυτά αναμένεται να αυξήσει την ορατότητα της εφαρμογής και τον βαθμό αξιοποίησης των καταχωρήσεων. Για παράδειγμα, η δυνατότητα να κοινοποιούνται οι εκδηλώσεις ως Facebook events και να αντλούνται από το Facebook αντίστοιχα προθέσεις συμμετοχής αυξάνει τον βαθμό αξιοποίησης των καταχωρήσεων. Παράλληλα, αν στην κοινοποίηση στο Facebook events παρέχονται σύνδεσμοι προς την πλατφόρμα (π.χ. για εισιτήρια ή λεπτομέρειες της εκδήλωσης), θα αυξηθεί η ορατότητα της πλατφόρμας και ανάλογα θα προωθηθεί η αύξηση της βάσης χρηστών της.

7.2.6 Διαχείριση Κατηγοριών από Διαχειριστή

Οι κατηγορίες των εκδηλώσεων είναι **προκαθορισμένες** και δεν τροποποιούνται δυναμικά μέσα από την εφαρμογή. Αυτό σημαίνει ότι οι διαθέσιμες κατηγορίες ορίζονται εξ αρχής στο σύστημα και χρησιμοποιούνται από τους διοργανωτές κατά τη δημιουργία ή επεξεργασία μιας εκδήλωσης.

Ωστόσο, σε μελλοντική επέκταση θα μπορούσε να προστεθεί ειδική λειτουργία διαχείρισης κατηγοριών από διαχειριστή. Μέσω αυτής της λειτουργίας, ένας διαχειριστής θα μπορούσε να δημιουργεί νέες κατηγορίες, να επεξεργάζεται υπάρχουσες, να απενεργοποιεί κατηγορίες που δεν χρησιμοποιούνται πλέον και να οργανώνει καλύτερα τη θεματολογία των εκδηλώσεων. Η προσθήκη αυτής της δυνατότητας θα έκανε την πλατφόρμα πιο προσαρμόσιμη σε διαφορετικές ανάγκες και σε νέους τύπους εκδηλώσεων που μπορεί να προκύψουν μελλοντικά.

7.2.7 Διαχείριση Εισιτηρίων μέσω ξεχωριστού API

Οι πληροφορίες που σχετίζονται με τα εισιτήρια μιας εκδήλωσης καταχωρίζονται στο πλαίσιο της δημιουργίας ή επεξεργασίας της εκδήλωσης, χωρίς να υπάρχει ανεξάρτητο

σύνολο διεπαφών για λειτουργίες όπως δημιουργία, ενημέρωση ή διαγραφή εισιτηρίων.

Ένα τέτοιο API θα μπορούσε να υποστηρίζει λειτουργίες όπως δημιουργία διαφορετικών τύπων εισιτηρίων, ενημέρωση τιμών και διαθεσιμότητας, διαγραφή εισιτηρίων, παρακολούθηση κρατήσεων και πιθανή διασύνδεση με σύστημα ηλεκτρονικών πληρωμών. Με αυτόν τον τρόπο, η πλατφόρμα θα μπορούσε να εξελιχθεί από ένα σύστημα προβολής και οργάνωσης εκδηλώσεων σε ένα πιο ολοκληρωμένο εργαλείο διαχείρισης συμμετοχών και εισιτηρίων.

ΑΝΑΦΟΡΕΣ

Chęć, D., & Nowak, Z. (2018). The Performance Analysis of Web Applications Based on Virtual DOM and Reactive User Interfaces. In *Advances in Intelligent Systems and Computing* (pp. 119–134). Springer International Publishing. https://doi.org/10.1007/978-3-319-99617-2_8

Ferrari, L., & Pirozzi, E. (2023). *Learn PostgreSQL* (2nd edn). Birmingham, England: Packt Publishing.

Fink, G., & Flatow, I. (2014). Introducing Single Page Applications. In *Pro Single Page Application Development* (pp. 3–13). Apress. https://doi.org/10.1007/978-1-4302-6674-7_1

Gupta, S., & Gupta, B. B. (2015). Cross-Site Scripting (XSS) attacks and defense mechanisms: classification and state-of-the-art. *International Journal of System Assurance Engineering and Management*, 8(S1), 512–530. <https://doi.org/10.1007/s13198-015-0376-0>

Haklay, M., & Weber, P. (2008). OpenStreetMap: User-Generated Street Maps. *IEEE Pervasive Computing*, 7(4), 12–18. <https://doi.org/10.1109/mprv.2008.80>

Huang, H., Gartner, G., Krisp, J. M., Raubal, M., & Van de Weghe, N. (2018). Location based services: ongoing evolution and research agenda. *Journal of Location Based Services*, 12(2), 63–93. <https://doi.org/10.1080/17489725.2018.1508763>

Jakob Michal, Pěchouček Michal, Miles Simon & Luck Michael (2008). Case studies for contract-based systems. In *Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems: industrial track (AAMAS '08)*. International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, 55–62.

Masse, M. (2011). *REST API design rulebook: Designing consistent RESTful web service interfaces*. O'Reilly Media.

Oeggel, B., & Kofler, M. (2023). *Docker*. USA: Rheinwerk Publishing.

Peyrott, S. E. (2024). *The JWT handbook* (Version 0.14.2). <https://www.jre-training.earl-family.net/NextGen/Documents/jwt-handbook.pdf>

Sharma, N., & Agarwal, R. (2023). HTTP, WebSocket, and SignalR: A Comparison of Real-Time Online Communication Protocols. In *Lecture Notes in Computer Science* (pp. 128–135). Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-44084-7_13

VMware, Inc. (2026). *Spring Boot Reference Documentation* (Version 3.x). Spring Boot Documentation

Winarno, E., Hadikurniawati, W., & Rosso, R. N. (2017). Location based service for presence system using haversine method. In *2017 International Conference on Innovative and Creative Information Technology (ICITech)* (pp. 1–4). IEEE. 2017 International Conference on Innovative and Creative Information Technology (ICITech). <https://doi.org/10.1109/innocit.2017.8319153>