

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ
Τμήμα Πληροφορικής



Εργασία Μαθήματος
Εκπαιδευτικό Λογισμικό

Αριθμός εργασίας – Τίτλος εργασίας	Εφαρμογή Επαγγελματικού Προσανατολισμού
Μέλη Ομάδας	Μιχάλης Στυλιανίδης - Π19165, Λευτέρης Κοντούρης - Π19077, Κωνσταντίνος Καλογερόπουλος - Π19057, Σοφία Μαρκοπούλου - Π19097
Ημερομηνία παράδοσης	Εαρινή Εξεταστική, 2023



Εκφώνηση εργασίας

Θέμα εργασίας: «Εκπαιδευτικό Λογισμικό
για επαγγελματικό προσανατολισμό φοιτητών και αποφοίτων
Τμημάτων Πληροφορικής»

Ζητείται να γίνει ένα αλληλεπιδραστικό λογισμικό εκπαίδευσης για το παρακάτω πεδίο:

«Επαγγελματικός προσανατολισμός φοιτητών και αποφοίτων τμημάτων Πληροφορικής».

Η Πληροφορική είναι μια ταχύτατα αναπτυσσόμενη επιστήμη, η οποία είναι απαραίτητη για την στελέχωση κάθε είδους επιχειρήσεων και οργανισμών. Είναι πάρα πολλές οι κατευθύνσεις, οι οποίες μπορεί να ενδιαφέρουν τους νέους επιστήμονες και καθώς η Πληροφορική εξελίσσεται, οι κατευθύνσεις πολλαπλασιάζονται. Επομένως έχει νόημα να υπάρχει εκπαιδευτικό λογισμικό που να διδάσκει τον επαγγελματικό προσανατολισμό της Πληροφορικής σε ειδικότητες όπως Προγραμματιστής Λογισμικού (Software programmer), Μηχανικός Λογισμικού (Software Engineer), Ειδικός Μηχανικής Μάθησης και Τεχνητής Νοημοσύνης (AI/Machine learning Specialist), Σχεδιαστής Εμπειρίας Χρηστών (UX designer), Διαχειριστής έργων πληροφορικής (Informatics Project Manager), Προγραμματιστής Web (Web developer), Ειδικός Ασφάλειας και Προστασίας Δεδομένων (Cybersecurity and data protection specialist), Cloud architect κ.λπ. Το εκπαιδευτικό υλικό θα αποτελείται από πληροφορίες για μαθήματα που διδάσκονται σε Τμήματα Πληροφορικής, καθώς και πληροφορίες για τα μαθησιακά αποτελέσματα των μαθημάτων και για τις ικανότητες που αναπτύσσονται στους φοιτητές που τα παρακολουθούν αλλά και για πιθανούς τρόπους που αυτά συνδυάζονται με επαγγελματικές κατευθύνσεις. Ως μια πρώτη προσέγγιση θα μπορούσε να χρησιμοποιηθεί, ως εκπαιδευτικό υλικό, το πρόγραμμα σπουδών (ή μέρος αυτού) του Τμήματος Πληροφορικής και οι προπτυχιακές ή μεταπτυχιακές κατευθύνσεις του ίδιου Τμήματος.

Η εργασία αυτή θα περιλαμβάνει τρόπους παρουσίασης του διδακτικού υλικού με στόχο να γίνει το θέμα κατανοητό. Στη συνέχεια, αφού οι εκπαιδευόμενοι θα έχουν διαβάσει το διδακτικό υλικό, το σύστημα θα παρέχει δυο ειδών τεστ διαφορετικού σκοπού:

Α. Ερωτήσεις και τεστ αξιολόγησης των εκπαιδευόμενων σχετικά με το πόσο καλά έχουν κατανοήσει τα αντικείμενα και τις κατευθύνσεις της Πληροφορικής, π.χ. Ποια είναι τα μαθησιακά αποτελέσματα του μαθήματος Τεχνολογίας Λογισμικού; Ποιες είναι οι



προαπαιτούμενες γνώσεις για τα Σήματα και Συστήματα; Ποια μαθήματα χρειάζονται για την καριέρα του UX designer; Ποιες ικανότητες απαιτούνται για τον Ειδικό Μηχανικής Μάθησης;

Β. Ερωτηματολόγια στο πλαίσιο ενός μικρού υποσυστήματος συστάσεων προκειμένου να προτείνουν στους εκπαιδευόμενους κατευθύνσεις επαγγελματών συναφών με την Πληροφορική και κατάλληλες μεταπτυχιακές σπουδές ή επαγγελματικούς χώρους. Για να προταθεί κάποια επιστημονική περιοχή ή ειδικότητα θα πρέπει να ληφθούν υπόψη οι προτιμήσεις των εκπαιδευόμενων, βαθμοί σε μαθήματα (που θα εισάγονται στο σύστημα) καθώς και οι ικανότητες και τα μαθησιακά αποτελέσματα των μαθημάτων αυτών.

Ο κύριος σκοπός της εργασίας είναι ο καλός σχεδιασμός και υλοποίηση του εκπαιδευτικού λογισμικού (διδασκαλία – αξιολόγηση του μαθητή- σύστημα προτάσεων) και όχι η εισαγωγή μεγάλου μέρους υλικού.

Συγκεκριμένα ζητούνται τα παρακάτω:

ΒΑΣΙΚΕΣ ΛΕΙΤΟΥΡΓΙΕΣ

1. ΠΑΡΟΥΣΙΑΣΗ ΔΙΔΑΣΚΑΛΙΑΣ

2. ΕΡΩΤΗΣΕΙΣ / ΤΕΣΤ ΑΥΤΟΑΞΙΟΛΟΓΗΣΗΣ

2.1 Κατασκευή των τεστ.

- Θα πρέπει να κατασκευάζονται τεστ για την αυτοαξιολόγηση των μαθητών, για κάθε ενότητα διδασκαλίας.
- Στα επαναληπτικά τεστ, θα πρέπει να παρουσιάζονται ερωτήσεις απ'όλες τις ενότητες.
- Η μορφή των ασκήσεων μπορεί να είναι πολλαπλών επιλογών ή άλλης μορφής ανάλογα με τη δική σας ανάλυση απαιτήσεων και σχεδιασμό.

2.2 Αποθήκευση στατιστικών στοιχείων προόδου του μαθητή.

Θα πρέπει να υπάρχει μια βάση δεδομένων όπου να αποθηκεύονται στοιχεία για κάθε μαθητή σχετικά με την πρόοδό του. Τα στοιχεία θα βασίζονται στην απόδοση των μαθητών στα τεστ αυτοαξιολόγησης αλλά και στα στατιστικά επισκεψιμότητας της παρουσίασης του θέματος.

3. ΠΑΡΟΧΗ ΣΥΣΤΑΣΕΩΝ

Το μικρό υποσύστημα συστάσεων θα λειτουργεί με την συλλογή πληροφοριών για τους εκπαιδευόμενους και τις προτιμήσεις τους, όπως οι βαθμοί τους σε συναφή μαθήματα και οι ικανότητες που έχουν αποκτήσει σε αυτά τα μαθήματα.



Στη συνέχεια, το υποσύστημα θα προσπαθεί να βρει τις κατευθύνσεις επαγγελματικής ανάπτυξης και των μεταπτυχιακών σπουδών που είναι πιο συναφείς με τις προτιμήσεις και τις ικανότητες του κάθε εκπαιδευόμενου. Τα αποτελέσματα της ανάλυσης θα παρουσιάζονται στους εκπαιδευόμενους σε μια λίστα προτεινόμενων κατευθύνσεων και μεταπτυχιακών σπουδών που ταιριάζουν καλύτερα στις προτιμήσεις και τις ικανότητες τους.

Η εφαρμογή θα πρέπει να συνοδεύεται από τα εξής εγχειρίδια:

1. Εγχειρίδιο χρήστη (user manual)
2. On-line help
3. Εγχειρίδιο Ανάλυσης και Σχεδιασμού της εφαρμογής (Τεχνικό Εγχειρίδιο).



ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

1	Εισαγωγή	6
2	Περιγραφή της υλοποίησης	6
2.1	Σχεδίαση της εφαρμογής	6
2.2	Δημιουργία και συλλογή των δεδομένων	6
2.3	Δομή της βάσης δεδομένων	7
2.4	Υλοποίηση του API service	9
2.5	Υλοποίηση της Web εφαρμογής	11
3	Επίδειξη της λύσης	13
3.1	Οδηγίες Εγκατάστασης	13
3.2	Παρουσίασης της εφαρμογής	14
4	Βιβλιογραφικές Πηγές	17



1 Εισαγωγή

Ο κύριος στόχος της εργασίας είναι η δημιουργία μιας εφαρμογής που θα δίνει προτάσεις σε μαθητές, φοιτητές και αποφοίτους για την επαγγελματική καριέρα που θα ακολουθήσουν. Οι προτάσεις αυτές θα βασίζονται σε τεστ αυτοαξιολόγησης με ερωτήσεις γενικών γνώσεων και ερωτήσεις προτιμήσεων.

2 Περιγραφή της υλοποίησης

2.1 Σχεδίαση της εφαρμογής

Η σχεδίαση της εφαρμογής έγινε με τη χρήση του εργαλείου σχεδίασης Figma. Το Figma μας επέτρεψε να κατανοήσουμε καλύτερα τις απαιτήσεις του χρήστη, να σχεδιάσουμε τις διεπαφές χρήστη με ακρίβεια και να ορίσουμε την αλληλεπίδραση μεταξύ των διάφορων στοιχείων της εφαρμογής.

2.2 Δημιουργία και συλλογή των δεδομένων

Η διαδικασία δημιουργίας και συλλογής των δεδομένων ήταν ουσιαστικό μέρος της υλοποίησης. Τα δεδομένα που απαιτούνταν για την λειτουργία της εφαρμογής, όπως οι πληροφορίες για τα μαθήματα, τις ερωτήσεις των τεστ, τις βαθμολογίες και τα στατιστικά στοιχεία, συλλέχθηκαν από την ιστοσελίδα του πανεπιστημίου. Έπειτα, η μορφοποίηση των δεδομένων έγινε σε JSON, προκειμένου να μπορούν να ανταλλάσσονται εύκολα μεταξύ του front-end και του back-end, και να υποστηρίζουν την ανάκτηση, ενημέρωση και επεξεργασία των δεδομένων μέσα στην εφαρμογή.

Εικόνα 1: Δείγμα από τα JSON αρχεία δεδομένων.

2.3 Δομή της βάσης δεδομένων

Ακολουθεί η περιγραφή της δομής της Postgres (βάση δεδομένων)

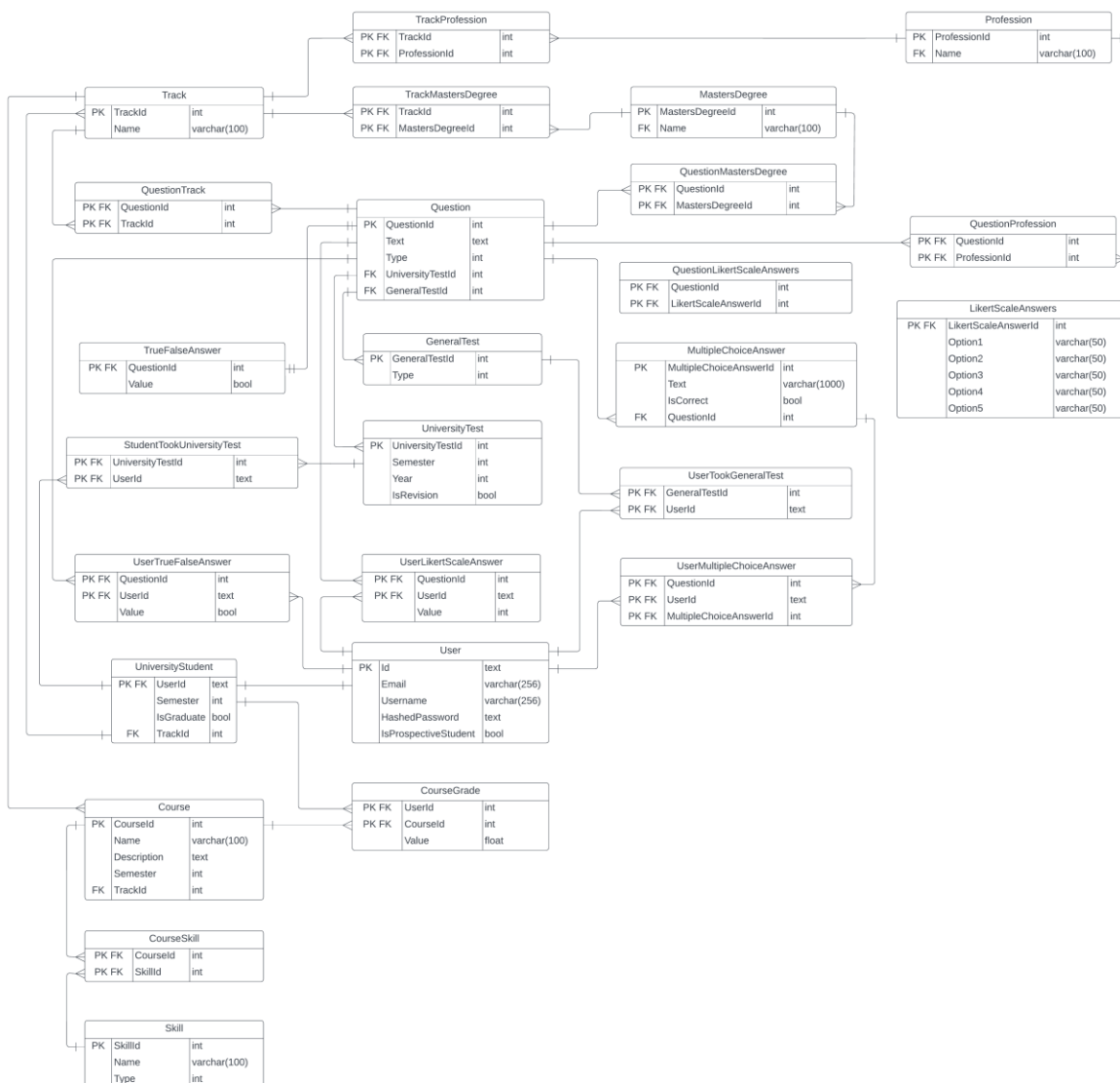
- **Specialties**
 - MastersDegrees: Αναπαριστά τα μεταπτυχιακά προγράμματα τα οποία μπορεί να ακολουθήσει ένας φοιτητής.
 - Professions: Περιγράφει τις επαγγελματικές διαδρομές που μπορεί να ακολουθήσει ένας φοιτητής.
 - Tracks: Περιγράφει τους διάφορους τομείς ειδίκευσης.
- **Tests**
 - GeneralTests: Τα γενικά τεστ αυτοαξιολόγησης.
 - UniversityTests: Τεστ αυτοαξιολόγησης σχεδιασμένα για φοιτητές πανεπιστημίου.
 - Questions: Οι ερωτήσεις των τεστ.
 - MultipleChoiceAnswers: Οι απαντήσεις πολλαπλής επιλογής στις ερωτήσεις.
 - TrueFalseAnswers: Οι απαντήσεις σωστό/λάθος στις ερωτήσεις.
- **Specialty relations**
 - TrackMastersDegrees: Αναπαριστά τη σχέση μεταξύ των τομέων και των μεταπτυχιακών προγραμμάτων.
 - TrackProfessions: Αναπαριστά τη σχέση μεταξύ των τομέων και των επαγγελματικών διαδρομών.
- **Tests - Specialty relations**



- QuestionMastersDegrees: Αναπαριστά τη σχέση μεταξύ των ερωτήσεων και των μεταπτυχιακών προγραμμάτων.
- QuestionProfessions: Αναπαριστά τη σχέση μεταξύ των ερωτήσεων και των επαγγελματικών διαδρομών.
- QuestionTracks: Αναπαριστά τη σχέση μεταξύ των ερωτήσεων και των τομέων.
- **Answers - Users relations**
 - UserLikertScaleAnswers: Αναπαριστά τις απαντήσεις των χρηστών στις ερωτήσεις κλίμακας Likert.
 - UserMultipleChoiceAnswers: Αναπαριστά τις απαντήσεις πολλαπλής επιλογής των χρηστών.
 - UserTrueFalseAnswers: Αναπαριστά τις απαντήσεις σωστό/λάθος των χρηστών.
- **Tests - Users relations**
 - UsersTookGeneralTests: Παρακολουθεί ποιοι χρήστες έχουν λάβει τα γενικά τεστ.
 - StudentsTookUniversityTests: Παρακολουθεί ποιοι φοιτητές έχουν λάβει τα τεστ του πανεπιστημίου.
- **Users**
 - UniversityStudents: Αναπαριστά τους φοιτητές του πανεπιστημίου.
- **Courses**
 - Course: Περιγράφει τα μαθήματα που παρέχονται.
 - Skill: Οι δεξιότητες που μπορούν να αποκτηθούν μέσω των μαθημάτων.
 - CourseSkill: Αναπαριστά τη σχέση μεταξύ των μαθημάτων και των δεξιοτήτων.
 - UserCourseGrade: Καταγράφει τις βαθμολογίες των χρηστών στα μαθήματα.



Διάγραμμα Βάσης



2.4 Υλοποίηση του API service

Το API service που έχει υλοποιηθεί είναι γραμμένο σε .NET 7.

Παρακάτω ακολουθεί η περιγραφή κάθε τελικού σημείου του API για κάθε Controller.

Courses

1. **Endpoint: Get Courses** URL: **/api/Courses** Μέθοδος: GET

Αυτό το endpoint επιστρέφει τα μαθήματα του δοσμένου εξαμήνου. Για εξάμηνα 1-4, η τροχιά πρέπει να είναι null, ενώ για εξάμηνα 5+ είναι υποχρεωτική.

Grades



2. **Endpoint: Get Grades** URL: `/api/Grades` Μέθοδος: GET

Αυτό το endpoint επιστρέφει τους βαθμούς όλων των μαθημάτων του συνδεδεμένου φοιτητή.

ProspectiveStudentTests

3. **Endpoint: Get General Test** URL: `/api/ProspectiveStudentTests/generalTestId`
Μέθοδος: GET

Αυτό το endpoint επιστρέφει τις ερωτήσεις και τις απαντήσεις του γενικού τεστ του οποίου το ID παρέχεται.

4. **Endpoint: Get Completed Test**

URL: `/api/ProspectiveStudentTests/Completed/{generalTestId}` Μέθοδος: GET

Αυτό το endpoint επιστρέφει ένα boolean που δείχνει εάν ο συνδεδεμένος χρήστης έχει ολοκληρώσει το παρεχόμενο γενικό τεστ.

5. **Endpoint: Post Prospective Student Test** URL: `/api/ProspectiveStudentTests` Μέθοδος: POST

Αυτό το endpoint υποβάλλει τις απαντήσεις του τεστ του προοπτικού φοιτητή.

Statistics

6. **Endpoint: Post Teaching Access Stats** URL: `/api/Statistics/TeachingAccessStatistics`
Μέθοδος: POST

Αυτό το endpoint αυξάνει τον αριθμό των στατιστικών πρόσβασης στο εξάμηνο ή στο έτος αναθεώρησης για τον συνδεδεμένο χρήστη κατά ένα.

7. **Endpoint: Get Teaching Access Stats** URL: `/api/Statistics/TeachingAccessStatistics`
Μέθοδος: GET

Αυτό το endpoint επιστρέφει όλα τα στατιστικά στοιχεία πρόσβασης στη διδασκαλία, τόσο για τα εξάμηνα όσο και για τα έτη αναθεώρησης.

StudentTests

8. **Endpoint: Get Student Test** URL: `/api/StudentTests` Μέθοδος: GET

Αυτό το endpoint επιστρέφει τις ερωτήσεις και τις απαντήσεις του τεστ είτε του δοσμένου εξαμήνου ή της αναθεώρησης ενός έτους.

9. **Endpoint: Post Student Test** URL: `/api/StudentTests` Μέθοδος: POST

Αυτό το endpoint υποβάλλει τις απαντήσεις του τεστ του φοιτητή.

10. **Endpoint: Get Completed University Test** URL:
`/api/StudentTests/Completed/{universityTestId}` Μέθοδος: GET

Αυτό το endpoint επιστρέφει ένα boolean που δείχνει εάν ο συνδεδεμένος χρήστης έχει ολοκληρώσει το παρεχόμενο τεστ πανεπιστημίου.

Users

11. **Endpoint: Get User** URL: `/api/Users/{userId}` Μέθοδος: GET

Αυτό το endpoint επιστρέφει τον χρήστη με το συγκεκριμένο id. Απαιτείται εξουσιοδότηση.

12. **Endpoint: Register User** URL: `/api/Users/Register` Μέθοδος: POST

Αυτό το endpoint εγγράφει έναν χρήστη στον identity server.

13. **Endpoint: Login User** URL: `/api/Users/Login` Μέθοδος: POST

Αυτό το endpoint ταυτοποιεί τον χρήστη.

Courses			^
GET	/api/Courses	Gets the courses of the given semester. For semesters 1-4 track must be null, while for semesters 5+ it is mandatory.	✓ 🔒
Grades			^
GET	/api/Grades	Gets the grades of all the courses of the logged in student	✓ 🔒
ProspectiveStudentTests			^
GET	/api/ProspectiveStudentTests/generalTestId	Gets the questions and answers of the general test whose id was provided	✓ 🔒
GET	/api/ProspectiveStudentTests/Completed/{generalTestId}	Gets a bool that indicates whether the logged in user has completed the supplied general test	✓ 🔒
POST	/api/ProspectiveStudentTests	Submits the test answers of the prospective student	✓ 🔒
Statistics			^
POST	/api/Statistics/TeachingAccessStatistics	Increments the semester or the revision year access count statistic for the logged in user by one	✓ 🔒
GET	/api/Statistics/TeachingAccessStatistics	Gets all the teaching access statistics for both the semesters and the revision years	✓ 🔒
StudentTests			^
GET	/api/StudentTests	Gets the questions and answers of the test either of the given semester or the revision of a year.	✓ 🔒
POST	/api/StudentTests	Submits the test answers of the student	✓ 🔒
GET	/api/StudentTests/Completed/{universityTestId}	Gets a bool that indicates whether the logged in user has completed the supplied university test	✓ 🔒
Users			^
GET	/api/Users/{userId}	Gets the user with the specified id. Authorization is required	✓ 🔒
POST	/api/Users/Register	Registers a user to the identity server.	✓ 🔒
POST	/api/Users/Login	Authenticates the user.	✓ 🔒

Εικόνα 2: Documentation του API μέσω του swagger.

2.5 Υλοποίηση της Web εφαρμογής

Η υλοποίηση της web εφαρμογής έγινε με τη χρήση της βιβλιοθήκης React NextJs, η οποία παρέχει προηγμένες λειτουργίες για τη δημιουργία εφαρμογών μονής σελίδας (SPA), καθώς και για την υποστήριξη server-side rendering (SSR) και static site generation (SSG), για βελτιωμένη απόδοση και SEO.



Η εφαρμογή περιλαμβάνει μια σειρά από σελίδες, όπως την αρχική σελίδα, τη σελίδα των μαθημάτων, τη σελίδα των βαθμολογιών και τις σελίδες των τεστ. Οι σελίδες αυτές επικοινωνούν με το back-end API για να ανακτήσουν τα απαραίτητα δεδομένα και να προβάλουν τα αποτελέσματα στον χρήστη.

Η εφαρμογή έχει υλοποιηθεί με απόλυτη συμμόρφωση προς τις αρχές της αποκρίσεως και της προσβασιμότητας, ώστε να είναι ευχρήστη και να παρέχει μια ομαλή εμπειρία σε όλους τους χρήστες, ανεξαρτήτως της συσκευής ή της τεχνολογίας περιήγησης που χρησιμοποιούν.

Επίσης, υλοποιήθηκαν λειτουργίες ασφαλείας, όπως η επαλήθευση των χρηστών και η προστασία των δεδομένων, μέσω της ενσωμάτωσης με τον server πιστοποίησης, για την ασφάλεια των δεδομένων και την προστασία των χρηστών.

Τέλος, όλο το front-end έχει δομηθεί με μια σαφή και ευέλικτη αρχιτεκτονική, που διευκολύνει τη διαχείριση και την επέκταση του κώδικα, καθώς και την προσθήκη νέων λειτουργιών στο μέλλον.



3 Επίδειξη της λύσης

3.1 Οδηγίες Εγκατάστασης

Back-end (.NET 7 API)

Οι απαραίτητες ρυθμίσεις για το .NET 7 API πραγματοποιούνται στο αρχείο **appsettings.json**. Σημειώνεται ότι οι παρακάτω οδηγίες αφορούν μια τυπική εγκατάσταση, και ενδέχεται να χρειάζονται προσαρμογές ανάλογα με την περιβάλλοντα εγκατάστασης:

1. Ανοίξτε το αρχείο **appsettings.json** στον editor της επιλογής σας.
2. Αναζητήστε την ενότητα **ConnectionStrings**. Θα πρέπει να ορίσετε το string σύνδεσης για τη βάση δεδομένων σας εδώ. Για παράδειγμα:

```
"ConnectionStrings": {  
  "DefaultConnection": "Server=myServerAddress;Database=myDataBase;User  
Id=myUsername;Password=myPassword;"
```

Προσθέστε οποιεσδήποτε άλλες απαραίτητες ρυθμίσεις, όπως τις ρυθμίσεις JWT για το authentication της εφαρμογής.

Μετά την ρύθμιση του **appsettings.json**, μπορείτε να ξεκινήσετε το API μέσω του τερματικού πληκτρολογώντας **dotnet run** στον κατάλογο του project.

Front-end (React NextJs Web App)

Για να εγκαταστήσετε και να τρέξετε το front-end της εφαρμογής, ακολουθήστε τα παρακάτω βήματα:

1. Εγκαταστήστε την τελευταία έκδοση του Node.js και του npm (Node Package Manager). Μπορείτε να το κάνετε από την [επίσημη ιστοσελίδα Node.js](https://nodejs.org/en/).
2. Ανοίξτε το τερματικό σας και μεταβείτε στον κατάλογο της εφαρμογής NextJs μέσω της εντολής **cd**.
3. Εγκαταστήστε τα απαραίτητα πακέτα μέσω npm, εκτελώντας την εντολή **npm install**.
4. Για να ξεκινήσετε την εφαρμογή, τρέξτε την εντολή **npm run dev**. Αυτό θα ξεκινήσει τον διακομιστή.

Αφού τρέξει επιτυχώς, θα μπορείτε να περιηγηθείτε στην εφαρμογή στον browser σας στην διεύθυνση **http://localhost:3000** ή την διεύθυνση που θα αναφέρει το τερματικό σας.

3.2 Παρουσίασης της εφαρμογής



[Αρχική Σελίδα](#) [Σχετικά με εμάς](#) [Επικοινωνία](#) [Κεντρικός Πίνακας](#) [Αποσύνδεση](#)

Επαγγελματικός προσανατολισμός για το τμήμα
Πληροφορικής του Πανεπιστημίου Πειραιώς



"It doesn't matter how slowly you go as long as you do not stop."

Created by WEBFLEXERS

Εικόνα 3: Αρχική σελίδα



Κεντρικός Πίνακας



Welcome, Sofia
Φοιτητής

Κεντρικός Πίνακας

Διδασκαλία

Τεστ Αυτοαξιολόγησης

Πρόσδος

Ρυθμίσεις

Έξοδος

Βαθμοί

Εξάμηνο 1	
ΜΑΘΗΜΑΤΙΚΑ ΤΩΝ ΥΠΟΛΟΓΙΣΤΩΝ	8
ΤΕΧΝΟΛΟΓΙΕΣ ΔΙΑΔΙΚΤΥΟΥ	6
ΛΟΓΙΚΗ ΣΧΕΔΙΑΣΗ ΨΗΦΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ	9
ΕΙΣΑΓΩΓΗ ΣΤΗΝ ΕΠΙΣΤΗΜΗ ΤΩΝ ΥΠΟΛΟΓΙΣΤΩΝ	6
ΔΙΑΚΡΙΤΑ ΜΑΘΗΜΑΤΙΚΑ	6
Εξάμηνο 2	
ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ	8
ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΕΦΗΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ	10
ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΥΠΟΛΟΓΙΣΤΩΝ	8
Εξάμηνο 3	
ΜΑΘΗΜΑΤΙΚΟΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ	7
ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΕΦΗΣ ΑΝΑΠΤΥΞΗ ΕΦΑΡΜΟΓΩΝ	5
ΛΕΙΤΟΥΡΓΙΚΑ ΣΥΣΤΗΜΑΤΑ	6
ΜΕΤΑΓΛΩΤΤΙΣΤΕΣ	6
ΒΙΟΛΟΓΙΚΕΣ ΚΑΙ ΣΤΑΤΙΣΤΙΚΕΣ	5



Διδασκαλία

Welcome, Sofia
Φοιτήτης

Κεντρικός Πίνακας

Διδασκαλία

Τεστ Αυτοαξιολόγησης

Πρόσδος

Ρυθμίσεις

Έξοδος

Εξάμηνο 1

1. ΛΟΓΙΚΗ ΣΧΕΔΙΑΣΗ ΨΗΦΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ (Κορμού)

Περιγραφή

Το μάθημα "Λογική Σχεδίαση Ψηφιακών Συστημάτων" εξετάζει τη σχεδίαση και λειτουργία των ψηφιακών συστημάτων. Καλύπτει θέματα όπως το δυαδικό σύστημα αρίθμησης, άλγεβρα Boole και εφαρμογές της, αριθμητικά κυκλώματα, σύγχρονα ακολουθιακά κυκλώματα, μηχανές καταστάσεων, μονάδες μνήμης, ψηφιακά ολοκληρωμένα κυκλώματα, προσομοίωση κυκλωμάτων και γλώσσες περιγραφής υλικού.

Hard Skills

- Κατανόηση των βασικών εννοιών του σχεδιασμού ψηφιακών συστημάτων και της λειτουργίας τους
- Κατανόηση της αναπαράστασης των αριθμών και των δεδομένων στα ψηφιακά συστήματα
- Ικανότητα ανάλυσης και βελτίωσης της απόδοσης ενός ψηφιακού κυκλώματος
- Εξοικείωση με εργαλεία σχεδίασης και προσομοίωσης ψηφιακών κυκλωμάτων

Soft Skills

- Προσαρμοστικότητα
- Δημιουργική σκέψη
- Διαχείριση χρόνου
- Επίλυση προβλημάτων
- Κριτική σκέψη

2. ΜΑΘΗΜΑΤΙΚΑ ΤΩΝ ΥΠΟΛΟΓΙΣΤΩΝ (Κορμού)

Περιγραφή

Το μάθημα "Μαθηματικά των υπολογιστών" προσφέρει μια εισαγωγή στα θεμελιώδη μαθηματικά και λογικά εργαλεία που χρησιμοποιούνται στην πληροφορική.



4 Βιβλιογραφικές Πηγές

1. **Menezes, Vandstone.** *Handbook of Applied Cryptography*. New York : CRC PResS, 1996.