



Εκπαιδευτικό Λογισμικό

ΤΕΧΝΙΚΟ ΕΓΧΕΙΡΙΔΙΟ – TECHNICAL MANUAL

Ακαδημαϊκό Έτος 2022-2023

ΟΜΑΔΑ

Πετροπούλου Μαρία


Π19140

marianapetropoulou@gmail.com

Σπάθα Μυρτώ – Μαρία

Π19156

mirtospatha173@gmail.com



Περιεχόμενα

| | |
|---|---|
| 1. Εισαγωγή..... | 3 |
| 2. Τεχνολογίες..... | 4 |
| 3. Σχεδιασμός και Σχήμα Βάσης Δεδομένων | 5 |
| 4. Διαγράμματα UML | 6 |
| 4.1 Use Case Diagram | 6 |
| 5 Βιβλιογραφία..... | 6 |

1. Εισαγωγή

Η παρούσα εργασία αφορά το μάθημα «Εκπαιδευτικό Λογισμικό», το οποίο διδάσκεται στο 8^ο εξάμηνο του τμήματος Πληροφορικής του Πανεπιστημίου Πειραιώς. Σκοπός της εργασίας είναι ο σχεδιασμός ενός εκπαιδευτικού λογισμικού, το οποίο θα συμβάλλει στον επαγγελματικό προσανατολισμό προπτυχιακών φοιτητών και αποφοίτων του τμήματος. Η εφαρμογή περιέχει εκπαιδευτικό υλικό, το οποίο ενημερώνει τις παραπάνω ομάδες αναφορικά με πληροφορίες για τα διδασκόμενα μαθήματα, για τα μαθησιακά αποτελέσματα τους και τις ικανότητες που αποκτούν οι φοιτητές, οι οποίοι τα παρακολουθούν, καθώς και για επαγγελματικές κατευθύνσεις.

Για την υλοποίηση του συγκεκριμένου ζητήματος, χρησιμοποιήθηκε το υλικό του προγράμματος προπτυχιακών σπουδών του τμήματος μας. Αφού μελετηθεί το υλικό, οι χρήστες καλούνται να πραγματοποιήσουν τεστ αξιολόγησης, τα οποία αποδεικνύουν την κατανόηση των δεδομένων, όπως «Ποια είναι τα μαθησιακά αποτελέσματα του μαθήματος Τεχνολογίας Λογισμικού;». Στη συνέχεια, χρειάζεται η καταχώρηση των βαθμολογιών του πανεπιστημίου στο σύστημα καθώς και η συμπλήρωση ερωτηματολογίων, με στόχο την εξαγωγή κατάλληλων συμπερασμάτων για την σύσταση επαγγελματικών κατευθύνσεων.

Στο παρών τεχνικό εγχειρίδιο αναλύονται οι τεχνολογίες οι οποίες χρησιμοποιήθηκαν, ο σχεδιασμός και το σχήμα της βάσης δεδομένων, η αρχιτεκτονική του συστήματος και τα βασικά διαγράμματα UML.

2. Τεχνολογίες

Κατά την υλοποίηση της εφαρμογής, η γλώσσα προγραμματισμού η οποία χρησιμοποιήθηκε, κυρίως, είναι η C# ενώ η ανάπτυξη του προγράμματος έγινε μέσω του περιβάλλοντος Visual Studio της Microsoft. Για την διαμόρφωση του user interface, βέβαια, χρησιμοποιήθηκαν οι γλώσσες προγραμματισμού HTML5, CSS και σε μικρότερη κλίμακα η Javascript, με σκοπό την δημιουργία μίας φιλικής-προς-τον-χρήστη εφαρμογής.

Αξίζει να σημειωθεί πως χρησιμοποιήθηκε η προσέγγιση Code-First με το Entity Framework, μία ακόμη υλοποίηση της Microsoft. Σύμφωνα με την συγκεκριμένη, η προσοχή εστιάζεται στην εφαρμογή και σχεδιάζονται οι κατάλληλες κλάσεις (χρήστης, μαθήματα, πρόοδος, τεστ κλπ.), βάσει των οποίων θα δημιουργηθεί η βάση δεδομένων μέσω του EF. Στην αντίθετη περίπτωση, σχεδιάζεται πρώτα η ΒΔ και ύστερα οι κλάσεις, οι οποίες ταιριάζουν στους υπάρχοντες πίνακες. Έτσι, μετά την δημιουργία ή τροποποίηση των κλάσεων, πραγματοποιούμε την διαδικασία Migration, μέσω του Package Manager Console στο VS. Με αυτή την μεθοδολογία, απαιτείται αποκλειστικά η γνώση C#, αποφεύγοντας την SQL.

Περαιτέρω, για την ανάπτυξη της εργασίας χρησιμοποιήθηκε το πρόσθετο εργαλείο Language Integrated Query (LINQ), το οποίο συμβάλλει στον χειρισμό μεγάλου πλήθους δεδομένων με την χρήση ερωτημάτων, όπως τα ερωτήματα της SQL. Στην ουσία, αποτελεί μία μέθοδο σύνταξης, με σκοπό την ανάκτηση δεδομένων από διαφορετικές πηγές, για παράδειγμα μία Βάση Δεδομένων. Μάλιστα, ένα LINQ query επιστρέφει τα αποτελέσματα ως αντικείμενα, με αποτέλεσμα να επιτρέπει την αντικειμενοστραφή διαχείριση των αποτελεσμάτων. Το LINQ, εφαρμόστηκε στον Controller, με σκοπό την πρόσβαση σε δεδομένα σχετικά με τα μαθήματα, τους χρήστες, την πρόοδο και τους βαθμούς τους καθώς και τα τεστ, τα οποία καταχώρησαν στο σύστημα.

Παρακάτω έχουμε συμπεριλάβει ένα απόσπασμα κώδικα, στο οποίο έχει χρησιμοποιηθεί LINQ και στο οποίο επιλέγεται από την βάση ο χρήστης με το όνομα χρήστη το οποίο έχει αποθηκευτεί στην μεταβλητή username.

```
var user = _context.users.FirstOrDefault x => x.username == username
```

Τέλος, τα μέλη της ομάδας μας χρησιμοποίησαν το εργαλείο pgAdmin4, για την διαχείριση της βάσης δεδομένων.

3. Αρχιτεκτονική Συστήματος

Όσον αφορά την αρχιτεκτονική του συστήματος, χρησιμοποιήθηκε το μοντέλο αρχιτεκτονικής λογισμικού Model-View-Controller (MVC) και υλοποιήθηκε, τελικά, μία web εφαρμογή. Όσον αφορά την συγκεκριμένη τεχνολογία, η εργασία διαχωρίζεται σε τρία μοντέλα (Model, View, Controller), τα οποία επικοινωνούν μεταξύ τους για την επίτευξη των εκάστοτε λειτουργιών. Το Model είναι υπεύθυνο για την διαχείριση των δεδομένων, τα οποία μεταφέρονται μεταξύ της βάσης δεδομένων για της διεπαφής του χρήστη, το View χειρίζεται την διάταξη και την μορφή προβολής των δεδομένων, ενώ ο Controller ευθύνεται για την δρομολόγηση των commands στο Model και στο View.

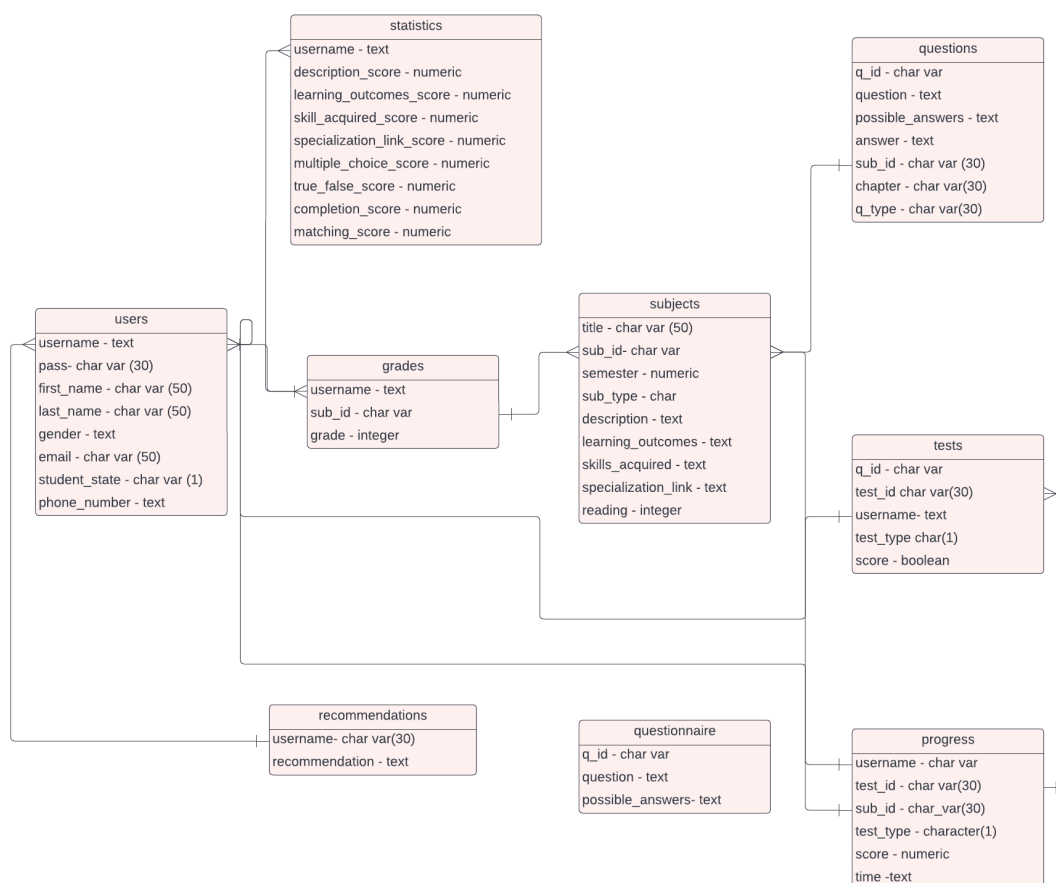
4. Σχεδιασμός και Σχήμα Βάσης Δεδομένων

Για τον σχεδιασμό της βάσης δεδομένων, αναλύθηκαν και επεξεργάστηκαν όλα τα δεδομένα, τα οποία θα ήταν χρήσιμα για την εφαρμογή. Πιο συγκεκριμένα, δημιουργήθηκε, αρχικά, ο πίνακας χρηστών όπου αποθηκεύεται κάθε χρήστης ο οποίος κάνει registration.

Έπειτα, ο πίνακας subjects αφορά το σύνολο των μαθημάτων και περιέχει το αντίστοιχο εκπαιδευτικό υλικό ενώ ο questions εμπεριέχει τις ερωτήσεις οι οποίες προβάλλονται τυχαία στα τεστ και ακριβέστερα, περιέχει την εκφώνηση, τις σωστές απαντήσεις, το μάθημα και την υποενότητα στα οποία αντιστοιχούν, καθώς και τον τύπο ερώτησης (πχ *πολλαπλής επιλογής*). Αντίστοιχα, έχει υλοποιηθεί και το ερωτηματολόγιο ενδιαφερόντων, το οποίο, βέβαια, δεν χρειάζεται έλεγχο ορθότητας των απαντήσεων. Μετά την ολοκλήρωση κάθε τεστ, ακολουθεί η αποθήκευση κάθε απάντησης στον πίνακα tests, από τον οποίο λαμβάνει δεδομένα και ο πίνακας statistics.

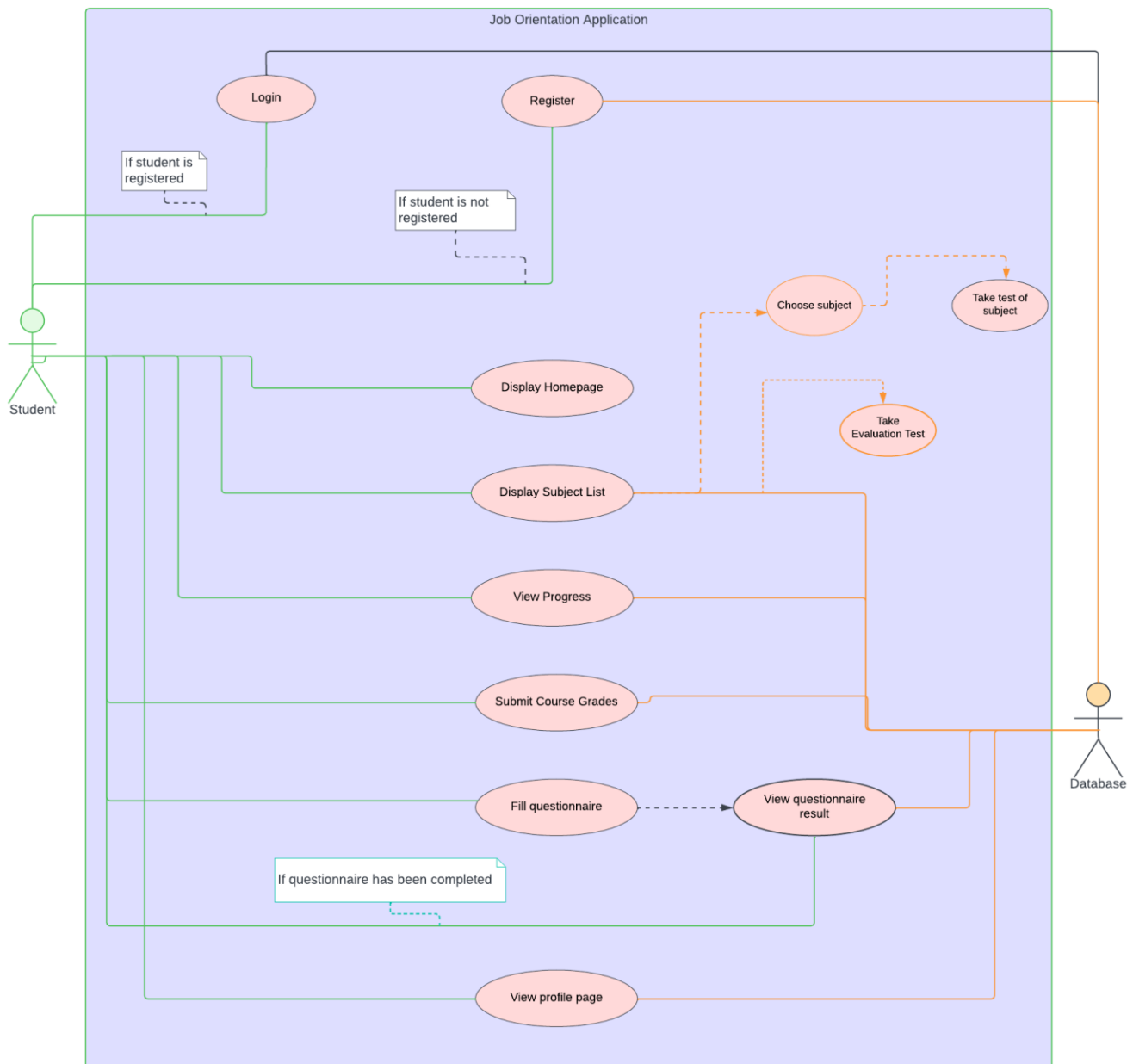
Ο τελευταίος εμφανίζεται στην σελίδα προόδου, για παρακολούθηση της πορείας του χρήστη. Για την ανάγκη καταγραφής παρελθοντικών βαθμών στα μαθήματα, υλοποιήθηκε και ο πίνακας grades ενώ μετά την ολοκλήρωση του ερωτηματολογίου συμπληρώνεται και ο τελευταίος πίνακας, ο οποίος ονομάστηκε recommendations και περιέχει την τελική πρόταση επαγγελματικού προσανατολισμού.

Παρακάτω έχει ενσωματωθεί το σχήμα βάσης δεδομένων.



5. Διαγράμματα UML

4.1 Use Case Diagram



5 Βιβλιογραφία

[1] Scott Addie, Rick Anderson, Tom Dykstra, Andy Pasic, Nick Schonning και Dean Simpson (2022, 7 Ιουλίου) "Entity Framework", <https://learn.microsoft.com/en-us/aspnet/entity-framework>, [Accessed 13 Ιουνίου, 2023].

[2] David Coulter, Steve Hoag, Luke Latham, Nick Schonning, Zev Spitz, Bill Wagner, Genevieve Warren και Maira Wenzel (2023, 9 Μαρτίου) Language Integrated Query (LINQ), <https://learn.microsoft.com/el-gr/dotnet/csharp/linq/linq-in-csharp> [Accessed 13 Ιουνίου, 2023].

[3] Brice Lambson, Smit Patel, Shay Rojansky, Arthur Vickers, Maira Wenzel, Tom Dykstra και Diego Vega (2020, 14 Οκτωβρίου), “Code First to a New Database”, <https://learn.microsoft.com/en-us/ef/ef6/modeling/code-first/workflows/new-database> , [Accessed 13 Ιουνίου, 2023].