

**ΠΜΣ «ΠΛΗΡΟΦΟΡΙΚΗ»**  
**ΤΑΧΕΙΑ ΑΝΑΠΤΥΞΗ ΕΦΑΡΜΟΓΩΝ**  
**ΘΕΜΑ ΑΠΑΛΛΑΚΤΙΚΗΣ ΕΡΓΑΣΙΑΣ 2019**

**Μονάδες: 10**

**Περιγραφή:**

Η εργασία του μαθήματος μπορεί να εκπονηθεί από ομάδες φοιτητών 2 ατόμων. Θα πρέπει να συνεννοηθείτε μεταξύ σας για την σύνθεση της ομάδας σας.

Ως εργασία θα πρέπει να αναπτύξετε σε γλώσσα προγραμματισμού C# μια εφαρμογή ενός τουριστικού οδηγού, για κάποιον προορισμό της επιλογής σας. Η τελική εφαρμογή μπορεί να είναι windows form application, όπως έχετε διδαχθεί και στα εργαστήρια που έχουν γίνει στο μάθημα (μπορείτε να την υλοποιήσετε, εφόσον το θέλετε και σε άλλα είδη projects που υποστηρίζει το Visual Studio). Θα επιλέξετε έναν τουριστικό προορισμό (π.χ. κάποιο νησί) για τον οποίο μπορείτε να συγκεντρώσετε το απαραίτητο υλικό (π.χ. αξιοθέατα, παραλίες, παραστάσεις, εστιατόρια, εκδηλώσεις, κ.λπ.) και στη συνέχεια θα χρησιμοποιήσετε το υλικό αυτό για να «γεμίσετε» και να «χτίσετε» την εφαρμογή σας.

Βασικά χαρακτηριστικά εφαρμογής<sup>1</sup>:

- Στην εφαρμογή σας μπορούν να εισέλθουν είτε επισκέπτες είτε εγγεγραμμένοι χρήστες. Οι εγγεγραμμένοι χρήστες θα έχουν περισσότερες δυνατότητες από τους επισκέπτες (π.χ. παρουσίαση πρόσθετων πληροφοριών, πλοήγηση σε περισσότερες φόρμες, κ.λπ.)
- Στην περίπτωση των εγγεγραμμένων χρηστών η εφαρμογή θα πρέπει να διαθέτει μια απλή φόρμα login για την είσοδο (φυσικά με τον απαραίτητο έλεγχο). Δεν απαιτείται εγγραφή χρήστη, οι χρήστες μπορούν να είναι ήδη «περασμένοι» μέσα στην εφαρμογή σας. (Μια ορθή προσέγγιση στο πρόβλημα της ύπαρξης/διαχείρισης χρηστών περιλαμβάνει την ενσωμάτωση μιας βάσης δεδομένων, κάτι το οποίο έχει καλυφθεί από την ύλη μας)
- Θα πρέπει για κάθε χρήστη να τηρείται ιστορικό (έστω και προσωρινό, όσο χρησιμοποιεί το πρόγραμμα), δηλαδή να καταγράφονται όσα μέρη του τουριστικού οδηγού έχει ήδη επισκεφτεί (π.χ ποιες φόρμες έχει δει) και να διακρίνονται από αυτά που δεν έχει ακόμα επισκεφτεί. Επίσης να υπάρχει η δυνατότητα καθαρισμού του ιστορικού του
- Θα πρέπει να διαθέτει πολλαπλά παράθυρα (forms) στα οποία θα υπάρχουν οι διάφορες ενότητες/περιοχές του τουριστικού οδηγού
- Το περιεχόμενο του οδηγού θα πρέπει να περιλαμβάνει κείμενο στην ελληνική γλώσσα, εικόνες και αρχεία πολυμέσων (ήχος, animation)
- Θα πρέπει σε κάθε φόρμα να υπάρχει μενού επιλογών (εκτός της όποιας πλοήγησης που θα έχετε φτιάξει εσείς με κουμπιά)
- Σε μια από τις διαθέσιμες φόρμες θα πρέπει να υπάρχει ένα ημερολόγιο, το οποίο θα πρέπει να χρησιμοποιήσετε (π.χ. για την ανάγνωση κάποιων εκδηλώσεων, ή για την καταχώρηση σχολίων από τους χρήστες)

---

<sup>1</sup> Φυσικά, είστε ελεύθεροι εκτός από αυτά που σας ζητάμε εμείς ως απαιτούμενα να υλοποιήσετε και επιπλέον λειτουργίες, προς όφελος της εργασίας σας. Όλες οι προσπάθειες θα εκτιμηθούν.

- Σε κάθε φόρμα θα πρέπει να υπάρχει κουμπί για την έξοδο από την εφαρμογή, καθώς και ένα κουμπί “About” όπου θα παραπέμπει στα στοιχεία των δημιουργών της εφαρμογής
- Θα πρέπει με τη χρήση ενός timer να υλοποιήσετε (με χρονική καθυστέρηση) μια διαδοχή φωτογραφιών (σαν slideshow)
- Θα πρέπει να γίνεται ο απαραίτητος χειρισμός εξαιρέσεων από την εφαρμογή σας (exception handling), ώστε κατά την λανθασμένη εισαγωγή δεδομένων από τον χρήστη, η εφαρμογή να μην καταρρέει.
- Θα πρέπει η εφαρμογή σας να υποστηρίζει κάποια επιπλέον events, εκτός από το «κλασικό» πάτημα του κουμπιού. Μπορείτε εσείς να επιλέξετε τα events αυτά.
- Οι χρήστες της εφαρμογής θα πρέπει να μπορούν να κάνουν export σε ένα αρχείο κειμένου τις διαθέσιμες πληροφορίες για ένα κάποια καταχώρηση του τουριστικού οδηγού, όπως π.χ. τα στοιχεία ενός εστιατορίου, ή τα στοιχεία για μια παράσταση.

#### Σημαντικά:

- Η χρήση της C# ως γλώσσας προγραμματισμού είναι υποχρεωτική.
- Η ύπαρξη ΒΔ είναι προαιρετική (μπορούν να χρησιμοποιηθούν κατάλληλα αρχεία, π.χ. κειμένου)
- Οι φοιτητές καλούνται να υλοποιήσουν όσο το δυνατόν περισσότερες τάξεις (όπου είναι αυτό εφικτό), οι οποίες να περιλαμβάνουν λειτουργίες και ιδιότητες. Σε κάθε εφαρμογή πρέπει να έχει υλοποιηθεί **τουλάχιστον 1** τάξη δημιουργημένη από τους φοιτητές, ακόμα και αν αυτή χρησιμοποιεί ή καλεί τμήματα άλλων τάξεων. Η τάξη αυτή θα πρέπει να χρησιμοποιείται κατά τη διάρκεια εκτέλεσης της εφαρμογής. Η χρήση «χαρακτηριστικών» του αντικειμενοστρεφούς προγραμματισμού (π.χ. κληρονομικότητα μεταξύ τάξεων) στις δικές σας τάξεις δεν είναι υποχρεωτική, ωστόσο θα εκτιμηθεί ιδιαίτερα εφόσον υλοποιηθεί από εσάς.
- Το εργαλείο ανάπτυξης των εφαρμογών είναι της επιλογής σας. Όμως συνίσταται η χρήση κάποιας έκδοσης Visual Studio, όσο πιο νέας και πληρέστερης γίνεται (υπενθυμίζεται ότι όλο το απαραίτητο software είναι διαθέσιμο στους φοιτητές μέσω της υπηρεσίας msdnaa).
- Εντός της κάθε ομάδας τα μέλη καλούνται να συνεργαστούν και να αλληλοβοηθηθούν. Κατά της τελική εξέταση, το κάθε μέλος της ομάδας πρέπει να γνωρίζει το πώς υλοποιήθηκε η εφαρμογή και να γνωρίζει όλο τον κώδικά της. **Κατά την εξέταση, παράλληλα με την παρουσίαση/επίδειξη της εκτελέσιμης εφαρμογής γίνονται και ερωτήσεις που αφορούν στον κώδικα και στην αντιστοίχιση λειτουργιών με τμήματα κώδικα. Παράλληλα γίνονται και ερωτήσεις σε σχέση με ό,τι έχετε διδαχθεί στο εν λόγω μάθημα.**
- Εξετάζονται και οι 2 φοιτητές και μάλιστα είναι πολύ πιθανό να πάρουν και διαφορετικό βαθμό, ανάλογα με το πώς θα απαντήσουν στις ερωτήσεις των εξεταστών. Η παρουσία όλων των υπό εξέταση φοιτητών είναι υποχρεωτική, όπως και στις γραπτές εξετάσεις (δεν γίνεται να παρουσιάσει εργασία το ένα μέλος της ομάδας και να βαθμολογηθεί ένα άλλο μέλος το οποίο λείπει).
- Δεν επιτρέπεται η συνεργασία με άλλες ομάδες.
- Η εξέταση θα γίνει στα εργαστήρια του τμήματος. Για την ακριβή ημερομηνία/ώρα θα ενημερωθείτε με σχετική ανακοίνωση στο e-class.

**Καλή επιτυχία!!**