

CSYM025: Visual Object Software			
Ημερομηνία Δημιουργίας:	20/03/2023	Καταληκτική Ημερομηνία Παράδοσης:	10/07/2023
		Υπεύθυνος Μαθήματος:	Dr Θεόδωρος Μανάβης

Οδηγίες – Παρακαλείστε να διαβάσετε προσεκτικά:

- I. Η Πολιτική του Πανεπιστημίου του Northampton για την ακαδημαϊκή ακεραιότητα και την ανάρμοστη συμπεριφορά θα εφαρμοστεί αυστηρά.
- II. Η παρούσα εργασία δεν είναι ομαδική και υποβάλλοντας την δηλώνετε ότι δημιουργήθηκε με αποκλειστικά δική σας προσπάθεια. Μπορείτε να συζητήσετε την εργασία με συμφοιτητές σας, αλλά οποιοσδήποτε γραμμένος κώδικας θα πρέπει να είναι δικός σας. Η κοινή χρήση της εργασίας σας με άλλον μαθητή ή η υποβολή κώδικα που γράφτηκε από κάποιον άλλον μπορεί να θεωρηθεί ακαδημαϊκό παράπτωμα.
- III. Εάν έχετε χρησιμοποιήσει οποιονδήποτε κώδικα από εξωτερική πηγή, τον οποίο δεν γράψατε εσείς, πρέπει να το αναφέρετε με σαφήνεια με κάποια παραπομπή. Δείτε εδώ: <https://libguides.northampton.ac.uk/c.php?g=683675&p=4918336>
- IV. Πρέπει να υποβάλετε όλα τα ζητούμενα της αξιολόγησης σύμφωνα με τη διαδικασία υποβολής που αναφέρεται σε αυτό το έγγραφο. **Η μη τήρηση της διαδικασίας υποβολής ενδέχεται να οδηγήσει σε ποινή ή σε μείωση του βαθμού.**
- V. Αυτή η εργασία/μάθημα χρησιμοποιείται για την αξιολόγηση των σχεδιαστικών και προγραμματιστικών δεξιοτήτων σας στην αντικειμενοστραφή φιλοσοφία. Ως εκ τούτου, **οποιαδήποτε χρήση βάσεων δεδομένων και γλωσσών ερωτημάτων (π.χ. SQL) απαγορεύεται αυστηρά και θα οδηγούσε σε αποτυχία στο μάθημα.**

Εκφώνηση:

Σχεδιάστε, υλοποιήστε και δοκιμάστε ένα σύστημα διαχείρισης για έναν πάροχο ενέργειας στο Northampton χρησιμοποιώντας αντικειμενοστραφείς αρχές με χρήση Java FX. Ο σχεδιασμός πρέπει να περιλαμβάνει διαγράμματα κλάσεων. Η δοκιμή (testing) θα πρέπει να περιλαμβάνει τόσο “white box” (δοκιμές JUnit) όσο και “black box” (καταγραφή αποτελεσμάτων των τεστ). Το σύστημα θα πρέπει να επιτρέπει στον προμηθευτή να καταγράφει λογαριασμούς πελατών, τιμολογήσεις ενέργειας, να δημιουργεί τιμολόγια πελατών και ούτω καθεξής. Ακολουθούν οι λεπτομερείς απαιτήσεις:

Βασικές απαιτήσεις συστήματος:

Το σύστημα πρέπει να επιτρέπει στον προμηθευτή να:

1. Καταγράφει τα στοιχεία ενός νέου πελάτη (όνομα, αριθμός τηλεφώνου, τρέχουσα διεύθυνση, τιμολόγηση ενέργειας, τύπος μετρητή, κ.λπ.).

2. Εμφανίζει τον πίνακα ελέγχου χρήσης ενέργειας ενός συγκεκριμένου πελάτη (π.χ. ενδείξεις μετρητών, περίληψη λογαριασμού (πληρωμές, λογαριασμοί, λεπτομέρειες τιμολογήσεων κ.λπ.).
3. Δημιουργεί ένα μηνιαίο τιμολόγιο/λογαριασμό για κάθε πελάτη. Ο λογαριασμός θα πρέπει επίσης να δείχνει λεπτομερείς υπολογισμούς των ενεργειακών χρεώσεων (δηλαδή, μετατροπή των ενδείξεων του μετρητή σε χρεώσεις).
4. Καταγράφει μια πληρωμή για κάθε πελάτη (δηλαδή πληρωμένη/μη πληρωμένη).

Βελτιώσεις (κατά σειρά σπουδαιότητας – υψηλή προς χαμηλή):

Οι πρόσθετες λειτουργίες που μπορείτε να συμπεριλάβετε είναι οι εξής:

1. Μόνιμη αποθήκευση δεδομένων και CRUD [Create, Read, Update and Delete] (object serialisation) ([Δημιουργία, ανάγνωση, ενημέρωση και διαγραφή] (σειριοποίηση αντικειμένων)).
2. Εισαγωγή/Ανάγνωση ενδείξεων μετρητή ενέργειας από αρχείο (σειριοποίηση αντικειμένων ή αρχεία κειμένου/csv).
3. Αναζήτηση πελάτη με αριθμό λογαριασμού ή όνομα.
4. Εισαγωγή/ενημέρωση πληροφοριών τιμολόγησης (δηλαδή προσθήκη/τροποποίηση μιας νέας/υφιστάμενης προσφερόμενης τιμολόγησης).
5. Εμφάνιση γραφήματος ετήσιας χρήσης ενέργειας και προβλεπόμενης μηνιαίας χρήσης.
6. Αποστολή email που περιέχει το τιμολόγιο ως συνημμένο (αρχείο pdf) σε έναν πελάτη.
7. Δυνατότητα σύνδεσης (login) στο σύστημα.

Αυτές οι πρόσθετες λειτουργίες είναι μόνο προτάσεις και δεν είναι εξαντλητικές. Μπορείτε να συμπεριλάβετε οποιεσδήποτε άλλες χρήσιμες λειτουργίες που σχετίζονται με αυτήν την εφαρμογή. Πρέπει να διευκρινίσετε/παράξετε πιο λεπτομερείς απαιτήσεις από τον καθηγητή και ενθαρρύνετε να χρησιμοποιήσετε μια ευέλικτη (agile) μεθοδολογία για την ανάπτυξη λογισμικού.

Παραδοτέα:

Όλες οι απαιτήσεις (A, B και C που ακολουθούν) **ΠΡΕΠΕΙ** να παραδοθούν για να επιτευχθεί ένας προβιβάσιμος βαθμός για αυτήν την εργασία.

A) Τεχνική έκθεση

Η αναφορά πρέπει να αποτελείται από τις ακόλουθες ενότητες (με την ίδια σειρά):

- a. Όνομα χρήστη και κωδικός πρόσβασης για όλους τους σχετικούς λογαριασμούς (εάν εφαρμόζονται).
- b. UML διαγράμματα κλάσεων που δείχνουν τις συσχετίσεις μεταξύ των κύριων κλάσεων στο μοντέλο.
- c. Μια λίστα σε μορφή πίνακα, με όλες τις λειτουργίες που υλοποιούνται στην εφαρμογή. Για παράδειγμα:

Λειτουργία	Υλοποιήθηκε (Μερικώς/Πλήρως)	Σχόλια
Καταγραφή στοιχείων νέου πελάτη	Πλήρως	Δεν πραγματοποιήθηκε επικύρωση σφαλμάτων.
Εμφάνιση πίνακα ελέγχου χρήσης ενέργειας	Πλήρως	

1. Επεξήγηση των κύριων ενοτήτων/τμημάτων του κώδικα. Παρέχετε πληροφορίες που θα ήταν χρήσιμες για έναν άλλο προγραμματιστή (όχι τελικό χρήστη!) που μπορεί να θέλει να επεκτείνει/συντηρήσει το σύστημά σας. Ίσως χρειαστεί να ανατρέξετε στα διαγράμματα κλάσεων για να εξηγήσετε τον κώδικα.
2. Στιγμιότυπα οθόνης του συστήματος που δείχνουν όλες τις βασικές λειτουργίες.
3. Αποδείξεις των Δοκιμών (Testing):
 - a. Δοκιμές Blackbox: Αρχεία καταγραφής δοκιμών που παρέχουν πληροφορίες για όλες τις δοκιμές που πραγματοποιήθηκαν (συμπεριλαμβανομένων τυχόν αποτυχημένων δοκιμών για λειτουργικότητα που δεν υλοποιήθηκε).
 - b. Δοκιμές Whitebox: **Παράθεση Κώδικα** όλων των περιπτώσεων δοκιμών JUnit.
 - c. Λίστα τυχόν σφαλμάτων και/ή αδυναμιών στο σύστημά σας (αν δεν πιστεύετε ότι υπάρχουν, τότε πείτε το). Τα σφάλματα που δηλώνονται σε αυτήν τη λίστα θα σας στερήσουν λιγότερους βαθμούς από αυτά που δεν δηλώνετε.
4. Παραπομπές εντός Κώδικα:
Εάν έχετε δανειστεί κώδικα από αλλού (π.χ. από ένα βιβλίο ή κάποια πηγή στον ιστό), πρέπει να υποδείξετε με σαφήνεια τι είναι και να συμπεριλάβετε παραπομπές στον κώδικά σας σύμφωνα με τις οδηγίες αναφοράς κώδικα (βλ. οδηγίες παραπάνω).

B) Πηγαίος Κώδικας

Ο πηγαίος κώδικας πρέπει να είναι καλά τεκμηριωμένος με τα απαραίτητα σχόλια. Η συνεπής και σαφής «στοίχιση» του κώδικα είναι επίσης σημαντική. Ο πηγαίος κώδικας πρέπει να υποβληθεί σε δύο μορφές:

- (i) Ως ενιαίο αρχείο ZIP (αρχείο .zip που αποτελείται από όλα τα αρχεία ".java", δοκιμές μονάδων, αρχεία δεδομένων, εκτελέσιμο jar).
- (ii) Μια πλήρης καταχώριση με σχόλια σε ένα ξεχωριστό έγγραφο του Word με το όνομα "Πλήρης καταχώριση πηγαίου κώδικα".

C) Επίδειξη με Βίντεο

Εκτός από την αναφορά, πρέπει να υποβάλετε μια επίδειξη βίντεο (URL) της εργασίας σας. Το demo θα πρέπει να έχει διάρκεια περίπου 10 λεπτά (μέγιστο: 15 λεπτά) και να καλύπτει όλη την εργασία σας με λογικό τρόπο. Θα πρέπει να εξηγήσετε τις κύριες φάσεις του σχεδιασμού και της υλοποίησης καλύπτοντας τα κύρια τμήματα του κώδικα. Το πρόσωπο και η φωνή σας πρέπει να είναι καθαρά. Θα πρέπει επίσης να περιλαμβάνει μια βήμα προς βήμα περιγραφή της χρήσης του λογισμικού και πρέπει να επιδεικνύει τα βασικά χαρακτηριστικά. Ο υπεύθυνος του μαθήματος διατηρεί το δικαίωμα να σας προσκαλέσει σε δια

ζώσης παρουσίαση. Τυχόν ανεπαρκής επίδειξη, παρουσίαση ή αναφορά θα μπορούσε να επηρεάσει αρνητικά όλα τα κριτήρια βαθμολόγησης και να οδηγήσει σε μη προβιβάσιμο βαθμό.

Διαδικασία Υποβολής:

- Ηλεκτρονική υποβολή των εγγράφων μέσω Turnitin στο NILE ως ΔΥΟ ξεχωριστά έγγραφα WORD, με τα ακόλουθα ονόματα:
[Έγγραφο 1 = **Report** & Έγγραφο 2 = **FullSourceCodeListing**].
Για να το κάνετε αυτό, μεταβείτε στον ιστότοπο NILE για αυτό το μάθημα και χρησιμοποιήστε τον σύνδεσμο με την ένδειξη «Submit your work».
- Ηλεκτρονική υποβολή ενός ενιαίου αρχείου ZIP που περιέχει όλα τα αρχεία πηγαίου κώδικα (.java), δοκιμές μονάδων, αρχεία δεδομένων, αρχείο ReadMe και εκτελέσιμο αρχείο (jar). Ονομάστε το αρχείο με τη φοιτητική σας ταυτότητα, π.χ. 12345678.zip όπου το 12345678 είναι η φοιτητική σας ταυτότητα. Για να το κάνετε αυτό, μεταβείτε στον ιστότοπο NILE για αυτό το μάθημα και χρησιμοποιήστε τον σύνδεσμο με την ένδειξη «Submit your work». Κάνοντας κλικ στον σύνδεσμο (SourceCodeSubmission), θα μεταφερθείτε στη φόρμα υποβολής, όπου μπορείτε να ανεβάσετε το αρχείο ZIP χρησιμοποιώντας το κουμπί «Attach File» (Browse for Local File). Τέλος, κάντε κλικ στο κουμπί “Submit”.
- Κατά την υποβολή της επίδειξης βίντεο, συνιστάται η χρήση του Kaltura (<https://video.northampton.ac.uk/>). Πρέπει να βεβαιωθείτε ότι ο σύνδεσμος βίντεο είναι προσβάσιμος στον διορθωτή (μην τον ορίσετε σε ιδιωτική πρόσβαση).
- Η μη τήρηση των παραπάνω οδηγιών υποβολής ενδέχεται να οδηγήσει σε μειωμένο βαθμό ή αποτυχία.

Κριτήρια Βαθμολόγησης:

Ο βαθμός για αυτήν την εργασία θα αποτελεί το 100% του συνολικού βαθμού για το μάθημα. Οι βαθμοί θα κατανεμηθούν σύμφωνα με το ακόλουθο σχήμα. Γενικά, τα ακόλουθα κριτήρια θα λειτουργήσουν ως οδηγός για το τι πρέπει να περιμένετε:

	A	B	C	F	G
Σχεδιασμός (20%)	Εξαιρετικός σχεδιασμός εφαρμογής και διεπαφής χρήστη. Τήρηση αντικειμενοστραφών αρχών. Τα διαγράμματα κλάσεων είναι πολύ καλά σχεδιασμένα και παρουσιασμένα.	Καλής ποιότητας σχεδιασμός προγράμματος και διεπαφής χρήστη. Επαρκής υιοθέτηση αντικειμενοστραφών αρχών. Τα διαγράμματα κλάσεων είναι καλά σχεδιασμένα και παρουσιασμένα.	Ικανοποιητικός σχεδιασμός προγράμματος και διεπαφής χρήστη. Τα διαγράμματα κλάσεων είναι καλά σχεδιασμένα και παρουσιασμένα.	Εσφαλμένη σχεδίαση διαγραμμάτων κλάσεων. Πολύ σύντομος σχολιασμός της συνολικής σχεδίασης.	Καμία υποβολή ή υποβολή χωρίς περιεχόμενο.
Λειτουργικότητα (35%)	Όλα τα κριτήρια για το (B) και πολλά σημαντικά πρόσθετα χαρακτηριστικά.	Όλα τα κριτήρια για το (C) και ορισμένα σημαντικά πρόσθετα χαρακτηριστικά.	Οι περισσότερες βασικές απαιτήσεις του συστήματος πληρούνται.	Οι περισσότερες βασικές απαιτήσεις του συστήματος δεν πληρούνται.	Καμία υποβολή ή υποβολή χωρίς περιεχόμενο.
Δοκιμές (20%)	Υπάρχουν αποδείξεις δοκιμών white box και black box με εκτεταμένη κάλυψη του κώδικα.	Υπάρχουν αποδείξεις δοκιμών white box και black box με καλή κάλυψη του κώδικα.	Υπάρχουν αποδείξεις δοκιμών είτε white box είτε black box με ικανοποιητική κάλυψη του κώδικα.	Δεν υπάρχουν αποδείξεις δοκιμών είτε white box είτε black box.	Καμία υποβολή ή υποβολή χωρίς περιεχόμενο.
Τεχνική Έκθεση (10%)	Άριστη περιγραφή των κύριων τμημάτων του κώδικα. Όλα τα τμήματα καλύπτονται με τη σωστή σειρά. Οποιοσδήποτε παραδοχές και σφάλματα/αδυναμίες αναφέρονται σαφώς.	Καλή περιγραφή των κύριων τμημάτων του κώδικα. Όλα τα τμήματα καλύπτονται με τη σωστή σειρά.	Ικανοποιητική περιγραφή των κύριων τμημάτων του κώδικα. Καλύπτονται τα περισσότερα τμήματα.	Καμία περιγραφή των τμημάτων του κώδικα.	Καμία υποβολή ή υποβολή χωρίς περιεχόμενο.
Ποιότητα κώδικα και αποδοτικότητα (15%)	Ο κώδικας είναι πολύ καλά δομημένος ώστε να επιτρέπει τη δοκιμή white box, την επαναχρησιμοποίηση και τον εντοπισμό σφαλμάτων. Άριστη δουλειά στον χειρισμό σφαλμάτων.	Ο κώδικας είναι καλά δομημένος ώστε να επιτρέπει τη δοκιμή white box, την επαναχρησιμοποίηση και τον εντοπισμό σφαλμάτων. Καλή δουλειά στον χειρισμό σφαλμάτων.	Καταβλήθηκε κάποια προσπάθεια στη δημιουργία κατάλληλης δομής του κώδικα και της διαχείρισης σφαλμάτων.	Καταβλήθηκε ελάχιστη προσπάθεια στη δημιουργία κατάλληλης δομής του κώδικα.	Καμία υποβολή ή υποβολή χωρίς περιεχόμενο.