ΣΤΟΙΧΕΙΑ ΠΟΥ ΣΥΜΠΛΗΡΩΝΕΙ Ο ΦΟΙΤΗΤΗΣ / Η ΦΟΙΤΗΤΡΙΑ

|  |  |
| --- | --- |
| Ονοματεπώνυμο φοιτητή |  |
| Αριθμός Μητρώου |  |
| Κωδικός Θ.Ε. | ΠΛΗ24 |
| Κωδικός Τμήματος |  |
| Α/Α Γραπτής εργασίας | 2η ΕΡΓΑΣΙΑ |
| Ονοματεπώνυμο καθηγητή |  |
| Σχόλια προς καθηγητή |  |

***Υπεύθυνη Δήλωση Φοιτητή****: Βεβαιώνω ότι είμαι συγγραφέας αυτής της εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία αυτής της εργασίας, είναι πλήρως αναγνωρισμένη και αναφέρεται, είτε στο σημείο «Σχόλια προς καθηγητή», είτε μέσα στην εργασία. Επίσης, έχω αναφέρει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων, είτε αυτές αναφέρονται ακριβώς, είτε παραφρασμένες. Επίσης, βεβαιώνω ότι αυτή η εργασία προετοιμάστηκε από εμένα προσωπικά ειδικά για τη συγκεκριμένη Θεματική Ενότητα.*

|  |  |
| --- | --- |
|  | Συμφωνώ και αποδέχομαι την ανωτέρω δήλωση |
|  | Δε συμφωνώ και δεν αποδέχομαι την ανωτέρω δήλωση (στην περίπτωση αυτή, ο Κ-Σ έχει δικαίωμα να μην αξιολογήσει την εργασία του φοιτητή) |

|  |
| --- |
| **Ημερομηνία ανακοίνωσης εργασίας:** Τετάρτη, 07/12/2021  **Ημερομηνία υποβολής εργασίας:** Τετάρτη, 19/01/2022, 11:59 μ.μ.  **Ημερομηνία υποβολής εργασίας με ατομική παράταση:** Τετάρτη 26/1/2022, 11:59 μ.μ.  *Ο κάθε φοιτητής δικαιούται μια ατομική παράταση σε όλη τη διάρκεια του ακαδημαϊκού έτους* |

ΣΤΟΙΧΕΙΑ ΠΟΥ ΣΥΜΠΛΗΡΩΝΕΙ Ο ΚΑΘΗΓΗΤΗΣ

|  |  |
| --- | --- |
| **Ημερομηνία αξιολόγησης** |  |
| **Τελικός βαθμός** |  |

**Σχόλια προς το φοιτητή**

**Αναλυτική αξιολόγηση**

|  |  |  |  |
| --- | --- | --- | --- |
| **Άσκηση** | **Περιγραφή** | **Ποσοστό %** | **Βαθμός** |
| 1.A | Δημιουργία Διαγράμματος Ακολουθίας | 10 |  |
| 1.B | Δημιουργία Διαγράμματος Κλάσεων | 5 |  |
| 2 | Χρήση Προτύπων Σχεδίασης | 10 |  |
| 3.Α | Κληρονομικότητα κλάσεων και υποσκέλιση μεθόδων | 15 |  |
| 3.Β | Υλοποίηση διαδικασιών | 25 |  |
| 3.Γ | Υλοποίηση προσομοίωσης με τα δεδομένα χρήστη | 10 |  |
| 3.Δ | Χειρισμός αλφαριθμητικών και εκτύπωση σε αρχείο κειμένου | 15 |  |
| 4 | Συμμόρφωση με τους κανόνες συγγραφής | 10 |  |
|  | Σύνολο | 100 |  |

**Εκφώνηση 1ης Άσκησης**

**Πληροφοριακό Σύστημα Πωλήσεων και Προμηθειών της εταιρίας Nature’s Liqueurs.**

Το σύστημα χρησιμοποιείται από τη startup εταιρία Nature’s Liqueurs, η οποία παράγει βιολογικά λικέρ για διαβητικούς και τα πουλά σε εμπόρους (π.χ. κάβες, super market, καταστήματα βιολογικών προϊόντων). Η εταιρία χρησιμοποιεί το σύστημα για να καταγράφει τις παραγγελίες των πελατών της (τηλεφωνικές και ηλεκτρονικές), για να κάνει παραγγελίες στους προμηθευτές της και για να καταγράφει τα αποθέματά της. **Από την ανάλυση του πεδίου προβλήματος όπως έχει γίνει έως τώρα, έχει παραχθεί το διάγραμμα Περιπτώσεων Χρήσης (ΠΧ), καθώς και η λεκτική προδιαγραφή της ΠΧ, όπως δίνονται παρακάτω.**

**Διάγραμμα Περιπτώσεων Χρήσης**

Diagram

Description automatically generated

**Λεκτική Περιγραφή Περίπτωση Χρήσης «Καταχώρηση Τηλεφωνικής Παραγγελίας»**

|  |  |  |  |
| --- | --- | --- | --- |
| **Όνομα ΠΧ** | | Καταχώρηση Τηλεφωνικής Παραγγελίας | |
| **Χειριστής** | | Υπάλληλος Πωλήσεων | |
| **Σύντομη Περιγραφή** | | | |
| Ο Υπάλληλος Πωλήσεων επιλέγει στο σύστημα τον πελάτη που έδωσε τηλεφωνική παραγγελία και καταχωρεί εκ μέρους του μια νέα παραγγελία βιολογικών λικέρ Nature’s Liqueurs | | | |
| **Ροή Γεγονότων** | | | |
| **Βασική Ροή** | | | |
| 1 | Ο Υπάλληλος Πωλήσεων επιλέγει στην Αρχική Οθόνη «Καταχώρηση Νέας Παραγγελίας» | | |
| 2 | Το σύστημα εμφανίζει την οθόνη επιλογής Πελάτη | | |
| 3 | Ο Υπάλληλος Πωλήσεων εισάγει το ΑΦΜ του Πελάτη | | |
| 4 | Το σύστημα εντοπίζει τον Πελάτη στον Κατάλογο Πελατών και εμφανίζει τα στοιχεία του στην οθόνη Καταχώρησης Παραγγελίας **[ΕΝΑΛΛΑΚΤΙΚΗ ΡΟΗ Α] [ΕΝΑΛΛΑΚΤΙΚΗ ΡΟΗ Β][ΕΝΑΛΛΑΚΤΙΚΗ ΡΟΗ Α] [ΕΝΑΛΛΑΚΤΙΚΗ ΡΟΗ Β]** | | |
| 5 | Ο Υπάλληλος Πωλήσεων επιλέγει προϊόν και ποσότητα και πατάει το κουμπί “Προσθήκη” | | |
| 6 | Το σύστημα καταχωρεί στην Παραγγελία προϊόν και ποσότητα  (τα βήματα 5 και 6 επαναλαμβάνονται για όλα τα είδη της παραγγελίας) | | |
| 7 | Ο Υπάλληλος Πωλήσεων επιλέγει «Καταχώρηση Παραγγελίας» | | |
| 8 | Το σύστημα καλεί την ΠΧ «Check Financial Capacity» για να ελεγχθεί το ιστορικό συναλλαγών με το συγκεκριμένο Πελάτη | | |
| 9 | Το σύστημα καλεί την ΠΧ «Check Supplies» για να ελεγχθεί αν υπάρχουν τα απαιτούμενα αποθέματα για τη συγκεκριμένη Παραγγελία | | |
| 10 | Το σύστημα καταχωρεί την Παραγγελία στον Κατάλογο Παραγγελιών και στις παραγγελίες του Πελάτη που την υπέβαλε και εμφανίζει μήνυμα επιτυχούς καταχώρησης **[ΕΝΑΛΛΑΚΤΙΚΗ ΡΟΗ Γ]** **[ΕΝΑΛΛΑΚΤΙΚΗ ΡΟΗ Δ]** | | |
| 11 | Ο Υπάλληλος Πωλήσεων επιβεβαιώνει το μήνυμα | | |
| 12 | Το σύστημα εμφανίζει την Αρχική Οθόνη | | |
| **Εναλλακτικές Ροές** | | | |
| **Εναλλακτική Ροή Α**: ΔΕ ΒΡΕΘΗΚΕ ΤΟ ΑΦΜ – ΑΚΥΡΩΣΗ ΚΑΤΑΧΩΡΗΣΗΣ | | | |
| 1 | Το σύστημα δεν εντόπισε Πελάτη με το δοσμένο ΑΦΜ και εμφανίζει ενημερωτικό μήνυμα στον Υπάλληλο Πωλήσεων | | |
| 2 | Ο Υπάλληλος Πωλήσεων επιλέγει να ακυρώσει τη διαδικασία (επιλέγει «Ακύρωση») | | |
| 3 | Η εκτέλεση επιστρέφει στο βήμα 2 της βασικής ροής | | |
| **Εναλλακτική Ροή Β**: ΔΕΝ ΒΡΕΘΗΚΕ ΤΟ ΑΦΜ – ΚΑΤΑΧΩΡΗΣΗ ΝΕΟΥ ΠΕΛΑΤΗ | | | |
| 1 | Το σύστημα δεν εντόπισε Πελάτη με το δοσμένο ΑΦΜ και εμφανίζει ενημερωτικό μήνυμα στον Υπάλληλο Πωλήσεων | | |
| 2 | Ο Υπάλληλος Πωλήσεων επιλέγει να καταχωρήσει ένα νέο Πελάτη | | |
| 3 | Καλείται η ΠΧ «Add New Customer» | | |
| **Εναλλακτική Ροή Γ**: ΑΠΟΤΥΧΙΑ ΕΛΕΓΧΟΥ ΑΠΟΘΕΜΑΤΟΣ | | | |
| 1 | Το σύστημα διαπίστωσε ότι δεν υπάρχουν τα απαραίτητα αποθέματα για τα είδη της παραγγελίας και εμφανίζει σχετικό ενημερωτικό μήνυμα ακύρωσης της καταχώρησης παραγγελίας | | |
| 2 | Ο Υπάλληλος Πωλήσεων επιβεβαιώνει το μήνυμα | | |
| 3 | Το σύστημα επιστρέφει στην Αρχική Οθόνη | | |
| **Εναλλακτική Ροή Δ**: ΒΕΒΑΡΗΜΕΝΟ ΙΣΤΟΡΙΚΟ ΣΥΝΑΛΛΑΓΩΝ | | | |
| 1 | Το σύστημα μετά τον έλεγχο διαπιστώνει ότι ο συγκεκριμένος Πελάτης έχει βεβαρημένο ιστορικό προηγούμενων συναλλαγών και εμφανίζει μήνυμα ακύρωσης της καταχώρησης παραγγελίας | | |
| 2 | Ο Υπάλληλος Πωλήσεων επιβεβαιώνει το μήνυμα | | |
| 3 | Το σύστημα επιστρέφει στην Αρχική Οθόνη | | |
| **Προϋποθέσεις** | | | Καμία προϋπόθεση |
| **Μετασυνθήκες** | | | Χαρακτηρισμός μιας παραγγελίας ως «Under processing»  Κλήση της ΠΧ «Check Financial Capacity»  Κλήση της ΠΧ «Add New Customer»  Κλήση της ΠΧ «Έλεγχος Αποθέματος» |
| **Εξαιρέσεις (δεν συμπεριλαμβάνεται στο διάγραμμα ακολουθίας)** | | | - |

**Ερώτημα Α – Δημιουργία Διαγράμματος Ακολουθίας**

**Καλείστε να δημιουργήσετε το διάγραμμα ακολουθίας για την ΠΧ: «Καταχώρηση Τηλεφωνικής Παραγγελίας» απεικονίζοντας μόνο τη Βασική Ροή και την Εναλλακτική Ροή Β.** Από το διάγραμμα ΠΧ, παρακαλείστε να επικεντρωθείτε στις σχέσεις της ΠΧ «Καταχώρηση Τηλεφωνικής Παραγγελίας» (Submit Telephone Order).

|  |
| --- |
| ***Μαθησιακά Αποτελέσματα***  Στην άσκηση 1.A. θα σας δοθεί η δυνατότητα να κατανοήσετε την κατασκευή του «Διαγράμματος Ακολουθίας». Πιο συγκεκριμένα θα μπορέσετε:   * να περιγράψετε το Διάγραμμα Ακολουθίας, * να περιγράψετε τα μέλη που αυτό περιλαμβάνει, * να σχεδιάσετε τα σύμβολα που αυτό περιλαμβάνει, * να αντιστοιχίσετε τα μέλη με τα σύμβολα που αυτό περιλαμβάνει, * να συνδέσετε την λειτουργικότητα του κάθε συμβάντος με την έννοια της ιχνηλασιμότητας |

**Ερώτημα B – Δημιουργία Διαγράμματος Κλάσεων**

Με βάση το διάγραμμα ακολουθίας της άσκησης 1.Α, να σχεδιάσετε το λεπτομερές Διάγραμμα Κλάσεων. *Σημειώνουμε ότι το παραχθέν διάγραμμα κλάσεων θα πρέπει να είναι σε πλήρη συμφωνία με το διάγραμμα ακολουθίας*. Περιγράψτε τις κλάσεις του συστήματος, τις αρμοδιότητες τους (μέθοδοι και ιδιότητες), τις σχέσεις τους με άλλες κλάσεις και επεξηγήσετε γιατί επιλέξατε κάποια συγκεκριμένη σχέση. Όπου θεωρείτε ότι απαιτείται, να κάνετε τις αναγκαίες παραδοχές ώστε το διάγραμμα να είναι όσο το δυνατόν πλήρες.

|  |
| --- |
| ***Μαθησιακά Αποτελέσματα***  Στην άσκηση 1.B θα σας δοθεί η δυνατότητα να κατανοήσετε την κατασκευή του «Διαγράμματος Κλάσεων». Πιο συγκεκριμένα θα μπορέσετε:   * να παραθέσετε τα κύρια μέλη που αυτό περιλαμβάνει, * να παραθέσετε τα είδη των σχέσεων μεταξύ των κλάσεων, * να αντιστοιχίσετε το κάθε είδος σχέσης με το σύμβολο που την απεικονίζει, * να περιγράψετε τον τρόπο σύνταξης των ονομάτων των κλάσεων, * να περιγράψετε τον τρόπο σύνταξης των ιδιοτήτων των κλάσεων, * να περιγράψετε τον τρόπο σύνταξης των μεθόδων των κλάσεων και * να περιγράψετε με παράδειγμα το κάθε είδος σχέσης μεταξύ των κλάσεων. |

**Εκφώνηση 2ης Άσκησης**

Χρησιμοποιείστε το πρότυπο σχεδίασης (design pattern) Visitor και υλοποιήστε την παρακάτω απαίτηση μέσω διαγράμματος κλάσεων. Επεξηγείστε τις κλάσεις, τα πεδία, τις μεθόδους και τις συσχετίσεις μεταξύ των κλάσεων καθώς και τον τρόπο που ενσωματώσατε το πρότυπο στη σχεδίασή σας. Το πρότυπο σχεδίασης Visitor (δείτε πληροφορίες για το Visitor Design Pattern στο <https://refactoring.guru/design-patterns/visitor>) έχει σα στόχο την αναπαράσταση μιας λειτουργίας που πρόκειται να εφαρμοστεί στα στοιχεία μιας δομής αντικειμένων. Το πρότυπο επιτρέπει τον ορισμό μιας νέας λειτουργίας χωρίς την τροποποίηση των κλάσεων των στοιχείων στα οποία επιδρά.

***Απαίτηση***: Σχεδιάστε τα απαραίτητα αντικείμενα μιας εφαρμογής μισθοδοσίας σε ένα ιδιωτικό σχολείο όπου καταγράφεται η μηνιαία αμοιβή των εργαζομένων, το ποσό των επιδομάτων και οι μέρες αδειών του προσωπικού. Θεωρείστε πως το σχολείο έχει μόνο Διοικητικούς Υπαλλήλους και Εκπαιδευτικούς. Στους δεύτερους μόνο δίνεται η δυνατότητα να έχουν και υπερωριακή αμοιβή (overtime). Είναι λογικό πως ανάλογα με την εκάστοτε νομοθεσία είναι απαίτηση να τροποποιείται η μισθοδοτική πολιτική αλλά και οι άδειες σε τακτά χρονικά διαστήματα. Σχεδιάστε την απαίτηση αυτή αξιοποιώντας το πρότυπο σχεδίασης Visitor.

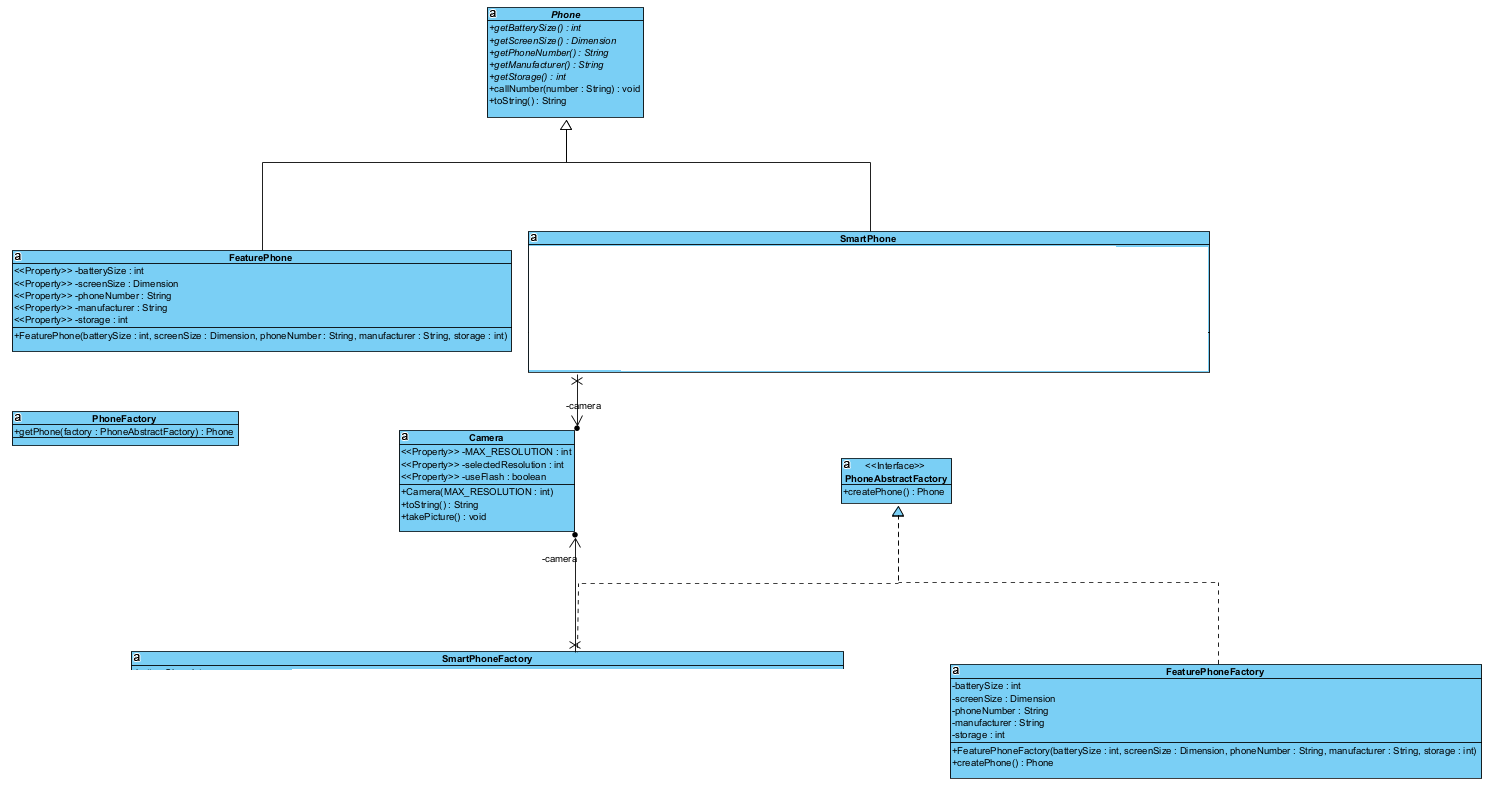
|  |
| --- |
| ***Μαθησιακά Αποτελέσματα***  Στην άσκηση 2 θα σας δοθεί η δυνατότητα να κατανοήσετε την εφαρμογή ενός προτύπου σχεδίασης. Πιο συγκεκριμένα θα μπορέσετε:   * να ταυτίσετε τους ρόλους ενός προτύπου σχεδίασης με κλάσεις ενός πραγματικού παραδείγματος, * να κατανοήσετε τη λειτουργία του προτύπου, * να περιγράψετε τον τρόπο με τον οποίο προσθέτουμε λειτουργικότητα σε ένα πρότυπο, * να κατανοήσετε τον τρόπο λειτουργίας της αρχής ανοιχτής-κλειστής σχεδίασης |

**Εκφώνηση 3ης Άσκησης**

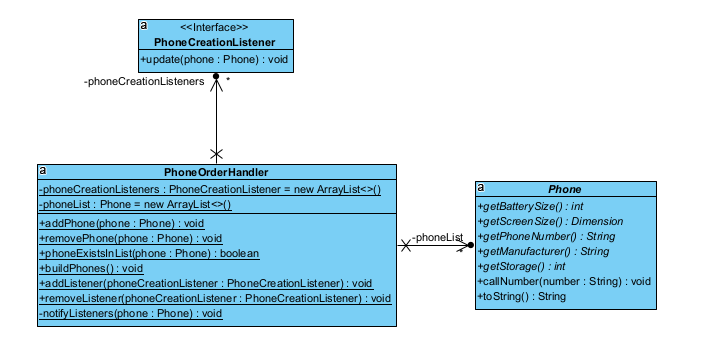
Σας δίνεται το ημιτελές Java Project “plh24ge2.zip”, το οποίο διαθέτει κλάσεις και διεπαφές χωρισμένα σε 3 πακέτα, σχετικά με ένα απλοποιημένο σύστημα προσομοίωσης γραμμής παραγωγής κινητών τηλεφώνων και πελατών.

Χρησιμοποιήστε τις δοθείσες κλάσεις / διεπαφές και συμπληρώστε ό,τι κρίνετε απαραίτητο (νέες κλάσεις / διεπαφές, πεδία, μεθόδους) ώστε να υλοποιήσετε το εν λόγω Project, απαντώντας στα ερωτήματα που ακολουθούν.

Περιγραφή: Στο εν λόγω project υλοποιούνται δύο πρότυπα σχεδίασης, συγκεκριμένα το “Abstract Factory” (<https://refactoring.guru/design-patterns/abstract-factory>) και το “Observer” (<https://refactoring.guru/design-patterns/observer>). Το πρότυπο Abstract Factory χρησιμοποιείται, τροποποιημένο, για την παραγωγή κινητών τηλεφώνων οποιουδήποτε τύπου, όπου στο συγκεκριμένο project εξειδικευόμαστε σε Featurephones και Smartphones (και γενικά μπορεί να χρησιμοποιηθεί σε έναν μεγάλο αριθμό αντίστοιχων προβλημάτων/προδιαγραφών, όπου από γενικούς τύπους θέλουμε να φτάσουμε στην κατασκευή ειδικότερων τύπων). Η εν λόγω χρήση του προτύπου στο project αναπαρίσταται με τη χρήση διαγράμματος κλάσεων στην επόμενη εικόνα (πακέτο eap.abstractfactory). Τα περιεχόμενα των κλάσεων SmartPhone και SmartPhoneFactory δεν φαίνονται στην εικόνα λόγω του ότι αποτελούν ζητούμενα.



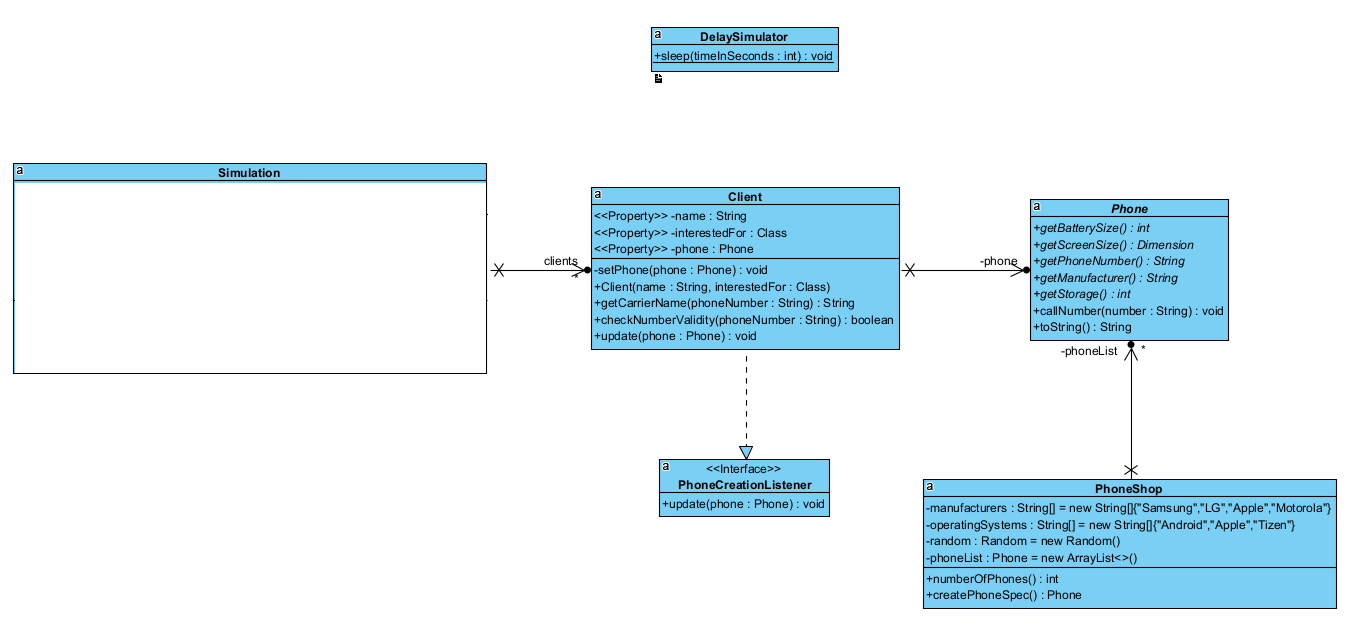
Το πρότυπο Observer χρησιμοποιείται, τροποποιημένο, για την «εγγραφή» των πελατών ως «ενδιαφερόμενους» για την απόκτηση κινητών τηλεφώνων, με αποτέλεσμα την άμεση ενημέρωσή τους όποτε προκύψει ένα τηλέφωνο. Η εν λόγω χρήση του προτύπου στο project αναπαρίσταται με τη χρήση διαγράμματος κλάσεων στην επόμενη εικόνα (πακέτο eap.observer).



Και για τα 2 παραπάνω πρότυπα που αναφέρονται δεν χρειάζεται να υλοποιήσετε νέα αρχεία, είναι ολοκληρωμένα σχεδιαστικά, σχετικά με τα πρότυπα που εξυπηρετούν. Θα χρειαστεί να προσθέσετε κώδικα, όπου ζητείται εντός των κλάσεων που τα χρησιμοποιούν, σύμφωνα με τα ερωτήματα της άσκησης.

Λογική: Η λογική εκτέλεσης του εν λόγω project έχει ως εξής: Το πρόγραμμα (εκκίνηση του προγράμματος από τη μέθοδο main της κλάσης Simulation, πακέτο eap.simulate) αρχικά ζητά από τον χρήστη να δηλώσει τον αριθμό των κινητών τηλεφώνων που πρόκειται να δοθούν στη συνέχεια στη γραμμή παραγωγής, καθώς και τον αριθμό των πελατών που θα δημιουργηθούν και θα αναμείνουν για την ενδεχόμενη παραγωγή των τηλεφώνων που τους ενδιαφέρουν. Οι πελάτες, με τυχαίο τρόπο, ενδιαφέρονται είτε για Featurephone, είτε για Smartphone (ένα ο καθένας). Εν συνεχεία, δημιουργείται ένα αντικείμενο της κλάσης “PhoneShop”, το οποίο είναι ουσιαστικά υπεύθυνο για τη δημιουργία των προδιαγραφών, οι οποίες πρέπει να είναι τυχαίες, των κινητών που θέλουμε να παραχθούν. Οι εν λόγω προδιαγραφές τηλεφώνων μεταφέρονται στην κλάση “PhoneOrderHandler”, η οποία πρόκειται να εκτελέσει τις κατασκευές αυτών, καθώς και να ενημερώσει τους ενδιαφερόμενους πελάτες (χρήση της μεθόδου “buildPhones”). Ο κάθε πελάτης ο οποίος έχει τελικά λάβει ένα κινητό τηλέφωνο, το χρησιμοποιεί, στην περίπτωση Featurephone κάνοντας μια κλήση σε έναν αριθμό, ενώ στην περίπτωση του Smartphone τραβώντας μια φωτογραφία. Το πρόγραμμα ολοκληρώνεται αναγράφοντας τον αριθμό των πελατών που, ενδεχομένως, έμειναν χωρίς να πάρουν ένα κινητό τηλέφωνο.

Ακολουθεί και το διάγραμμα κλάσεων του πακέτου eap.simulate, για λόγους πληρότητας. Τα περιεχόμενα της κλάσης Simulate δεν φαίνονται στην εικόνα λόγω του ότι αποτελούν ζητούμενα.



Σημειώνεται ότι τα πεδία εντός των κλάσεων (χαρακτηριστικά και λειτουργίες) στα δοθέντα διαγράμματα κλάσεων ενδεχομένως να είναι διαφορετικά στην τελική, δική σας υλοποίηση, χωρίς αυτό να είναι λάθος εφόσον υλοποιήσετε σωστά τα δεδομένα που σας έχουν δοθεί από τις εκφωνήσεις των ερωτημάτων.

Στη συνέχεια ακολουθεί ένας αριθμός από απαιτήσεις και παραδοχές που αφορούν στο συγκεκριμένο project:

1. Η κλάση “Simulation” περιέχει τη μέθοδο main, από την οποία ξεκινάει την εκτέλεσή του το πρόγραμμα και κάνει την προσομοίωση του project:
   1. Διαθέτει έναν πίνακα (names) με τα ονόματα χρηστών (όσων) που θα χρησιμοποιηθούν (μπορείτε να αφήσετε τον πίνακα ως έχει)
   2. Ζητά από τον χρήση να εισάγει τον αριθμό των τηλεφώνων που πρόκειται να δοθούν ως παραγγελίες για να φτιαχτούν
   3. Ζητά από τον χρήστη να εισάγει τον αριθμό των πελατών που θα περιμένουν για να πάρουν ένα κινητό (μέγιστος αριθμός 10)
   4. Στη συνέχεια προσομοιώνει το κατάστημα PhoneShop, καθώς και τους πελάτες, επιλέγοντας ένα τυχαίο όνομα για τον κάθε πελάτη. Επιπλέον, όπως προαναφέρθηκε, σε κάθε πελάτη θα πρέπει να αντιστοιχηθεί ένα είδος τηλεφώνου (FeaturePhone ή SmartPhone) με τυχαίο τρόπο. Κάθε ένας πελάτης (άρα κάθε όνομα) μπορεί να λάβει ένα τηλέφωνο
   5. Τέλος εκτυπώνει ως αναφορά το πόσοι πελάτες δεν έλαβαν κάποιο τηλέφωνο (σύμφωνα με τα χαρακτηριστικά που ζήτησαν, ή αν ήταν περισσότεροι από τα τηλέφωνα που παραγγέλθηκαν)
2. Η γραμμή παραγωγής κινητών τηλεφώνων υποστηρίζει στην τρέχουσα υλοποίηση τα Featurephones (παλαιότερου τύπου κινητά τηλέφωνα, χωρίς λειτουργικό και κάμερα) και τα Smartphones (σύγχρονα κινητά τηλέφωνα, με λειτουργικό σύστημα και κάμερα). Συγκεκριμένα, τα Smartphones έχουν όλες τις δυνατότητες των Featurephones, ενώ επιπλέον διαθέτουν λειτουργικό σύστημα και κάμερα.
3. Τα αρχεία κλάσεων/διεπαφών που αφορούν στο πρότυπο Abstract Factory βρίσκονται στο πακέτο “eap.abstractfactory”, τα αρχεία που αφορούν στο πρότυπο Observer βρίσκονται στο πακέτο “eap.observer”, ενώ τα υπόλοιπα αρχεία, που αφορούν στις προσομοιώσεις, στο πακέτο “ eap.simulate”.
4. Στο project, όπου είναι δυνατόν, χρησιμοποιούνται private fields σε συνδυασμό με getters/setters (όπου χρειάζονται), για τον καλύτερο και ασφαλέστερο χειρισμό των δεδομένων.
5. Η κλάση “DelaySimulator” διαθέτει τη στατική μέθοδο sleep(int timeInSeconds) με την οποία προσομοιώνει μια χρονική καθυστέρηση για τον δοθέντα χρόνο σε δευτερόλεπτα, χωρίς τη χρήση νημάτων (threads).
6. Η διεπαφή “PhoneCreationListener” χρησιμοποιείται αφενός στην κλάση “PhoneOrderHandler”, όσο και σε όποια άλλη κλάση θέλει να δηλώσει «ενδιαφέρον παρακολούθησης/παρατήρησης» πάνω σε συγκεκριμένα αντικείμενα. Ως εκ τούτου, οι ενδιαφερόμενοι για κινητά τηλέφωνα Clients, υλοποιούν το εν λόγω Interface, ώστε στη συνέχεια να μπορούν να ενημερώνονται, μέσω της κλάσης “PhoneOrderHandler”, για όποτε αυτό συμβεί.
7. Η κλάση “PhoneOrderHandler” είναι υπεύθυνη για τον χειρισμό των παραγγελιών ως προς την υλοποίηση των παραγγελιών, καθώς και για την ενημέρωση των ενδιαφερόμενων πελατών. Εντός της κλάσης, η δημιουργία ενός νέου τηλεφώνου γίνεται με την κλήση της μεθόδου “PhoneFactory.getPhone”. Επιπλέον, διαθέτει τη μέθοδο “printCreatedPhoneSpecsToTextFile”, η οποία σώζει τα βασικά χαρακτηριστικά των τηλεφώνων που παράχθηκαν σε αρχείο κειμένου.
8. Η κλάση “PhoneShop” είναι υπεύθυνη ώστε οι προδιαγραφές των κινητών τηλεφώνων (όσα πεδία διαθέτουν) να παράγονται με τυχαίο τρόπο. Επιπλέον, διατηρεί και μια λίστα στην οποία καταγράφονται τα τηλέφωνα για τα οποία παρήχθησαν προδιαγραφές (και πρόκειται να κατασκευαστούν).
9. Η κλάση “Client” που αφορά στον πελάτη, αναλαμβάνει μεταξύ άλλων, όποτε ειδοποιείται για την παραγωγή νέου κινητού τηλεφώνου να κάνει τα εξής:
   1. Να ελέγχει αν το κινητό που παράχθηκε είναι τέτοιου τύπου που να ενδιαφέρει τον πελάτη (Featurephone ή Smartphone). Έχοντας στη διάθεσή του το τηλέφωνο που τον ενδιαφέρει, στη περίπτωση που είναι Featurephone πραγματοποιεί μια κλήση σε ένα τηλέφωνο (δεν χρειάζεται να είναι τυχαίο), ενώ στην περίπτωση που διαθέτει Smartphone τραβάει μια φωτογραφία σε ανάλυση και χρήση φλας της επιλογής του (δεν χρειάζεται να είναι τυχαία) (μέθοδος usePhone)
   2. Εφόσον ο πελάτης βρίσκει το τηλέφωνο που τον ενδιαφέρει, πρέπει να αφαιρεί το τηλέφωνο από το σύνολο των διαθέσιμων τηλεφώνων (για να μην το πάρει κάποιος άλλος), να απεγγράφεται από τη λίστα των ενδιαφερόμενων πελατών (ώστε να μην ενημερωθεί ξανά), καθώς και να αφαιρείται από το σύνολο των πελατών που αναμένουν για κινητό (μέθοδος update)
   3. Να ελέγχει και να εκτυπώνει κατά πόσο ο τηλεφωνικός αριθμός του τηλεφώνου που έλαβε είναι έγκυρος (οι πληροφορίες βρίσκονται στην περιγραφή της μεθόδου “checkNumberValidity”)
   4. Να ελέγχει και να εκτυπώνει το όνομα του παρόχου κινητής τηλεφωνίας για τον τηλεφωνικό αριθμό που έλαβε (πληροφορίες στις περιγραφές του πίνακα «carriers» και της μεθόδου ”getCarrierName” αντίστοιχα

Ενδεικτική εκτέλεση του προγράμματος (ενδεικτικά δοκιμαστικά δεδομένα):

Please enter number of phones to be ordered:3

Please enter number of clients waiting for new phones (MAX:10):2

3 phone orders have been placed by the PhoneShop!

2 clients are waiting to buy a new phone!

##########-Phone Build Process Started-##########

$$$$-New Phone-$$$$

Battery(mAh): 4400

Screen: java.awt.Dimension[width=2928,height=1581]

Number: +306981066085

Manufacturer: Apple

Storage(GB): 144

Camera: 72 MP

Operating System: Tizen

Has been created

$$$$-----------$$$$

$$$$-New Phone-$$$$

Battery(mAh): 1445

Screen: java.awt.Dimension[width=1951,height=2214]

Number: +306998519113

Manufacturer: Samsung

Storage(GB): 12

Camera: 90 MP

Operating System: Android

Has been created

$$$$-----------$$$$

$$$$-New Phone-$$$$

Battery(mAh): 3067

Screen: java.awt.Dimension[width=1032,height=2126]

Number: +306981911515

Manufacturer: Motorola

Storage(GB): 5

Has been created

$$$$-----------$$$$

Hi, I am Eleni(FeaturePhone) and I got my new phone!

Phone specs:

Battery(mAh): 3067

Screen: java.awt.Dimension[width=1032,height=2126]

Number: +306981911515

Manufacturer: Motorola

Storage(GB): 5

Phone number valid:true

Phone number carrier:Cosmote

------------------------

Phone call from +306981911515 to +30123456789

Call in progress...

Call terminated.

------------------------

########-Phone Specs saved to output.txt-########

###########-Phone Build Process Ended-###########

%%%%%----Report----%%%%%

1 clients did not get a phone...

%%%%%--------------%%%%%

Ενδεικτικό αρχείο output.txt, για την παραπάνω εκτέλεση (από τη μέθοδο printCreatedPhoneSpecsToTextFile):

1.

Battery(mAh): 4400

Screen: java.awt.Dimension[width=2928,height=1581]

Number: +306981066085

Manufacturer: Apple

Storage(GB): 144

Camera: 72 MP

Operating System: Tizen

2.

Battery(mAh): 1445

Screen: java.awt.Dimension[width=1951,height=2214]

Number: +306998519113

Manufacturer: Samsung

Storage(GB): 12

Camera: 90 MP

Operating System: Android

3.

Battery(mAh): 3067

Screen: java.awt.Dimension[width=1032,height=2126]

Number: +306981911515

Manufacturer: Motorola

Storage(GB): 5

**Ερώτημα Α – Κληρονομικότητα κλάσεων και υποσκέλιση (override) μεθόδων**

Με βάση τις απαιτήσεις/προδιαγραφές, να υλοποιήσετε τον κώδικα που λείπει για τις κλάσεις SmartPhone και SmartPhoneFactory, λαμβάνοντας υπόψη και τα όποια σχόλια υπάρχουν εντός του κώδικα που σας δίνεται.

|  |
| --- |
| ***Μαθησιακά Αποτελέσματα***  Στην άσκηση 3.Α. θα μάθετε:   * να ορίσετε με ακρίβεια την έννοια της κλάσης, * να κατανοήσετε τα βασικά δομικά στοιχεία των κλάσεων και των διεπαφών * να χρησιμοποιήσετε κλάση η οποία υλοποιεί μια διεπαφή * να κατανοήσετε την έννοια της μεθόδου * να εξηγήσετε τις βασικές διαφορές ενός κατασκευαστή από μια απλή μέθοδο, * να υλοποιήσετε κληρονομικότητα και υποσκέλιση |

**Απάντηση**

*Εισάγετε τον κώδικα των δύο κλάσεων Java που φτιάξατε. Φροντίστε ο κώδικας να είναι μορφοποιημένος κατάλληλα και να είναι ευανάγνωστος. Θα πρέπει να* ***υπάρχει τεκμηρίωση με μορφή σχολίων****.*

*Εάν δεν έχετε δώσει απάντηση γράψτε με κεφαλαία γράμματα, ΔΕΝ ΑΠΑΝΤΗΘΗΚΕ.*

*Εάν εν γνώση σας δίνετε ελλιπή απάντηση γράψτε με κεφαλαία γράμματα, ΕΛΛΙΠΗΣ ΑΠΑΝΤΗΣΗ.*

**Ερώτημα Β – Υλοποίηση διαδικασιών**

Με βάση τις απαιτήσεις/προδιαγραφές, τον κώδικα που έχετε λάβει, καθώς και τα σχόλια εντός των αρχείων κώδικα Java, να υλοποιήστε τις παρακάτω μεθόδους:

* 1. Στην κλάση **Client** τη μέθοδο “**update”** (μελετήστε την απαίτηση/παραδοχή 9.b)μέσα στην οποία θα καλέσετεκαι τις μεθόδους **“getCarrierName”** και **“checkNumberValidity”,** οι υλοποιήσεις των οποίων ζητούνται στο ερώτημα Δ.
  2. Στην κλάση **PhoneShop** τη μέθοδο “**createPhoneSpec”,** η οποία παράγει τα χαρακτηριστικά των τηλεφώνων (ανάλογα με το είδος του τηλεφώνου) με τυχαίο τρόπο για κάθε τηλέφωνο και ενημερώνει και τη λίστα τηλεφώνων με τις προδιαγραφές τους (λίστα phoneList).

|  |
| --- |
| ***Μαθησιακά Αποτελέσματα***  Στην άσκηση 3.Β. θα μάθετε:   * Να υλοποιείτε μεθόδους ως υλοποίηση διεπαφών (Interfaces) * Να προσθαφαιρείτε στοιχεία από λίστες αντικειμένων * Να παράγετε αντικείμενα με τυχαία χαρακτηριστικά * Να κατανοείτε τη λογική πίσω από πρότυπα σχεδίασης της Java |

**Απάντηση**

*Εισάγετε τον κώδικα των δύο μεθόδων Java που φτιάξατε. Φροντίστε ο κώδικας να είναι μορφοποιημένος κατάλληλα και να είναι ευανάγνωστος. Θα πρέπει να* ***υπάρχει τεκμηρίωση με μορφή σχολίων****.*

*Εάν δεν έχετε δώσει απάντηση γράψτε με κεφαλαία γράμματα, ΔΕΝ ΑΠΑΝΤΗΘΗΚΕ.*

*Εάν εν γνώση σας δίνετε ελλιπή απάντηση γράψτε με κεφαλαία γράμματα, ΕΛΛΙΠΗΣ ΑΠΑΝΤΗΣΗ.*

**Ερώτημα Γ – Υλοποίηση προσομοίωσης με τα δεδομένα χρήστη**

Να συμπληρώσετε, όπου απαιτείται, τον κώδικα εντός της κλάσης Simulation με βάση τις απαιτήσεις/προδιαγραφές (απαίτηση/παραδοχή 1), τον κώδικα που έχετε λάβει, την εκτύπωση της **«Ενδεικτικής εκτέλεσης του προγράμματος»,** καθώς και τα σχόλια εντός των αρχείων κώδικα Java.

|  |
| --- |
| ***Μαθησιακά Αποτελέσματα***  Στην άσκηση 3.Γ. θα μάθετε:   * Να κάνετε προσομοίωση ζητώντας input από τον χρήστη * Να δημιουργείτε και να χειρίζεστε πίνακες ή/και λίστες αντικειμένων |

**Απάντηση**

*Εισάγετε τον κώδικα της κλάσης Simulation Java που φτιάξατε. Φροντίστε ο κώδικας να είναι μορφοποιημένος κατάλληλα και να είναι ευανάγνωστος. Θα πρέπει να* ***υπάρχει τεκμηρίωση με μορφή σχολίων****.*

*Εάν δεν έχετε δώσει απάντηση γράψτε με κεφαλαία γράμματα, ΔΕΝ ΑΠΑΝΤΗΘΗΚΕ.*

*Εάν εν γνώση σας δίνετε ελλιπή απάντηση γράψτε με κεφαλαία γράμματα, ΕΛΛΙΠΗΣ ΑΠΑΝΤΗΣΗ.*

**Ερώτημα Δ – Χειρισμός αλφαριθμητικών και εκτύπωση σε αρχείο κειμένου**

Με βάση τις απαιτήσεις/προδιαγραφές, τον κώδικα που έχετε λάβει, καθώς και τα σχόλια εντός των αρχείων κώδικα Java, να υλοποιήστε τις παρακάτω μεθόδους:

1. Στην κλάση **Client** τις μεθόδους **“getCarrierName”** και **“checkNumberValidity”** (χειρισμός αλφαριθμητικών)
2. Στην κλάση **PhoneOrderHandler** τη μέθοδο “ **printCreatedPhoneSpecsToTextFile”** η οποία εκτυπώνει τα βασικά χαρακτηριστικά (μόνο από την κλάση Phone) όσων κινητών παράχθηκαν, στο αρχείο “output.txt”

|  |
| --- |
| ***Μαθησιακά Αποτελέσματα***  Στην άσκηση 3.Δ. θα μάθετε:   * Να γράφετε δεδομένα σε αρχεία, * Να χειρίζεστε αλφαριθμητικά |

**Απάντηση**

*Εισάγετε τον κώδικα των μεθόδων σε Java που φτιάξατε. Φροντίστε ο κώδικας να είναι μορφοποιημένος κατάλληλα και να είναι ευανάγνωστος. Στο φάκελο με το Project υπάρχουν τα αρχεία της βιβλιοθήκης GSON την οποία μπορείτε να χρησιμοποιήσετε προαιρετικά για να παράγεται το αποτέλεσμα. Εναλλακτικά μπορείτε να παράγεται το αποτέλεσμα με χρήση String τα οποία θα γράφετε στο αρχείο. Θα πρέπει να* ***υπάρχει τεκμηρίωση με μορφή σχολίων****.*

*Εάν δεν έχετε δώσει απάντηση γράψτε με κεφαλαία γράμματα, ΔΕΝ ΑΠΑΝΤΗΘΗΚΕ.*

*Εάν εν γνώση σας δίνετε ελλιπή απάντηση γράψτε με κεφαλαία γράμματα, ΕΛΛΙΠΗΣ ΑΠΑΝΤΗΣΗ.*

**Υποδείξεις για τη συγγραφή της εργασίας**

1) Για την απάντηση της εργασίας θα πρέπει να χρησιμοποιηθεί το υπόδειγμα της εργασίας. Στο υπόδειγμα:

- Συμπληρώστε όλα τα στοιχεία με κίτρινο.

- Μην ξεχάσετε να δηλώσετε εάν η εργασία αποτελεί προϊόν αποκλειστικά δικής σας εργασίας.

- Ενσωματώστε τις απαντήσεις (διαγράμματα ή/και κώδικα) μετά το κάθε ερώτημα. Δεν θα πρέπει να κάνετε παραπομπές της μορφής «βλέπε αρχείο…».

- Αν δεν έχετε απαντήσει σε ένα ερώτημα γράψτε «**ΔΕΝ ΑΠΑΝΤΗΘΗΚΕ**».

- Αν απαντήσατε με ελλείψεις σε ένα ερώτημα γράψτε «**ΕΛΛΙΠΗΣ ΑΠΑΝΤΗΣΗ**».

2) Η συνεργασία στην ανάλυση της εργασίας επιτρέπεται, αλλά καλό είναι να αναφερθεί στον ειδικό χώρο στην πρώτη σελίδα της εργασίας. Η συνεργασία δεν πρέπει να οδηγεί σε από κοινού επίλυση και συγγραφή της εργασίας. Η υποβολή κοινών απαντήσεων από διαφορετικούς φοιτητές που συνεργάστηκαν δεν επιτρέπεται και θεωρείται ως **ΑΝΤΙΓΡΑΦΗ**. Οι απαντήσεις ελέγχονται, τόσο μεταξύ των φοιτητών του ιδίου τμήματος, όσο και μεταξύ φοιτητών διαφορετικών τμημάτων. Η αντιγραφή έχει ως αποτέλεσμα το **ΜΗΔΕΝΙΣΜΟ ΤΗΣ ΕΡΓΑΣΙΑΣ ΣΥΝΟΛΙΚΑ** και την παραπομπή των παραβατών στην Κοσμητεία της Σχολής Θετικών Επιστημών & Τεχνολογίας, σύμφωνα με τον εσωτερικό κανονισμό του ΕΑΠ.

3) Η εκπόνηση της εργασίας θα πρέπει να γίνει **αποκλειστικά και υποχρεωτικά** με το εργαλείο Visual Paradigm για τη UML και Netbeans για τη Java.

4) Η εργασία θα υποβληθεί στο σύστημα υποβολή εργασιών του study.eap.gr.

5) Ο φοιτητής θα πρέπει να στείλει την σε δύο αρχεία

* **Το 1ο αρχείο θα έχει όνομα PLH24\_2ERG\_EPITHETO\_ONOMA.doc** και είναι το παρόν αρχείο συμπληρωμένο με την απάντησή σας.
* **Το 2ο αρχείο είναι ένα συμπιεσμένο αρχείου zip ή rar** με όνομα PLH24\_2ERG\_EPITHETO\_ONOMA.<rar|zip>. Το συμπιεσμένο αρχείο θα πρέπει να αποτελείται από:

i. Ένα κατάλογο που θα περιέχει τα διαγράμματα που ζητά η κάθε άσκηση.

ii. Ο κατάλογος με τον κώδικα Java (άσκηση 3) θα πρέπει να περιλαμβάνει το project όπως αυτό δημιουργείται από το εργαλείο Netbeans και θα πρέπει να μπορεί να εκτελείται χωρίς αλλαγές από τον καθηγητή.

iii. Να γίνει χρήση λατινικών χαρακτήρων **ΑΠΟΚΛΕΙΣΤΙΚΑ** για την αποφυγή προβλημάτων με το moodle.

H εφαρμογή των παραπάνω κανόνων είναι **ΥΠΟΧΡΕΩΤΙΚΗ** και βαθμολογείται σύμφωνα με το αντίστοιχο κριτήριο αξιολόγησης. Η μη εφαρμογή **των παραπάνω κανόνων** μπορεί να οδηγήσει σε συνολική απόρριψη της εργασίας (π.χ. να δηλώσετε εάν η εργασία αποτελεί προϊόν αποκλειστικά δικής σας εργασίας, να μην αποστείλετε τον κώδικα σε netbeans, να μην υποβάλετε την εργασία ως doc αλλά ως pdf).

**ΚΑΛΗ ΕΠΙΤΥΧΙΑ!**