

# Αυτοματοποιημένη ενσωμάτωση μοτίβων σχεδίασης σε πηγαίο κώδικα Java

Αναστάσιος Λιόντος

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Τμήμα Μηχανικών Η/Υ και Πληροφορικής  
Πολυτεχνική Σχολή  
Πανεπιστήμιο Ιωαννίνων

Ιούλιος 2023

# ΑΦΙΕΡΩΣΗ

---

ΕΥΧΑΡΙΣΤΙΕΣ

---

# ΠΕΡΙΕΧΟΜΕΝΑ

---

Κατάλογος Σχημάτων	iii
Κατάλογος Πινάκων	iv
Περίληψη	v
Abstract	vi
<b>1 Εισαγωγή</b>	<b>1</b>
1.1 Σχεδιαστικά μοτίβα . . . . .	1
1.2 Στόχοι . . . . .	1
1.3 Δομή της διπλωματικής εργασίας . . . . .	2
<b>2 Σχετική Δουλειά</b>	<b>3</b>
2.1 Pattern Wizard . . . . .	3
2.2 Patternbox . . . . .	3
2.3 DESIGN PATTERN AUTOMATION TOOLKIT . . . . .	3
2.4 ALPHAWORKS DESIGN PATTERN TOOLKIT . . . . .	4
<b>3 Ανάλυση Απαιτήσεων</b>	<b>5</b>
3.1 Ιστορίες Χρήστη . . . . .	5
3.2 Περιπτώσεις χρήσης . . . . .	6
<b>4 Σχεδίαση και αρχιτεκτονική λογισμικού</b>	<b>16</b>
4.1 Πακέτα συστήματος . . . . .	16
4.2 Κλάσεις συστήματος . . . . .	16
4.2.1 Ανάλυση κλάσεων . . . . .	16
4.2.2 Διάγραμμα κλάσεων . . . . .	16
4.3 Κάρτες αρμοδιοτήτων και συνεργασιών κλάσεων . . . . .	16

<b>5 Έλεγχος</b>	<b>17</b>
5.1 Έλεγχος δομής σχεδιαστικών μοτίβων . . . . .	17
5.2 Έλεγχος μεθόδων . . . . .	17
5.3 Έλεγχος μεθόδων δημιουργίας πηγαίου κώδικα java . . . . .	17
<b>6 Οδηγός Χρήσης Design Pattern Builder</b>	<b>18</b>
6.1 Λειτουργίες χρήστη . . . . .	18
<b>7 Επίλογος</b>	<b>19</b>
7.1 Μελλοντικές επεκτάσεις . . . . .	19
<b>Βιβλιογραφία</b>	<b>20</b>

# ΚΑΤΑΛΟΓΟΣ ΣΧΗΜΑΤΩΝ

---

# ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

---

3.1	Ιστορίες χρήστη . . . . .	6
3.2	Επιλογή κατηγορίας μοτίβου. . . . .	7
3.3	Επιλογή μοτίβου. . . . .	7
3.4	Καθορισμός μεθόδων κλάσης. . . . .	8
3.5	Ονοματοδοσία κλάσης. . . . .	9
3.6	Καθορισμός πεδίων κλάσης. . . . .	10
3.7	Ονοματοδοσία διεπαφής. . . . .	11
3.8	Καθορισμός μεθόδων διεπαφής. . . . .	12
3.9	Εξαγωγή σκελετού επιλεγμένου μοτίβου. . . . .	13
3.10	Ακύρωση διαδικασίας. . . . .	13
3.11	Προσθήκη νέας κλάσης. . . . .	14
3.12	Καθορισμός υλοποιημένης διεπαφής. . . . .	15

# ΠΕΡΙΛΗΨΗ

---

Αναστάσιος Λιόντος, Δίπλωμα, Τμήμα Μηχανικών Η/Υ και Πληροφορικής, Πολυτεχνική Σχολή, Πανεπιστήμιο Ιωαννίνων, Ιούλιος 2023.

Αυτοματοποιημένη ενσωμάτωση μοτίβων σχεδίασης σε πηγαίο κώδικα Java.

Επιβλέπων: Απόστολος Ζάρρας, Καθηγητής.



# ABSTRACT

---

Anastasios Lontos, Diploma, Department of Computer Science and Engineering,  
School of Engineering, University of Ioannina, Greece, July 2023.

Automated incorporation of design patterns in Java source code.

Advisor: Apostolos Zarras, Professor.

# ΚΕΦΑΛΑΙΟ 1

## Εισαγωγή

### 1.1 Σχεδιαστικά μοτίβα

Ένα σχεδιαστικό μοτίβο ορίζει μια γενική λύση σε ένα συχνά εμφανιζόμενο πρόβλημα, γραμμένο ως πρότυπο ή ως σύνολο σχέσεων. Στην επιστήμη των υπολογιστών, αυτό μεταφράζεται σε ένα προγραμματιστικό μοτίβο ή ένα διάγραμμα αντικειμένων που αναπαριστά μια γνωστή λύση σε μια κοινή εργασία.

Όπως οι αλγόριθμοι, έτσι και τα πρότυπα σχεδίασης είναι μια δοκιμασμένη και αποδεκτή λύση- ως εκ τούτου, οι άνθρωποι τα αναπτύσσουν και τα προσαρμόζουν αντί να τα ανακαλύπτουν. Και τα δύο αντιπροσωπεύουν μια προσέγγιση ενός προβλήματος περισσότερο από την ειδική υλοποίηση της λύσης. Ωστόσο, διαφέρουν ως προς τον σκοπό τους: οι αλγόριθμοι βελτιστοποιούν το κόστος μιας λύσης, ενώ τα πρότυπα σχεδίασης βελτιστοποιούν τη σαφήνειά της.

### 1.2 Στόχοι

Αυτή η διπλωματική επιδιώκει την ανάπτυξη ενός εργαλείου που επιτρέπει που επιτρέπει σε αρχάριους προγραμματιστές να εισάγουν σχεδιαστικά μοτίβα στον κώδικά τους σε Java με ένα απλό βήμα. Το εργαλείο παρέχει μια διεπαφή, όπου ο χρήστης μπορεί να επιλέξει το επιθυμητό σχεδιαστικό μοτίβο, να κάνει τις απαραίτητες ρυθμίσεις και να εισάγει αυτόματα τον αντίστοιχο κώδικα στον δικό του κώδικα. Το εργαλείο περιλαμβάνει μια συλλογή από τα σχεδιαστικά μοτίβα που προτείνονται από την ομάδα των GOF, όπως το Singleton, το Factory, το Observer

και άλλα. Ο χρήστης έχει τη δυνατότητα να εξερευνήσει αυτήν τη συλλογή, να επιλέξει το επιθυμητό μοτίβο και να εισάγει αυτόματα τον απαιτούμενο κώδικα στον κώδικά του. Ο στόχος είναι να διευκολυνθεί ο αρχάριος προγραμματιστής στη χρήση σχεδιαστικών μοτίβων και να ενισχυθεί η προγραμματιστική του απόδοση και ακρίβεια, επιτρέποντάς του να εισάγει εύκολα τα απαραίτητα σχεδιαστικά μοτίβα στον κώδικά του, χωρίς την ανάγκη για χειροκίνητη υλοποίηση. Αυτή η προσαρμογή περιγράφει το εργαλείο σου ως ένα απλό εργαλείο που επιτρέπει στους χρήστες να εισάγουν τον κώδικα των σχεδιαστικών μοτίβων που επιθυμούν, χωρίς την ανάγκη για χειροκίνητη υλοποίηση. Συνολικά, ο στόχος είναι να δημιουργηθεί ένα εύχρηστο και βοηθητικό εργαλείο που θα επιτρέπει στους αρχάριους προγραμματιστές να αξιοποιούν τα σχεδιαστικά μοτίβα.

### 1.3 Δομή της διπλωματικής εργασίας

Η διπλωματική εργασία περιέχει 7 κεφάλαια. Το υπόλοιπο της διπλωματικής εργασίας περιλαμβάνει τα εξής κεφάλαια: Στο κεφάλαιο 2, γίνεται ανάλυση της παρελθοντικής δουλειάς που έχει γίνει και είναι σχετική με το εργαλείο που πραγματεύεται η παρούσα διπλωματική. Το κεφάλαιο 3, περιέχει τις ιστορίες χρήστη καθώς, και τις περιπτώσεις χρήσης του εργαλείου. Το κεφάλαιο 4, εξετάζει την αρχιτεκτονική του εργαλείου, καθώς παρουσιάζονται λεπτομερώς οι κλάσεις και τα πακέτα που αποτελούν το εργαλείο. Στο κεφάλαιο 5, περιέχονται λεπτομέρειες σχετικά με τον αυτοματοποιημένο έλεγχο του εργαλείου. Στο κεφάλαιο 6, παρουσιάζονται οι οδηγίες χρήσης του εργαλείου. Τέλος, στο κεφάλαιο 7 παρουσιάζονται οι μελλοντικές επεκτάσεις του εργαλείου. <sup>1</sup>

---

<sup>1</sup>[1]

# ΚΕΦΑΛΑΙΟ 2

## Σχετική Δουλειά

### 2.1 Pattern Wizard

Αυτό το εργαλείο μετατρέπει τον βασικό κώδικα σε μία μορφή κώδικα που χρησιμοποιεί ένα σχεδιαστικό πρότυπο, ένας προγραμματιστής μπορεί να αναλύσει τον τρόπο που λειτουργεί ένα μοτίβο, χωρίς να χρειάζεται να υλοποιήσει τον κώδικα. Το εργαλείο αυτό παρέχει τα μοτίβα Adapter, Abstract factory και Observer για την γλώσσα προγραμματισμού java.

### 2.2 Patternbox

Το εργαλείο αυτό δημιουργεί ένα ειδικό αρχείο το οποίο αναπαριστά το μοτίβο. Μέσα από το αρχείο μπορεί κάποιος να επιλέξει κάποιο στοιχείο του μοτίβου και να επιλέξει την τοποθεσία του αντικειμένου. Ένα αντικείμενο μπορεί να είναι είτε μία κλάση είτε μία διεπαφή. Αφού ο χρήστης ολοκληρώσει την διαδικασία, το εργαλείο παράγει ένα πρότυπο κώδικα.

### 2.3 DESIGN PATTERN AUTOMATION TOOLKIT

Το εργαλείο αυτό εμφανίζει τη χρήση προτύπων σχεδίασης σε πολλές γλώσσες. Διαθέτει επίσης έναν τρόπο δημιουργίας νέων προτύπων σχεδίασης (εκτός από τα πρότυπα των GoF) από την πλευρά του χρήστη χρησιμοποιώντας μια γλώσσα περιγραφής προτύπων. Ωστόσο, αυτό το εργαλείο δεν τροποποιεί τον υπάρχοντα κώδικα

ώστε να ταιριάζει με το πρότυπο, απλώς τον παράγει.

## 2.4 ALPHAWORKS DESIGN PATTERN TOOLKIT

Το εργαλείο αυτό είναι μία επέκταση του Eclipse και μοιάζει με το εργαλείο 2.2, επίσης και αυτό το εργαλείο δεν τροποποιεί τον κώδικα.<sup>1</sup>

---

<sup>1</sup>[2]

## ΚΕΦΑΛΑΙΟ 3

### Ανάλυση Απαιτήσεων

Στο κεφάλαιο αυτό θα δοθούν οι ιστορίες χρήστη που αφορούν το εργαλείο, σε μορφή πινάκων, καθώς και οι περιπτώσεις χρήσης του.

#### 3.1 Ιστορίες Χρήστη

Οι ιστορίες χρήστη αποτελούν άτυπες περιγραφές των χαρακτηριστικών του εργαλείου μας και των δυνατοτήτων του σε φυσική γλώσσα. Γράφονται από την πλευρά του χρήστη του εργαλείου σε μορφή καρτών.

Πίνακας 3.1: Ιστορίες χρήστη

Ιστορία Χρήστη	Σαν [τύπος χρήστη]	Θέλω να [πραγματοποιήσω ένα έργο]	Ώστε να μπορώ [να πετύχω έναν στόχο]
IX1	Προγραμματιστής	Να μπορώ να επιλέξω κατηγορία μοτίβων.	Έτσι ώστε να εισάγω αυτόματα στον κώδικά μου το σκελετό ενός μοτίβου της κατηγορίας αυτής.
IX2	Προγραμματιστής	Να μπορώ να επιλέγω ένα μοτίβο μιας κατηγορίας.	Έτσι ώστε να εισάγω αυτόματα στον κώδικά μου το σκελετό του μοτίβου αυτού.
IX3	Προγραμματιστής	Να μπορώ να καθορίσω τα ονόματα των κλάσεων που θα δημιουργηθούν αυτόματα.	Έτσι ώστε να προσαρμώσω τις κλάσεις αυτές στον κώδικά μου και στις ανάγκες του μοτίβου.
IX4	Προγραμματιστής	Να μπορώ να καθορίσω πεδία που θα προστεθούν στις νέες κλάσεις.	Έτσι ώστε να προσαρμώσω τις κλάσεις αυτές στον κώδικά μου και στις ανάγκες του μοτίβου.
IX5	Προγραμματιστής	Να μπορώ να καθορίσω μεθόδους που θα προστεθούν στις νέες κλάσεις.	Έτσι ώστε να προσαρμώσω τις μεθόδους αυτές στον κώδικά μου και στις ανάγκες του μοτίβου.
IX6	Προγραμματιστής	Να μπορώ να δημιουργήσω αυτόματα τον κώδικα του μοτίβου με βάση τις όποιες παραμετροποιήσεις έχουν γίνει.	Έτσι ώστε να εισάγω αυτόματα στον κώδικά μου το σκελετό του μοτίβου.
IX7	Προγραμματιστής	Να μπορώ να καθορίσω τα ονόματα των διεπαφών που θα δημιουργηθούν αυτόματα.	Έτσι ώστε να προσαρμώσω τις διεπαφές αυτές στον κώδικά μου και στις ανάγκες του μοτίβου.
IX8	Προγραμματιστής	Να μπορώ να καθορίσω μεθόδους που θα προστεθούν στις νέες διεπαφές.	Έτσι ώστε να προσαρμώσω τις διεπαφές αυτές στον κώδικά μου και στις ανάγκες του μοτίβου.
IX9	Προγραμματιστής	Να μπορώ να ακυρώσω την διαδικασία.	Έτσι ώστε να επιστρέψω σε αυτό που έκανα χωρίς να αλλάξω τον κώδικά μου.
IX10	Προγραμματιστής	Να μπορώ να προσθέσω νέες κλάσεις.	Έτσι ώστε να προσαρμώσω το μοτίβο στον κώδικά μου.
IX11	Προγραμματιστής	Να μπορώ να καθορίσω ποιες διεπαφές θα υλοποιούν οι νέες κλάσεις.	Έτσι ώστε να προσαρμώσω τις κλάσεις αυτές στον κώδικά μου και στις ανάγκες του μοτίβου.

## 3.2 Περιπτώσεις χρήσης

Οι Περιπτώσεις Χρήσης, αφορούν σύνολα διαδοχικών ενεργειών που προσδιορίζουν τη συμπεριφορά του συστήματος και τις λειτουργικές του απαιτήσεις. Αποτελούν μία πιο λεπτομερειακή προσέγγιση των ιστοριών χρήστη. Κάθε περίπτωση χρήσης πρέπει να διαθέτει τουλάχιστον έναν Actor, δηλαδή κάποιον που παίζει έναν ρόλο και αλληλεπιδρά με το σύστημα με τον τρόπο που ορίζει το περιεχόμενο της

περίπτωσης χρήσης.

Περίπτωση χρήσης: Επιλογή κατηγορίας μοτίβου.
Αναγνωριστικό: PX1
Προϋποθέσεις:  1. Ο προγραμματιστής χρειάζεται να έχει επιλέξει το έργο που θα εργαστεί.
Ροή γεγονότων:  1. Η περίπτωση χρήσης ξεκινά όταν ο προγραμματιστής επιλέξει Import pattern, κάτω από το μενού Design pattern builder στο παράθυρο New του eclipse.  2. Το σύστημα εμφανίζει έναν οδηγό.  3. Ο προγραμματιστής επιλέγει την κατηγορία μοτίβου που επιθυμεί.
Μετα-συνθήκες:

Πίνακας 3.2: Επιλογή κατηγορίας μοτίβου.

Περίπτωση χρήσης: Επιλογή μοτίβου
Αναγνωριστικό: PX2
Προϋποθέσεις:  1. Ο προγραμματιστής χρειάζεται να έχει επιλέξει κατηγορία μοτίβου.
Ροή γεγονότων:  1. Η περίπτωση χρήσης ξεκινάει όταν ο προγραμματιστής κάνει κλικ στην πτυσσόμενη λίστα.  2. Το σύστημα εμφανίζει τα διαθέσιμα μοτίβα.  3. Ο προγραμματιστής επιλέγει το μοτίβο που επιθυμεί.
Μετα-συνθήκες:

Πίνακας 3.3: Επιλογή μοτίβου.



<b>Περίπτωση χρήσης:</b> Καθορισμός μεθόδων κλάσης
<b>Αναγνωριστικό:</b> ΠΧ3
<p><b>Προϋποθέσεις:</b></p> <ol style="list-style-type: none"> <li>1. Ο προγραμματιστής χρειάζεται να είναι στο παράθυρο επεξεργασίας της κλάσης.</li> </ol>
<p><b>Ροή γεγονότων:</b></p> <ol style="list-style-type: none"> <li>1. Η περίπτωση χρήσης ξεκινά όταν ο προγραμματιστής κάνει κλικ στο κουμπί Next του παραθύρου επεξεργασίας πεδίων της κλάσης.</li> <li>2. Το σύστημα εμφανίζει ένα νέο παράθυρο με τις μεθόδους της τρέχουσας κλάσης.</li> <li>3. Για κάθε μέθοδο: <ol style="list-style-type: none"> <li>(α) Ο προγραμματιστής επεξεργάζεται το όνομα της μεθόδου.</li> <li>(β) Ο προγραμματιστής επεξεργάζεται τον επιστρεφόμενο τύπο της μεθόδου.</li> <li>(γ) Ο προγραμματιστής επεξεργάζεται την ορατότητα της μεθόδου.</li> </ol> </li> <li>4. Ο προγραμματιστής κάνει κλικ στο κουμπί finish.</li> <li>5. Το σύστημα κλείνει το παράθυρο.</li> </ol>
<p><b>Μετα-συνθήκες:</b></p> <ol style="list-style-type: none"> <li>1. Το σύστημα ρυθμίζει το όνομα της κλάσης.</li> <li>2. Το σύστημα ρυθμίζει τα πεδία της κλάσης.</li> <li>3. Το σύστημα ρυθμίζει τις μεθόδους της κλάσης.</li> </ol>

Πίνακας 3.4: Καθορισμός μεθόδων κλάσης.

Περίπτωση χρήσης: Ονοματοδοσία κλάσης.
Αναγνωριστικό: ΠΧ4
<p>Προϋποθέσεις:</p> <ol style="list-style-type: none"> <li>1. Ο προγραμματιστής χρειάζεται να έχει επιλέξει μοτίβο.</li> </ol>
<p>Ροή γεγονότων:</p> <ol style="list-style-type: none"> <li>1. Η περίπτωση χρήσης ξεκινά όταν ο προγραμματιστής κάνει κλικ στο κουμπί Next του παραθύρου επιλογής μοτίβου.</li> <li>2. Το σύστημα εμφανίζει ένα νέο παράθυρο με τις κλάσεις και τις διεπαφές του μοτίβου.</li> <li>3. Ο προγραμματιστής επιλέγει την κλάση που επιθυμεί.</li> <li>4. Ο προγραμματιστής κάνει κλικ στο κουμπί edit class.</li> <li>5. Το σύστημα εμφανίζει ένα νέο παράθυρο όπου ο χρήστης μπορεί να επεξεργαστεί το όνομα της κλάσης.</li> <li>6. Ο προγραμματιστής επεξεργάζεται το όνομα της κλάσης.</li> <li>7. Ο προγραμματιστής κάνει κλικ στο κουμπί Next.</li> <li>8. Το σύστημα εμφανίζει ένα νέο παράθυρο.</li> </ol>
Μετα-συνθήκες:

Πίνακας 3.5: Ονοματοδοσία κλάσης.

Περίπτωση χρήσης: Καθορισμός πεδίων κλάσης
Αναγνωριστικό: ΠΧ5
<p>Προϋποθέσεις:</p> <ol style="list-style-type: none"> <li>1. Ο προγραμματιστής χρειάζεται να έχει επεξεργαστεί το όνομα της κλάσης</li> </ol>
<p>Ροή γεγονότων:</p> <ol style="list-style-type: none"> <li>1. Η περίπτωση χρήσης ξεκινά όταν ο προγραμματιστής κάνει κλικ στο κουμπί Next του παραθύρου επεξεργασίας ονόματος της κλάσης.</li> <li>2. Το σύστημα εμφανίζει ένα νέο παράθυρο με τα πεδία της τρέχουσας κλάσης.</li> <li>3. Για κάθε μέθοδο: <ol style="list-style-type: none"> <li>(α) Ο προγραμματιστής επεξεργάζεται το όνομα του πεδίου.</li> <li>(β) Ο προγραμματιστής επεξεργάζεται τον τύπο του πεδίου.</li> <li>(γ) Ο προγραμματιστής επεξεργάζεται την ορατότητα του πεδίου.</li> </ol> </li> <li>4. Ο προγραμματιστής κάνει κλικ στο κουμπί Next.</li> <li>5. Το σύστημα εμφανίζει ένα νέο παράθυρο.</li> </ol>
Μετα-συνθήκες:

Πίνακας 3.6: Καθορισμός πεδίων κλάσης.

Περίπτωση χρήσης: Ονοματοδοσία διεπαφής
Αναγνωριστικό: ΠΧ6
<p>Προϋποθέσεις:</p> <ol style="list-style-type: none"> <li>1. Ο προγραμματιστής χρειάζεται να έχει επιλέξει μοτίβο.</li> </ol>
<p>Ροή γεγονότων:</p> <ol style="list-style-type: none"> <li>1. Η περίπτωση χρήσης ξεκινά όταν ο προγραμματιστής κάνει κλικ στο κουμπί Next του παραθύρου επιλογής μοτίβου.</li> <li>2. Το σύστημα εμφανίζει ένα νέο παράθυρο με τις κλάσεις και τις διεπαφές του μοτίβου.</li> <li>3. Ο προγραμματιστής επιλέγει την διεπαφή που επιθυμεί.</li> <li>4. Ο προγραμματιστής κάνει κλικ στο κουμπί edit interface.</li> <li>5. Το σύστημα εμφανίζει ένα νέο παράθυρο όπου ο χρήστης μπορεί να επεξεργαστεί το όνομα της διεπαφής.</li> <li>6. Ο προγραμματιστής επεξεργάζεται το όνομα της διεπαφής.</li> <li>7. Ο προγραμματιστής κάνει κλικ στο κουμπί Next.</li> <li>8. Το σύστημα εμφανίζει ένα νέο παράθυρο.</li> </ol>
Μετα-συνθήκες:

Πίνακας 3.7: Ονοματοδοσία διεπαφής.

<b>Περίπτωση χρήσης:</b> Καθορισμός μεθόδων διεπαφής
<b>Αναγνωριστικό:</b> ΠΧ7
<p><b>Προϋποθέσεις:</b></p> <ol style="list-style-type: none"> <li>1. Η περίπτωση χρήσης ξεκινά όταν ο προγραμματιστής κάνει κλικ στο κουμπί Next του παραθύρου επεξεργασίας ονόματος της διεπαφής.</li> <li>2. Το σύστημα εμφανίζει ένα νέο παράθυρο με τις μεθόδους της τρέχουσας διεπαφής.</li> <li>3. Για κάθε μέθοδο: <ol style="list-style-type: none"> <li>(α) Ο προγραμματιστής επεξεργάζεται το όνομα της μεθόδου.</li> <li>(β) Ο προγραμματιστής επεξεργάζεται τον επιστρεφόμενο τύπο της μεθόδου.</li> <li>(γ) Ο προγραμματιστής επεξεργάζεται την ορατότητα της μεθόδου.</li> </ol> </li> <li>4. Ο προγραμματιστής κάνει κλικ στο κουμπί finish.</li> <li>5. Το σύστημα κλείνει το παράθυρο.</li> </ol>
<p><b>Μετα-συνθήκες:</b></p> <ol style="list-style-type: none"> <li>1. Το σύστημα ρυθμίζει το όνομα της διεπαφής.</li> <li>2. Το σύστημα ρυθμίζει τις μεθόδους της διεπαφής.</li> </ol>

Πίνακας 3.8: Καθορισμός μεθόδων διεπαφής.

Περίπτωση χρήσης: Εξαγωγή σκελετού επιλεγμένου μοτίβου.
Αναγνωριστικό: PX8
<p>Προϋποθέσεις:</p> <ol style="list-style-type: none"> <li>1. Ο προγραμματιστής χρειάζεται να έχει επιλέξει κατηγορία μοτίβου.</li> </ol>
<p>Ροή γεγονότων:</p> <ol style="list-style-type: none"> <li>1. Η Περίπτωση χρήσης ξεκινά όταν ο προγραμματιστής κάνει κλικ στο κουμπί finish στο παράθυρο επιλογής κλάσης ή διεπαφής.</li> <li>2. Το σύστημα δημιουργεί τα απαραίτητα πηγαία αρχεία java στο επιλεγμένο πακέτο.</li> <li>3. Το σύστημα τερματίζει.</li> </ol>
Μετα-συνθήκες:

Πίνακας 3.9: Εξαγωγή σκελετού επιλεγμένου μοτίβου.

Περίπτωση χρήσης: Ακύρωση διαδικασίας
Αναγνωριστικό: PX9
<p>Προϋποθέσεις:</p>
<p>Ροή γεγονότων:</p> <ol style="list-style-type: none"> <li>1. Ο προγραμματιστής μπορεί ανα πάσα στιγμή να ακυρώσει την διαδικασία.</li> <li>2. Το σύστημα τερματίζει τα παράθυρα.</li> </ol>
Μετα-συνθήκες:

Πίνακας 3.10: Ακύρωση διαδικασίας.

<b>Περίπτωση χρήσης:</b> Προσθήκη νέας κλάσης
<b>Αναγνωριστικό:</b> PX10
<b>Προϋποθέσεις:</b> <ol style="list-style-type: none"> <li>1. Ο προγραμματιστής χρειάζεται να έχει επιλέξει μοτίβο.</li> <li>2. Το μοτίβο πρέπει να επιτρέπει την εισαγωγή καινούργιας κλάσης.</li> </ol>
<b>Ροή γεγονότων:</b> <ol style="list-style-type: none"> <li>1. Η περίπτωση χρήσης ξεκινά όταν ο προγραμματιστής κάνει κλικ στο κουμπί Next του παραθύρου επιλογής μοτίβου.</li> <li>2. Το σύστημα εμφανίζει ένα νέο παράθυρο με τις κλάσεις και τις διεπαφές του μοτίβου.</li> <li>3. Ο προγραμματιστής Κάνει κλικ στο κουμπί Add Class.</li> <li>4. Το σύστημα προσθέτει μία νέα κλάση.</li> </ol>
<b>Μετα-συνθήκες:</b>

Πίνακας 3.11: Προσθήκη νέας κλάσης.

<b>Περίπτωση χρήσης:</b> Καθορισμός υλοποιημένης διεπαφής
<b>Αναγνωριστικό:</b> ΠΧ11
<b>Προϋποθέσεις:</b> <ol style="list-style-type: none"> <li>1. Ο προγραμματιστής χρειάζεται να έχει επιλέξει μοτίβο.</li> </ol>
<b>Ροή γεγονότων:</b> <ol style="list-style-type: none"> <li>1. Η περίπτωση χρήσης ξεκινά όταν ο προγραμματιστής κάνει κλικ στο κουμπί Next του παραθύρου επιλογής μοτίβου.</li> <li>2. Το σύστημα εμφανίζει ένα νέο παράθυρο με τις κλάσεις και τις διεπαφές του μοτίβου.</li> <li>3. Ο προγραμματιστής επιλέγει μία κλάση που έχει προσθέσει ο ίδιος.</li> <li>4. Ο προγραμματιστής κάνει κλικ στο κουμπί edit class.</li> <li>5. Το σύστημα εμφανίζει ένα νέο παράθυρο όπου ο χρήστης μπορεί να επεξεργαστεί την διεπαφή που μπορεί να υλοποιεί η κλάση αυτή.</li> <li>6. Ο προγραμματιστής επεξεργάζεται την διεπαφή που μπορεί να υλοποιεί η κλάση αυτή.</li> <li>7. Ο προγραμματιστής κάνει κλικ στο κουμπί Next.</li> <li>8. Το σύστημα εμφανίζει ένα νέο παράθυρο.</li> </ol>
<b>Μετα-συνθήκες:</b>

Πίνακας 3.12: Καθορισμός υλοποιημένης διεπαφής.



## ΚΕΦΑΛΑΙΟ 4

### Σχεδίαση και αρχιτεκτονική λογισμικού

#### 4.1 Πακέτα συστήματος

#### 4.2 Κλάσεις συστήματος

##### 4.2.1 Ανάλυση κλάσεων

##### 4.2.2 Διάγραμμα κλάσεων

#### 4.3 Κάρτες αρμοδιοτήτων και συνεργασιών κλάσεων

# ΚΕΦΑΛΑΙΟ 5

## Έλεγχος

Στο κεφάλαιο αυτό θα αναλυθούν οι έλεγχοι που υλοποιήθηκαν για τον κώδικα της εφαρμογής. Ο έλεγχος χωρίστηκε σε δύο επιμέρους κατηγορίες, στον έλεγχο του εργαλείου και στον έλεγχο των μοτίβων που υλοποιεί το εργαλείο. Για την υλοποίησή τους χρησιμοποιήθηκε η βιβλιοθήκη Junit 4.

### 5.1 Έλεγχος δομής σχεδιαστικών μοτίβων

### 5.2 Έλεγχος μεθόδων

### 5.3 Έλεγχος μεθόδων δημιουργίας πηγαίου κώδικα java

## ΚΕΦΑΛΑΙΟ 6

# Οδηγός Χρήσης Design Pattern Builder

### 6.1 Λειτουργίες χρήστη

# ΚΕΦΑΛΑΙΟ 7

## Επίλογος

### 7.1 Μελλοντικές επεκτάσεις

# ΒΙΒΛΙΟΓΡΑΦΙΑ

---

- [1] E. Gamma, R. Helm, R. Johnson, and J. Vlissides, *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley Professional, 1994.
- [2] R. K. Jain, D.-M. W. Chiu, and W. R. Hawe, “A design pattern generation tool,” Tech. Rep. GFP 0801, Faculty of Worcester Polytechnic Insitute, 2009.