



Πανεπιστήμιο Πειραιώς – Τμήμα Πληροφορικής
Πρόγραμμα Μεταπτυχιακών Σπουδών
«Προηγμένα Συστήματα Πληροφορικής»

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	Πλατφόρμα Αποθετηρίου Εργασιών υλοποιημένη με Java EE Project Repository Platform developed in Java EE
Όνοματεπώνυμο Φοιτητή	Εμμανουήλ Λελεκάκης
Πατρώνυμο	Αντώνιος
Αριθμός Μητρώου	ΜΠΣΠ / 17038
Επιβλέπων	Ευθύμιος Αλέπης, Επίκουρος Καθηγητής

Ημερομηνία Παράδοσης **Δεκέμβριος 2019**

Τριμελής Εξεταστική Επιτροπή

(υπογραφή)

(υπογραφή)

(υπογραφή)

Ευθύμιος Αλέπης
Επίκουρος Καθηγητής

Μαρία Βίρβου
Καθηγητής

Κωνσταντίνος Πατσάκης
Επίκουρος Καθηγητής

1 Περιεχόμενα

2	Περίληψη	5
3	Abstract	6
1.	Εισαγωγή.....	7
2.	Ανασκόπηση πεδίου	8
2.1.	App Ideas Collection.....	8
2.2.	Quirky	8
3.	Παρουσίαση και χρήση εφαρμογής	9
4.	Αρχιτεκτονική Συστήματος.....	20
4.1.	JAVA EE.....	20
4.2.	Bootstrap.....	20
4.3.	Expression Language (EL)	20
4.4.	Δομή της εργασίας.....	20
4.4.1.	author-details.xhtml.....	21
4.4.2.	clone.xhtml.....	21
4.4.3.	console.xhtml	21
4.4.4.	deposit.xhtml	21
4.4.5.	edit.xhtml	21
4.4.6.	errorpage.xhtml	21
4.4.7.	filenotfound.xhtml	22
4.4.8.	index.xhtml.....	22
4.4.9.	response.xhtml.....	22
4.4.10.	show-ergasia.xhtml	22
4.4.11.	Author.java.....	22
4.4.12.	AuthorBean.java.....	22
4.4.13.	CurrentErgasiaBean.java	23
4.4.14.	EditBean.java.....	23
4.4.15.	Ergasia.java.....	23
4.4.16.	ErgasiaBean.java.....	23
4.4.17.	FilterBean.java.....	23
4.4.18.	Login.java.....	25
4.4.19.	SendMail.java	25
4.4.20.	User.java.....	25

4.5.	Ανάλυση – Σχεδιασμός με διαγράμματα UML	26
4.5.1.	Διάγραμμα Περιπτώσεων Χρήσης	26
4.5.2.	Διάγραμμα Καταστάσεων	27
4.6.	MySQL	27
5.	Συμπεράσματα και μελλοντικές επεκτάσεις	30
4	Βιβλιογραφία	31

2 Περίληψη

Ο μεγάλος όγκος των φοιτητών που σπουδάζουν στα πανεπιστήμια της χώρας και αποφοιτούν από αυτά καθώς επίσης και ο όγκος των εργασιών που τους χορηγούνται στα πλαίσια της ακαδημαϊκής τους σταδιοδρομίας, είναι μεγάλος. Συχνά οι καθηγητές πασχίζουν να βρουν νέα θέματα εργασιών για τους φοιτητές τους. Έτσι δημιουργήθηκε η ανάγκη μιας εφαρμογής που θα παίζει το ρόλο ενός αποθετηρίου ιδεών το οποίο όμως ενεργό ρόλο θα έχει και ο πολίτης αφού αυτός κατά κύριο ρόλο θα είναι που θα δημιουργεί τα θέματα εργασιών. Η ιδέα είναι μια διαδικτυακή (web) εφαρμογή όπου ο πολίτης θα καταθέτει μια ιδέα που έχει που θα μπορούσε στη συνέχεια να αποτελέσει θέμα εργασίας για τους φοιτητές. Φυσικά στο ρόλο του πολίτη μπορεί να είναι οι ίδιοι φοιτητές και καθηγητές. Οι πιστοποιημένοι καθηγητές-χρήστες της εφαρμογής που θα συνδέονται με τα διαπιστευτήριά τους, θα έχουν στη διάθεση τους όλη τη συλλογή των ιδεών ώστε να τις δουν και επεξεργαστούν για να τις μετασχηματίσουν κατάλληλα σε θέματα εργασιών για τους φοιτητές τους.

3 Abstract

The large volume of undergraduate students in the country's universities, as well as the amount of work assigned to them as part of their academic careers, is large. Teachers often struggle to find new work topics for their students. Thus, it is created the need for an application that would play the role of a repository of ideas. Citizens will have an active role as it will be their job to create the issues. The idea is a web application where the citizen will submit an idea that he or she has that could then be a topic of work for students. Of course, the role of the citizen can be the same students and teachers. Certified teachers, application users, will have the whole collection of ideas to view and process to transform them into work-related topics for their students.

1. Εισαγωγή

Καθώς όλο και περισσότερο υπάρχει η ανάγκη για μεγαλύτερη ποικιλία θεμάτων για εργασίες φοιτητών πανεπιστημίου, αλλά και ένας κοινός χώρος όπου θα συλλέγονται, θα επεξεργάζονται και θα διαμοιράζονται, κρίθηκε απαραίτητο η δημιουργία μιας web εφαρμογής που αντιμετωπίζει αυτό ακριβώς το θέμα.

Η εφαρμογή έχει ως χρήστες δύο κατηγορίες ανθρώπων, τους πολίτες που έχουν μία ιδέα και απλά θέλουν να την καταθέσουν και τους καθηγητές που διαχειρίζονται τις ιδέες αυτές. Στην κατηγορία των πολιτών που καταθέτουν τις ιδέες τους, μπορούν κάλλιστα να συμμετάσχουν και οι ίδιοι καθηγητές ή φοιτητές ως πολίτες.

Αρχικά ο πολίτης επισκέπτεται την κατάλληλη σελίδα που φιλοξενεί τη φόρμα με τα απαραίτητα πεδία που χρειάζεται να συμπληρώσει προκειμένου να καταθέσει την ιδέα του. Τα πεδία αυτά είναι το ονοματεπώνυμό του, η ηλεκτρονική του διεύθυνση, ο τίτλος και η περιγραφή της ιδέας του. Εναλλακτικά μπορεί να μεταφορτώσει την ιδέα του σε ένα αρχείο μορφής pdf. Αφού έχει ολοκληρώσει με την συμπλήρωση των πεδίων αυτών, μπορεί να την αποθηκεύσει.

Από την στιγμή που γίνεται η κατάθεση της ιδέας από τον πολίτη, αυτή αντιμετωπίζεται ως τέτοια μέχρι την στιγμή που θα την παραλάβει ο καθηγητής και είτε θα την μετασχηματίσει κατάλληλα σε εργασία ώστε να μπορεί να ανατεθεί σε φοιτητή, είτε θα την απορρίψει.

Η «ιδέα» ως οντότητα έχει κάποιες ιδιότητες μερικές από τις οποίες είναι ο «Τίτλος», το «Θέμα», ο «Κωδικός του συγγραφέα, το «Επίπεδο Δυσκολίας» και το «Ακαδημαϊκό Επίπεδο» που αναφέρεται στο είδος του φοιτητή, δηλαδή αν προορίζεται για Προπτυχιακούς, Μεταπτυχιακούς ή Διδακτορικούς φοιτητές. Κατά τη δημιουργία μιας ιδέας κάποιες από αυτές τις ιδιότητες λαμβάνουν προεπιλεγμένες τιμές από το σύστημα και είναι στην κρίση του καθηγητή να τις αλλάξει. Έτσι λοιπόν για να γίνει ένας αρχικός διαχωρισμός των ιδεών που έχουν αξιολογηθεί ή όχι από τον καθηγητή, υπάρχει ακόμα μία ιδιότητα της οντότητας «ιδέα» η οποία είναι μία δυαδική τιμή που χαρακτηρίζει αν μια ιδέα είναι νέα ή παλιά. Από τη στιγμή που ο καθηγητής θα προβάλλει μία νέα ιδέα και θα την επεξεργαστεί προκειμένου να μετασχηματιστεί κατάλληλα έτσι ώστε να μπορεί να ανατεθεί σε φοιτητές, τότε πλέον μιλάμε για εργασία και όχι για ιδέα. Μία ιδέα μπορεί να απορριφθεί από τον καθηγητή για λόγους μη συμβατούς με τα ακαδημαϊκά δεδομένα.

Οι ενέργειες που επιτρέπονται επί των εργασιών είναι συνολικά πέντε. Επεξεργασία, Κλωνοποίηση, Αποστολή με e-mail, Διαγραφή και Επαναφορά.

- Με την **επεξεργασία** ο καθηγητής είναι ελεύθερος να επεξεργαστεί οποιοδήποτε τμήμα της εργασίας το οποίο κρίνει ότι πρέπει να αλλάξει. Μπορεί για παράδειγμα να επεξεργαστεί το κυρίως κείμενο της εργασίας για να γίνει πιο κατανοητό ένα συγκεκριμένο τμήμα της ή και το επίπεδο δυσκολίας της.
- Με την **κλωνοποίηση** ο καθηγητής μπορεί να δημιουργήσει ένα αντίγραφο μιας εργασίας και κατόπιν να επεξεργαστεί το θέμα της προσθέτοντας επιπλέον απαιτήσεις στην εργασία προκειμένου να ανεβάσει το επίπεδο δυσκολίας ή αντίθετα να το μειώσει, αφαιρώντας απαιτήσεις.
- Όταν μία εργασία θεωρείται πως είναι κατάλληλη να δοθεί σε φοιτητή, τότε ο καθηγητής μέσα από το σύστημα μπορεί προαιρετικά να συμπεριλάβει ένα προσωπικό μήνυμα στον φοιτητή και να του την στείλει με **e-mail**.
- Μια εργασία μπορεί να **διαγραφεί** αν δεν πληροί κάποιες προϋποθέσεις ή αν θεωρείται πως είναι ξεπερασμένη.
- Ενώ αντίθετα μπορεί να γίνει **επαναφορά** μιας διαγραμμένης εργασίας.

Η αναζήτηση μιας εργασίας είναι επίσης εφικτή μέσω φίλτρων, όπου ο καθηγητής μπορεί συνδυάζοντάς τα μεταξύ τους να πάρει ένα ακριβές αποτέλεσμα. Για παράδειγμα μπορεί να αναζητήσει όλες τις εργασίες που έχουν επίπεδο δυσκολίας «Δύσκολο» και προορίζονται για του μεταπτυχιακούς ή που περιλαμβάνουν στον τίτλο τους τη λέξη «java» και προορίζονται για τους προπτυχιακούς.

Ο καθηγητής τέλος έχει την δυνατότητα να δει το προφίλ του συγγραφέα μιας εργασίας όπου μεταξύ άλλων περιλαμβάνεται το ονοματεπώνυμό του, το e-mail του, πόσες συνολικά ιδέες έχει καταθέσει και ποιες είναι αυτές.

2. Ανασκόπηση πεδίου

Σχεδόν για κάθε εφαρμογή που κυκλοφορεί υπάρχει τουλάχιστον και άλλη μία παρόμοιά της. Στην περίπτωση μας δε βρέθηκε άλλη εφαρμογή που να εξυπηρετεί ακριβώς τον ακριβώς ίδιο σκοπό αλλά παρόμοιο. Οι εφαρμογές αυτές λειτουργούν επίσης ως ένα αποθετήριο ιδεών όπου ο πολίτης μπορεί να καταθέσει την ιδέα του όπως συμβαίνει και με την εφαρμογή μας, αλλά διαφέρουν στο είδος της ιδέας και στο λόγο για τον οποίο κάποιος θα συνεισφέρει σε αυτό. Για παράδειγμα, η μία αφορά ένα αποθετήριο ιδεών για προγραμματιστές, ενώ η άλλη για ιδέες προϊόντων.

2.1. App Ideas Collection

Το συγκεκριμένο project αναφέρεται σε ένα αποθετήριο ιδεών, όπως και η παρούσα εργασία, όπου οποιοσδήποτε επιθυμεί μπορεί να καταγράψει μια ιδέα που έχει, με τη διαφορά ότι δεν απευθύνεται σε καθηγητές και φοιτητές, αλλά σε προγραμματιστές. [1] Είναι χαρακτηριστικό το μήνυμα που γράφουν στη σελίδα τους.

Θέλατε ποτέ να κατασκευάσετε κάτι αλλά δεν είχατε ιδέα τι να κάνετε; Ακριβώς όπως οι συγγραφείς μπλοκάρουν μερικές φορές, το ίδιο συμβαίνει και με τους προγραμματιστές. Αυτή η λίστα έχει σκοπό να λύσει αυτό το ζήτημα μια για πάντα!

Ο σκοπός είναι να βοηθήσει ανθρώπους που έχουν όρεξη να προγραμματίζουν αλλά για κάποιο λόγο δεν έχουν την κατάλληλη έμπνευση. Συνεπώς, αυτή η σελίδα μπορεί να βοηθήσει αυτούς τους προγραμματιστές παρέχοντάς τους μια μεγάλη ποικιλία από ιδέες προγραμμάτων που περιμένουν να υλοποιηθούν. Αυτές οι εφαρμογές είναι κατάλληλες για τη βελτίωση των προγραμματιστικών τους δεξιοτήτων, μπορούν να χρησιμοποιηθούν για να πειραματιστούν πάνω σε νέες τεχνολογίες, μπορούν επίσης να τις συμπεριλάβουν στο βιογραφικό τους, αλλά επίσης μπορούν να χρησιμοποιηθούν ως διδακτικό υλικό.

Κάθενα από αυτά τα προγράμματα συνοδεύεται από ένα σαφή και περιγραφικό στόχο ώστε να γίνει κατανοητό από τον υποψήφιο προγραμματιστή που επιθυμεί να την αναλάβει το τι απαιτήσεις έχει. Επίσης, χωρίζονται σε τρεις μεγάλες κατηγορίες ανάλογα το επίπεδο γνώσης του προγραμματιστή. Υπάρχει η κατηγορία για τους προγραμματιστές με μικρή εμπειρία, με μέτρια εμπειρία που θεωρούνται αυτοί που έχουν μία άνεση στο να δημιουργούν εφαρμογές με κάπως πιο σύνθετο UI και τέλος υπάρχει και η κατηγορία των έμπειρων προγραμματιστών που εδώ συγκαταλέγονται εφαρμογές που χρησιμοποιούν πέρα του σύνθετου UI και προηγμένες τεχνικές για backend και υπηρεσιών Βάσεων Δεδομένων.

2.2. Quirky

Το Quirky είναι επίσης ένα αποθετήριο ιδεών αλλά στόχος του είναι οι ιδέες να γίνουν πρωτότυπα εμπορικά προϊόντα, όπως πατίνια, τσάντες και ότι άλλο σκεφτείς κανείς. [2] Το Quirky λειτουργεί ως εξής. Καταθέτεις μια πρωτότυπη ιδέα που έχεις και το Quirky κάθε μήνα επιλέγει μία που θεωρεί ότι είναι εφικτή να πραγματοποιηθεί, είναι πρωτότυπη και θα έχει εμπορικό ενδιαφέρον. Η ιδέα που τελικά επιλέγεται περνάει από περαιτέρω σχεδιασμό, ανάπτυξη, προτυποποίηση, δοκιμή, κατασκευή και διανομή μέχρι το ράφι των μεγάλων καταστημάτων. Ο εμπνευστής της ιδέας του οποίου το προϊόν επιλέχθηκε και έφτασε μέχρι τα ράφια των καταστημάτων παίρνει μερίδιο επί των πωλήσεων και συγκεκριμένα 1,5% επί της χονδρικής τιμής πώλησης για κάθε προϊόν που πωλείται.

3. Παρουσίαση και χρήση εφαρμογής

Στην κεντρική σελίδα της εφαρμογής κυριαρχεί το κίτρινο χρώμα και αποτελείται από ένα μήνυμα καλωσορίσματος και μία εικόνα μολυβιών που προδιαθέτει το επισκέπτη. Αμέσως μετά από το μήνυμα καλωσορίσματος ακολουθούν κάποιες ερωτο-απαντήσεις (QA) που βοηθούν στην κατανόηση του σκοπού της εφαρμογής αυτής και τη χρήση της για τον επισκέπτη που φθάνει εκεί την πρώτη φορά.



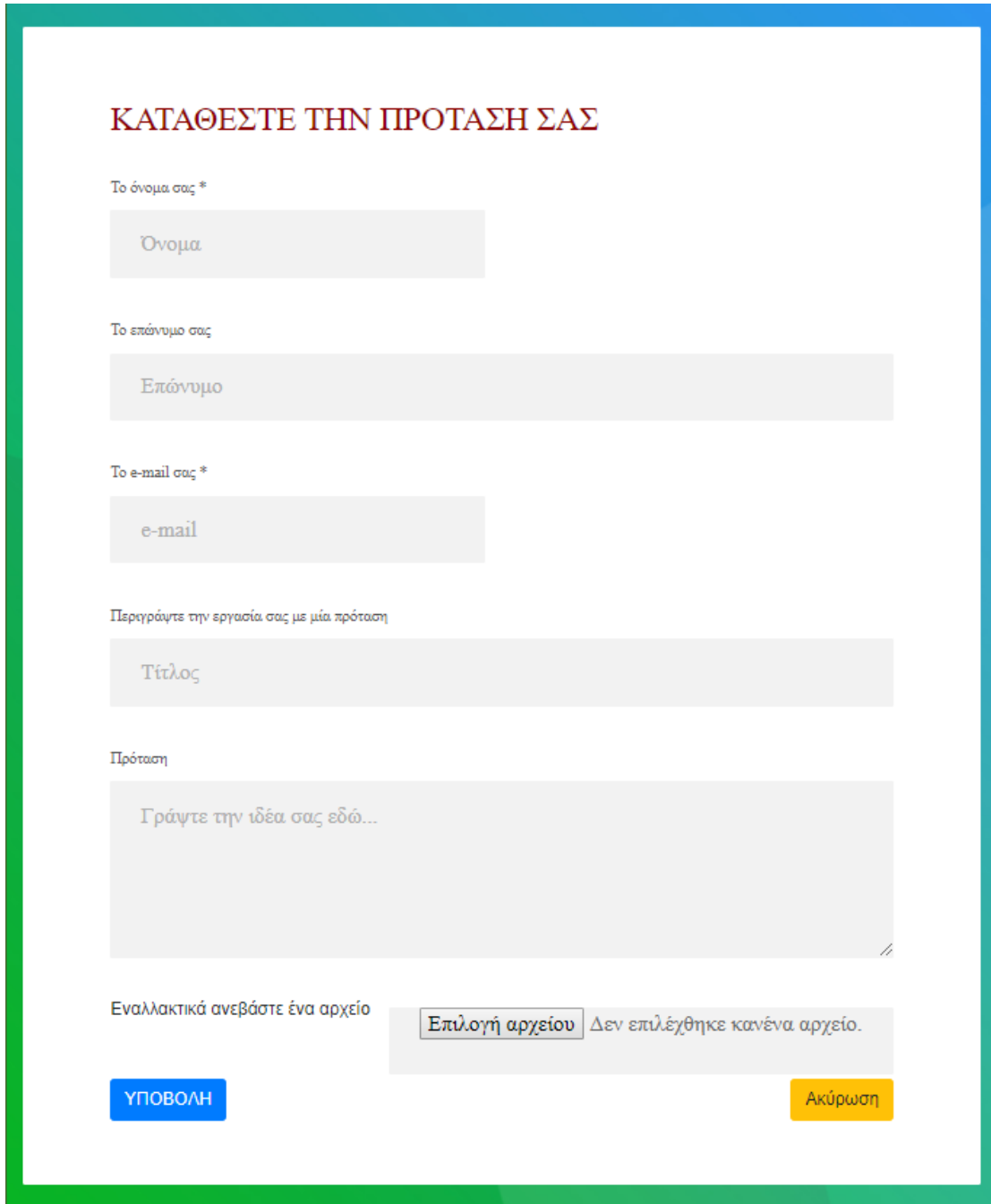
Εικόνα 1. Αρχική οθόνη (index.xhtml)

Ένα μικρό μενού επιλογών βρίσκεται στην επάνω αριστερή γωνία της σελίδας όπου ο επισκέπτης μπορεί να πατήσει και να εμφανιστεί η επιλογή της συγγραφής της ιδέας. Αντίστοιχα, ο εξουσιοδοτημένος καθηγητής μπορεί να το επιλέξει προκειμένου να εισέλθει στο σύστημα.



Εικόνα 2. Αρχική οθόνη με αναπτυγμένο το μενού (index.xhtml)

Ο χρήστης που θέλει να επιλέξει να καταθέσει την ιδέα του, επιλέγει από το μενού το «Συγγραφή ιδέας» και μεταφέρεται άμεσα στη σελίδα που φιλοξενεί την αντίστοιχη φόρμα για να την συμπληρώσει. Εδώ παρατηρούμε ότι πέρα από τη δυνατότητα που έχει ο χρήστης να πληκτρολογήσει την ιδέα που έχει, μπορεί εναλλακτικά να ανεβάσει ένα αρχείο pdf. Το επιτρεπόμενο μέγεθος αρχείου είναι μέχρι 5 MB.



ΚΑΤΑΘΕΣΤΕ ΤΗΝ ΠΡΟΤΑΣΗ ΣΑΣ

Το όνομα σας *

Το επώνυμο σας

Το e-mail σας *

Περιγράψτε την εργασία σας με μία πρόταση

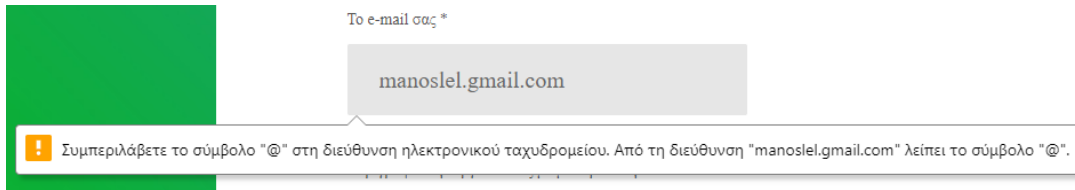
Πρόταση

Εναλλακτικά ανεβάστε ένα αρχείο

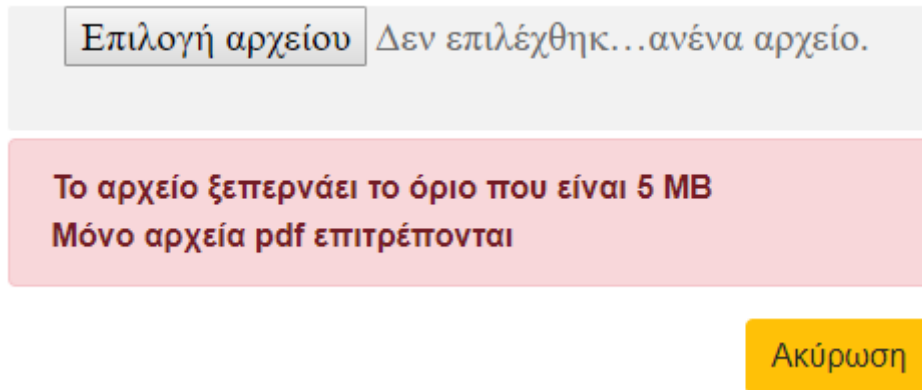
 Δεν επιλέχθηκε κανένα αρχείο.

Εικόνα 3. Οθόνη κατάθεσης ιδέας (deposit.xhtml)

Σε περίπτωση λάθους στη μορφή του e-mail ή στον τύπο και το μέγεθος του αρχείου για μεταφόρτωση, εμφανίζονται ανάλογα μηνύματα.

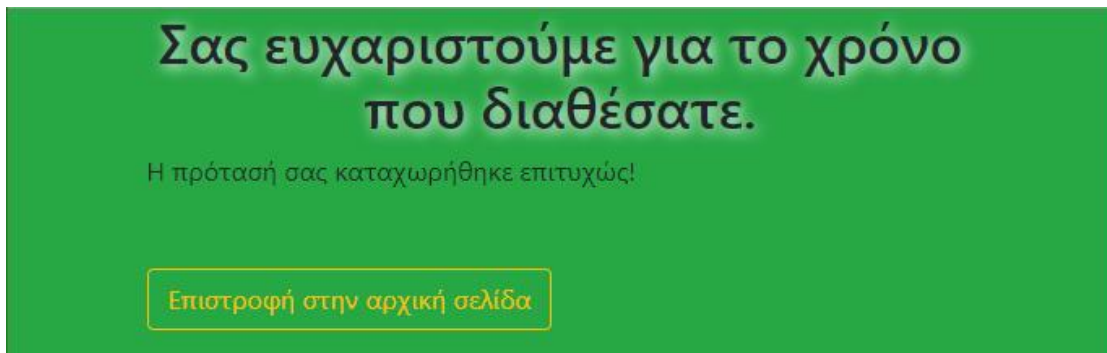


Εικόνα 4. Σφάλμα στη μορφή email (deposit.xhtml)



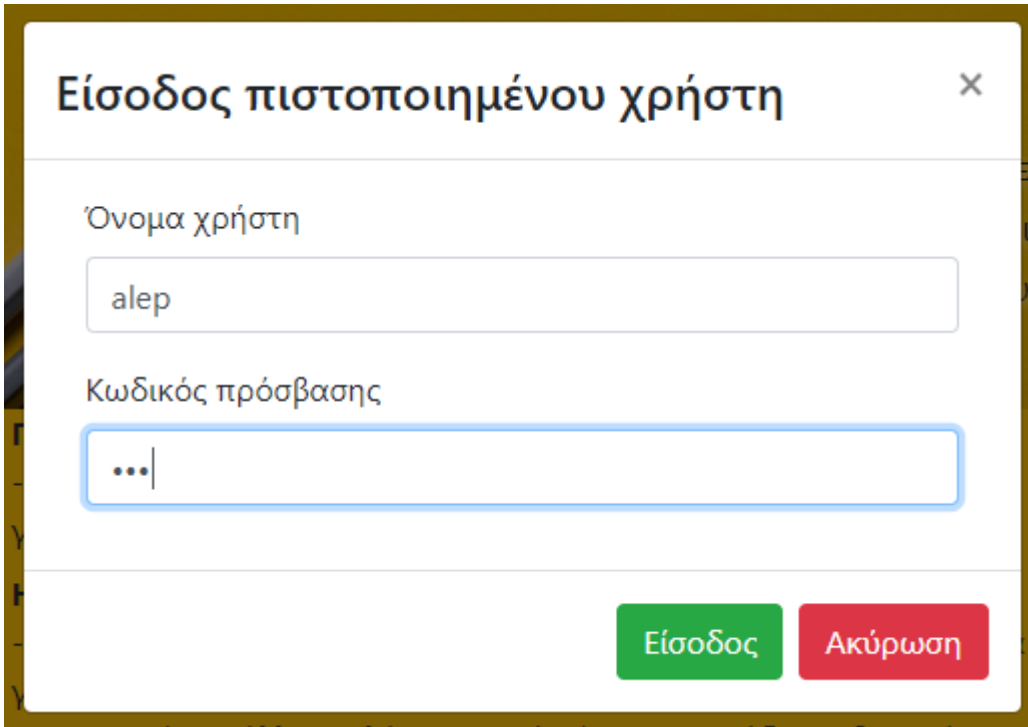
Εικόνα 5. Σφάλμα στον τύπο αρχείου αλλά και στο μέγεθος (deposit.xhtml)

Μετά από μία επιτυχημένη καταχώρηση, το σύστημα αποκρίνεται με ένα ευχαριστήριο μήνυμα και με ένα κουμπί πλοήγησης μας δίνει τη δυνατότητα να επιστρέψουμε στην αρχική σελίδα της εφαρμογής.



Εικόνα 6. Επιτυχημένη καταχώρηση ιδέας (response.xhtml)

Αντίστοιχα τώρα, για τον καθηγητή που θέλει να εισέλθει στο σύστημα, θα πρέπει από το μενού της αρχικής σελίδας να επιλέξει την επιλογή «Είσοδος». Αμέσως το σύστημα θα αποκριθεί με μία φόρμα όπου θα του ζητήσει τα διαπιστευτήριά του.



Είσοδος πιστοποιημένου χρήστη

Όνομα χρήστη

alep

Κωδικός πρόσβασης

...

Είσοδος Ακύρωση

Εικόνα 7. Προτροπή για σύνδεση καθηγητή (index.xhtml)

Μετά την επιτυχημένη σύνδεση του καθηγητή στο σύστημα, η πρώτη οθόνη που συναντά είναι το διαχειριστικό περιβάλλον των εργασιών. Από αυτήν την οθόνη ο καθηγητής έχει μια προεπισκόπηση των εργασιών σε μορφή πίνακα όπου για κάθε εργασία στη δεξιά πλευρά του πίνακα και σε μορφή κουμπιών εντολής, του δίνεται η δυνατότητα της Επεξεργασίας, Κλωνοποίησης ή Διαγραφής μιας εργασίας. Αναλόγως, στις δύο αριστερές στήλες του πίνακα φαίνονται ο κωδικός εργασίας και ο κωδικός του συγγραφέα αντίστοιχα όπου έχουν τη μορφή συνδέσμου για μετάβαση σε οθόνη με περισσότερες λεπτομέρειες. Επιπλέον, άλλες πληροφορίες που λαμβάνει από αυτήν την οθόνη ο καθηγητής είναι ο τίτλος και η πλήρης περιγραφή της εργασίας, το επίπεδο δυσκολίας της καθώς και σε τι κατηγορία φοιτητών προορίζεται (Προπτυχιακό, Μεταπτυχιακό, Διδακτορικό). Μία τελευταία πληροφορία που είναι επίσης δυνατή από αυτήν την οθόνη είναι η ημερομηνία και ώρα δημιουργίας αυτής. Στη συνέχεια, πάνω από την λίστα των εργασιών παρατηρούμε σε γκρι πλαίσιο ότι υπάρχουν τα φίλτρα αναζήτησης και τέλος πάνω δεξιά στην οθόνη βλέπουμε το όνομα του συνδεδεμένου χρήστη και την επιλογή που του δίνεται για αποσύνδεση.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ
UNIVERSITY OF PIRAEUS

Αποθετήριο για θέματα εργασιών

Δείτε, επεξεργαστείτε και μοιράστε τις εργασίες σας

Σύνδεση ως alep [Αποσύνδεση](#)

Φίλτρα αναζήτησης

Αποστάται 1ο
Κωδ. Εργασίας

Αποστάται 2ο
Κωδ. Συγγραφέα

Τίτλος

Θέμα

Δυσκολία
Οποιοδήποτε ▾

Επίπεδο
Οποιοδήποτε ▾


Άλλα κριτήρια
Παλιές ▾

Εφαρμογή φίλτρων

Κωδ. Εργασίας	Κωδ. Συγγραφέα	Τίτλος	Θέμα	Δυσκολία	Επίπεδο	Ημ/νία Δημιουργίας	Επεξεργασία	Κλώνος	Διαγραφή
4	1	Εφαρμογές αυτοματισμού εργασιών	Ανάλυση διαθέσιμων εφαρμογών αυτοματισμού εργασιών (IFTTT, Zapier, κ.α.) Αυτοματισμός καθημερινών εργασιών με χρήση εφαρμογής αυτού του είδους.	Εύκολη	Προπτυχιακό	2019-11-13 14:13:48	Επεξεργασία	Κλώνος	Διαγραφή
5	4	Δημιουργία βάσης δεδομένων των μεγαλύτερων επιχειρήσεων της Περιφέρειας Κεντρικής Μακεδονίας	Κύριος στόχος της πτυχιακής αυτής εργασίας είναι η δημιουργία μίας βάσης δεδομένων με τα στοιχεία των μεγαλύτερων επιχειρήσεων που δραστηριοποιούνται στην	Μέτρια	Προπτυχιακό	2019-11-13 14:13:48	Επεξεργασία	Κλώνος	Διαγραφή

Εικόνα 8. Κεντρική κονσόλα διαχείρισης εργασιών (console.xhtml)

Επιλέγοντας ο καθηγητής να επεξεργαστεί μία εργασία, πατάει το ανάλογο κουμπί και μεταφέρεται στην οθόνη επεξεργασίας. Από εδώ μπορεί να επεξεργαστεί τον τίτλο ή το θέμα της εργασίας, τη δυσκολία και το επίπεδο ενώ παρατηρούμε ότι ο κωδικός εργασίας και ο κωδικός συγγραφέα είναι δύο μη επεξεργάσιμα στοιχεία.



Επεξεργασία εργασίας

Κάντε διορθώσεις ορθογραφικών ή και συντακτικών λαθών στον τίτλο ή στο θέμα της εργασίας καθώς επίσης αλλαγή στο επίπεδο δυσκολίας ή στην κατηγορία (Προπτ. Μετ. Διδ.)

Κωδ. Εργασίας	Κωδ. Συγγραφέα
4	1

Τίτλος

Εφαρμογές αυτοματοποίησης εργασιών

Θεμα

Ανάλυση διαθεσιμων εφαρμογών αυτοματοποίησης εργασιών (IFTTT, Zapier, κ.α.) Αυτοματοποίηση καθημερινών εργασιών με χρήση εφαρμογής αυτού του είδους.

Δυσκολία

Δύσκολη ▾

Επίπεδο

Μεταπτυχιακό ▾

Υποβολή αλλαγών

Ακύρωση

Εικόνα 9. Επεξεργασία εργασίας (edit.xhtml)

Ευρισκόμενος ακόμα στην κεντρική οθόνη, ο καθηγητής μπορεί να επιλέξει να κλωνοποίηση την εργασία οπότε πατάει το κουμπί κλώνος της αντίστοιχης εργασίας. Αμέσως μεταφέρεται στην ανάλογη οθόνη όπου και πάλι εδώ μπορεί να την επεξεργαστεί όπως και στην περίπτωση που θέλει μόνο να την επεξεργαστεί, με την διαφορά ότι σε αυτή την περίπτωση η αρχική εργασία θα μείνει ανέπαφη και θα δημιουργηθεί ένα αντίγραφο αυτής με τις επιπλέον τροποποιήσεις που θα έχει υποστεί. Ο μόνος περιορισμός εδώ προκειμένου να δημιουργηθεί το αντίγραφο, είναι να γίνει τουλάχιστον επεξεργασία στο θέμα της εργασίας.

Δημιουργία κλώνου

Με τη δημιουργία κλώνου δημιουργείτε ένα αντίγραφο μιας εργασίας με σκοπό να επεξεργαστείτε το αντίγραφο και να το κάνετε ελαφρώς διαφορετικό της αρχικής της μορφής προκειμένου να αλλάξετε το επίπεδο δυσκολίας ή την κατηγορία στην οποία ανήκει.

Προσοχή! Προκειμένου να δημιουργηθεί η νέα εργασία, θα πρέπει να επεξεργαστείτε τουλάχιστον το θέμα της.

Ο συγγραφέας της νέας εργασίας που προκύπτει από την επεξεργασία του κλώνου, παραμένει ο αρχικός.

Κωδ. Συγγραφέα

Τίτλος

Θεμα

δημιουργίας της. Στη συνέχεια αναλύεται το σύστημα υπηρεσιών της εφοδιαστικής, ερευνάται που εφαρμόζεται και μελετώνται οι δραστηριότητές της. Στο τρίτο κεφάλαιο αναλύονται οι κατηγορίες των πληροφοριακών συστημάτων στα logistics και στη συνέχεια δίνονται τα συγκεντρωτικά χαρακτηριστικά των τύπων των πληροφοριακών συστημάτων. Τέλος παρατίθενται οι σύγχρονες τάσεις που υπάρχουν.


Στο τέταρτο κεφάλαιο παρουσιάζεται η εταιρία SARMED A.E. και ποια πληροφοριακά συστήματα αυτή χρησιμοποιεί.

Δυσκολία Επίπεδο

Εικόνα 10. Κλωνοποίηση εργασίας (clone.html)

Τέλος, από την σειρά των κουμπιών εντολής που βρίσκονται στην κεντρική κοσσόλα διαχείρισης είναι το διαγραφή, όπου πατώντας το για κάποια εργασία, αυτή μεταφέρεται αυτόματα στις διαγραμμένες.

Η επόμενη σειρά ενεργειών βρίσκεται στις δύο αριστερές στήλες της λίστας εργασιών της κεντρικής κοσσόλας. Η πρώτη από αυτές που υποδεικνύει τον κωδικό της εργασίας, χρησιμοποιείται απλά για προβολή της εργασίας είτε και για αποστολή/ανάθεση της εργασίας σε κάποιο φοιτητή μέσω e-mail. Εφόσον επιθυμεί ο καθηγητής να αναθέσει τη συγκεκριμένη εργασία σε ένα φοιτητή του, εισάγει το e-mail του και προαιρετικά ένα συνοδευτικό μήνυμα προς τον φοιτητή με οδηγίες.



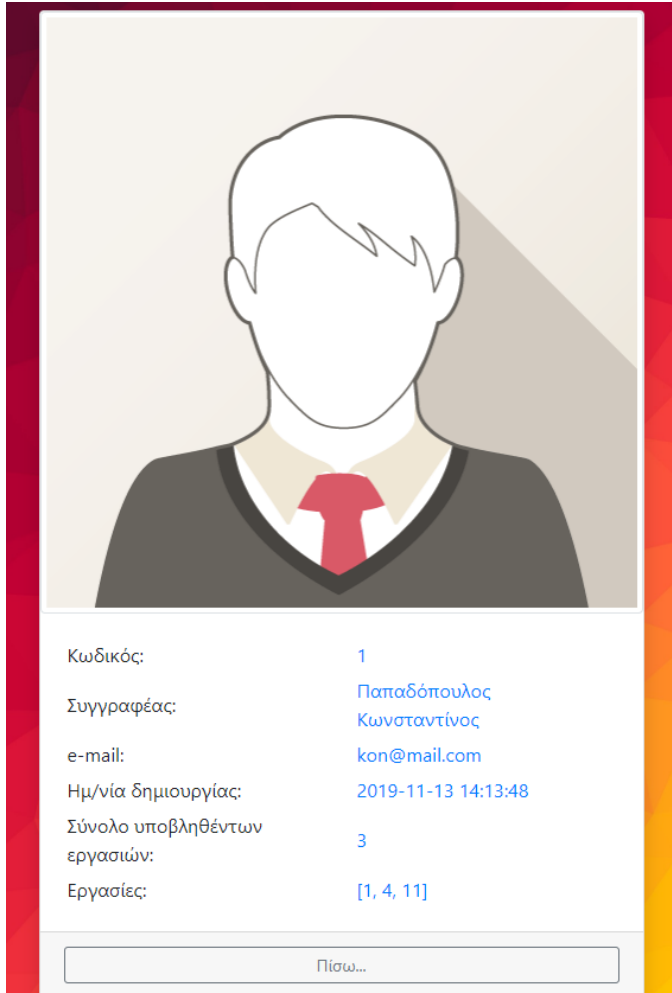
Προβολή και αποστολή εργασίας

Δείτε την εργασία και εισάγετε το e-mail του φοιτητή που θέλετε να του την αναθέσετε προκειμένου να του αποσταλεί.

Κωδ. Εργασίας	Κωδ. Συγγραφέα
1	1
Τίτλος	
Πληροφοριακά συστήματα διοίκησης στα logistics	
Θεμα	
<p>Η πτυχιακή αυτή εργασία ως σκοπό της έχει τη μελέτη των πληροφοριακών συστημάτων στα logistics.</p> <p>Για να επιτευχθεί αυτός ο σκοπός η εργασία χωρίστηκε σε τέσσερα κεφάλαια. Στο πρώτο κεφάλαιο δίνεται η έννοια και ο ορισμός των πληροφοριακών συστημάτων και ο σκοπός τους.</p> <p>Αναλύεται η σημασία που αυτά έχουν για την εφοδιαστική αλυσίδα και μελετώνται οι προκλήσεις και οι συνέπειές τους. Στο δεύτερο κεφάλαιο ερευνάται η προέλευση της λέξεως logistics, δίνεται ο ορισμός και η έννοια και μελετάται ο σκοπός της δημιουργίας της. Στη συνέχεια αναλύεται το σύστημα</p>	
Δυσκολία	Επίπεδο
Δύσκολη ▾	Προπτυχιακό ▾
email φοιτητή	
manoslel@gmail.com	
Συμπεριλάβετε ένα δικό σας μήνυμα στον φοιτητή	
Γεια σου Μανώλη, σου στέλνω την εργασία που θα ασχοληθείς το επόμενο διάστημα. Καλή συνέχεια!	
<input type="button" value="Αποστολή"/>	<input type="button" value="Ακύρωση"/>

Εικόνα 11. Προβολή και αποστολή εργασίας σε φοιτητή (show-ergasia.xhtml)

Από την κεντρική κονσόλα, πατώντας πάνω στον κωδικό του συγγραφέα, το σύστημα φορτώνει κάποιες βασικές πληροφορίες γι' αυτόν όπως είναι το ονοματεπώνυμό του, το e-mail του, πότε κατέθεσε πρώτη φορά μια ιδέα, πόσες ιδέες έχει καταθέσει στο σύνολο και τους κωδικούς αυτών. Επιπλέον, εφόσον το επιθυμεί ο καθηγητής μπορεί να επικοινωνήσει μαζί του ηλεκτρονικά πατώντας πάνω στο e-mail του συγγραφέα.



The image shows a user interface for an author's profile. At the top is a placeholder for a profile picture, represented by a simple line drawing of a person's head and shoulders. Below the picture is a white box containing the following information:

Κωδικός:	1
Συγγραφέας:	Παπαδόπουλος Κωνσταντίνος
e-mail:	kon@mail.com
Ημ/νία δημιουργίας:	2019-11-13 14:13:48
Σύνολο υποβληθέντων εργασιών:	3
Εργασίες:	[1, 4, 11]

At the bottom of the white box is a button labeled "Πίσω..." (Back...).

Εικόνα 12. Προβολή προφίλ καταθέτη (author-details.xhtml)

Ως τελευταία λειτουργία είναι αυτή της αναζήτησης εργασιών μέσω φίλτρων. Ο καθηγητής μπορεί να κάνει οποιονδήποτε συνδυασμό φίλτρων για να γίνει το ερώτημά του πιο συγκεκριμένο. Για παράδειγμα αν θέλει να προβάλει τις διαγραμμένες εργασίες, επιλέγει το ανάλογο φίλτρο. Προβάλλοντας τις διαγραμμένες εργασίες, ο καθηγητής έχει την δυνατότητα να επαναφέρει κάποια εργασία πατώντας το κουμπί της επαναφοράς.



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ
UNIVERSITY OF PIRAEUS

Αποθετήριο για θέματα εργασιών

Σύνδεση ως alep [Αποσύνδεση](#)

Δείτε, επεξεργαστείτε και μοιράστε τις εργασίες σας

Φίλτρα αναζήτησης

Αποιμάται το

Κωδ. Εργασίας

Αποιμάται το

Κωδ. Συγγραφέα

Τίτλος

Θέμα

Δυσκολία

Οποιοδήποτε ▾

Επίπεδο

Οποιοδήποτε ▾

Άλλα κριτήρια

Διαγραμμένες ▾

Εφαρμογή φίλτρων

Κωδ. Εργασίας	Κωδ. Συγγραφέα	Τίτλος	Θέμα	Δυσκολία	Επίπεδο	Ημ/νία Δημιουργίας	✎	📄	🗑️
1	1	Πληροφοριακά συστήματα διοίκησης στα logistics	Η πτυχιακή αυτή εργασία ως σκοπό της έχει τη μελέτη των πληροφοριακών συστημάτων στα logistics. Για να επιτευχθεί αυτός ο σκοπός η εργασία χωρίστηκε σε τέσσερα κεφάλαια.	Δύσκολη	Προπτυχιακό	2019-11-13 14:13:48	Επεξεργασία	Κλώνος	Επαναφορά

Εικόνα 13. Χρήση των φίλτρων. Προβολή διαγραμμένων εργασιών (console.xhtml)

4. Αρχιτεκτονική Συστήματος

Στην παρούσα web εφαρμογή χρησιμοποιήθηκαν κυρίως τεχνολογίες όπως Java EE ως γλώσσα προγραμματισμού και MySQL για τη σχεσιακή βάση δεδομένων. Για το οπτικό αποτέλεσμα έγινε χρήση της HTML και Bootstrap αλλά κυρίως του JSF, ένα Framework που ενσωματώνεται στην πλατφόρμα της Java EE.

4.1. JAVA EE

Η πλατφόρμα Java, Enterprise Edition (Java EE), είναι μια ευρέως διαδεδομένη πλατφόρμα για προγραμματισμό σε Servers που στηρίζεται πάνω στην γλώσσα προγραμματισμού Java. Η Java EE αναπτύσσεται χρησιμοποιώντας την Java Community Process (JCP), με συμβολές από ειδικούς της βιομηχανίας, εμπορικούς οργανισμούς, οργανισμούς ανοιχτού κώδικα, ομάδες χρηστών Java και αμέτρητα άτομα. Κάθε έκδοση ενσωματώνει νέες λειτουργίες που ευθυγραμμίζονται με τις ανάγκες της βιομηχανίας, βελτιώνουν τη φορητότητα εφαρμογών και αυξάνουν την παραγωγικότητα των προγραμματιστών. Η Java EE διαφέρει από την βασική έκδοση της Java SE από το γεγονός ότι προσθέτει βιβλιοθήκες οι οποίες παρέχουν λειτουργικότητα ώστε να αναπτυχθεί διαδικτυακό, πολυμερές λογισμικό σε Java, βασισμένο σε αυτοτελή μέρη που τρέχουν σε Servers. [3] [4]

Η JavaServer Faces (JSF) είναι μια τεχνολογία που βασίζεται στη διαδικασία JCP για τη δημιουργία συστατικών όπου απλοποιεί τις διεπαφές χρήστη για τις εφαρμογές JavaServer. Οι προγραμματιστές μπορούν να δημιουργήσουν εφαρμογές ιστού με τη σύνθεση επαναχρησιμοποιούμενων στοιχείων UI σε μια σελίδα, σύνδεση αυτών των στοιχείων σε μια πηγή δεδομένων εφαρμογών και την σύνδεση συμβάντων που δημιουργούνται από τους πελάτες σε χειριστές συμβάντων από πλευράς διακομιστή. [5] [6]

4.2. Bootstrap

Το Bootstrap είναι από τα πιο δημοφιλέ CSS Framework για την ανάπτυξη ανταποκρίσιμων (responsive) και mobile-first ιστοτόπων. [7] Περιέχει πρότυπα σχεδίασης HTML και CSS για φόρμες, κουμπιά πλοήγηση και άλλα στοιχεία διεπαφής, καθώς και προαιρετικές επεκτάσεις JavaScript. Χρησιμοποιήθηκε εκτενέστατα προκειμένου η εφαρμογή να είναι responsive και mobile-first ώστε να μην χάνεται καμία πληροφορία σε όποιο μέσο και αν προβληθεί, από έξυπνα κινητά τηλέφωνα μέχρι μεγάλες οθόνες τηλεοράσεων.

4.3. Expression Language (EL)

Η Java Expression Language (EL) είναι ένας συμπαγής και ισχυρός μηχανισμός που επιτρέπει δυναμική επικοινωνία σε εφαρμογές JSP και JSF (συμπεριλαμβανομένων frameworks ανάπτυξης που βασίζονται σε JSF όπως τα PrimeFaces, ICEfaces και RichFaces). Με την EL ενσωματώνουμε εκφράσεις στο επίπεδο παρουσίασης (σελίδες xhtml) για να επικοινωνήσουμε με το επίπεδο λογικής της εφαρμογής (Managed Beans). Η EL παρέχει αμφίδρομη επικοινωνία, πράγμα που σημαίνει ότι μπορούμε να εκθέσουμε τα δεδομένα λογικής της εφαρμογής στον χρήστη, αλλά μπορούμε επίσης να υποβάλουμε δεδομένα χρήστη πίσω στο επίπεδο λογικής της εφαρμογής. Γενικά, η EL μπορεί να χρησιμοποιηθεί για την κάλυψη αιτημάτων HTTP με δεδομένα χρήστη, για την εξαγωγή και έκθεση δεδομένων από τις απαντήσεις HTTP, για την ενημέρωση του HTML DOM, για την επεξεργασία δεδομένων υπό όρους και για πολλά άλλα. [8] Η EL έχει χρησιμοποιηθεί εκτενέστατα στην παρούσα εργασία. Κάθε διαφορετική σελίδα xhtml που περιλαμβάνεται στη εργασία περιέχει και EL.

4.4. Δομή της εργασίας

Η εφαρμογή αποτελείται συνολικά από 10 xhtml αρχεία που αντιπροσωπεύουν τις οθόνες (UI) με τις οποίες αλληλοεπιδρά ο χρήστης και 10 τάξεις από τις οποίες οι 8 είναι Managed Beans.

Οι 10 σελίδες αλφαβητικά είναι οι ακόλουθες:

4.4.1. author-details.xhtml

Η σελίδα αυτή προβάλλει το προφίλ του συγγραφέα της ιδέας. Περιλαμβάνει μεταξύ άλλων το ονοματεπώνυμο του, το e-mail του, το σύνολο των υποβληθέντων εργασιών και ποιες συγκεκριμένα είναι αυτές. (εικόνα 12)

4.4.2. clone.xhtml

Σε αυτή τη σελίδα προβάλλεται η εργασία που επιθυμεί ο καθηγητής να κλωνοποιήσει επιτρέποντάς του να κάνει οσεσδήποτε αλλαγές και στη συνέχεια να αποθηκεύσει το αντίγραφο. (Εικόνα 10)

4.4.3. console.xhtml

Είναι η κεντρική σελίδα που εμφανίζεται στον καθηγητή αμέσως μετά την είσοδό του στο σύστημα. Είναι η κεντρική κονσόλα διαχείρισης εργασιών με όλες τις απαραίτητες λειτουργίες να ξεκινούν από εδώ. (Εικόνες 8 και 13)

4.4.4. deposit.xhtml

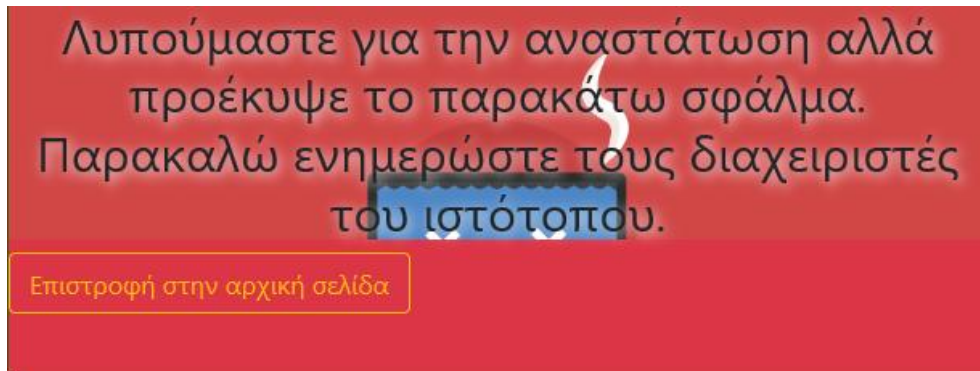
Είναι η σελίδα με τη φόρμα κατάθεσης της ιδέας. (Εικόνες 3, 4 και 5)

4.4.5. edit.xhtml

Μέσω αυτής της σελίδα ο καθηγητής μπορεί να επεξεργαστεί την εργασία αλλάζοντας το επίπεδο δυσκολίας ή ότι άλλο επιθυμεί. (Εικόνα 9)

4.4.6. errorpage.xhtml

Η σελίδα που εμφανίζεται στο χρήστη σε περίπτωση κάποιου σφάλματος στην επεξεργασία της εντολής που έδωσε.

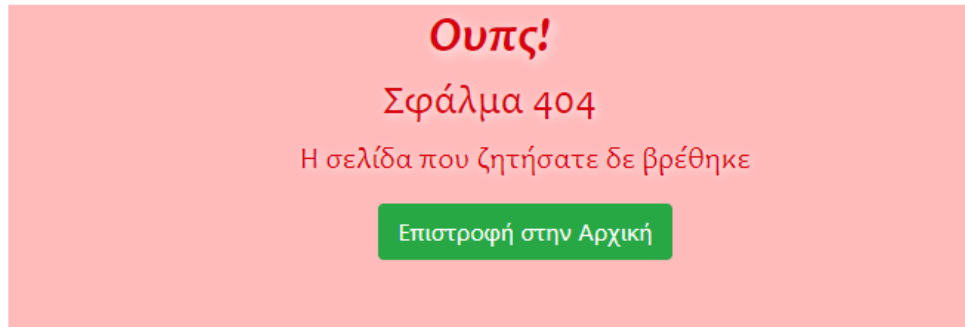


Εικόνα 14. Σφάλμα στην επεξεργασία της εντολής του χρήστη (errorpage.xhtml)

4.4.7. **filenotfound.xhtml**

Επειδή πολλές φορές ίσως από κεκτημένη ταχύτητα πληκτρολογούμε λάθος μια διεύθυνση ιστοτόπου, ο εξυπηρετητής μας απαντάει με ένα όχι και τόσο φιλικό μήνυμα 404 not found. Στην παρούσα εφαρμογή δημιουργήθηκε μια προσαρμοσμένη σελίδα για αυτό το σκοπό και ρυθμίστηκε κατάλληλα το αρχείο web.xml ώστε να εμφανίζεται σε τέτοιες περιπτώσεις. Η καταχώρηση που έγινε στο web.xml είναι η ακόλουθη.

```
<error-page>
  <error-code>404</error-code>
  <location>/filenotfound.xhtml</location>
</error-page>
```



Εικόνα 15. Προσαρμοσμένη σελίδα για σφάλματα 404 (filenotfound.xhtml)

4.4.8. **index.xhtml**

Η κεντρική σελίδα της εφαρμογής μέσω της οποίας ο χρήστης μπορεί είτε να πλοηγηθεί προς την σελίδα κατάθεσης της ιδέας του ή στην περίπτωση του καθηγητή να συνδεθεί στο σύστημα. (Εικόνες 1, 2 και 7)

4.4.9. **response.xhtml**

Αμέσως μετά την επιτυχημένη κατάθεση της ιδέας του, ο χρήστης λαμβάνει ένα ευχαριστήριο μήνυμα μέσω αυτής της σελίδας. (Εικόνα 6)

4.4.10. **show-ergasia.xhtml**

Σε αυτήν την σελίδα προβάλλεται η εργασία στον καθηγητή χωρίς να έχει δυνατότητα οποιασδήποτε επεξεργασίας πάνω της. Λειτουργεί ως μια εναλλακτική προβολή πέρα από αυτήν του πίνακα, και του δίνεται η δυνατότητα να την στείλει ηλεκτρονικά στον φοιτητή του μαζί με ένα προσωποποιημένο μήνυμα. (Εικόνα 11)

Οι 10 τάξεις αλφαβητικά είναι οι ακόλουθες:

4.4.11. **Author.java**

Χρησιμοποιείται για την επικοινωνία με την ΒΔ για να αντλήσει δεδομένα σχετικά με τον συγγραφέα της ιδέας. Αποτελείται από μία μέθοδο την `public static HashMap<String, String> getAuthor(Integer kat_id)` όπου δεχεται ως όρισμα τον κωδικό ενός καταθέτη και τον αναζητά στην ΒΔ. Εφόσον βρεθεί, επιστρέφει ένα HashMap με τα στοιχεία του.

4.4.12. **AuthorBean.java**

Είναι το Bean που χρησιμοποιεί το JSF για να προβάλει τις πληροφορίες του συγγραφέα στο χρήστη. Η τάξη Author επικοινωνεί με την AuthorBean ανταλλάσσοντας δεδομένα.

4.4.13. CurrentErgasiaBean.java

Είναι το Bean που κρατάει πληροφορίες για την τρέχουσα εργασία που έχει επιλέξει ο καθηγητής

4.4.14. EditBean.java

Είναι το Bean που έχει το ρόλο να αποθηκεύει τις αλλαγές που γίνονται σε μία εργασία κατά την επεξεργασία της.

4.4.15. Ergasia.java

Χρησιμοποιείται για την επικοινωνία με την ΒΔ για να αντλήσει δεδομένα σχετικά με τον εργασία.

4.4.16. ErgasiaBean.java

Είναι το Bean που χρησιμοποιεί το JSF για να προβάλει τις πληροφορίες της εργασίας στο χρήστη. Η τάξη Ergasia επικοινωνεί με την ErgasiaBean ανταλλάσσοντας δεδομένα.

4.4.17. FilterBean.java

Είναι το Bean που υλοποιεί την λογική των φίλτρων της κεντρικής κονσόλας εργασιών που βλέπει ο καθηγητής. Μία σημαντική μέθοδος της τάξης αυτής είναι η **public void applyFilters()** όπου είναι υπεύθυνη για την κύρια λειτουργία της τάξης. Ο κώδικας που περιέχει είναι ο ακόλουθος.

```

1 delete = false;
2 label = "Διαγραφή";
3 boolean flagWhere = false;
4 sql = "SELECT erg_id, kat_id, erg_title, erg_text, erg_dif,
      erg_goal, erg_create FROM ergasia ";
5 if (!search_erg_id.equals("")) {
6     sql = sql + "WHERE erg_id = " + search_erg_id + " ";
7     search_kat_id = "";
8     search_title = "";
9     search_text = "";
10    search_dif = "Οποιοδήποτε";
11    search_goal = "Οποιοδήποτε";
12    search_other = "Παλιές";
13 } else {
14     if (!search_kat_id.equals("")) {
15         flagWhere = true;
16         sql = sql + "WHERE kat_id = " + search_kat_id + " ";
17     }
18     if (!search_title.equals("")) {
19         if (flagWhere) {
20             sql = sql + "AND erg_title LIKE '%" + search_title +
21             "%' ";
22         } else {
23             flagWhere = true;
24             sql = sql + "WHERE erg_title LIKE '%" + search_title + '%"
25             ";
26         }
27     }
28     if (!search_text.equals("")) {
29         if (flagWhere) {
30             sql = sql + "AND erg_text LIKE '%" + search_text + '%"
31             ";
32         } else {

```

```
30         flagWhere = true;
31         sql = sql + "WHERE erg_text LIKE '%" + search_text +
"%' ";
32     }
33 }
34 if (!search_dif.equals("Οποιοδήποτε")) {
35     if (flagWhere) {
36         sql = sql + "AND erg_dif = '" + search_dif + "' ";
37     } else {
38         flagWhere = true;
39         sql = sql + "WHERE erg_dif = '" + search_dif + "' ";
40     }
41 }
42 if (!search_goal.equals("Οποιοδήποτε")) {
43     if (flagWhere) {
44         sql = sql + "AND erg_goal = '" + search_goal + "' ";
45     } else {
46         flagWhere = true;
47         sql = sql + "WHERE erg_goal = '" + search_goal + "' ";
48     }
49 }
50 if (search_other.equals("Παλιές")) {
51     if (flagWhere) {
52         sql = sql + "AND erg_new IS FALSE AND erg_active IS
TRUE ";
53     } else {
54         flagWhere = true;
55         sql = sql + "WHERE erg_new IS FALSE AND erg_active IS
TRUE ";
56     }
57 } else if (search_other.equals("Νέες")) {
58     if (flagWhere) {
59         sql = sql + "AND erg_new IS TRUE AND erg_active IS
TRUE ";
60     } else {
61         flagWhere = true;
62         sql = sql + "WHERE erg_new IS TRUE AND erg_active IS
TRUE ";
63     }
64 } else if (search_other.equals("Διαγραφμένες")) {
65     delete = true;
66     label = "Επαναφορά";
67     if (flagWhere) {
68         sql = sql + "AND erg_active IS FALSE ";
69     } else {
70         flagWhere = true;
71         sql = sql + "WHERE erg_active IS FALSE ";
72     }
73 }
74 }
75 sql = sql + "ORDER BY erg_id";
76 LOGGER.log(Level.INFO, "SQL statement: " + sql);
```


4.4.18. Login.java

Αυτό το Bean έχει ως ρόλο να ελέγξει αν τα διαπιστευτήρια που έδωσε ο χρήστης αντιστοιχούν σε κάποια που βρίσκονται στη ΒΔ. Επειδή ο κωδικός του χρήστη-καθηγητή είναι κρυπτογραφημένος στην ΒΔ με τον αλγόριθμο MD5, για να διαπιστώσουμε ότι ο χρήστης εισήγαγε τον σωστό κωδικό για να εισέλθει, θα πρέπει πρώτα να τον κρυπτογραφήσουμε και έπειτα να ελέγξουμε αν το hash string που δημιουργήθηκε, αντιστοιχεί με αυτό που βρίσκεται αποθηκευμένο στη ΒΔ. Η μέθοδος που υλοποιεί την κρυπτογράφηση του κωδικού είναι η **private** String **getMd5**(String input) και ο κώδικας που περιέχει είναι ο εξής.

```
1 try {
2
3     // Static getInstance method is called with hashing MD5
4     MessageDigest md = MessageDigest.getInstance("MD5");
5
6     // digest() method is called to calculate message digest
7     // of an input digest() return array of byte
8     byte[] messageDigest = md.digest(input.getBytes());
9
10    // Convert byte array into signum representation
11    BigInteger no = new BigInteger(1, messageDigest);
12
13    // Convert message digest into hex value
14    String hashtext = no.toString(16);
15    while (hashtext.length() < 32) {
16        hashtext = "0" + hashtext;
17    }
18    return hashtext;
19 } // For specifying wrong message digest algorithms
20 catch (NoSuchAlgorithmException e) {
21     throw new RuntimeException(e);
22 }
```

4.4.19. SendMail.java

Είναι το Bean υλοποιεί την λογική της αποστολής μιας εργασίας σε έναν φοιτητή με email. Η βασική μέθοδος που είναι υπεύθυνη για αυτήν την λειτουργία είναι η **public void** **sendEmail**()

4.4.20. User.java

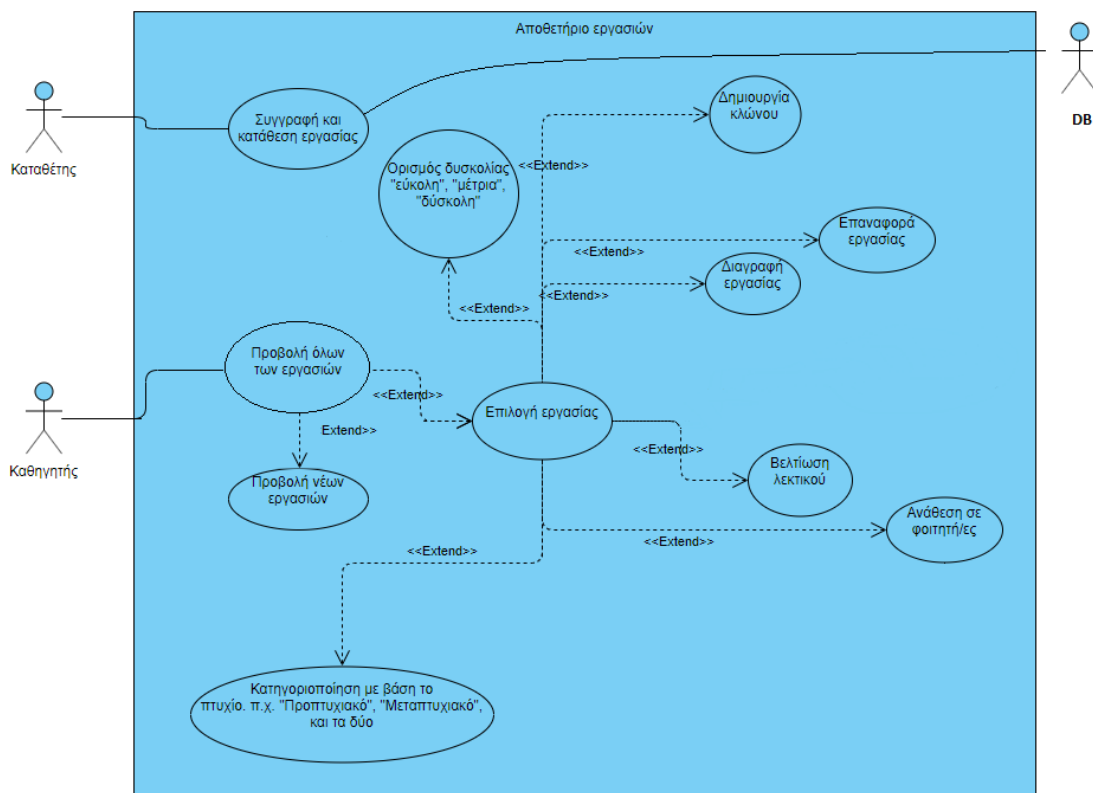
Χρησιμοποιείται για να αποθηκεύσει την ιδέα που κατέθεσε ο πολίτης καθώς και τα στοιχεία αυτού.

4.5. Ανάλυση – Σχεδιασμός με διαγράμματα UML

Θέλοντας να παρουσιάσουμε μια πιο παραστατική και φορμαλιστική απεικόνιση συνολικά αλλά και ειδικά της εφαρμογής, θα χρησιμοποιήσουμε διαγράμματα UML. Αρχικά, χρησιμοποιώντας το διάγραμμα περιπτώσεων χρήσης και έπειτα το διάγραμμα καταστάσεων.

4.5.1. Διάγραμμα Περιπτώσεων Χρήσης

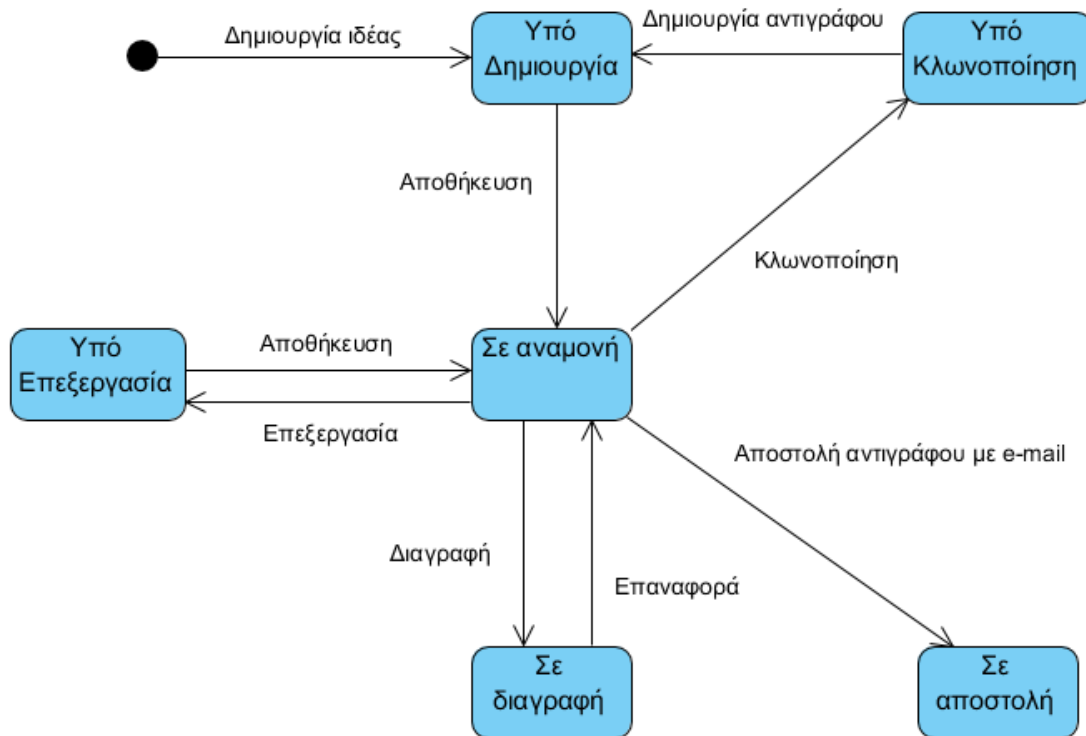
Το Διάγραμμα Περιπτώσεων Χρήσης είναι ένα σύνολο σεναρίων τα οποία συνδέονται από έναν κοινό, για το χρήστη, στόχο. Πρακτικά, μια περίπτωση χρήσης είναι μία δυνατότητα που έχει ένας χρήστης σε ένα σύστημα. Με μια πρώτη ματιά οι ενέργειες που έχει κάθε κατηγορία χρηστών στο υπό εξεταζόμενο σύστημα, είναι αυτές που φαίνονται στο παρακάτω διάγραμμα Π.Χ. ο Καταθέτης συγγράφει και καταθέτει την ιδέα του ενώ ο Καθηγητής ξεκινώντας από την προβολή των εργασιών και στην συνέχεια την επιλογή μίας εργασίας, μπορεί να κάνει μια σειρά από ενέργειες όπως είναι η Επεξεργασία, Διαγραφή, Κλωνοποίηση, Επαναφορά (διαγραμμένες) και Ανάθεση σε φοιτητή.



Εικόνα 16. Use Case diagram

4.5.2. Διάγραμμα Καταστάσεων

Το διάγραμμα καταστάσεων (State Chart ή State Machine) χρησιμοποιείται για να απεικονίσει όλες τις δυνατές καταστάσεις στις οποίες μπορεί να περιέλθει μία οντότητα ενός συστήματος καθώς και τις δυνατές μεταβάσεις μεταξύ αυτών. Εδώ ως τέτοια οντότητα έχουμε την εργασία και βλέπουμε ότι αρχικά κατά τη δημιουργία βρίσκεται σε κατάσταση «Υπό Δημιουργία» στη συνέχεια αφού την αποθηκεύσει ο συγγραφέας βρίσκεται σε αναμονή μέχρι ο καθηγητής που θα την παραλάβει να κάνει διάφορες ενέργειες πάνω της όπως Επεξεργασία, Διαγραφή, Κλωνοποίηση ή να την στείλει με e-mail σε κάποιον φοιτητή. Παρατηρούμε ότι από την κατάσταση «Σε αποστολή», δεν υπάρχει κάποια μετάβαση που να μας επιστρέφει πίσω. Αυτό συμβαίνει πολύ φυσικά διότι αυτό που στέλνεται με e-mail, είναι ένα αντίγραφο της εργασίας και όχι η ίδια η εργασία. Συνεπώς η εργασία δεν χάνεται και γι' αυτό δεν υπάρχει εξωτερική μετάβαση από την κατάσταση «Σε αποστολή». Μία άλλη παρατήρηση είναι ότι δεν υπάρχει κάποια μετάβαση που να οδηγεί σε τελική κατάσταση. Θα μπορούσε για παράδειγμα να υπήρχε μία εξωτερική μετάβαση από την κατάσταση «Σε διαγραφή» που να οδηγούσε σε τελική κατάσταση. Αυτό όμως που χρησιμοποιούμε ως σύμβαση στην παρούσα εφαρμογή είναι ότι στην πραγματικότητα οι διαγραμμένες εργασίες απλά μαρκάρονται ως διαγραμμένες για να υπάρχει η δυνατότητα μελλοντικής τους επαναφοράς.



Εικόνα 17. State Machine diagram με οντότητα την ιδέα-εργασία

4.6. MySQL

Η MySQL είναι από τα πιο δημοφιλή Συστήματα Διαχείρισης Σχεσιακών Βάσεων Δεδομένων (RDBMS) ανοικτού κώδικα όπου αναπτύσσεται, διανέμεται και υποστηρίζεται από την Oracle Corporation. [9] [10]

Παρακάτω παρουσιάζεται το scrip για τη δημιουργία της ΒΔ όπου βλέπουμε ότι αποτελείται από 3 πίνακες, τον professor, τον katathetis και τον ergasia. Ο πρώτος χρησιμοποιείται για την αποθήκευση των στοιχείων των καθηγητών. Στοιχεία όπως, όνομα, επώνυμο, όνομα χρήστη, κωδικός πρόσβασης και e-mail είναι μερικά από τα βασικά χαρακτηριστικά που αποθηκεύονται για τον καθηγητή. Μερικά στοιχεία που αποθηκεύονται για τον καταθέτη είναι το όνομά του, το επώνυμό του και το email του. Τέλος, μερικά από τα στοιχεία που αποθηκεύονται στον πίνακα

ergasia είναι ο τίτλος, η πλήρης περιγραφή της, το επίπεδο δυσκολίας, το επίπεδο στο οποίο στοχεύει η εργασία για παράδειγμα Προπτυχιακό, Μεταπτυχιακό ή Διδακτορικό, ημερομηνία δημιουργίας, το hash της εργασίας που είναι το αποτέλεσμα του MD5 αλγορίθμου κρυπτογράφησης και τέλος η θέση (path) του αρχείου pdf στο server σε περίπτωση που έχει επιλέξει ο καταθέτης να μεταφορτώσει ένα τέτοιο.

Αξίζει να σημειωθεί εδώ μία τεχνική που χρησιμοποιήθηκε προκειμένου να αποφεύγονται να αποθηκεύονται αντίγραφα της ίδιας εργασίας στη βάση. Η ιδέα είναι η εξής. Κάθε εργασία πριν καταχωρηθεί στην βάση, περνάει από τον αλγόριθμο κρυπτογράφησης MD5 που δημιουργεί ένα hash. Ένα MD5 hash δημιουργείται με τη λήψη μιας συμβολοσειράς οποιουδήποτε μήκους και την κωδικοποίησή του σε ένα αποτύπωμα 128 bit. Η κωδικοποίηση της ίδιας συμβολοσειράς με τον αλγόριθμο MD5 θα έχει πάντα ως αποτέλεσμα την ίδια έξοδο κατακερματισμού 128 bit. Εκμεταλλευόμενοι λοιπόν αυτό το χαρακτηριστικό δημιουργήθηκε ένας trigger όπου πριν την καταχώρηση του στη βάση, δημιουργεί το MD5 hash της περιγραφής της ιδέας-εργασίας.

```
create trigger tri_bi_erg_md5 before insert on ergasia
for each row
set new.erg_md5 = md5(new.erg_text);
```

Επιπλέον, στον πίνακα της εργασίας υπάρχει το **unique**(erg_md5), όπου αποτρέπει την αποθήκευση μιας ιδέας που παράγει το ίδιο MD5 hash. Έτσι εξασφαλίζουμε ότι για κάθε εγγραφή στη βάση θα υπάρχει ένα μοναδικό αντίγραφο αυτής.

Ο ίδιος αλγόριθμος MD5 χρησιμοποιήθηκε προκειμένου να αποθηκεύει το MD5 hash του κωδικού του καθηγητή και όχι τον πραγματικό κωδικό. Κατά την προσπάθεια εισόδου του καθηγητή στο σύστημα, μετατρέπεται ο κωδικός που δήλωσε ο χρήστης με χρήση java σε MD5 και ελέγχεται αν ο χρήστης με αυτό το όνομα χρήστη στη βάση έχουν το ίδιο hash, αν ισχύει αυτό τότε το σύστημα του επιτρέπει την είσοδο.

Παρακάτω παρουσιάζεται αναλυτικά στο script της sql για την δημιουργία της ΒΔ.

```
1 ##### Έναρξη Τμήματος δηλώσεων για αρχικοποίηση της ΒΔ #####
2 drop database if exists thesis;
3 CREATE DATABASE IF NOT EXISTS thesis DEFAULT CHARACTER SET greek
  COLLATE greek_general_ci;
4 use thesis;
5 SET GLOBAL event_scheduler = 1;
6 SET NAMES 'utf8';
7 SET CHARACTER SET 'utf8';
8 ##### Τέλος Τμήματος δηλώσεων για αρχικοποίηση της ΒΔ #####
9
10 ##### Έναρξη Τμήματος δηλώσεων Πινάκων #####
11 CREATE TABLE IF NOT EXISTS professor (
12     pro_id int unsigned not null auto_increment,
13     pro_name VARCHAR(20) not null,
14     pro_surname VARCHAR(20) not null,
15     pro_mail varchar(20) not null,
16     pro_un VARCHAR(20) not null, # user name
17     pro_pass VARCHAR(32) not null, # password
18     pro_active boolean default true, # Αν είναι ενεργός καθηγητής
19     του πανεπιστημίου
20     pro_create timestamp default current_timestamp,
21     pro_update timestamp on update current_timestamp,
22     primary key(pro_id),
23     unique (pro_mail, pro_un),
```

```

22     index(pro_id)
23 );
24
25 CREATE TABLE IF NOT EXISTS katathetis(
26     kat_id int unsigned not null auto_increment,
27     kat_name varchar(20),
28     kat_surname varchar(20),
29     kat_mail varchar(40) not null,
30     kat_create timestamp default current_timestamp,
31     kat_last_update timestamp on update current_timestamp,
32     primary key(kat_id),
33     unique (kat_mail),
34     index(kat_id, kat_mail)
35 );
36
37 CREATE TABLE IF NOT EXISTS ergasia(
38     erg_id int unsigned not null auto_increment,
39     kat_id int unsigned,
40     erg_title varchar(150),
41     erg_text varchar(2000),
42     erg_dif enum('Εύκολη', 'Μέτρια', 'Δύσκολη') default 'Μέτρια',
43     erg_goal enum('Προπτυχιακό', 'Μεταπτυχιακό', 'Διδακτορικό')
44 default 'Προπτυχιακό', # Σε ποιο ακαδημαϊκό επίπεδο απευθύνεται η
    εργασία. Μπορεί να ανήκει σε πάνω από ένα.
45     erg_md5 varchar(32),
46     erg_pdfpath varchar(255),
47     erg_active boolean default true,
48     erg_new boolean default true,
49     erg_create timestamp default current_timestamp,
50     erg_last_update timestamp on update current_timestamp, #
    Τελευταία ημερομηνία ανανέωσης
51     primary key(erg_id),
52     FOREIGN KEY (kat_id) REFERENCES katathetis(kat_id),
53     unique(erg_md5),
54     index(erg_id, kat_id)
55 );
56 ##### Τέλος Τμήματος δηλώσεων Πινάκων #####
57 ##### Έναρξη Τμήματος δηλώσεων Triggers #####
58 create trigger tri_bi_erg_md5 before insert on ergasia
59 for each row
60 set new.erg_md5 = md5(new.erg_text);
61 create trigger tri_bu_erg_md5 before update on ergasia
62 for each row
63 set new.erg_md5 = md5(new.erg_text);
64

```

```

65 create trigger tri_bi_pro_md5 before insert on professor
66 for each row
67 set new.pro_pass = md5(new.pro_pass);
68 ##### Τέλος Τμήματος δηλώσεων Triggers #####
69

```

5. Συμπεράσματα και μελλοντικές επεκτάσεις

Η εφαρμογή αυτή μπορεί να αποτελέσει ένα πολύ καλό εργαλείο στα χέρια του καθηγητή που θέλει να έχει ένα εύκολο σύστημα παραγωγής και συγκέντρωσης θεμάτων εργασιών σε ένα χώρο. Τον απαλλάσσει εν μέρει από το άγχος της συνεχής αναζήτησης νέων και πρωτότυπων ιδεών για εργασίες και επωφελείται από τον τρόπο και την οργάνωση που είναι συγκεντρωμένες σε ένα σημείο.

Πάντα βέβαια υπάρχουν περιθώρια βελτίωσης σε όλους τους τομείς. Έτσι κι εδώ, η εφαρμογή έχει πολλά περιθώρια επέκτασης των λειτουργιών της όπου οι απαιτήσεις μπορούν να διαμορφωθούν ως ακολούθως.

Η εφαρμογή αμέσως μετά την συγγραφή και αποθήκευση της ιδέας από τον συγγραφέα, του δίνει τη δυνατότητα της εκτύπωσης και αποθήκευσης σε μορφή pdf όπου εμφανίζεται και ο κωδικός της ιδέας.

Η εφαρμογή επικοινωνεί με το μητρώο φοιτητών του πανεπιστημίου έτσι ώστε οι φοιτητές να μπορούν να επιλέξουν μόνοι τους την εργασία που θέλουν να υλοποιήσουν. Κάθε εργασία μπορεί να χρησιμοποιηθεί μία μόνο φορά ανά Χ έτη ώστε να μην γίνεται συχνή ανακύκλωση. Το σύστημα προτείνει θέματα στους φοιτητές ανάλογα με το τι άλλες εργασίες είχαν αναλάβει στο παρελθόν και έτσι μιλάμε για εξατομίκευση λογισμικού. Ο φοιτητής μπορεί να κάνει κράτηση του θέματος για μία εβδομάδα. Αν στο διάστημα αυτό δεν επιβεβαιώσει ότι θα την αναλάβει, τότε το θέμα ελευθερώνεται και δε μπορεί να το κάνει κράτηση ξανά για μία εβδομάδα. Κατά το διάστημα που το έχει σε κράτηση, μπορεί να επιβεβαιώσει ότι θα την αναλάβει και τότε δε μπορεί να κάνει ξανά κράτηση μέχρι να την παραδώσει. Ο φοιτητής που έχει κατοχυρώσει θέμα και πρέπει να το αλλάξει, δε μπορεί ο ίδιος, παρά μόνο ο καθηγητής του μπορεί να το ακυρώσει. Ο φοιτητής μπορεί να αξιολογήσει μια εργασία ανεξαρτήτως αν την ανέλαβε ή όχι με μια κλίμακα από 1 (κακή) έως 5 (πολύ ενδιαφέρουσα) αστέρια (★★★★). Επομένως έτσι θα γινόταν αντιληπτό μεταξύ φοιτητών και καθηγητών κατά πόσο μια εργασία είναι δημοφιλής και ενδιαφέρουσα. Επιπλέον, οι συνδεδεμένοι φοιτητές μπορούν να καταχωρούν τις ιδέες τους επώνυμα και θα καταγράφεται από ποιον κατατέθηκαν. Έτσι, θα μπορεί να υπάρξει πρόγραμμα επιβράβευσης για τους φοιτητές που έχουν καταθέσει πολλές ιδέες με την μορφή κάποιου achievement. Θα υπάρχει ένα leaderboard με την γενική κατάταξη των φοιτητών που έχουν υποβάλει τις περισσότερες ιδέες.

Τέλος, ο καθηγητής μπορεί να δημιουργεί ομάδες φοιτητών και να τους αναθέτει μία εργασία (ομαδική εργασία).

Οι δυνατότητες που μπορείς πρακτικά να ενσωματώσεις σε μια εφαρμογή είναι ουσιαστικά πολλές. Με αυτό το σκεπτικό θα μπορούσε η εφαρμογή να ενσωματώσει επίσης λειτουργικότητα αποθετηρίου υλοποιημένων εργασιών, όπως αυτές που έχουν ήδη τα πανεπιστήμια.

Για να αντέχει μια εφαρμογή στο χρόνο θα πρέπει συνεχώς να ενημερώνεται με νέα χαρακτηριστικά καθώς οι ανάγκες των χρηστών μεταβάλλονται με τον καιρό. Κάτι που σήμερα θεωρείτε ασήμαντο, αύριο μπορεί να γίνει απαραίτητο.

4 Βιβλιογραφία

- [1] F. Pop, «App Ideas Collection,» 2019. [Ηλεκτρονικό]. Available: <https://github.com/florinpop17/app-ideas>.
- [2] Quirky, «Quirky,» [Ηλεκτρονικό]. Available: <https://quirky.com/>.
- [3] Oracle, Oracle, [Ηλεκτρονικό]. Available: <https://www.oracle.com/java/technologies/java-ee-glance.html>.
- [4] A. Goncalves, Beginning Java EE 7, apress.
- [5] Oracle, «javaxserverfaces-spec,» [Ηλεκτρονικό]. Available: <https://javaee.github.io/javaxserverfaces-spec/>.
- [6] A. Leonard, Mastering JavaServer 2.2, Birmingham: Packt Publishing, 2014.
- [7] Bootstrap, «Bootstrap,» [Ηλεκτρονικό]. Available: <https://getbootstrap.com/>.
- [8] L. Anghel, Mastering JavaServer 2.2, Birmingham: Packt Publishing, 2014.
- [9] MySQL, «MySQL,» [Ηλεκτρονικό]. Available: <https://dev.mysql.com/>.
- [10] P. DuBois, MySQL Cookbook, O'Reilly, 2014.