



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΛΟΠΟΝΝΗΣΟΥ
ΣΧΟΛΗ ΟΙΚΟΝΟΜΙΑΣ, ΔΙΟΙΚΗΣΗΣ ΚΑΙ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ



Ερευνητική Ομάδα Γνώσης και Αβεβαιότητας

Πτυχιακή εργασία

ΣΥΣΤΗΜΑ ΟΡΓΑΝΩΣΗΣ ΚΗΠΟΥ ΚΑΙ ΥΠΟΣΤΗΡΙΞΗΣ ΚΗΠΟΥΡΟΥ

ΙΩΑΝΝΗΣ ΗΛΙΑΣ ΠΛΗΚΑΣ
2024200900088

Επιβλέπων:

Μανόλης Γουάλλες
Λέκτορας

Τρίπολη, Ιούνιος 2015

Εγκρίθηκε από την εξεταστική επιτροπή την 10η Ιουνίου 2015.

Νικόλαος Τσελίκας
Επίκουρος Καθηγητής

Γεώργιος Λέπουρας
Αναπληρωτής Καθηγητής

.....

Ιωάννης-Ηλίας Πλήκας
Τμήμα Πληροφορικής και Τηλεπικοινωνιών
Πανεπιστήμιο Πελοποννήσου

Copyright © Ιωάννης-Ηλίας Πλήκας, 2015
Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα. Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευτεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Πανεπιστημίου Πελοποννήσου.

Η παρούσα εργασία εκπονήθηκε σε συνεργασία με την Ερευνητική Ομάδα Γνώσης και Αβεβαιότητας (FAB LAB)

Περίληψη

Στόχος της συγκεκριμένης πτυχιακής, ήταν η υλοποίηση μιας εφαρμογής γραμμένης σε JAVA που αφορά την παρακολούθηση καλλιεργειών σε ερασιτεχνικό επίπεδο. Έμφαση δόθηκε στην ευκολία χρήσης, έτσι ώστε να είναι προσιτή ακόμα και από άτομα που δεν έχουν ιδιαίτερη εξοικείωση με την τεχνολογία και τους ηλεκτρονικούς υπολογιστές.

Η μεθοδολογία που ακολουθήθηκε με σκοπό υλοποίησης της εφαρμογής ήταν αρχικά η επιλογή του τρόπου αποθήκευσης όλων των δεδομένων των καλλιεργειών, τα οποία θα χρησιμοποιούνταν για τη μετέπειτα διασύνδεση με την εφαρμογή. Έτσι επιλέχθηκαν τα αρχεία κειμένου να διαβάζονται διαμέσω της JAVA. Τα αρχεία και τα εικονίδια της εφαρμογής ήταν γνωστά, όπως επίσης και η τοποθεσία τους για υλοποίηση του προγράμματος. Όλα τα δεδομένα καθώς και οι εικόνες συνδέθηκαν μεταξύ τους με μεγάλη αρμονία για υλοποίηση των γραφικών αντικειμένων. Για να παρουσιαστούν τα δεδομένα αυτά στην οθόνη και συνεπώς στο χρήστη, χρησιμοποιήθηκαν τα εργαλεία της JAVA Swing για δημιουργία του γραφικού περιβάλλοντος. Τα κύρια βήματα που εκτελέστηκαν, ήταν αρχικά η δημιουργία του αρχείου config. ini στο οποίο σε κάθε σειρά αποθηκεύτηκαν όλες οι ενέργειες, οι οποίες χωρίστηκαν σε δράσεις και παρεμβάσεις. Κάθε δράση, αντιστοιχίστηκε σε εικονίδιο και χρώμα ενώ κάθε παρέμβαση μόνο σε εικονίδιο. Στη συνέχεια, δημιουργήθηκαν δύο φάκελοι ο user και ο recommended. Ο πρώτος αποθηκεύει όλα τα αρχεία κειμένου με δεδομένα των καλλιεργειών και συγκεκριμένη δομή. Κάθε αρχείο κειμένου αντιστοιχίστηκε με δεδομένα μίας καλλιέργειας. Ο δεύτερος φάκελος αφορούσε τα δεδομένα των προτύπων καλλιεργειών της εφαρμογής. Δημιουργήθηκε επίσης, ένας φάκελος icons ο οποίος περιέχει τους φακέλους actions, interfere και plants. Στον πρώτο αποθηκεύτηκαν όλες οι εικόνες των ενεργών και παλιών καλλιεργειών, ενώ στο φάκελο recommended όλες οι εικόνες των προτύπων καλλιεργειών.

Στο φάκελο plants αποθηκεύτηκαν όλες οι εικόνες που περιέγραφαν τις καλλιέργειες. Στο φάκελο αυτό δημιουργήθηκε, ακόμη ένα αρχείο κειμένου το οποίο χρησιμοποιήθηκε για να συνδεθούν τα ονόματα των καλλιεργειών με τις αντίστοιχες εικόνες που τα περιέγραφαν. Όλα αυτά τα δεδομένα μεταφέρθηκαν στο γραφικό περιβάλλον της εφαρμογής και συγκεκριμένα πάνω σε οριζόντια ορθογώνια. Έτσι δημιουργήθηκαν σειρές που περιέχουν η κάθε μια, μία μπάρα εξέλιξης κάποιας καλλιέργειας. Στη συνέχεια, δημιουργήθηκε μπάρα χρόνου πάνω από τις μπάρες εξέλιξης των καλλιεργειών και η κάθε μία ξεκίναγε πια από τη δική της ημερομηνία. Τέλος, προστέθηκαν επιπλέον λειτουργίες όπως μενού επιλογών και εισαγωγή νέου συμβάντος για καλύτερη διαχείριση της εφαρμογής και ευκολία αυτής προς τον ίδιο χρήστη. Η εφαρμογή έγινε πλήρως δυναμική. Οποιοσδήποτε αλλαγές σε αυτή διαμέσου των αρχείων επηρεάζουν και τη γραφική εμφάνιση της εφαρμογής. Κύρια αποτελέσματα των βημάτων αυτών ήταν η δημιουργία μιας πλήρως **δυναμικής εφαρμογής**, στην οποία ο συνδυασμός χρωμάτων και λειτουργιών βοηθάνε στη μέγιστη διαχείριση καλλιεργειών μικρής κλίμακας, ακόμη και για τον πιο αρχάριο του είδους.

Abstract

The aim of this thesis, was the implementation of an application written in JAVA for monitoring crops at amateur level. Emphasis was placed on ease of use, so that it is accessible even by people who are not very familiar with technology and computers. The Methodology followed to implement the application was initially to select the way to store all data of the cultures, which were used to interconnect with the application. Selected text files can be read through JAVA program. Files and icons of the application was known, as well as their location for the embodiment with the application. Every data and images were ligated together with great harmony for the embodiment of graphic objects. Aiming to present data on screen and then the user himself, we used tools from JAVA Swing GUI for creating graphical user interface . The main steps executed initially, was to create file config. ini in which each row stored all actions, which were divided into actions and interferes. Every action, was assigned an icon and color and every interfere was assigned only an icon. Then two folders were created, user and recommended folder. The first stores all text files with data and crop data following specific creating. Each text file is assigned with data representing each crop. The second file created to relate the suggested cropest of our application. There were also created a folder named icons, which contains the actions folder, interfere folder and plants folder. Actions stored all images of active and historical cropest while the recommended folder, stored all images of suggested cropest.

In folder named plants were stored all the images describing the cropest themselves. On that folder were created a text file that is used to connect the names of cropest, with the images that describe them. All these data are then transferred to the graphical interface of the application each one horizontal rectangles. Were created rows, each containing an evolution bar of each crop. Then were created a time bar above the crop evolution bars and each was able to start from their own time. Finally, additional features were added as, such as options menu and event insertion to better manage implementation and ease of use. The application was finally dynamic. Every changes to its files influence the appearance of the application. Primary results of these steps was the creation of a fully **dynamic application**, in which the combination of colors and functions help at maximum in managing small scale cropest, even for a newbie.

Περιεχόμενα

Πρόλογος	vii
1 Εισαγωγή	1
2 Καλλιέργειες και ηλεκτρονικοί υπολογιστές	3
2.1 Σημεία ενδιαφέροντος για τις καλλιέργειες γενικά	3
2.2 Τα τεχνολογικά μέσα των γεωργικών καλλιεργειών	4
2.3 Χρήση λογισμικού στις καλλιέργειες	4
3 Χρήσιμες τεχνολογίες	7
3.1 Τρόποι αποθήκευσης δεδομένων	7
3.1.1 Βάσεις δεδομένων	7
3.1.2 Αρχεία	8
3.1.3 Σύγκριση μεθόδων	9
3.2 Εμφάνιση δεδομένων	9
3.2.1 Εμφάνιση σε φόρμα	10
3.2.2 Εμφάνιση σε ιστοσελίδα	11
3.2.3 Σύγκριση μεθόδων	13
4 Δική μας επιλογή - επιλογή υλοποίησης	15
4.1 Αποθήκευση δεδομένων	15
4.2 Διεπαφή χρήστη	16
4.2.1 Αντικειμενοστρεφής προγραμματισμός	16
4.2.2 JAVA	16
4.2.3 JAVA και GUI	17
5 Τί μπορεί να κάνει κάποιος με τις καλλιέργειες γενικά	23
5.1 Αρχικοί ενδοιασμοί καλλιέργειας	23
5.2 Ενέργειες πάνω στις καλλιέργειες	24
5.3 Συνηθέστερος τρόπος καταγραφής καλλιέργειας	25
5.4 Αποτύπωση ενεργειών στο χαρτί	25
5.4.1 Καλλιέργειες μικρής κλίμακας, ποιός τρόπος είναι πιο ιδανικός και γιατί .	28
5.4.2 Βασικοί παράγοντες προσοχής	29
5.4.3 Θετικά και αρνητικά του χαρτιού ως ημερολόγιο καλλιεργειών	30

6 Προδιαγραφές	31
6.1 Τι καλύτερο από το χαρτί	31
6.2 Εμφάνιση και λειτουργία εφαρμογής	32
7 User Interface και πώς έγινε η υλοποίηση	35
7.1 Αρχικοί ενδοιασμοί υλοποίησης	35
7.2 Δράσεις-Παρεμβάσεις	36
7.3 Αρχικός καθορισμός του config. ini	37
7.4 Φάκελοι user, recommended και δομή αρχείων κειμένου	39
7.5 Εικόνες	43
7.5.1 Φάκελος icons	43
7.6 Προσχέδιο στο χαρτί του γραφικού περιβάλλοντος	43
7.7 Γραφικό περιβάλλον διεπαφής χρήστη - GUI	45
7.7.1 Υλοποίηση με JButtons	45
7.7.2 Υλοποίηση με Rectangles	47
8 Αρχιτεκτονική	55
8.1 Αρχεία	55
8.2 JAVA	59
9 Παραδείγματα χρήσης	71
10 Συμπεράσματα	81
11 Βιβλιογραφία	83
11.1 Βιβλιογραφικές Αναφορές	83
11.2 Ηλεκτρονικές διευθύνσεις	83

Πρόλογος

Ευχαριστίες

Στο σημείο αυτό θα ήθελα να ευχαριστήσω όλους τους ανθρώπους αυτούς, που βοήθησαν και συντέλεσαν στη διεκπεραίωση αυτής της πτυχιακής εργασίας. Θα ήθελα να ευχαριστήσω αρχικά τον καθηγητή κ. Γουάλλες Εμμανουήλ, διδάσκοντα του Πανεπιστημίου Πελλοποννήσου και επιβλέποντα, για την βοήθεια και καθοδήγηση του, την μεταδοτικότητα και τις γνώσεις που μου προσέφερε καθώς και την άριστη συνεργασία καθ' όλη τη διάρκεια αυτής της εργασίας. Και τέλος, ένα μεγάλο ευχαριστώ στους δικούς μου ανθρώπους, οικογένεια και φίλους που στάθηκαν δίπλα μου σε όλη τη διάρκεια της πορείας μου μέχρι εδώ και σε όλους εκείνους που βοήθησαν ο καθένας με τον δικό του τρόπο.

Κεφάλαιο 1

Εισαγωγή

Η σημερινή κοινωνία έχει αναπτυχθεί και εξοικειωθεί με τη συνεχώς αναπτυσσόμενη τεχνολογία. Παρ' όλη την εξέλιξή της, η κοινωνία και ο ανθρώπινος πολιτισμός βασίζει την σημερινή του εξέλιξη στον πρωτογενή τομέα. Από τις αρχές του χρόνου ο άνθρωπος χρησιμοποιούσε την καλλιέργεια διαφόρων προϊόντων για να καλύψει τις βασικές του ανάγκες. Με την πάροδο όμως του χρόνου και την ύπαρξη της τεχνολογίας, ο πρωτογενής τομέας παραμένει βάση της εξέλιξης και του σημερινού κρίκου του πολιτισμού μας και για το λόγο αυτό πρέπει να δίνεται ιδιαίτερη έμφαση στη γεωργία και κατ' επέκταση στις καλλιέργειες προϊόντων. Δίνεται όμως μία νέα δυνατότητα στις καλλιέργειες και αυτή είναι η σύνδεση της σημερινής τεχνολογίας με των καλλιεργειών είτε σε κλίμακα κήπου είτε χωραφιού, με σκοπό την καλύτερη και ευκολότερη διαχείρισή τους.

Όλο και περισσότεροι άνθρωποι καλλιεργούν σε μικρή κλίμακα φυτόρια όπως σε μία βεράντα σπιτιού, έχοντας βάση αυτού ως κέρδος την ύπαρξη πολλών καλλιεργειών συζευμένων σε ένα μικρό μέρος και έτσι την ευκολότερη διαχείρισή τους. Δημιουργούνται παρ' όλα αυτά κάποια προβλήματα και κάποιοι ενδοιασμοί. Κάποιος ο οποίος καλλιεργεί φυτά σε μικρή κλίμακα, θα πρέπει να έχει στο πίσω μέρος του κεφαλιού του διάφορες πληροφορίες σχετικά με τα αυτά και την καλλιέργειά τους, ειδικότερα όσο τα φυτόρια αυξάνονται καθιστώντας αυτές τις πληροφορίες αδύνατο να αφομοιωθούν, πόσο μάλλον να τις θυμάται κάποιος. Με γνώμονα, να μπορεί κάποιος να διαχειριστεί καλύτερα μικρότερης κλίμακας καλλιέργειες χωρίς να χρειάζεται να θυμάται τί έχει να κάνει, όπως και τις καλλιέργειες που έχουν ήδη αναπτυχθεί ή ακόμη την ύπαρξη πολλών καλλιεργειών σε κομβικά σημεία λόγω ελαχίστου χώρου αποθήκευσής τους, έπρεπε να βρεθεί κάποια λύση που να το επιτελεί για αυτόν. Αυτό ακριβώς πραγματεύεται η συγκεκριμένη πτυχιακή.

Σε γενικότερο επίπεδο, υπάρχουν διαφόρου τύπου καλλιέργειες τις οποίες μπορεί να φυτέψει και κατά συνέπεια να καλλιεργήσει κάποιος και δύο τρόποι να ξεκινήσει κάποιος την ανάπτυξη κάποιου φυτού ή φυτωρίου. Αυτοί είναι σπορά ο πρώτος και αγορά κάποιου έτοιμου, ήδη ανεπτυγμένου φυτού όπως σε μία γλάστρα ο δεύτερος και η εν συνεχεία ανάπτυξη της εκάστοτα καλλιέργειας από εκείνο το σημείο και έπειτα. Επίσης, υπάρχουν ήδη υπάρχουσες τεχνολογίες με σκοπό όπως προαναφέρθηκε, να υποβοηθήσουν την διαχείριση των καλλιεργειών. Όλοι λίγο πολύ είναι εξοικειωμένοι με τους ηλεκτρονικούς υπολογιστές οι οποίοι έχουν πια μεγάλες δυνατότητες, ειδικά αν χρησιμοποιηθούν σωστά.

Αυτό αναλύεται στο δεύτερο κεφάλαιο. Εκεί, αρχικά αναφερόμαστε σε σημεία που πρέπει να προσέξει κάποιος ο οποίος σκοπένει να απασχοληθεί με κάποια καλλιέργεια και την επίδραση ήδη υπαρχουσών λογισμικών στις καλλιέργειες. Στο τρίτο γίνεται μία αναφορά σε χρήσιμες τεχνολογίες πάνω στους υπολογιστές, με ποιόν τρόπο μπορούμε να αποθηκεύσουμε δεδομένα καθώς και με ποιούς τρόπους αυτά μπορούν να εμφανιστούν στο χρήστη. Επιπλέον, γίνεται αναφορά σε προ-

γράμματα εγκατάστασης και κλείνοντας το κεφάλαιο, γίνεται σύγκριση των παραπάνω μεθόδων. Στο τέταρτο κεφάλαιο δείχνεται η επιλογή υλοποίησης που προτιμήθηκε από τη σύγκριση του τρίτου κεφαλαίου, ως προς την αποθήκευση δεδομένων, τη διεπαφή του χρήστη και το πρόγραμμα εγκατάστασης. Στο πέμπτο κεφάλαιο αναλύεται εκτενέστερα τί μπορεί κάποιος να κάνει με τις καλλιέργειες και με ποιό τρόπο μπορεί να τις διαχειριστεί, με απουσία υπολογιστή και χρήσης λογισμικού. Το έκτο κεφάλαιο έρχεται να διαδεχθεί το πέμπτο, στο οποίο δείχνεται τί καλύτερο υπάρχει από το χαρτί καθώς η εμφάνιση και η λειτουργία της εφαρμογής. Στο έβδομο κεφάλαιο αναλύεται η διεπαφή χρήστη και με ποιό τρόπο έγινε η υλοποίηση της εφαρμογής. Συνέχεια έχει το όγδοο, η αρχιτεκτονική της εφαρμογής, απ' το πώς έγινε η εγκατάσταση και η δομή των αρχείων έως τη διασύνδεση των κλάσεων που διέπουν την εφαρμογή. Έπειτα, έχοντας δείξει όλα αυτά, στο επόμενο κεφάλαιο δίνεται ένα μεγάλο παράδειγμα χρήσης της εφαρμογής συμπεριλαμβάνοντας όλες τις επιμέρους λειτουργίες. Στο δέκατο κεφάλαιο δείχνουμε τα τελικά συμπεράσματα, τις αδυναμίες και τις επεκτάσεις και τέλος στο ενδέκατο συμπεριλαμβάνεται σχετική βιβλιογραφία.

Κεφάλαιο 2

Καλλιέργειες και ηλεκτρονικοί υπολογιστές

Ο σύγχρονος ρυθμός της ζωής των ανθρώπων και η ανάπτυξη της τεχνολογίας βοήθησαν αισθητά στην εξέλιξη ενός τεχνολογικού μέσου γνωστό ως ηλεκτρονικός υπολογιστής. Είναι γεγονός ότι αποτελεί πλέον αναπόσπαστο κομμάτι στην σημερινή κοινωνία και χρησιμοποιείται σε πολλούς τομείς, ένας από τους οποίους είναι οι καλλιέργειες προϊόντων. Αυτό το εργαλείο διαθέτει τεχνολογίες λογισμικού στο οποίο οποιοδήποτε άτομο, έμπειρο και μη θα μπορεί να ξεκινήσει τη δική του καλλιέργεια χωρίς να έχει τις απαραίτητες γνώσεις. Παρακάτω αναλύονται σημεία ενδιαφέροντος και προσοχής στις καλλιέργειες καθώς και η χρήση του ήδη υπάρχοντος λογισμικού σε αυτές.

2.1 Σημεία ενδιαφέροντος για τις καλλιέργειες γενικά

Προτού ξεκινήσουμε να καλλιεργούμε, σημαντικό είναι να γνωρίζουμε ορισμένα πράγματα τα οποία θα μας βοηθήσουν να κατανοήσουμε καλύτερα την καλλιέργεια. Για τη διευκόλυνση μας, μπορούμε να δημιουργήσουμε ένα γενικό πλάνο όπου θα αναφερόμαστε στο σκοπό και στον τρόπο με τον οποίο η καλλιέργεια θα διεξαχθεί. Βέβαια, τα σημεία τα οποία πρέπει να προσέξουμε ιδιαίτερα είναι αυτά που μπορούν να επηρεάσουν την πορεία της. Τέτοια σημεία είναι πράγματα όπως η τοποθεσία, οι καιρικές συνθήκες και ο αριθμός των καλλιεργειών που θα φυτευτούν.

Για να φυτέψουμε λοιπόν ένα σπόρο πρέπει να σιγουρευτούμε ότι βρίσκεται σε ιδανικό χώμα, πράγμα σημαντικό για την ανάπτυξή του. Όμως είναι απαραίτητο να σημειωθεί πως υπάρχει και απολυμασμένο μίγμα σπόρου και χώματος τα οποία χρησιμοποιούνται κυρίως από ερασιτέχνες, που γνωρίζουν ελάχιστα για τις καλλιέργειες. Ένας παράγοντας που επηρεάζει την παραγωγή των προϊόντων είναι τα καιρικά φαινόμενα. Είναι γνωστό πως το φυτό ή και το φυτώριο για να εξελιχθεί, είναι αναγκαίο να βρίσκεται στο σωστό περιβάλλον στο οποίο θα υπάρχει η κατάλληλη υγρασία που χρειάζεται το φυτό. Το νερό ακόμη, έχει τον κυριότερο ρόλο καθώς ο σπόρος πρέπει να ποτίζεται επαρκώς και να ελέγχεται ανά τακτά χρονικά διαστήματα με σκοπό την ισορρόπηση του πελά του χώματος και της υγρασίας του φυτού. Τέλος, το φυτό για να αναπτυχθεί σωστά πρέπει να βρίσκεται στην σωστή θερμοκρασία περιβάλλοντος και είναι αξιοσημείωτο να αναφερθεί ότι με την αγορά του σπόρου δίνονται ανάλογες οδηγίες που βοηθούν στη σωστή συντήρησή του.

2.2 Τα τεχνολογικά μέσα των γεωργικών καλλιεργειών

Σε παλαιότερες εποχές, οι άνθρωποι ασχολούνταν με τη γεωργία μέσω της οποίας αναπτύχθηκαν φυτικά προϊόντα. Με την όλο και περισσότερη αύξηση των αναγκαιοτήτων των ανθρώπων και τη θέλησή τους για βέλτιστη αποδοτικότητα των προϊόντων τους, σύμβαλλαν ως πρωτογενής τομέας στην ανάπτυξη της τεχνολογίας.

Σύμφωνα με τις μέχρι τώρα γνώσεις ο άνθρωπος καλλιεργούσε τη γη με απλά ξύλινα εργαλεία εκτελώντας όλες οι εργασίες με χρήση της μυϊκής του δύναμης. Βεβαίως, με την εμφάνιση της βιομηχανίας και την αύξηση των πληθυσμών των πόλεων άρχισε να αυξάνεται η ζήτηση γεωργικών προϊόντων με συνέπεια την πολλαπλή προσπάθεια των ανθρώπων να ανακαλύψουν νέα τεχνολογικά μέσα για να μπορέσουν να ανταπεξέλθουν. Αρχικά τους ώθησε να επισκευάσουν έναν ατμοκίνητο κινητήρα ιδανικό για το όργωμα. Στις αρχές του 20ου αιώνα άρχισαν να χρησιμοποιούνται τα πρώτα τρακτέρ με κινητήρες εσωτερικής καύσεως. Αργότερα, εφήυραν διάφορα εργαλεία και μηχανήματα τα οποία σύμβαλλαν στην κοπή και σπορά προϊόντων. Τέτοια εργαλεία είναι ο τυφώνας, ο καλλιεργητής, ο σπαστήρας. Είναι σημαντικό να σημειωθεί πως με αυτή την εξέλιξη οι άνθρωποι δεν χρονοτριβούσαν στα χωράφια και η εργασία τους αν και σκληρή έγινε πιο ευχάριστη. Η ανάπτυξη της τεχνολογίας σημείωσε πρόοδο και στην χημική βιομηχανία η οποία άρχισε να διαδίδει χημικά λιπάσματα και φάρμακα για την καταπολέμηση των γεωργικών ασθενειών των φυτών και άλλων αναγκών.

Συνοψίζοντας, η τεχνολογία συνδεθηκε άμεσα με την γεωργική απασχόληση υποβοηθώντας με αυτό τον τρόπο και την επιμέρους τεχνολογία ως πρωτογενής κλάδος. Η τεχνολογία όμως επέφερε και ένα άλλο μέσο, τον ηλεκτρονικό υπολογιστή το οποίο αποτελεί βασικό εργαλείο στις μέρες μας. Ο υπολογιστής μπορεί να μας ενημερώσει παγκοσμίως και να αποκτήσουμε μέσου αυτού γνώσεις σε πολλούς τομείς, ακόμη και με πολλούς τρόπους πάνω στις καλλιέργειες προϊόντων.

2.3 Χρήση λογισμικού στις καλλιέργειες

Οι ηλεκτρονικοί υπολογιστές αποτελούν ένα μοναδικό μέσο ενημέρωσης, συλλογής πληροφοριών και στατιστικών στοιχείων. Στη συγκεκριμένη περίπτωση αποτελούν συνδετικό κρίκο ανάμεσα στην πληροφορία και καλλιέργεια.

Υπάρχει ευρεία γκάμα δυνατοτήτων του ηλεκτρονικού υπολογιστή. Αξιοσημείωτη είναι η χρήση του ως μέσου καταγραφής δεδομένων, ως μέσου παρακολούθησης γεγονότων και στατιστικών στοιχείων και για εξίσου ενημέρωση μέσω ίντερνετ. Αυτές οι δυνατότητες, δημιουργήθηκαν από την διάφορες ανάγκες του ανθρώπου στην καθημερινή ζωή και σε αυτό το σημείο ήρθε ο υπολογιστής να υποβοηθήσει.

Ένα πολύ σημαντικό εργαλείο το οποίο αποτελεί πια καθημερινότητα στη ζωή του ανθρώπου, είναι η χρήση του διαδικτύου. Ένα εργαλείο, το οποίο χρησιμοποιείται να μόν για ψυχαγωγία, αλλά και για διαχείριση πληροφοριών. Η διαχείριση πληροφοριών εκτείνεται σε πολλούς τομείς. Στην εκπαίδευση, σε έναν εναλλακτικό τρόπο επικοινωνίας για τους ανθρώπους, καθώς και σε χώρους εργασίας. Στις καλλιέργειες τώρα μπορεί να χρησιμοποιηθεί με ποικίλους τρόπους. Ένας από αυτούς, είναι η ενημέρωση δεδομένων μέσω του ίντερνετ πάνω σε νεότερα για καλλιέργειες μέσω διαδικτυακών εφημερίδων και περιοδικών, καινούργιους τύπους φυτορίων και πολλά άλλα. Είναι σημαντικό να αναφερθεί πως στο διαδύκτιο υπάρχουν έτοιμα πρότυπα καλλιεργειών για την

ευκολία των έμπειρων η ακόμη και άπειρων με τον τομέα ανθρώπων. Μέσω στατιστικών στοιχείων, κάποιος μπορεί να δημιουργήσει μια ολοκληρωμένη γενική άποψη για την καλλιέργεια και να καταλήξει στα κατάλληλα συμπεράσματα ώστε να προβεί στην καλύτερη λύση. Επίσης υπάρχουν ορισμένες μηχανές αναζήτησης που διευκολύνουν κατά πολύ σε οποιαδήποτε αναζήτηση επιθυμούμε. Μία από αυτές είναι η google. Χάρη σε αυτή, μπορούμε να αναζητήσουμε στατιστικά στοιχεία και εικόνες πάνω στις καλλιέργειες. Αξιοσημείωτο είναι ότι μπορούν να αποθηκευτούν δεδομένα από το δίκτυο στον υπολογιστή και να χρησιμοποιηθούν όποτε ο χρήστης το επιθυμεί. Με τη χρήση του διαδικτύου μπορούμε να ενημερωθούμε και να δράσουμε άμεσα αν παρατηρήσουμε κάποια αλλαγή στα καιρικά φαινόμενα, ώστε να προλάβουμε κάποια τυχόν καταστροφή. Από την άλλη υπάρχει ένα ποσοστό ανθρώπων που δεν προτιμούν το ίντερνετ ως λύση και χρησιμοποιούν άλλα μέσα για καταγραφή δεδομένων των καλλιεργειών. Άλλα τέτοια μέσα είναι το σημειωματάριο και τα εργαλεία του office, εργαλεία εγκατεστημένα σε κάθε υπολογιστή. Ακόμη, μπορούμε να χρησιμοποιήσουμε το microsoft word για να καταγράψουμε λεπτομέρειες καλλιεργειών. Βεβαία, μοναδική είναι και η λειτουργία του microsoft excel με την οποία μπορούμε να διαχωρίσουμε τις καλλιέργειες και να καταγράψουμε δεδομένα πάνω σε αυτές, ακόμη και να καταγράψουμε στατιστικά στοιχεία τους. Τέλος, λογισμικό βάσεων δεδομένων μπορεί να χρησιμοποιηθεί μιας και έχει τη δυνατότητα αποθήκευσης πληθώρας καλλιεργειών καθεμιά με τα δικά της χαρακτηριστικά ιδανικά για το διαχωρισμό, τη διασύνδεση και την κατανόησή τους. Παρόμοιο λογισμικό αποτελεί το access.

Κεφάλαιο 3

Χρήσιμες τεχνολογίες

Είδαμε στο προηγούμενο κεφάλαιο, σημεία προσοχής για τις καλλιέργειες και λογισμικά που χρησιμοποιούνται ή μπορούν να χρησιμοποιηθούν σε αυτές. Εκτός αυτού, η σύγχρονη τεχνολογία δίνει τη δυνατότητα μέσω γλωσσών προγραμματισμού να αναπτύξει κάποιος το δικό του λογισμικό για να καλύψει τις δικές του ανάγκες. Οι γλώσσες προγραμματισμού επιτρέπουν την αναπαράσταση αντικειμένων, εικόνων σε εύχρηστο γραφικό περιβάλλον, ένα γραφικό περιβάλλον στο οποίο ο χρήστης μπορεί να αλληλοεπιδρά και να κάνει επιλογές αναλόγως του φάσματος των επιλογών που του δίνονται και αναλόγως των αναγκών του. Μερικά από τα περιβάλλοντα αυτά αποτελούν οι JAVA, Visual Basic, Delphi, C# ,PHP, Ruby On Rails. Το κάθε περιβάλλον έχει τη δική του λειτουργικότητα και ευκολία στον τρόπο διαχείρισης. Επίσης διαφέρουν και στον τρόπο αναπαράστασης των δεδομένων. Παραδείγματος χάρη, κάποιο εμφανίζει τα δεδομένα σε φόρμα και άλλο σε ιστοσελίδα. Σκοπός μας ήταν να χρησιμοποιηθεί ένας από τους πιο εύχρηστους και διαδεδομένους απο αυτούς.

3.1 Τρόποι αποθήκευσης δεδομένων

Δύο είναι οι επικρατέστεροι τρόποι αποθήκευσης δεδομένων και χωρίζονται στις βάσεις δεδομένων και τα αρχεία κειμένου. Οι δύο αυτοί τρόποι, έχουν διάφορα πλεονεκτήματα και μειονεκτήματα αναλόγως τη χρήση τους . Παρακάτω αναλύεται η λειτουργικότητα και η εφαρμογή τους και στη συνέχεια γίνεται σύγκριση των δύο αυτών μεθόδων.

3.1.1 Βάσεις δεδομένων

Οι βάσεις δεδομένων χρησιμοποιούνται για να μπορούμε να αποθηκεύσουμε, να επεξεργαστούμε αλλά και να εκμεταλλευτούμε το μεγάλο όγκο των πληροφοριών που αυξάνονται καθημερινά με αλματώδεις ρυθμούς. Το δεύτερο συνθετικό των Βάσεων δεδομένων είναι τα δεδομένα. Ακόμη, ο υπολογιστής απαρτίζεται ο ίδιος από δεδομένα. Είναι σημαντικό να επισημανθεί πως τα δεδομένα οργανώνονται καλύτερα με τη χρήση των ηλεκτρονικών υπολογιστών. Χρήση βάσης δεδομένων σε ηλεκτρονικό υπολογιστή, ταξινομεί τα δεδομένα σε αρχεία εγγραφών. Τα δεδομένα που συζητιούνται, είναι τρόποι αναπαράστασης εννοιών και γεγονότων που μπορούμε να υποβάλλουμε σε κατάλληλη διαχείριση και επεξεργασία. Με την συλλογή και την αποθήκευση των δεδομένων δεν μπορούμε να λύσουμε ακριβώς το πρόβλημα της σωστής οργάνωσης και ταξινό-

μησης των δεδομένων. Τα δεδομένα πρέπει να τα οργανώσουμε σωστά, ώστε να μπορούμε να τα εντοπίζουμε και να τα αξιοποιούμε γρηγορότερα και πιο αποτελεσματικά. Για παράδειγμα, εάν όλες μας οι καλλιέργειες δεν είναι καταχωρημένες αλφαβητικά σύμφωνα με το όνομά τους, αλλά σε τυχαίες θέσεις τότε έχουμε μη σωστή οργάνωση δεδομένων, εκτός και αν δεν υπάρχει μεγάλος αριθμός καλλιεργειών. Για την καλύτερη συνολική επεξεργασία των εγγραφών δημιουργήθηκαν τα συστήματα διαχείρισης βάσεων δεδομένων. Μεγάλο πλεονέκτημα είναι ότι οι βάσεις δεδομένων αποτελούν ελεύθερο λογισμικό για εγκατάσταση σε κάθε υπολογιστή και αυτό γίνεται με τη σωστή επιλογή του server.

RDBMS και χαρακτηριστικά τους

Τα συστήματα RDBMS ή αλλιώς Συστήματα Διαχείρισης Βάσεων Δεδομένων, είναι η κινητήρια δύναμη για τη σωστή οργάνωση των πληροφοριών των εφαρμογών του ηλεκτρονικού υπολογιστή. Βέβαια εξελίσσονται όλο και περισσότερο προσφέροντας πιο σύγχρονα συστήματα αποθήκευσης, ανάκτησης και διανομής δεδομένων. Οι βάσεις δεδομένων είναι μια ολοκληρωμένη συλλογή δεδομένων που επιτρέπει τη βέλτιστη πρόσβαση σε πληροφορίες που σχετίζονται με ένα συγκεκριμένο κομμάτι της. Αξιοσημείωτο είναι το γεγονός ότι η RDBMS είναι το λογισμικό που διαχειρίζεται τη σχεσιακή βάση δεδομένων. Να σημειωθεί ότι σχεσιακό μοντέλο είναι σήμερα το βασικό μοντέλο δεδομένων για εφαρμογές επεξεργασίας δεδομένων, το οποίο χρησιμοποιεί πίνακες, στήλες και βασικά γνωρίσματα για διασύνδεση των δεδομένων. Στο χώρο των RDBMS σήμερα τα πιο γνωστά από τα συστήματα που κυκλοφορούν είναι τα Oracle Database, DB2, Microsoft SQL Server, Teradata, Sybase, Informix, MySQL, PostgreSQL.

3.1.2 Αρχεία

Τα αρχεία είναι άλλη μια κατηγορία τρόπου αποθήκευσης δεδομένων όπου αποτελούν δεδομένα διατεταγμένα σε γραμμές. Τα αρχεία αποτελούν τον πιο συνηθισμένο τύπο αποθήκευσης δεδομένων. Στον υπολογιστή υπάρχουν ήδη εγκατεστημένα προγράμματα ειδικά για το άνοιγμα αυτού του είδους αρχείων, όπως το wordPad ή σημειωματάριο. Υπάρχει όμως μία παραπλανητικότητα στην ερμηνεία των όρων αρχείο κειμένου και ψηφιακό αρχείο δίνοντας την εντύπωση ότι οι πληροφορίες σε ένα αρχείο κειμένου δεν αποθηκεύονται ως δυαδικά δεδομένα. Όλα τα δεδομένα σε ένα υπολογιστή αποθηκεύονται ως δυαδικά ψηφία, αρχεία δηλαδή σε ψηφιακή μορφή. Οι όροι αναφέρονται στη δομή και τη μορφή της πληροφορίας σε bits. Υπάρχει πληθώρα τρόπων επεξεργασίας αρχείων κειμένου.

Είδη επεξεργασίας και χαρακτηριστικά αρχείων κειμένου

Κάποιοι τύποι επεξεργασίας αρχείων κειμένου είναι μικρά προγράμματα και εύκολα ενώ άλλα έχουν μεγαλύτερη συνθετικότητα. Κάποια από αυτά τα προγράμματα είναι ο επεξεργαστής pico που περιλαμβάνεται στο λειτουργικό Unix. Σημαντικός είναι ο Microsoft Word και ο WordPerfect, όπου τα δεδομένα που εισάγουμε αποτελούν διαμορφωμένο κείμενο αντίθετα με το απλό σημειωματάριο. Οι πιο πολλοί επεξεργαστές κειμένου μπορούν να διαβάσουν και να γράψουν αρχεία σε μορφή απλού κειμένου και να αποθηκευτούν από τους συντάκτες τους. Οι πιο απλοί επεξεργαστές κειμένου μπορούν να διαβάσουν τα αρχεία που βρίσκονται στην μνήμη του υπολογιστή. Βασικές

ιδιότητες των αρχείων είναι η αποκοπή, αντιγραφή, επικόλληση και αναίρεση. Τα αρχεία κειμένου υποδιαιρούνται σε δυαδικά ή ψηφιακά και μη δυαδικά. Το κάθε ένα από αυτά, παρέχει ένα συγκεκριμένο είδος πληροφορίας και απαιτεί διαφορετικό τρόπο αποθήκευσης και χειρισμό. Το δυαδικό αρχείο περιέχει δεδομένα σε ειδική μορφή και απαιτεί μια συγκεκριμένη ερμηνεία των bits του. Στα αρχεία κειμένου, τα δεδομένα σε bytes είναι χαρακτηριστές. Τα αρχεία κειμένου μπορεί να περιέχονται σε πρόγραμμα, όπως είναι η java. Οι δύο τύποι αρχείων έχουν δικές τους λειτουργίες. Για τα αρχεία κειμένου, μπορεί να γίνει δημιουργία αρχείου, διαγραφή, άνοιγμα αρχείου, κλείσιμο, διάβασμα δεδομένων και τέλος αποθήκευση δεδομένων. Ενώ για τα δυαδικά, μπορεί να γίνει γράψιμο δεδομένων στο τέλος του αρχείου, διαγραφή δεδομένων μετονομασία αρχείου και αντιγραφή. Βέβαια τα πιο διαδεδομένα είναι τα αρχεία κειμένου. Είναι μόνον τοπικά στον υπολογιστή αλλά και είναι ευκολότερα στο χειρισμό σε αντίθεση των δυαδικών.

3.1.3 Σύγκριση μεθόδων

Συγκρίνοντας τις δύο τεχνικές, συμπεραίνουμε ότι οι βάσεις δεδομένων αποτελούν ένα πολύ χρήσιμο μέσο αποθήκευσης και διαχείρισης δεδομένων γιατί δέχονται μεγαλύτερο όγκο δεδομένων. Τα αρχεία κειμένου είναι πολύ πιο απλοϊκά καθώς δεν απαιτείται σύνδεση στο δίκτυο σε αντίθεση με τις βάσεις δεδομένων που απαιτούν εγκατάσταση κάποιου server συστήματος. Ο χρήστης μπορεί ευκολότερα να παρέμβει στο σύστημα με αλλαγές στα αρχεία απλά αλλάζοντας τα δεδομένα του αρχείου και αποθηκεύοντάς το. Έτσι, τα αρχεία κειμένου αποτελούν μία πολύ πιο απλή και εύχρηστη λύση.

3.2 Εμφάνιση δεδομένων

Παραπάνω παρουσιάστηκαν οι δύο τρόποι αποθήκευσης δεδομένων και η χρησιμότητά τους. Η αποθήκευση των δεδομένων όμως είναι κάτι που συμβαίνει στο παρασκήνιο του υπολογιστή ή αλλιώς στο background. Κομμάτι ιδιαίτερης σημασίας αποτελεί το πώς φαίνονται τα δεδομένα ενός προγράμματος στο προσκήνιο δηλαδή στην οθόνη του υπολογιστή. Με άλλα λόγια μας ενδιαφέρει σημαντικά η διεπαφή μεταξύ υπολογιστή και χρήστη να είναι όσο το δυνατόν πιο απλοϊκή για τον ίδιο. Προφανώς μας ενδιαφέρει να εμφανίζονται τα αποτελέσματα με έναν οργανωμένο και εύληπτο για τον άνθρωπο τρόπο

ΔΙΕΠΑΦΗ ΧΡΗΣΤΗ:

Ο όρος διεπαφή χρήστη αναφέρεται στον τρόπο με τον οποίο αλληλεπιδρά κάποιος με τον υπολογιστή. Πιο συνηθισμένες διεπαφές είναι η γραφική, διεπαφή χρήστη και κονσόλας και η διεπαφή παγκόσμιου ιστού.

Η γραφική δέχεται πληροφορίες από συσκευές εισόδου όπως είναι το ποντίκι και το πληκτρολόγιο και εξάγει γραφικά τα δεδομένα στην οθόνη του υπολογιστή. Στη διεπαφή χρήστη και κονσόλας, χρησιμοποιούνται δεδομένα των οποίων είσοδος και έξοδος είναι το κείμενο και οι εφαρμογές τους βασίζονται στην κονσόλα του κάθε συστήματος. Η τελευταία διεπαφή είναι αυτή του παγκοσμίου ιστού ονομαζόμενη WUI, που απαρτίζεται από ιστοσελίδες και προβάλλονται μέσω της περιήγησης που είναι ένα πρόγραμμα του Διαδυκτίου. Βέβαια η τελευταία διεπαφή είναι η πιο εύχρηστη καθώς παρουσιάζει πολλά πλεονεκτήματα όπως συντηρησιμότητα.

Παρουσιάσαμε πιο πάνω τον ορισμό της διεπαφής χρήστη. Όσο αφορά την εμφάνιση δεδομένων, δύο είναι οι τρόποι εμφάνισης τους στην οθόνη ενός υπολογιστή, ο πρώτος σε φόρμα και ο δεύτερος σε ιστοσελίδα.

Δόθηκε πιο πάνω εξήγηση για το τί είναι διεπαφή χρήστη. Όσο αφορά την εμφάνιση δεδομένων, δύο είναι οι τρόποι εμφάνισης δεδομένων στην οθόνη ενός υπολογιστή. Εμφάνιση σε φόρμα και ο δεύτερος εμφάνιση σε ιστοσελίδα.

3.2.1 Εμφάνιση σε φόρμα

Υπάρχουν πολλοί τρόποι με τους οποίους μπορεί να αναπτυχθεί γραφική διεπαφή σε ηλεκτρονικό υπολογιστή. Η Java, η Visual Basic, η Delphi και η C# κυριαρχούν σε αυτούς τους τρόπους. Παρακάτω γίνεται μια αναφορά αυτά τα προγράμματα.

Η Java είναι μια αντικειμενοστρεφής γλώσσα προγραμματισμού και έχει μηχανισμούς που καθορίζουν κλάσεις και δημιουργούν αντικείμενα που είναι στιγμιότυπα αυτών. Μπορεί ακόμη, να εκτελέσει πρόγραμμα σε δύο μορφές. Την απλή τερματική μορφή και την μορφή γραφικού περιβάλλοντος. Η μορφή γραφικού περιβάλλοντος είναι γνωστή ως gui, όπου gui το graphical user interface. Ένα gui κάποιου προγράμματος, αποτελεί ένα εύκολο περιβάλλον χειρισμού για το χρήστη το οποίο αποτελείται από γραφικά αντικείμενα. Για συγγραφή σε γραφικό περιβάλλον της Java χρησιμοποιείται είτε Swing είτε JavaFx όπως επίσης και συγκεκριμένες βιβλιοθήκες για αυτό το σκοπό.

Η Visual Basic προέρχεται από τη BASIC και επιτρέπει την ταχύρυθμη ανάπτυξη εφαρμογών σε Gui, την πρόσβαση σε βάσεις δεδομένων χρησιμοποιώντας αντικείμενα και τη δημιουργία στοιχείων ελέγχου ActiveX και αντικειμένων. Η Visual Basic είναι εύκολη στην εκμάθηση και το χειρισμό. Η γλώσσα επιτρέπει την ανάπτυξη απλών εφαρμογών gui αλλά και πολύπλοκων εφαρμογών. Ο προγραμματισμός σε VB είναι συνδυασμός κάποιων στοιχείων σε μια φόρμα και έχει την δυνατότητα επιπλέον προγραμματισμού σε αυτά. Έχει μια μεγάλη βιβλιοθήκη με βοηθητικά αντικείμενα καθώς και βασική αντικειμενοστραφή υποστήριξη. Αλλά υπάρχουν βασικοί περιορισμοί στο περιβάλλον διαχείρισης οι οποίοι δεν επιτρέπουν την επίτευξη ορισμένων στόχων.

Η Delphi παρέχει ολοκληρωμένο περιβάλλον διαχείρισης για δημιουργία προγράμματος τύπου κονσόλας, γραφικού περιβάλλοντος, ίντερνετ και εφαρμογές κινητών. Οι compiler της Delphi χρησιμοποιούν τη δική του διάλεκτο βασισμένη στην Pascal και τα προγράμματα που δημιουργούνται από την Delphi υποστηρίζονται από πολλά περιβάλλοντα. Πιο συγκεκριμένα, 32- και 64-bit Windows, 32-bit Mac OS X, iOS και Android. Χαρακτηρίζεται ως υψηλού επιπέδου προγραμματιστική γλώσσα και είναι εύκολη στη χρήση. Τη διέπουν αποδοτικότητα και υποστήριξη πρόσβασης σε προγραμματισμό χαμηλού επιπέδου. Είναι αντικειμενοστρεφής ομοίως με τη Java και έτσι διακατέχεται από κλάσεις. Προσφέρει ταχεία ανάπτυξη λογισμικού και έχει μία βιβλιοθήκη με συστατικά που βοηθάνε σε αυτό το σκοπό. Μειονέκτημα αποτελεί το ότι καθώς ενημερώνονται οι εκδόσεις της Delphi, αυτή καθίσταται ασυμβίβαστη με ορισμένα λογισμικά. Επίσης, νεότερες εκδόσεις της delphi έχουν παρουσιάσει αστάθεια.

Τέλος, η C#. Βασίζεται στη C και C++ και περιέχει λειτουργίες παρόμοιες της Java όπως πολυμορφισμό. Χρησιμοποιείται σε Web υπηρεσίες στην .NET πλατφόρμα, που είναι βασισμένες στην XML και δημιουργήθηκε για να βελτιώσει κυρίως υπηρεσίες διαδικτύου. Επιτρέπει δημιουργία γραφικού περιβάλλοντος, χρησιμοποιώντας το Graphics Object. Το Graphics Object, απλοποιεί τη

δημιουργία γραφικού περιβάλλοντος, διαιρώντας όλες τις εντολές που δίνει ο χρήστης και επεξεργάζοντάς τα, το κάθε ένα χωριστά. Έτσι ο χρήστης το μόνο που κάνει είναι να πεί στο πρόγραμμα τί και πού να σχεδιάσει κάτι. Η C# όμως, δεν αποτελεί σύστημα πραγματικού χρόνου, δηλαδή κάνει compile πιο αργά. Επίσης, είναι λιγότερο ευέλικτη γιατί εξαρτάται κυρίως απ' το . NET framework.

3.2.2 Εμφάνιση σε ιστοσελίδα

Ανάπτυξη διαδικτυακών εφαρμογών σήμερα

Είναι αξιοσημείωτο το γεγονός ότι το ίντερνετ είναι πλέον αναγκαίο στην καθημερινότητα του ανθρώπου. Ο πλούτος της παρεχόμενης πληροφορίας, η αμεσότητα ενημέρωσης, οι υπηρεσίες και τα εργαλεία είναι αυτό που σήμερα αποκαλείται διαδίκτυο.

Το Internet απέκτησε το σημαντικότερο συστατικό του, το World Wide Web. Με τη διάδοσή του έδωσε νέα διάσταση στις διαδικασίες πρόσβασης πληροφοριών. Οι πληροφορίες πλέον δεν περιέχουν μόνο ένα απλό κείμενο αλλά εικόνες, ήχους, κίνηση και πολλά ακόμα χαρακτηριστικά. Μέσω του WWW ο άνθρωπος μπορεί να έχει πρόσβαση σε ένα τεράστιο όγκο πληροφοριών μέσω οποιουδήποτε υπολογιστή, έτσι ώστε η χρήση του να καθίσταται πιο εύκολη. Το διαδίκτυο έχει δημιουργήσει μία τεράστια αγορά πληροφοριών. Παρατηρείται μία συγκέντρωση γνώσης και πολιτιστικής κληρονομιάς που σχετίζεται άμεσα με το Ίντερνετ. Από τη στιγμή που το διαδίκτυο είναι ένα δίκτυο συνδεδεμένων υπολογιστών, κάθε χρήστης έχει την δυνατότητα να μοιραστεί πληροφορίες με άλλους χρήστες γενόμενος, πολλές φορές, ο ίδιος δημιουργός και πάροχος των πληροφοριών αυτών. Δεν υπάρχει άμεσος έλεγχος των πληροφοριών που ανεβαίνουν στο διαδίκτυο από κάποιον ιεραρχικά ανώτερο χρήστη ή οργανισμό. Ο όγκος της πληροφορίας στο διαδίκτυο είναι μεγάλος. Παρ' όλα αυτά, υπάρχουν πληροφορίες ευκολότερα και δυσκολότερα προσβάσιμες από τον χρήστη.

Η χρήση της Java οδήγησε στην αύξηση των εφαρμογών διαδικτύου. Αυτό είχε ως αποτέλεσμα την ανάπτυξη και εξέλιξη πολλών εργαλείων με αυτοσκοπό την συντομότερη και ευκολότερη δημιουργία εφαρμογών όπως είναι η JavaScript, η VBScript, η PHP, η C#, η ASP. Υπάρχουν και εφαρμογές οι οποίες δεν είναι αποκλειστικές μόνο του ίντερνετ. Κάποιες από αυτές είναι η MySQL, ο Microsoft SQL Server.

Η αναφορά που δώσαμε οδηγεί στο συμπέρασμα πως το Internet σήμερα δεν είναι μόνο εργαλείο ψυχαγωγίας, αλλά ένα εργαλείο που εξυπηρετεί σε σημαντικούς τομείς της ζωής, παρέχοντας μεγάλο όγκο πληροφοριών στο χρήστη χωρίς όμως άμεσο έλεγχο αυτών.

Τρόποι εμφάνισης δεδομένων σε ιστοσελίδα

Το ίντερνετ είναι πλέον μια αναγκαία καθημερινότητα των ανθρώπων. Αποτελείται από πολλές ιστοσελίδες, οι οποίες με τη σειρά τους περιλαμβάνουν ιδιαίτερα λειτουργικά εργαλεία για πιο γρήγορη περιήγηση. Στατικές ονομάζονται οι ιστοσελίδες, που έχουν μόνιμο περιεχόμενο. Αυτό δε σημαίνει ότι το περιεχόμενο τους πρέπει να είναι στατικό, αντιθέτως μπορεί να έχει πολλά κινούμενα γραφικά. Δυναμικές είναι οι ιστοσελίδες στις οποίες ο επισκέπτης αλληλεπιδρά με την ίδια τη σελίδα και σε αυτές περιλαμβάνονται Web κώδικας, scripting, όπως η PHP ή ASP. Οι δυναμικές ιστοσελίδες είναι ευκολότερο να συντηρηθούν σε σχέση με τις στατικές ιστοσελίδες, καθώς η καθεμία από τις στατικές περιέχει μοναδικό περιεχόμενο που μπορεί ανοιχτεί, επεξεργαστεί και δημοσιευτεί κάθε φορά που γίνεται μια αλλαγή σε αυτό. Επίσης, οι δυναμικές σελίδες από

την άλλη πλευρά έχουν πρόσβαση σε πληροφορίες από μία βάση δεδομένων. Ως έκ τούτου για να τροποποιηθεί το περιεχόμενο μιας δυναμικής σελίδας, ο διαχειριστής της ιστοσελίδας χρειάζεται μόνο να ενημερώσει ένα αρχείο βάσης δεδομένων. Αυτό είναι ιδιαίτερα χρήσιμο για τα μεγάλα sites που περιέχουν εκατοντάδες ή χιλιάδες σελίδες. Επιπλέον, παρέχει τη δυνατότητα σε πολλούς χρήστες να ενημερώσουν το περιεχόμενο μιας ιστοσελίδας χωρίς να επεξεργαστούν τη διάταξή τους.

Υπάρχουν δύο τρόποι για συγγραφή και εμφάνιση ιστοσελίδας. Η PHP που αναφέρθηκε και προηγουμένως και η Ruby On Rails.

Η PHP είναι μια γλώσσα προγραμματισμού για τη δημιουργία σελίδων web με δυναμικό περιεχόμενο. Μια σελίδα PHP περνά από επεξεργασία από ένα συμβατό διακομιστή του παγκοσμίου ιστού όπως ο Apache και παράγεται το τελικό περιεχόμενο, το οποίο εμφανίζεται σε μορφή κώδικα HTML στους χρήστες. Η PHP, χρησιμοποιείται ευρέως σήμερα για συγγραφή κώδικα HTML και δημιουργία δυναμικών σελίδων και είναι αρκετά εύκολος στην εκμάθηση. Όταν ένα αρχείο έχει επέκταση .php, θα πρέπει ο server να είναι ρυθμισμένος για να επεξεργάζεται κώδικα PHP με σκοπό τη σωστή εκτέλεση του αρχείου. Ο διακομιστής Apache, που χρησιμοποιείται σήμερα ευρέως σε συστήματα με λειτουργικό Linux και Microsoft Windows, υποστηρίζει εξ' ορισμού την εκτέλεση κώδικα PHP. Ένα μεγάλο θετικό της είναι ότι επιτρέπει τη διασύνδεσή της με βάση δεδομένων. Η PHP, είναι μια εύκολη γλώσσα αλλά και για αυτό το λόγο θεωρείται αδύναμη. Οι εντολές που την αποτελούν είναι ελάχιστες. Ακόμη, μειονέκτημα αποτελεί το γεγονός ότι δεν εξελίσσεται συνεχώς και για αυτό το λόγο δεν υπάρχει σταθερή έκδοσή της.

Το Ruby on Rails, συχνά Rails είναι ένα πλαίσιο ανάπτυξης Web εφαρμογών γραμμένο σε Ruby. Χρησιμοποιείται από τους προγραμματιστές για το λόγο ότι επιτρέπει ταχεία ανάπτυξη λογισμικού. Όντας πλαίσιο ανάπτυξης εφαρμογών γραμμένο σε Ruby κληρονομεί τα βασικά χαρακτηριστικά της Ruby. Το Rails είναι δυναμική, ανακλαστική, αντικειμενοστρεφής γλώσσα προγραμματισμού. Το Ruby on Rails περιλαμβάνει εργαλεία που διευκολύνουν κοινές προγραμματιστικές εργασίες, όπως η δημιουργία σκελετών προγραμμάτων που μπορεί να δημιουργήσει αυτόματα κάποια από τα μοντέλα και τις όψεις που χρειάζεται μια βασική σελίδα ιστού. Επίσης περιέχει τον WEBrick, έναν απλό εξυπηρετητή Ιστού σε Ruby και το Rake, ένα σύστημα κατασκευής προγραμμάτων. Αυτά τα εργαλεία, σε συνδυασμό με το Rails, προσφέρουν ένα βασικό περιβάλλον ανάπτυξης καθιστώντας εύκολη και ταχεία την ανάπτυξη εφαρμογών κυρίως σε επαγγελματικό επίπεδο. Βέβαια, βασίζεται σε έναν εξυπηρετητή ιστού όπως οι Mongrel και Apache για την εκτέλεσή του και προσφέρει σαν μορφές εξόδου HTML και XML. Ακόμη μία δυνατότητα του Ruby, είναι ότι μπορεί να συνδεθεί με μια βάση δεδομένων όπως η MySQL και έναν εξυπηρετητή ιστού όπως ο Apache. Είναι χωρισμένο σε διάφορα πακέτα, το ActiveRecord, το ActiveResource που παρέχει web services, το ActionPack, το ActiveSupport και το ActionMailer. Το Rails είναι πολύ χρήσιμη γλώσσα αλλά αρκετά δύσκολη στην εκμάθηση. Το ActiveRecord κάνει τη ζωή πιο εύκολη, αλλά θα μπορούσε να οδηγήσει σε κακό σχεδιασμό βάσεων δεδομένων, αν κάποιος πραγματικά δεν γνωρίζει τι κάνει. Εκτός των άλλων πρέπει να γίνουν κάποιες αρχικές χρονοβόρες ρυθμίσεις πριν κάποιος το χρησιμοποιήσει.

Εκ των δύο αυτών η PHP αποτελεί μια αρκετά απλή γλώσσα ανάπτυξης ιστοσελίδων, προσιτή σε όλους και χρησιμοποιείται αρκετά συχνά για αυτό το ακριβώς το λόγο. Το αρνητικό της

είναι ότι δεν έχει πολλά χαρακτηριστικά, αυτά της αντικειμενοστραφούς σχεδίασης ενώ το Ruby On Rails είναι αντικειμενοστραφές. Το αρνητικό του Rails είναι η εκμάθησή του, η οποία είναι δύσκολη και κάποιες αρχικές ρυθμίσεις πριν ο χρήστης το χρησιμοποιήσει. Αν κάποια από τις ρυθμίσεις δεν γίνει σωστά το Ruby δε θα εκτελεστεί. Όμως παρ' όλη τη δυσκολία του, το Ruby παρέχει γρήγορα αποτελέσματα και εύκολη διασύνδεση με τη βάση. Έτσι, μπορούμε να πούμε ότι μία πραγματική σύνδεση έκ των δύο δεν μπορεί να πραγματοποιηθεί καθώς το ένα έχει τα θετικά και τα αρνητικά του. Η επιλογή εξαρτάται καθαρά από τις προτιμήσεις του χρήστη και το είδος εργασίας που επιθυμεί να περατώσει.

3.2.3 Σύγκριση μεθόδων

Συγκρίνοντας τις παραπάνω μεθόδους, έχουμε :

Η PHP απαιτεί εγκατάσταση server, όπως παραδείγματος χάρη του Apache. Αποτελεί αδύναμη σε εισαγωγικά γλώσσα. Δηλαδή το προγραμματιστικό ύφος της είναι απλό και δεν ακολουθεί συγκεκριμένα πρότυπα όπως των άλλων γλωσσών προγραμματισμού. Αυτό μπορεί να κάνει την PHP πιο ελκυστική για τους προγραμματιστές αλλά απ' την άλλη μπορεί να περιπλέξει ορισμένα καθήκοντα. Ακόμη, η PHP αποτελεί open source λογισμικό και είναι platform independent. Χρησιμοποιείται ευρέως μαζί με βάσεις δεδομένων για συγκεκριμένους σκοπούς. Στην PHP, η Java Virtual Machine κάνει επανεκκίνηση μετά από κάθε αίτημα. Αυτό μπορεί να οδηγήσει σε επιπλέον προβλήματα απόδοσης. Κάποιος προγραμματιστής μπορεί να επιλέξει την PHP αν δεν έχει πολύ χρόνο για την ολοκλήρωση ενός έργου λαμβάνοντας υπόψη όμως ότι επεκτασιμότητα, επιδόσεις και ασφάλεια είναι ασθενή χαρακτηριστικά γνωρίσματα της PHP.

Η C#, δεν εκτελείται σε όλες τις πλατφόρμες. Μπορεί να τρέξει σε Windows και Mac OS - X, αλλά όχι σε Linux. Επίσης, έχει αρκετά παρόμοια σύνταξη και στυλ προγραμματισμού με της Java. Εάν κάποιος σκοπεύει να αναπτύξει εφαρμογή για επιφάνεια εργασίας των Windows ή για κάποιο Windows phone, η C#, αποτελεί ιδανική επιλογή.

Ακριβώς όπως η Python, ο μεγαλύτερος λόγος που οι προγραμματιστές προτιμούν τη Ruby πάνω από τις άλλες γλώσσες, είναι ότι μια λειτουργία που υλοποιείται σε Ruby θα πάρει λιγότερες γραμμές κώδικα σε σύγκριση με άλλες. Αυτό καθιστά ευκολότερο για τους προγραμματιστές να διαχειριστούν τον κώδικα που γράφουν σε Ruby, σε σχέση με άλλα προγράμματα. Η Ruby On Rails, είναι συνήθως προσανατολισμένη σε βάσεις δεδομένων για επαγγελματικούς σκοπούς.

Η Visual Basic διαφοροποιείται συντακτικά σε πολλά σημεία με τη C#, αλλά οι δύο αυτές γλώσσες αποτελούν μέρος της ίδιας . NET Framework πλατφόρμας ανάπτυξης και έχουν και οι δύο αναπτυχθεί από τη microsoft. Δεν αποτελεί συχνή επιλογή ανάπτυξης λογισμικού.

Java υπάρχει σε κάθε σχεδόν μηχανήμα, ευρέως χρησιμοποιούμενη. Δεν απαιτεί compile σε συγκεκριμένο λογισμικό σε αντίθεση με τις υπόλοιπες. Έχει τη δυνατότητα κάποιος να συγγράψει πρόγραμμα τόσο σε γραφικό επίπεδο και μή. Μπορεί είτε να συνδεθεί με βάσεις δεδομένων για δυναμική διαχείριση δεδομένων είτε να χρησιμοποιηθεί για διαβάσμα και επεξεργασία δεδομένων από αρχεία. Αποτελεί ελεύθερο λογισμικό για εγκατάσταση μέσω internet. Προϋπόθεση απαιτεί κατά την αρχική εγκατάσταση της Java, του Java runtime enviroment ή αλλιώς JRE και του Java virtual machine ή αλλιώς JVM. Java μπορεί να χρησιμοποιηθεί για χρησιμοποιηθεί για δημιουργία προγραμμάτων σε android, λογισμικό ευρέως εφαρμόσιμο σε κινητά. Τέλος, έχει πάρα πολλές βιβλιοθήκες διαθέσιμες προς το χρήστη. Χαρακτηριστικά αρνητικά αποτελούν ο χρόνος που κάνει compile και οι πάρα πολλές βιβλιοθήκες οι οποίες δυσκολεύουν το χρήστη. Αποτελεί τη συνηθέ-

στερη επιλογή του χρήστη.
,

Κεφάλαιο 4

Δική μας επιλογή - επιλογή υλοποίησης

Στα προηγούμενα κεφάλαια έγινε αναφορά σε γλώσσες προγραμματισμού στις οποίες δίνεται η δυνατότητα να χρησιμοποιηθούν για δημιουργία γραφικού περιβάλλοντος αλληλεπίδρασης ανθρώπου μηχανής και μερικά από τα προγράμματα εγκατάστασης που είναι ελεύθερα διαθέσιμα στην αγορά για τη δημιουργία του εκτελέσιμου αρχείου του προγράμματος ούτως ώστε ο κάθε χρήστης να μπορεί να εγκαταστήσει και συνεπώς να χρησιμοποιήσει το πρόγραμμα.

Η επιλογές που ακολουθήθηκαν για περάτωση της εφαρμογής, βασίστηκαν στη JAVA ως γλώσσα προγραμματισμού, σε αρχεία για διασύνδεση με τη JAVA και στην εφαρμογή install4j IDE version 6 για τη δημιουργία του προγράμματος εγκατάστασης.

4.1 Αποθήκευση δεδομένων

Επιλέχθηκαν τα **αρχεία** για αποθήκευση των δεδομένων του προγράμματος. Οι λόγοι που επιλέχθηκαν τα αρχεία και πιο συγκεκριμένα αρχεία τύπου **σημειωματάριο**, είναι αρχικά γιατί αποτελούσαν απλούστερη λύση. Ο χειρισμός των αρχείων από κάποιον δεν απαιτεί γνώσεις από χειριστή για εγκατάσταση βάσης δεδομένων. Η σύνδεση στο ίντερνετ ακόμη, δεν αποτελεί πρόβλημα μιάς και η συγγραφή σε αρχεία δεν απαιτεί σύνδεση στο ίντερνετ όπως οι βάσεις. Μία ακόμη πολύ σημαντική παράμετρος είναι ότι τα αρχεία αποτελούν πολύ εύκολη λύση για χειρισμό. Οποιοσδήποτε χρήστης μπορεί εύκολα να παρέμβει σε αυτά για αλλαγές. Το παρόν πρόγραμμα στοχεύει στην απλότητα σε όλους τους τομείς ούτως ώστε και ο πιο αρχάριος χρήστης να μπορεί να το χρησιμοποιήσει και αν το επιθυμεί να μπορεί εύκολα να παρέμβει στα αρχεία του προγράμματος για αλλαγές.

Επίσης αν είχε γίνει επιλογή βάσεων δεδομένων, αυτό θα είχε περιπλέξει τη διαδικασία εγκατάστασης για τον χρήστη και τελικά δεν προτιμήθηκε ως λύση. Έτσι, προτιμήθηκε η άλλη επιλογή.

Σκοπός της εφαρμογής είναι η δημιουργία **δυναμικού** περιβάλλοντος μέσω στατικών μεθόδων, όπως αυτής των αρχείων.

Σημειωματάριο

Το σημειωματάριο είναι ένας κοινός, πολύ απλός επεξεργαστής κειμένου που διακρίνεται για την απλότητα και ευκολία του. Τα αρχεία που προκύπτουν αποθηκεύονται με την επέκταση .txt και δεν έχουν ετικέτες ή στυλ μορφοποίησης. Βασικές λειτουργίες αποτελούν η αντιγραφή, επι-

κόλληση, αποκοπή, αναίρεση, αποθήκευση. Άλλες βασικές λειτουργίες αποτελούν αναζήτηση και αντικατάσταση, κράτηση κάποιων σημειώσεων σε αυτό ακόμη και συγγραφή κώδικα. Για τις ανάγκες του προγράμματος και για λόγους απλότητας όπως έχει αναφερθεί χρησιμοποιήθηκε αυτού του είδους επεξεργαστής κειμένου. Στο παρόν πρόγραμμα, όλες οι εγγραφές στα σημειωματάρια του προγράμματος, ακολουθούν κωδικοποίηση **UTF-8 without BOM**. Οποιαδήποτε άλλη κωδικοποίηση κειμένου δεν καθιστά το πρόγραμμα εκτελέσιμο και δημιουργεί παρανοήσεις.

4.2 Διεπαφή χρήστη

Επιλέχθηκε η JAVA ως πρόγραμμα υλοποίησης γιατί είναι η πιά διαδεδομένη λύση και δεν απαιτεί compile αλλά αντί αυτού απαιτεί την αρχική εγκατάσταση ενός προγράμματος συγγραφής για JAVA, του JRE και του JVM. Το JRE πιά γνωστό ως JAVA Runtime Environment είναι το περιβάλλον μέσα στο οποίο το εκτελείται το JVM. Το JVM, πιά γνωστό ως Java Virtual Machine, είναι ο compiler της JAVA. Ο JVM και το JRE περιέχονται και τα δύο στο Jdk του οποίου η τελευταία έκδοση βρίσκεται στην κεντρική σελίδα της ORACLE μαζί με την JAVA. Η JAVA διαθέτει τεράστιο όγκο βιβλιοθηκών, παρέχοντας πληθώρα έτοιμων συστατικών και ειδικά σε γραφικό επίπεδο για το χρήστη μιάς και η εφαρμογή αποτελείται εξολοκλήρου από γραφικά. Επίσης επιλέχθηκε ο NetBeans IDE για προγραμματισμό της εφαρμογής γιατί διαθέτει ενσωματωμένο Jdk και πολλά χρήσιμα εργαλεία καθιστώντας πιά εύκολη και διασκεδαστική τη συγγραφή κώδικα.

Χρησιμοποιήθηκε όπως αναφέρεται από πάνω η JAVA για υλοποίηση της εφαρμογής. Στο προηγούμενο κεφάλαιο είχε ειπωθεί ότι η JAVA μπορεί να εκτελέσει περιβάλλοντα τύπου κονσόλας και γραφικών. Η εφαρμογή αποτελείται από γραφικά στοιχεία. Για να εκτελέσει περιβάλλον γραφικών στοιχείων χρησιμοποιήθηκε η Swing κλάση της JAVA που επιτελεί αυτό το σκοπό.

4.2.1 Αντικειμενοστρεφής προγραμματισμός

Ένα αντικειμενοστρεφές πρόγραμμα δεν αποτελείται από τα δεδομένα και τον κώδικα που τα επεξεργάζεται αλλά από αντικείμενα τα οποία εμπεριέχουν τα δεδομένα και τα οποία ανταλλάσσουν μεταξύ τους πληροφορίες και μηνύματα προκειμένου να επιτευχθεί ο στόχος του προγράμματος. Ο κώδικας που ορίζει ένα αντικείμενο λέγεται κλάση του αντικειμένου αυτού και χρησιμοποιείται για την κατασκευή πανομοιότυπων αντικειμένων. Τα αντίγραφα, λέγονται στιγμιότυπα της κλάσης αυτής. Το αντικείμενο, είναι μια οντότητα στη μνήμη η οποία περιέχει δεδομένα καθώς και μεθόδους μέσω των οποίων μπορούμε να αλλάξουμε τα δεδομένα ή να επικοινωνήσουμε με το αντικείμενο. Αντίθετα η κλάση είναι απλώς ένα πρότυπο για την δημιουργία αντιγράφων του ίδιου αντικειμένου.

4.2.2 JAVA

Η JAVA δημιουργήθηκε, από την εταιρία Sun microsystems TM και σχεδιάστηκε για την ανάπτυξη εφαρμογών που θα τρέχουν σε ετερογενή δικτυακά περιβάλλοντα. Η JAVA είναι αντικειμενοστρεφής, κατανεμημένη, ασφαλής, multithreaded και interpreted γλώσσα. Κατανεμημένη, γιατί πρόγραμμα που έχει υλοποιηθεί σε JAVA, μπορεί να μεταφερθεί σε άλλο δίκτυο και να εκτελεστεί.

Ασφαλής, γιατί προστατεύει το πρόγραμμα από ιούς. Με τον όρο multithreaded, εννοείται ότι η JAVA χρησιμοποιεί πολυεπεξεργασία των προγραμμάτων που εκτελούνται σε αυτή. Με αυτό τον τρόπο τα διαχειρίζεται ευκολότερα. Interpreted γιατί δεν παράγεται εκτελέσιμος κώδικας αλλά μια μορφή ψευδοκώδικα το οποίο από μόνο του δεν εκτελείται σε καμία μηχανή. Προκειμένου λοιπόν να εκτελεστεί απαιτείται η χρήση ενός διερμηνέα. Αυτό το χαρακτηριστικό δίνει τη δυνατότητα στα java bytecodes να μπορούν να εκτελεστούν σε οποιοδήποτε μηχανήμα κάτω από οποιοδήποτε λειτουργικό, αρκεί να έχει εγκατασταθεί ένας διερμηνέας. Επίσης ένα άλλο χαρακτηριστικό του java bytecode είναι το μικρό του μέγεθος, όπου είναι μόλις λίγα Kilobytes. Αυτό το κάνει ιδανικό για μετάδοση μέσω του δικτύου. Ακόμη, η JAVA υποστηρίζει δημιουργία multimedia εφαρμογών και αυτό χάρη στον μεγάλο όγκο των βιβλιοθηκών που διαθέτει. Επιτρέπει τη δημιουργία ανεξάρτητων εφαρμογών και applets. Βασικά χαρακτηριστικά της JAVA αποτελούν ο πολυμορφισμός και η κληρονομικότητα. Κληρονομικότητα γιατί μία κλάση της μπορεί να κληρονομήσει χαρακτηριστικά από άλλη κλάση της και πολυμορφισμό γιατί μία μέθοδος έχει τη δυνατότητα να συνδεθεί με την αντίστοιχη κλάση κατά τη διάρκεια της εκτέλεσης. Αυτή η δυναμική διασύνδεση λέγεται πολυμορφική συμπεριφορά.

4.2.3 JAVA και GUI

Παραπάνω αναφέρθηκε ότι η Swing είναι η βασική κλάση της JAVA για ανάπτυξη εφαρμογών σε γραφικό επίπεδο. Η Swing είναι ένα GUI widget toolkit για τη JAVA, που προσφέρει γραφικό περιβάλλον διαχείρισης για εφαρμογές JAVA. Δημιουργήθηκε, για να παρέχει ένα πιά ανεπτυγμένο σέτ απο GUI συστατικά όπως είναι τα tabbed panel, scroll panes, από τον προκάτοχό του το Abstract Window Toolkit. Αξίζει να σημειωθεί ότι η Swing μπορεί να εκτελεστεί σε οποιοδήποτε λογισμικό.

Βασικά στοιχεία GUI

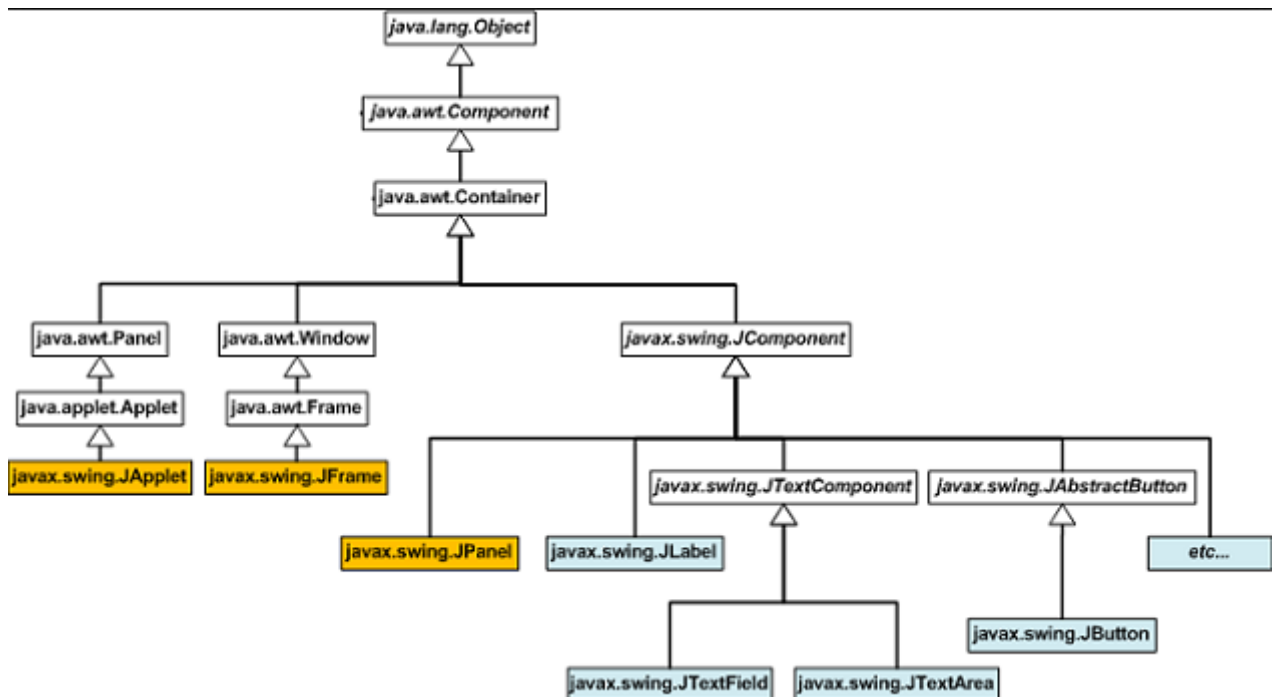
Το GUI, αφορά τον τρόπο εμφάνισης και χειρισμού του προγράμματος από τον χρήστη και αποτελείται από κάποια βασικά συστατικά. Τα βασικά αυτά συστατικά ονομάζονται GUI components, δηλαδή συστατικά του GUI. Οι κλάσεις που χρησιμοποιούνται για την κατασκευή τέτοιου είδους αντικειμένων, ανήκουν στο java. awt package. Οι βασικότερες από αυτές είναι η κλάση Component και η κλάση Container. Κάθε κλάση που κληρονομεί την κλάση Container είναι και αυτή ένα Container και κάθε κλάση που κληρονομεί την κλάση Component είναι και αυτή ένα Component. Στο σχήμα 4.1, περιγράφονται οι συνηθέστερες κλάσεις που χρησιμοποιούνται για την κατασκευή συστατικών GUI. Στο σχήμα 4.2, εμφανίζεται η ιεραρχία των Components αυτών και άλλων κλάσεων που διαχειρίζονται ένα GUI.

Δομή ενός GUI

Για την κατασκευή ενός GUI, συνήθως ακολουθείται μία λογική. Ένα μεγάλο GUI χωρίζεται σε μικρότερες υποπεριοχές. Οι υποπεριοχές αυτές είναι Panel components. Σημειώνεται ότι η κλάση Panel κληρονομεί την κλάση Container και η κλάση Applet την κλάση Panel. Έτσι τα Panels και τα Applets είναι Containers και μπορούν να έχουν Components, ακόμα και τύπου Panel. Σε κάθε Panel τα Components που περιέχει είναι τοποθετημένα με ένα συγκεκριμένο σχέδιο, το layout. Η Java παρέχει διαχειριστές αυτών των σχεδίων, δηλαδή Layout Managers για την διευ-

Label	Εμφανίζει κείμενο που δεν μπορεί να τροποποιηθεί από τον χρήστη.
Button	Περιοχή που προξενεί ένα γεγονός (event) όταν επιλέγεται με το ποντίκι.
TextField	Περιοχή όπου ο χρήστης εισάγει δεδομένα από το πληκτρολόγιο. Σε ένα TextField μπορούμε επίσης να εμφανίζουμε πληροφορίες.
TextArea	Περιοχή όπου ο χρήστης εισάγει δεδομένα από το πληκτρολόγιο. Σε ένα TextArea μπορούμε να έχουμε πολλές γραμμές κειμένου σε αντίθεση με το TextField που μπορούμε να έχουμε μόνο μία.
Choice	Μια λίστα στοιχείων από τα οποία ο χρήστης μπορεί να επιλέξει ένα από αυτά με το ποντίκι.
Checkbox	Ένα boolean component που είναι επιλεγμένο ή μη επιλεγμένο. Με αυτό υλοποιούνται και τα Radio buttons (αμοιβαίως αποκλειόμενες επιλογές).
List	Λίστα στοιχείων από τα οποία ο χρήστης μπορεί να επιλέξει ένα από αυτά με το ποντίκι. Διπλό κλικ σε ένα στοιχείο της λίστας προξενεί ένα action event.
Panel	Είναι ένα container αντικείμενο στο οποίο μπορούν να τοποθετηθούν component αντικείμενα.
ScrollPane	Είναι ένα container αντικείμενο στο οποίο μπορεί να τοποθετηθεί ένα component, συνήθως ένα Panel, το οποίο θα εμφανίζεται στον χρήστη αλλά κι όταν αυτό δεν είναι εφικτό θα εμφανίζονται Scrollbars που θα επιτρέψουν την διολίσθηση του component έτσι ώστε να γίνεται ορατό και το τμήμα του που δεν φαίνεται.

Σχήμα 4.1: Συνηθέστερες κλάσεις για την κατασκευή GUI-components



Σχήμα 4.2: Ιεραρχία των GUI-components κλάσεων

θέτηση αντικειμένων μέσα σε ένα Applet ή Panel. Στο σχήμα 4.3, παρουσιάζονται οι πιο συνηθισμένες κλάσεις.

GUI Events

Σε γραφικό περιβάλλον της JAVA κάποιος μπορεί να εκτελέσει βασικές λειτουργίες και επιλογές από τις μονάδες εισόδου του υπολογιστή και αυτές να φανούν στο γραφικό επίπεδο. Για

FlowLayout	Είναι ο default Layout Manager για τα Applets και τα Panels. Τοποθετεί τα Component αντικείμενα από αριστερά προς τα δεξιά με την σειρά που προστίθενται
BorderLayout	Τοποθετεί τα Component αντικείμενα σε πέντε περιοχές: Βόρεια, Νότια, Ανατολική, Δυτική και Κεντρική (North, South, East, West, Center).
GridLayout	Τοποθετεί τα Component αντικείμενα σε γραμμές και στήλες με καθορισμένη σειρά (πρώτα γεμίζει η πρώτη γραμμή μετά η δεύτερη γραμμή κλπ).
CardLayout	Τοποθετεί τα Component αντικείμενα σε στοίβα. Κάθε Container αντικείμενο στη στοίβα μπορεί να χρησιμοποιήσει οποιονδήποτε Layout Manager. Μόνο το Container αντικείμενο που βρίσκεται στην κορυφή της στοίβας είναι ορατό.
GridBagLayout	Είναι παρόμοιος με τον GridLayout Manager. Διαφέρει από αυτόν στο ότι κάθε Component αντικείμενο που του προστίθεται μπορεί να έχει οποιοδήποτε μέγεθος (δηλαδή να καλύπτει περισσότερες από μια γραμμές και στήλες). Επίσης η σειρά με την οποία προστίθενται τα Component αντικείμενα στον GridBagLayout Manager μπορεί να είναι οποιαδήποτε.

Σχήμα 4.3: Layout Managers

να συμβεί δηλαδή κάποιο γεγονός ή καλύτερα ένα event στη γραφική διεπαφή, πρέπει να ακολουθηθεί ένα σύνολο βημάτων. Αρχικά να δηλωθεί ένας ακροατής γεγονότων που θα ακούσει το event και ένας χειριστής γεγονότων που θα χειριστεί το event. Σε γλώσσα της JAVA, ένας event listener και event handler αντίστοιχα. Στον χειριστή γεγονότων γράφεται κώδικας που αφορά την ενέργεια που θέλει να πραγματοποιήσει ο χρήστης. Ένας event listener ανήκει στο πακέτο java. awt. event. Το γεγονός που λαμβάνει χώρα χρειάζεται στη συνέχεια κάπου να αποθηκευτεί. Έτσι, αποθηκεύεται σε κάποιο αντικείμενο της AWTEvent. Η AWTEvent είναι ορισμένη στο πακέτο java. awt. event. Το σχήμα 4.4, εμφανίζει την ιεραρχία των κλάσεων του πακέτου java. awt. event και το σχήμα 4.5, εμφανίζει την ιεραρχία των event-listener interfaces του πακέτου java. awt. event.

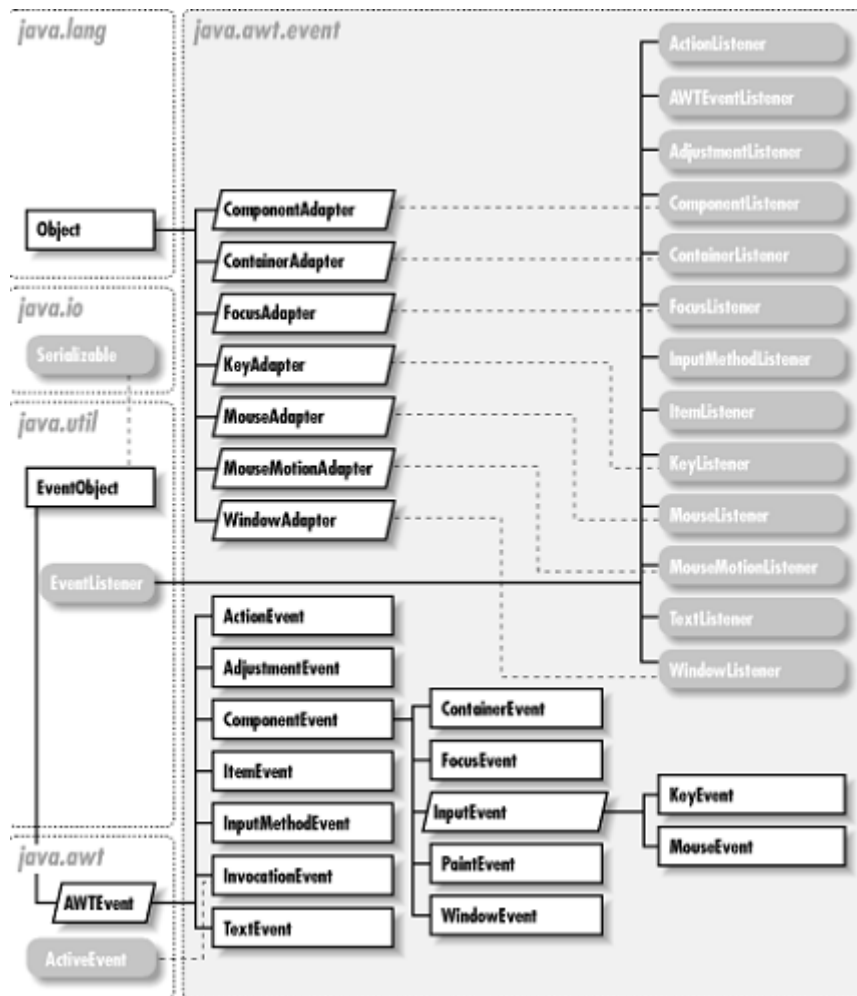
Αξιοσημείωτες είναι οι **Adapter Classes**, οι οποίες περιέχουν όλες τις συναρτήσεις ενός event listener interface με κενό σώμα. Κάποιος μπορεί απλά να παρακάμψει μόνο τις μεθόδους της adapter κλάσης που τον ενδιαφέρουν. Παραδείγματα adapter κλάσεων είναι η MouseAdapter που υλοποιεί το interface MouseListener, η MouseMotionAdapter που υλοποιεί το interface MouseMotionListener και η MouseMotionAdapter που υλοποιεί το interface MouseMotionAdapterListener.

Swing GUIs

Τα πακέτα που αρχίζουν με το πρόθεμα javax. swing παρέχουν ευέλικτα και ισχυρά εργαλεία ανάπτυξης GUI. Ιδιαίτερα το πακέτο javax. swing αναπτύχθηκε κυρίως λόγω των ανεπαρειών του Abstract Windows Toolkit ή απλώς AWT. .

Top-Level Swing Containers

Κάθε διεπαφή Swing πρέπει να έχει τουλάχιστον ένα top-level Swing container. Ένα top-level Swing container παρέχει την απαραίτητη υποστήριξη που χρειάζονται τα Swing components για την εμφάνισή τους και την διαχείριση των γεγονότων που αυτά παράγουν. Υπάρχουν τρία top-level Swing containers. Αυτά είναι το JFrame, το JDialog, και για applets το JApplet. Κάθε JFrame αντικείμενο δημιουργεί ένα κύριο γραφικό παράθυρο, κάθε JDialog αντικείμενο δημιουργεί ένα δευτερεύον παράθυρο, δηλαδή παράθυρο που εξαρτάται από κάποιο άλλο παράθυρο. Κάθε JApplet αντικείμενο δημιουργεί την περιοχή εμφάνισης ενός applet στο παράθυρο του Web Browser. Αυτά



Σχήμα 4.4: Ιεραρχία κλάσεων του πακέτου java. awt. event

φαίνονται και στο σχήμα 4.6.

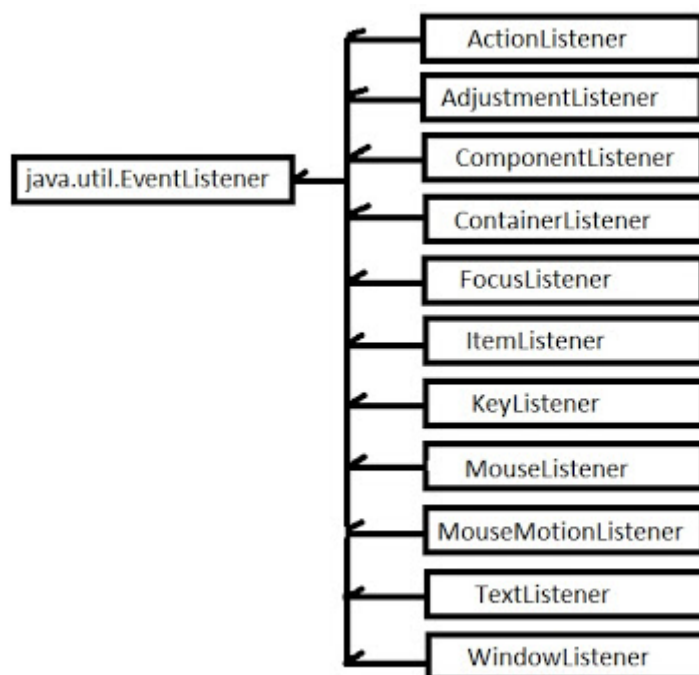
Swing Components

Με εξαίρεση τα top-level containers, όλα τα Swing components είναι υποκλάσεις της κλάσης JComponent. Η ιεραρχία της JComponent φαίνεται στο σχήμα 4.7.

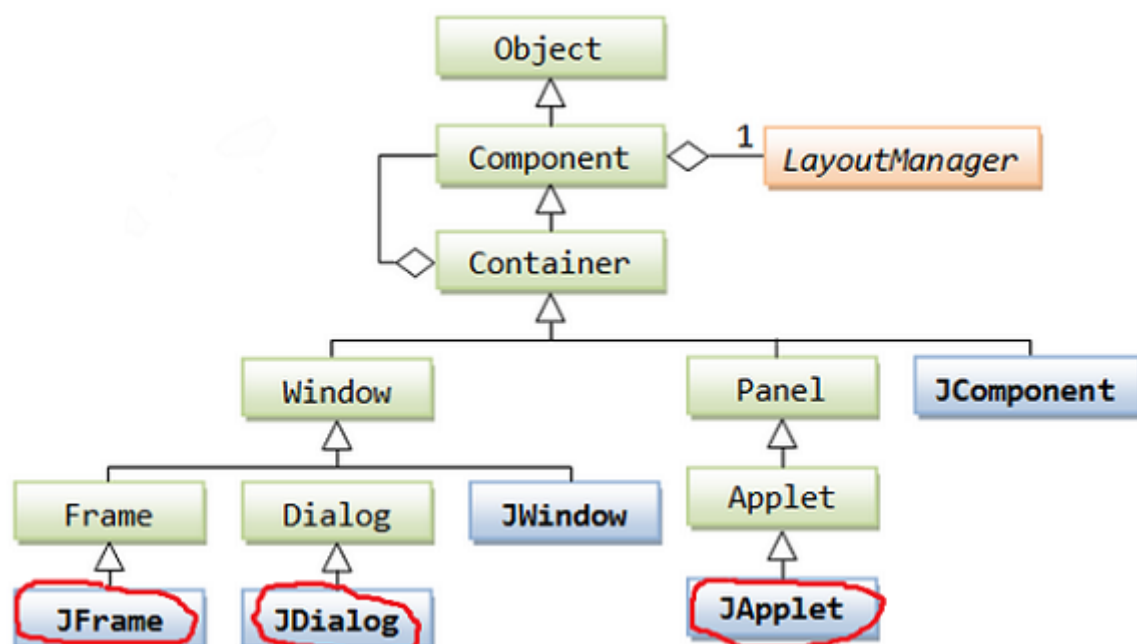
Συνήθως, όταν υλοποιείται μια GUI εφαρμογή σε Swing, δημιουργούμε ένα JFrame αντικείμενο και επιλέγουμε συγκεκριμένο σχέδιο ή layout για αυτό. Κατόπιν τοποθετούμε ένα ή περισσότερα JPanels στο JFrame. Τα JPanels διαθέτουν επίσης επιλογές layout όπως και τα Jframes.

Διαχειριστές Γεγονότων

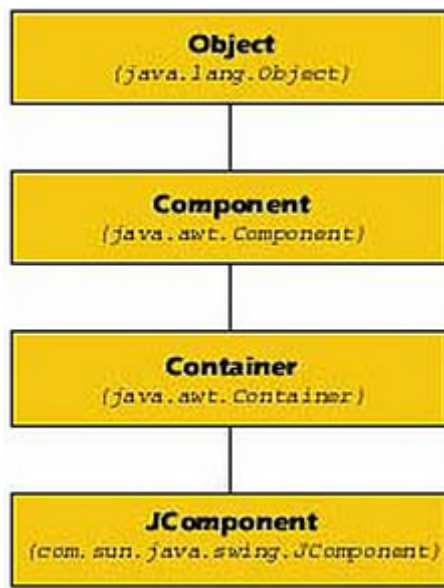
Τα Swing components παράγουν διάφορους τύπους γεγονότων. Ο πίνακας του σχήματος 4.8, περιέχει τα πιο συνηθισμένα από αυτά.



Σχήμα 4.5: Ιεραρχία των event-listener interfaces του πακέτου `java.awt.event`.



Σχήμα 4.6: `JFrame`, `JDialog`, `JApplet`



Σχήμα 4.7: JComponent

Some Events and Their Associated Event Listeners	
Act that Results in the Event	Listener Type
User clicks a button, presses Enter while typing in a text field, or chooses a menu item	ActionListener
User closes a frame (main window)	WindowListener
User presses a mouse button while the cursor is over a component	MouseListener
User moves the mouse over a component	MouseMotionListener
Component becomes visible	ComponentListener
Component gets the keyboard focus	FocusListener
Table or list selection changes	ListSelectionListener
Any property in a component changes such as the text on a label	PropertyChangeListener

Σχήμα 4.8: Πιο συνηθισμένοι τύποι γεγονότων

Κεφάλαιο 5

Τί μπορεί να κάνει κάποιος με τις καλλιέργειες γενικά

Στα προηγούμενα κεφάλαια έγινε αρχικά αναφορά στους τρόπους τους οποίους η καλλιέργεια σε συνδυασμό με την τεχνολογία και τους ηλεκτρονικούς υπολογιστές μπορεί να αναβαθμιστεί σημαντικά. Παρουσιάστηκαν χρήσιμες τεχνολογίες ως προς τον τρόπο αποθήκευσης δεδομένων, εμφάνισης αυτών στην οθόνη του υπολογιστή και τρόποι εγκατάστασης κάποιου προγράμματος. Επίσης δείχθηκε η επιλογή υλοποίησης και εγκατάστασης, βάση των προηγούμενων.

Δημιουργούνται όμως κάποια ερωτήματα. Πώς κάποιος μπορεί να διαχειρίζεται καλλιέργειες χωρίς τη χρήση του ηλεκτρονικού υπολογιστή, και πώς κάποιος μπορεί να οργανώσει τις καλλιέργειές του ώστε να τις διαχειριστεί όσο το δυνατόν πιο αποδοτικά. Αυτό αποτελεί αξιοσημείωτο ενδιαφέρον.

5.1 Αρχικοί ενδοιασμοί καλλιέργειας

Ξεκινώντας κάποιος να ασχοληθεί με την καλλιέργεια και ανάπτυξη κάποιων προϊόντων, διάφορα ζητήματα και ενδοιασμοί δημιουργούνται στο μυαλό του.

Οι αρχικές αποφάσεις του βασίζονται στο τί αρχικά θα καλλιεργήσει αναλόγως τις ανάγκες του. Εφόσον αποφασίσει και ξεκινήσει την καλλιέργειά του, κάποιος, καλό θα ήταν να γνωρίζει τί ενέργειες κάνει στην καλλιέργειά του. Αυτό γιατί στην πορεία της εξέλιξης της καλλιέργειάς του, το πιο πιθανό είναι να ξεχάσει τί έκανε την τελευταία φορά και κάνοντας λάθος χρήση κάποιου υλικού στην καλλιέργεια ή λάθος εκτίμηση, να τη βλάψει. Ακόμη και σε περίπτωση υπερβολικού ποτίσματος αυτής να μην έχουμε το επιθυμητό αποτέλεσμα. Ειδικότερα στην πιο συνηθισμένη περίπτωση που κάποιος καλλιεργεί άνω της μίας καλλιέργειας γίνεται όλο και πιο περίπλοκη η διαχείρισή τους. Με το σημερινό φόρτο εργασίας του κάθε ανθρώπου καθίσταται όλο και πιο δύσκολο να θυμάται κάποιος τις ενέργειες που έχει κάνει στις καλλιέργειές του.

Οι βασικότεροι ενδοιασμοί που θα έχει κάποιος είναι το που βρίσκεται που από τις καλλιέργειές του, τι υλικό να χρησιμοποιήσει και πότε, που θα ήταν καλύτερο να τοποθετήσει κάποια νέα καλλιέργεια και πώς θα ξέρει σε πιο στάδιο βρίσκονται σε δεδομένη στιγμή.

5.2 Ενέργειες πάνω στις καλλιέργειες

Αναφέρθηκαν πιο πάνω βασικοί ενδοιασμοί, έγνοιες που κάποιος μπορεί να έχει αρχικά προτού ενασχοληθεί με τις καλλιέργειες. Ξεκινώντας κάποιος να ενασχοληθεί με αυτές και καθόλη την πορεία διαχείρισής τους, κάνει κάποιες ενέργειες πάνω τους. Οι ενέργειες αυτές είτε θα αλλάζουν τη φάση της καλλιέργειας είτε θα παρεμβαίνουν σε αυτή. Σε αυτή την παράγραφο αναφέρονται τέτοιες ενέργειες.

Αρχικά κάποια καλλιέργεια, πρωτού αναπτυχθεί είναι σε μορφή σπόρου. Έτσι, πρέπει να φυτευτεί σε χώμα ώστε ο σπόρος να αρχίσει την ανάπτυξή του. Αντιστοίχως, σπόρος μπορεί να φυτευτεί αντί σε χώμα σε ήδη υπάρχων φυτώριο. Κάποιος όμως μπορεί να προτιμήσει την αγορά έτοιμου φυτωρίου σε χώμα ή ακόμη την αγορά έτοιμου φυτωρίου σε γλαστράκι και να συνεχίσει την καλλιέργειά τους από τη φάση ανάπτυξης που θα βρίσκονται και έπειτα. Επιπλέον χώμα σε καλλιέργεια μπορεί να προστεθεί ανά πάσα στιγμή. Όμως κάποιος μπορεί να μεταφυτεύσει κάποια από τις καλλιέργειές του σε άλλο σημείο του κήπου του ή καλύτερα σε άλλο γλαστράκι, μιάς και κύριο μέλημά μας είναι η καλλιέργεια φυτών μικρής κλίμακας. Συνήθως μεταφύτευση γίνεται στο φυτώριο, γιατί είναι μεγάλο και καταλαμβάνει χώρο. Κατά τη διάρκεια της ανάπτυξης του φυτού, επέρχεται η φάση όπου το φυτό βλασταίνει και μετά από κάποιο χρονικό διάστημα, στην ιδανική περίπτωση δίχως ύπαρξη επιπλοκών μπορεί να ανθοφορήσει. Εν συνεχεία των προηγούμενων, όταν κάποιος καλλιεργεί και ειδικά σε μικρή κλίμακα, όπως σε μία βεράντα, μπορεί ανά πάσα στιγμή μια επικείμενη καταστροφή να λάβει χώρα. Αυτό λόγω χάρη ακραίων καιρικών συνθηκών που θα επιφέρουν καταστροφή ενός ή περισσότερων ειδών καλλιεργειών ή φυτωρίου. Καιρικές συνθήκες όπως υγρασία, χαλάζι, ξηρασία, χιόνι, πλημμύρα, ψύχος ανήκουν σε αυτή την κατηγορία. Με την ολοκλήρωση της ανάπτυξης κάποιας καλλιέργειας, όταν αυτή έχει πιά ωριμάσει αρχίζει πια η συγκομιδή η οποία διαρκεί κάποιο χρονικό διάστημα, αναλόγως το φυτώριο και με το πέρας του χρονικού αυτού διαστήματος τελειώνει και η συγκομιδή του.

Αναφέρθηκαν οι κυριότερες ενέργειες που κάποιος μπορεί να κάνει πάνω στις καλλιέργειες, μαζί και η απρόσμενη καταστροφή. Υπάρχουν επίσης εκτός αυτών και άλλες παρεμβάσεις που κάνει κάποιος πάνω σε καλλιέργειές του, κατά τη διάρκεια ανάπτυξής τους. Μερικές απ' τις παρεμβάσεις αυτές είναι η προσθήκη λιπάσματος, προσθήκη φαρμάκων φυτοπροστασίας, κλάδεμα κάποιου φυτού όποτε χρειαστεί. Επίσης, αραίωση του χώματος στο οποίο έχουν φυτευτεί οι κάθε καλλιέργειες, αραίωση δηλαδή της πυκνότητας της καλλιέργειας. Βασικό μέλημα είναι το πότισμα το οποίο πρέπει να γίνεται σε τακτά χρονικά διαστήματα ειδικά στα πρώτα στάδια ανάπτυξης της κάθε καλλιέργειας. Παράγοντες που επηρεάζουν τη συχνότητα ποτίσματος είναι η τοποθεσία όπου βρίσκεται φυτευμένο το φυτό το οποίο είτε μπορεί να φωτίζεται αρκετά είτε όχι, η ξηρότητα του χώματος που έχουμε τοποθετήσει το κάθε φυτώριο λόγω ξηρότητας και το είδος του φυτού. Κυριότερο ρόλο στη συχνότητα ποτίσματος κατέχει το είδος του φυτού. Αναφέρθηκε η επικείμενη καταστροφή παραπάνω ως κακός παράγοντας καλλιέργειας. Σαν κακούς παράγοντες, σε πολύ μικρότερης κλίμακας καταστροφικούς θα ορίζαμε διαφόρους τύπους εντόμων όπως οι μελίγκρα, κάμπια, ακρίδα, ποντίκι, σκουλήκι σαλιγκάρι. Αυτά παρεμβαίνουν στις καλλιέργειες με κακή συνέπεια. Ο χρήστης μπορεί να παρέμβει με τη σειρά του στο φυτώριό του χρησιμοποιώντας εντομοαπωθητικό, εντομοκτόνο. Η τοποθέτηση κάποιου είδους φράχτη γύρω από κάποιο φυτώριο, είναι μια συνηθισμένη παρέμβαση κάποιου με σκοπό την επιπρόσθετη προστασία των καλλιεργειών του.

5.3 Συνηθέστερος τρόπος καταγραφής καλλιέργειας

Η σημερινή τεχνολογία, έχει εξελιχθεί σε σημαντικό βαθμό και απόρροια αυτής είναι η ένταξη του ηλεκτρονικού υπολογιστή στις ζωές των ανθρώπων καθιστώντας ακόμη και για τον πιο αρχάριο την εύκολη χρήση του. Από την άλλη όμως, υπάρχει μία πολύ μεγάλη ομάδα ανθρώπων οι οποίοι δεν έχουν ιδιαίτερη ευχέρεια με τον υπολογιστή και γενικότερα με την τεχνολογία.

Ο πρώτος και απλοϊκός τρόπος που μπορεί κάποιος να σκεφτεί για την καταγραφή των καλλιεργειών είναι κάτι σε όλους διαθέσιμο, το χαρτί. Ξεκινώντας κάποιος την ενασχόλησή του με έναν αριθμό καλλιεργειών, μπορεί να χρησιμοποιήσει έναν πολύ απλό τρόπο καταγραφής των καλλιεργειών που κατέχει. Ο τρόπος αυτός είναι το χαρτί, το οποίο μπορεί να χρησιμοποιηθεί ως ημερολόγιο καταγραφής ενεργειών του πάνω σε καλλιέργειες. Αυτά που πρέπει να σημειωθούν στο ημερολόγιο- χαρτί, είναι οι ημερομηνίες κάτω από τις οποίες θα καταχωρεί πληροφορίες σχετικά με την πρόοδο των καλλιεργειών του.

5.4 Αποτύπωση ενεργειών στο χαρτί

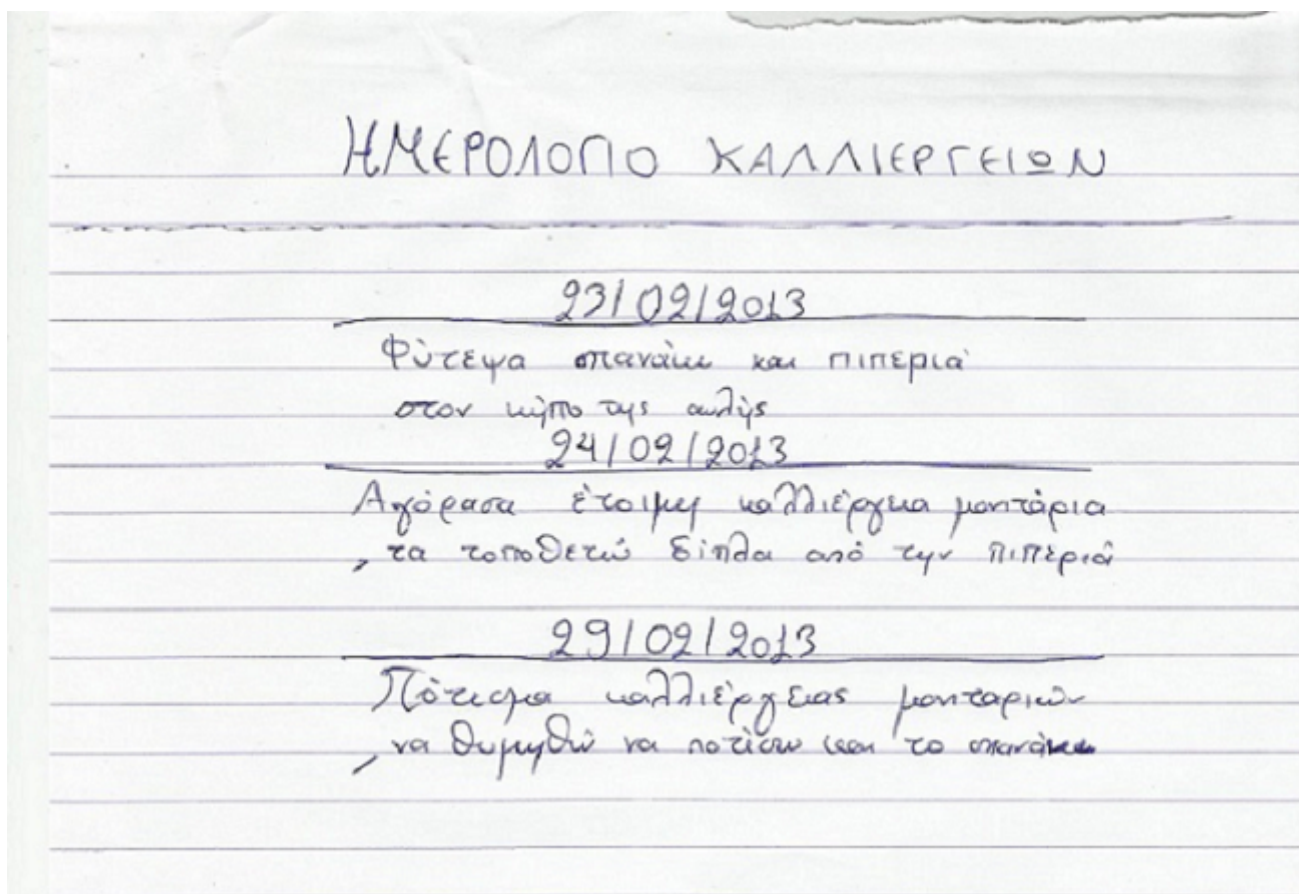
Παραπάνω αναφέρθηκαν οι κυριότερες ενέργειες που κάποιος μπορεί να κάνει σε μια καλλιέργεια, διάφορες φυσικές παρεμβάσεις και καταστροφές και οι κυριότερες φάσεις καλλιέργειας. Όμως ερώτημα αποτελεί με πιά τρόπο τα παραπάνω αποτυπώνονται στο χαρτί και με ποιιά σειρά. Αυτό εξαρτάται από τον άνθρωπο. Όμως γεγονός αποτελεί ότι οι περισσότεροι άνθρωποι επιλέγουν πάντα την πιο απλή και κατανοητή απεικόνιση. Τα δεδομένα που θα αποτυπωθούν στο ημερολόγιο είναι η ημερομηνία, ώστε κάθε μέρα που σημειώνεται στο χαρτί να αποτελεί την ημερομηνία εκείνης της ημέρας, και τα άλλα δεδομένα αποτελούνε οι παραπάνω ενέργειες που αναφέρθηκαν.

Πρώτο πρότυπο ημερολογίου καλλιεργειών με χρονολογία

Για να καταγράψει κάποιος καλλιέργειες στο χαρτί χρειάζεται να σημειώσει κάποια βασικά πράγματα. Το πρώτο είναι η ημερομηνία, το δεύτερο οι καλλιέργειες, το τρίτο οι ενέργειες που κάνει σε αυτές και τέταρτο τα σχετικά σχόλια. Ενέργειες και σχόλια σημειώνονται με τον ίδιο τρόπο στο χαρτί. Ο πιο απλός τρόπος να κάνει κάποιος αυτό είναι να σημειώσει αρχικά τον τίτλο του προσωπικού ημερολογίου, παραδείγματος χάρη ΗΜΕΡΟΛΟΓΙΟ ΚΑΛΛΙΕΡΓΕΙΩΝ. Απο κάτω στο κύριο μέρος του χαρτιού θα βρίσκονται οι λοιπές πληροφορίες. Η πιο απλή διάταξη που κάποιος θα ακολουθούσε είναι να καταγράφεται η κάθε ημερομηνία κάθετα και κάτω από κάθε ημερομηνία να καταγράφονται οι καλλιέργειες και τι έχει κάνει σε αυτές. Βέβαια, την ημερομηνία μπορεί να τη σημειώσει κάποιος στο χαρτί με τρεις τρόπους. Ο πρώτος με μορφή μέρας, μήνα και χρόνια, ο δεύτερος με τη μορφή μέρας και μήνα και ο τρίτος με τη μορφή ημέρας.

Ημέρα/Μήνας/Χρόνια:

Ο τρόπος αυτός είναι ιδανικός για κάποιον που κρατάει αρχείο καλλιεργειών, με σκοπό να θυμάται τι έχει κάνει σε κάποιο δεδομένο σημείο στο παρελθόν. Παραθέτεται σχετικό παράδειγμα αυτού του τρόπου στην εικόνα 5.1.



Σχήμα 5.1: Πρώτο πρότυπο μοτίβο ημερολογίου - Ημέρα/Μήνας/Χρόνια, καταγραφή καλλιεργειών και ενέργειες σε αυτές

Ημέρα/Μήνας:

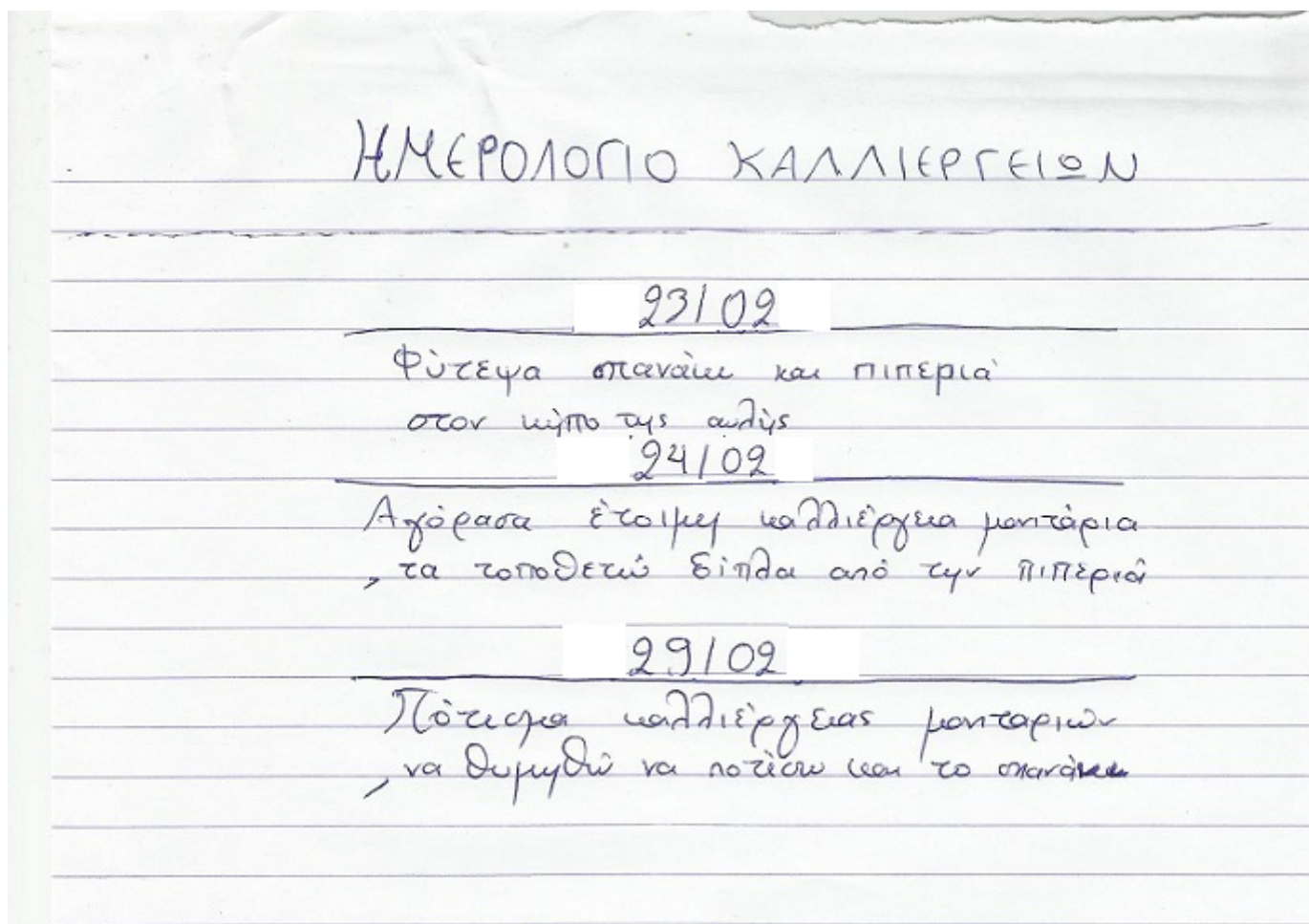
Ο δεύτερος τρόπος αποτύπωσης της ημερομηνίας στο χαρτί είναι χωρίς αναφορά σε χρόνια. Το χαρτί είναι αναλώσιμο και δεν κατέχει επαρκή χώρο. Οι περισσότεροι άνθρωποι χρειάζονται το χαρτί για να σημειώσουν απλά δεδομένα τα οποία να θυμούνται μελλοντικά σε βραχυπρόθεσμο στάδιο. Κάποιος μπορεί να προτιμήσει αυτό τον τρόπο αποτύπωσης σημειώνοντας απλά μέρες και μήνα για να θυμάται τι έκανε. Παρατίθεται σχετικό παράδειγμα στην εικόνα 5.2.

Δεύτερο πρότυπο ημερολογίου καλλιεργειών με χρονολογία

Παραπάνω παρουσιάστηκε πρότυπο ημερολογίου καλλιεργειών σύμφωνα με συγκεκριμένη διάταξη. Το δεύτερο πρότυπο που κάποιος μπορούσε να ακολουθήσει για καταγραφή, είναι να σημειώνει ανα σειρά τις ημερομηνίες και δίπλα από κάθε εκάστοτε ημερομηνία ποιές ενέργειες, τί έκανε σε αυτές.

Τρίτο πρότυπο ημερολογίου καλλιεργειών με χρονολογία

Αυτό το πρότυπο δεν είναι και το πιο ιδανικό για το λόγο του ότι αν κάποιος έχει πολλές καλλιέργειες και τις καλλιεργεί για μήνες θα χρειαστεί ναί μεν περισσότερα του ενός χαρτιά αλλά



Σχήμα 5.2: Πρώτο πρότυπο μοτίβο ημερολογίου - Ημέρα/Μήνας, καταγραφή καλλιεργειών και ενέργειες σε αυτές

ενδεχομένως να μπερδέψει τις ημέρες.

Ειπώθηκε προηγουμένως ότι η ημερομηνία μπορεί να σημειωθεί κυρίως με δύο τρόπους. Στην εικόνα 5.3, δίνεται παράδειγμα πρότυπου μοτίβου ημερολογίου με μορφή ημερομηνίας **Ημέρα/Μήνας/Χρόνια** και στην εικόνα 5.4, δίνεται παράδειγμα πρότυπου μοτίβου ημερολογίου με μορφή ημερομηνίας **Ημέρα/Μήνας**.

Οι παραπάνω εικόνες δείχνουν το ημερολόγιο που αναφέρθηκε, απλοϊκά γραμμένο με τον τρόπο που κάποιος θα τα έγραφε στο χαρτί.

Σύγκριση των δύο μορφών ημερομηνίας

Δείχθηκαν παραδείγματα πάνω στις δύο μορφές που κάποιος μπορεί να ακολουθήσει. Η τρίτη συνήθως, δεν προτιμάται ως λύση. Συγκρίνοντας τις δύο πρώτες μορφές, αν κάποιος κατέχει πολλές καλλιέργειες και καλλιεργεί για χρόνια ή ακόμη προσδοκεί να τις καλλιεργήσει για χρόνια θα χρειαστεί στην χρονολογική κλίμακά του να συμπεριλαμβάνει και χρόνια. Σημειώνοντας χρόνια

ΗΜΕΡΟΛΟΓΙΟ ΚΑΛΛΙΕΡΓΕΙΩΝ	
23/04/2025	Φύτεψα φράουλα στον κήπο της αυλής, η πρώτη καλλιέργεια που φύτεψα.
24/04/2025	Φύτευση μαντίνων αριστερά της φράουλας
25/04/2025	Φύτεψη φράουλας και μαντίνων
26/04/2025	Προσθήκη λιπασματος στη φράουλα
30/05/2025	Ποτίστος, καταστροφή καλλιέργειας μαντίνων
03/06/2025	Τοποθέτηση μικρού φράχτη χύδην στο την καλλιέργεια φράουλας, αγορά έτοιμης καλλιέργειας γαστρολιάς σε χλασέραι και τοποθέτηση στη γηροαιτή βεράντα.

Σχήμα 5.3: Δεύτερο πρότυπο μοτίβο ημερολογίου - Ημέρα/Μήνας/Χρόνια, καταγραφή καλλιεργειών και ενέργειες σε αυτές

ώς κλίμακα σημαίνει χρήση πλήθος χαρτιών, γεγονός που δυσκολεύει τον άνθρωπο. Ο μέσος άνθρωπος θα ακολουθούσε το δεύτερο τροπο γιατί αρχικά ξεκινώντας κάποιες καλλιέργειες κάποιος δε γνωρίζει πόσο καιρό θα καλλιεργεί έως ότου γίνει η συγκομιδή. Σημειώνοντας μέρες και μήνες ως χρονολογική κλίμακα συνεπάγεται λιγότερο πλήθος χαρτιού καθώς και λιγότερη δυσκολία για τον άνθρωπο.

5.4.1 Καλλιέργειες μικρής κλίμακας, ποιός τρόπος είναι πιο ιδανικός και γιατί

Σε αυτή την παράγραφο γίνεται αναφορά γενικότερα στις καλλιέργειες. Εδώ γίνεται μία παράθεση των καλλιεργειών μικρότερης κλίμακας, όπως η καλλιέργεια κάποιων φυτών σε βεράντα καθώς η παρούσα εφαρμογή αφορά κυρίως καλλιέργειες μικρής κλίμακας.

Στις καλλιέργειες μικρής κλίμακας, ο ιδανικότερος τρόπος είναι η καταγραφή σε Ημέρα/Μήνα.

ΗΜΕΡΟΛΟΓΙΟ ΚΑΛΛΙΕΡΓΕΙΩΝ	
23/04	Φύτευση φράουλα στον κήπο της αυλής, η πρώτη καλλιέργεια που φύτεψα.
24/04	Φύτευση μανάνιας αριστερά της φράουλας
25/04	Πότισμα φράουλας και μανάνιας
26/04	Προσθήκη λιπασματος στη φράουλα.
30/05	Παχέτος, καταστροφή καλλιέργειας μανάνιας
03/06	Τοποθέτηση μικρού φράχτη γύρω από την καλλιέργεια φράουλας, αγορά έτοιμης καλλιέργειας γασολιάς σε χλασεράκι και τοποθέτηση στη μισοειρή βεράντα.

Σχήμα 5.4: Δεύτερο πρότυπο μοτίβο ημερολογίου - Ημέρα/Μήνας, καταγραφή καλλιεργειών και ενέργειες σε αυτές

Αυτο γιατί όντας μικρής κλίμακας, η πλειοψηφία τους δε διαρκεί πολύ μεγάλης διάρκειας από τη σπορά έως και το τέλος της συγκομιδής. Έτσι η κλίμακα Ημέρα/Μήνας για τέτοιας κλίμακας καλλιέργειες, είναι επαρκής.

5.4.2 Βασικοί παράγοντες προσοχής

Στην αρχή του κεφαλαίου αυτού αναφέρθηκαν κάποιοι βασικοί παράγοντες οι οποίοι θα πρέπει να ικανοποιούνται. Τα ερωτήματα αυτά αφορούσαν τον τρόπο που κάποιος μπορούσε να διαχειριστεί τις καλλιέργειές του χωρίς τη χρήση ηλεκτρονικού υπολογιστή και τον τρόπο με τον οποίο μπορεί να οργανώσει τις καλλιέργειές του ώστε να τις διαχειρίζεται όσο το δυνατόν πιο αποδοτικά, όπου αναφέρθηκε το χαρτί-ημερολόγιο. Βέβαια, βασικό παράγοντα προσοχής αποτελούν επίσης το υλικό που κάποιος χρησιμοποιεί, ειδικά κάποιος αρχάριος που μόλις έχει ξεκινήσει να ενασχολείται με τις καλλιέργειες. Πράγματα που θα πρέπει να σημειώνει, είναι το μέρος όπου θα τοποθετήσει την κάθε καλλιέργειά του και τελευταίο και σημαντικό να θυμάται σε πίο στάδιο βρίσκεται η κάθε καλλιέργειά του.

Στο χαρτί ως ημερολόγιο κάποιος μπορεί να σημειώσει το υλικό που χρησιμοποιεί, με τον ίδιο τρόπο ως σχολιασμό. Κάποιος αρχάριος που πρόσφατα έχει ξεκινήσει την ενασχόλησή του με τις καλλιέργειες, δεν έχει τη δυνατότητα να παραδειγματιστεί από κάποιες πρότυπες καλλιέργειες παρά μόνο να σημειώσει τις ενέργειες και τους σχολιασμούς της ημέρας. Βέβαια, υπάρχει η δυνατότητα να σημειώνει στο χαρτί το μέρος που τοποθέτησε την κάθε καλλιέργεια σε περίπτωση που έχει ίδιες καλλιέργειες να μην τις μπερδεύει, αλλά πολλοί σχολιασμοί στο χαρτί μπερδεύουν τον άνθρωπο. Θα μπορούσε κάλλιστα να χρησιμοποιηθεί κάποιος χρωματισμός για σημείωση τοποθεσίας κάθε καλλιέργειας. Το τελευταίο που αναφέρθηκε πιο πάνω, δηλαδή το στάδιο όπου βρίσκεται η καλλιέργεια αν και πολύ σημαντικό δεν μπορεί να το παρατηρεί άμεσα. Στο χαρτί, έχει τη δυνατότητα να σημειώνει τις ενέργειες που αλλάζουν τη φάση τις καλλιέργειας μία μόνο φορά. Για να παρατηρήσει κάποια προηγούμενη φάση της καλλιέργειας ή να θυμηθεί κάποια, θα πρέπει να ανατρέξει στο παρελθόν, πράγμα κουραστικό.

5.4.3 Θετικά και αρνητικά του χαρτιού ως ημερολόγιο καλλιερειών

Το χαρτί είναι ένα πολύ απλό μέσο καταγραφής καλλιερειών, ενεργειών και σχολίων σε αυτές. Μπορεί κάποιος να το πάρει μαζί του όπου θελήσει πολύ εύκολα, να σημειώσει με απλό τρόπο πάνω του και να παρατηρήσει τι έκανε την τελευταία φορά. Αποτελεί όμως μειονεκτικό τρόπο διότι δεν υπάρχει η δυνατότητα κάποιος να συμβουλευτεί πρότυπες καλλιέργειες ούτε να θυμάται παρελθοντικές φάσεις καλλιερειών, παρά μόνο όταν ανατρέξει σε αυτές. Θα έπρεπε δηλαδή με κάποιο τρόπο να διαχωρίζονται οι φάσεις των καλλιερειών από τις παρεμβάσεις που κάνει σε αυτές. Ακόμη όταν καλλιεργούνται πολλά φυτά, δημιουργείται πληθώρα σελίδων ημερολογίου γεγονός που δυσχεραίνει τον άνθρωπο.

Παρατηρείται λοιπόν ότι αυτός ο τρόπος καταγραφής καλλιέρειας αν και απλός, δεν είναι αποδοτικός. Θα έπρεπε να βρεθεί κάτι καλύτερο που να ικανοποιεί τους παράγοντες προσοχής που αναφέρθηκαν, να ευκολύνει και να κάνει πιο διασκεδαστική την ενασχόληση των ανθρώπων με τις καλλιέργειες.

Κεφάλαιο 6

Προδιαγραφές

Στο προηγούμενο κεφάλαιο δείχθηκε με ποιό τρόπο κάποιος μπορεί να διαχειριστεί τις καλλιέργειές του, χωρίς τη χρήση λογισμικού και τεχνολογίας. Παρουσιάζονται όμως αρκετά προβλήματα με χρήση του χαρτιού ως ημερολογίου καταγραφής καλλιεργειών. Η υλοποίηση του παρόντος προγράμματος όμως, καθιστά πολύ πιο εύκολη την οργάνωση και τη διαχείριση καλλιεργειών μικρής κλίμακας καθώς διαθέτει συγκεκριμένες λειτουργίες για το σκοπό αυτό.

6.1 Τι καλύτερο από το χαρτί

Διαφορές εφαρμογής και χαρτιού

Ξεκινώντας, στην εφαρμογή οι καλλιέργειες αναπαριστώνται με αντίστοιχα εικονίδια η κάθε μία αντιθέτως του χαρτιού όπου χρησιμοποιεί προτάσεις για να τις ξεχωρίζει. Πάντα οι εικόνες αποτυπώνονται ευκολότερα στην ανθρώπινη όραση παρότι τα γράμματα. Κατά δεύτερον, αναφέρθηκε στο προηγούμενο κεφάλαιο ότι κάποιος μπορεί να κάνει κάποιες δράσεις που αλλάζουν τη βασική φάση της καλλιέργειας, όπως λόγου χάρη η φύτευση της καλλιέργειας σε χώμα και κάποιες παρεμβάσεις οι οποίες δεν αλλάζουν όμως τη βασική της φάση. Στο χαρτί οι ενέργειες αυτές απλά αναπαριστούνται με την περιγραφή τους. Η κάθε περιγραφή στο χαρτί-ημερολόγιο συνδέεται με μία ημερομηνία, σε κλίμακα ημέρας και μηνών. Στο πρόγραμμα για να αναπαρασταθούν οι ενέργειες που αλλάζουν τη φάση της καλλιέργειας και οι αντίστοιχες παρεμβάσεις συμβαίνει κάτι πιο εντυπωσιακό. Το εντυπωσιακό είναι ότι μία μπάρα εξέλιξης αντιστοιχεί σε κάθε καλλιέργεια και κάθε καλλιέργεια αντιστοιχεί σε μία εγγραφή σε ένα αρχείο κειμένου. Άνω των καλλιεργειών βρίσκεται η μπάρα χρόνου που είναι παρόμοια του χαρτιού και κάθε καλλιέργεια ξεκινάει από τη δική της ημερομηνία, κλίμακας μέρας και μήνα ομοίως του χαρτιού και εξελίσσεται προς τα δεξιά. Για κάθε ενέργεια που αλλάζει τη φάση της καλλιέργειας υπάρχει ένα συγκεκριμένο εικονίδιο και ένα χρώμα φόντου. Για κάθε ενέργεια που δεν αλλάζει τη φάση της καλλιέργειας υπάρχει ένα συγκεκριμένο εικονίδιο, όχι όμως χρώμα φόντου. Έτσι στο πρόγραμμα μπορεί κάποιος να ξεχωρίσει μέσω χρωμάτων πολύ εύκολα σε ποιά φάση βρίσκεται η κάθε του καλλιέργεια τώρα, σε παρελθοντικό στάδιο, ακόμη και όταν ξεκίνησε η καλλιέργειά του. Οι παρεμβάσεις επίσης διαχωρίζονται πολύ εύκολα πάνω στη μπάρα. Τα προηγούμενα δεν διαχωρίζονται εύκολα στο χαρτί αλλά αντιθέτως πολύ πιο δύσκολα, μιάς και κάποιος θα πρέπει να διατρέξει τις σημειώσεις του με πολύ προσοχή με σκοπό την εύρεση κάποιας παρελθοντικής ενέργειάς του. Ένας άλλος λόγος που κάποιος θα χρειαστεί να γνωρίζει παρελθοντικές ενέργειες, είναι γιατί στην καλλιέργεια κάποιου

προϊόντος μπορεί παρελθοντικά να έχει γίνει κάποια ενέργεια την οποία θα χρειαστεί να γνωρίζει είτε για να ξέρει τι θα χρησιμοποιήσει στη συνέχεια είτε για να μη χρησιμοποιήσει κάποιο υλικό που δεν αντιστοιχεί στην παρούσα φάση της καλλιέργειας. Μία ακόμη μεγάλη διαφορά χαρτιού και προγράμματος είναι η στατιστική διαφοροποίηση των καλλιεργειών. Με τον όρο αυτό εννοούμε ότι μέσω του προγράμματος ο χρήστης μπορεί και παρατηρεί τις καλλιέργειές του και να τις συγκρίνει μιάς και η μία βρίσκεται κάτω από την άλλη και διαφέρει χρωματικά. Στο χαρτί αυτό όμως είναι αδύνατο. Νέα καλλιέργεια μπορεί να προστεθεί στις ήδη υπάρχουσες. Στο χαρτί αυτό μπορεί να σημειωθεί αποτυπώνοντας το γεγονός στην αντίστοιχη ημέρα. Άν κάποιος επιθυμεί να προσθέσει πάρα πολλές καλλιέργειες, θα χρειαζόταν πολλά χαρτιά. Το χαρτί επομένως αποτελεί περιορισμό. Στο πρόγραμμα υπάρχει η λειτουργία δημιουργία καλλιέργειας όπου ο χρήστης όχι μόνο έχει τη δυνατότητα να προσθέσει νέα καλλιέργεια αλλά και οι ήδη υπάρχουσες προτείνονται για εισαγωγή. Θετικό αποτελεί ότι στο πρόγραμμα μπορούν να προστεθούν όσες καλλιέργειες επιθυμείται δίχως περιορισμό. Σύν της άλλης, βασική λειτουργία αποτελεί η μεγένθυση και σμίκρυνση της εφαρμογής. Σκοπός, η καλύτερη οπτική σύγκριση των καλλιεργειών ως προς τη φάση τους σε διάφορα επίπεδα λεπτομέρειας.

6.2 Εμφάνιση και λειτουργία εφαρμογής

Στο κεντρικό παράθυρο ανοίγοντας την εφαρμογή, παρατηρεί κάποιος διάφορα χρώματα, εικονίδια καθώς και μια μπάρα χρόνου. Σε αυτή την παράγραφο θα δωθεί εξήγηση της εμφάνισης των γραφικών στοιχείων της εφαρμογής.

Μπάρα χρόνου

Στο κέντρο της μπάρας χρόνου, όταν ανοίξει η εφαρμογή παρατηρείται ένα κόκκινο χρώμα φόντου και σε αυτό η σημερινή ημερομηνία. Δεξιά της σημερινής ημερομηνίας η μπάρα χρόνου αυξάνεται προοδευτικά με βήμα ημέρας, ενώ αριστερά της μειώνεται προοδευτικά με βήμα ημέρας. Οι ημερομηνίες της μπάρας είναι της μορφής Ημέρα/Μήνας και δεν παρουσιάζονται χρόνια. Αυτό γιατί δεν υπάρχει ουσιαστικά αρχή και τέλος του χρόνου. Μετά το τέλος της μπάρας αυτή ξαναρχίζει από την ημερομηνία που είχε εξαρχής.

Χρωματισμένες μπάρες εξέλιξης

Κάτω από τις ημερομηνίες της μπάρας χρόνου βρίσκονται οι μπάρες εξέλιξης των καλλιεργειών που κάθε μία αποτελείται από Rectangles, στα Ελληνικά ορθογώνιο. Κάθε σειρά και συνεπώς, κάθε μπάρα εξέλιξης φανερώνει μία καλλιέργεια. Το κάθε Rectangle, της κάθε μπάρας εξέλιξης αντιστοιχεί κάτω από μία ημερομηνία της μπάρας χρόνου. Αντικατοπτρίζουν όπως φανερώνει και το όνομά τους, την εξέλιξη των καλλιεργειών με βήμα ημέρας. Τις μπάρες αυτές χρόνου αποτελούν ένα εικονίδιο αριστερά της το οποίο φανερώνει την καλλιέργεια. Στο εικονίδιο αυτό, δε λαμβάνεται υπόψη ο χρόνος κάτω από τον οποίο βρίσκεται, καθώς υπάρχει μόνο για να μας δείχνει τι καλλιέργεια έχει ο χρήστης. Δεξιά του εικονιδίου αρχίζει η μπάρα εξέλιξης. Εσωτερικά της την αποτελούν χρώματα και εικονίδια. Παρατηρείται ότι το χρώμα αλλάζει στη μπάρα ΜΟΝΟ όταν συναντηθούν συγκεκριμένα εικονίδια. Αυτό γιατί τα εικονίδια έχουν χωριστεί σε δράσεις και παρεμβάσεις, στα οποία αναφορά έγινε στο προηγούμενο κεφάλαιο. Εικόνες δράσεων αντικατο-

πυρίζουν τη βασική φάση που βρίσκεται η της καλλιέργεια, ενώ παρέμβαση, μία παρέμβαση πάνω στη βασική της φάση που δεν αλλάζει όμως τη φάση αυτή. Έτσι η φάση αλλάζει μόνο όταν συναντάται εικονίδιο άλλης δράσης ή καλύτερα όταν συναντάται βασικής φάσεως.

Πρότυπες καλλιέργειες

Στο κεντρικό παράθυρο της εφαρμογής, αναφέρθηκαν οι μπάρες εξέλιξης, όπου η κάθε μπάρα δείχνει τη χρονική εξέλιξη της καλλιέργειας. Βέβαια, η εφαρμογή παραθέτει στο ίδιο παράθυρο και κάποιες ενδεικτικές πρότυπες καλλιέργειες τις οποίες μπορεί να συμβουλευτεί ο οποιοσδήποτε χρήστης. Συγκεκριμένα παραθέτονται δύο πρότυπες καλλιέργειες. Για να ξεχωρίσουν οι πρότυπες καλλιέργειες από τις υπόλοιπες, έχει προστεθεί μπορντό χρώμα φόντου στα πάνελτα οποία περιέχουν τις μπάρες των προτύπων καλλιεργειών.

Κάθετη περιγραφή καλλιεργειών

Στο κεντρικό παράθυρο της εφαρμογής, κάνοντας κάποιος κύλιση τελείως αριστερά παρατηρεί κάθετα τα ονόματα των καλλιεργειών. Κάθε όνομα καλλιέργειας περιγράφει την καλλιέργεια εκείνης της συγκεκριμένης γραμμής.

ScrollPane

Η εφαρμογή έχει μπάρες κύλισης. Για να συμβεί αυτό χρησιμοποιήθηκε το αντικείμενο της JAVA Swing που ονομάζεται ScrollPane. Ανοίγοντας, η εφαρμογή εμφανίζεται στη μπάρα χρόνου όπου αναγράφεται η σημερινή ημερομηνία, όπως αναφέρθηκε. Με τη μπάρα χρόνου υπάρχει η δυνατότητα κύλισης αριστερά των ορίων της μπάρας χρόνου και δεξιά αυτών προσφέροντας πλήρη άνεση στο χρήστη.

Ονόματα καλλιεργειών σε κάθετη ταξινόμηση

Κάνοντας κάποιος κύλιση αριστερά στα όρια του παραθύρου, παρατηρεί κάποια ονόματα σε κάθετη ταξινόμηση. Τα ονόματα αυτά φανερώνουν την καλλιέργεια στην οποία σε εκείνη τη σειρά βρίσκεται η δική τους μπάρα εξέλιξης.

Μενού επιλογών

Στο κεντρικό παράθυρο της εφαρμογής, θα παρατηρούσε πολύ εύκολα κάποιος ένα μενού επιλογών στο οποίο βρίσκονται κάποιες λειτουργίες. Εκεί, υπάρχουν οι επιλογές Δημιουργία, Αποθήκευση, Μεγένθυνη, Σμίκρυνση, Βοήθεια.

Δημιουργία

Η επιλογή αυτή χρησιμεύει για δημιουργία νέας καλλιέργειας. Για να συμβεί αυτό, νέο παράθυρο επιλογών εμφανίζεται στο χρήστη όπου εκείνος μπορεί να διαλέξει κάποια από τις διαθέσιμες καλλιέργειες και να την εισάγει στην ημερομηνία που επιθυμεί. Πάντα η σημερινή ημερομηνία εμφανίζεται ως προτεινόμενη. Βέβαια, μπορεί ακόμη να ονομάσει μία δική του νέα καλλιέργεια με

νέο εικονίδιο που ο ίδιος θα επιλέξει. Για δημιουργία νέας καλλιέργειας, νέα γραμμή που θα την περιέχει θα εμφανίζεται κάτω από τις προηγούμενες αλλά πάνω όμως από τις πρότυπες καλλιέργειες.

Αποθήκευση

Χρησιμοποιείται για αποθήκευση των αλλαγών. Ο χρήστης έχει την επιλογή να αποθηκεύσει τις αλλαγές που έκανε, είτε να μην τις αποθηκεύσει, λόγου χάρη σε περίπτωση λάθους.

Μεγένθυση

Αρχικά η επιλογή αυτή βρίσκεται γκριζαρισμένη. Ο λόγος είναι γιατί υπάρχει μία μέγιστη μεγένθυση στην οποία θα παρατηρούνται τα εικονίδια στο κανονικό τους μέγεθος. Μεγένθυση μπορεί να γίνει αφότου έχει γίνει Σμίκρυνση, για επιστροφή σε μεγαλύτερα επίπεδα λεπτομέρειας καθώς και επιστροφή στο αρχικό επίπεδο αυτής.

Σμίκρυνση

Η επιλογή αυτή θα μπορούσε να χαρακτηριστεί καλύτερα ως σύμπτυξη, μιάς και συμπίπτει ανά τρεις ημέρες τη μπάρα χρόνου καθώς και τα αντίστοιχα εικονίδια που την αντιπροσωπεύουν. Οι επιλογές Μεγένθυση και Σμίκρυνση γίνονται με σκοπό να βλέπει ο χρήστης διάφορα επίπεδα λεπτομέρειας. Με τη Σμίκρυνση ή σύμπτυξη θα μπορεί να παρατηρεί σε μεγαλύτερη κλίμακα χρόνου τις καλλιέργειές του, ενώ με τη Μεγένθυση σε μικρότερη κλίμακα χρόνου. Σμικρύνοντας συνεχώς συμπίπτουν ανά τρία τα εικονίδια όπως αναφέρθηκε, αλλά ταυτόχρονα συμπίπτουν και τα εικονίδια κάτω από τους αντίστοιχους χρόνους. Το πιο εικονίδιο θα κυριαρχήσει έναντι των άλλων καθορίζεται σε προγραμματιστικό επίπεδο λαμβάνοντας υπόψη το config.ini.

Βοήθεια

Σε αυτή την επιλογή παρουσιάζεται ένα παράθυρο το οποίο δίνει πληροφορίες σχετικά με την εφαρμογή.

Επιλογή για τερματισμό παραθύρου

Όταν ο χρήστης σκοπεύει να κλείσει την εφαρμογή του εμφανίζεται σχετικό παραθυράκι, δίνοντάς του την επιλογή είτε να αποθηκεύσει πιθανές αλλαγές είτε όχι.

Κεφάλαιο 7

User Interface και πώς έγινε η υλοποίηση

Στο κεφάλαιο αυτό παρουσιάζονται τα βήματα υλοποίησης της εφαρμογής από τους αρχικούς προβληματισμούς έως και την τελική διεπαφή χρήστη.

7.1 Αρχικοί ενδοιασμοί υλοποίησης

Αρχικά καθορίστηκαν τα Windows ως το λειτουργικό σύστημα στο οποίο η εφαρμογή θα γινόταν εκτελέσιμη. Έχοντας επιλέξει τη JAVA ως πρόγραμμα υλοποίησης για τους λόγους που αναφέρθηκαν στο κεφάλαιο τέσσερα, απαραίτητο για την υλοποίηση της εφαρμογής σε JAVA ήταν να βρεθεί πρόγραμμα απαραίτητο πάνω στο οποίο μπορούσε, αυτός ο προγραμματιστικός τρόπος να γραφτεί. Το Netbeans IDE 8. 0, κρίθηκε κατάλληλο για αυτό το σκοπό.

Η εφαρμογή θα μπορούσε να αναπτυχθεί με δύο τρόπους. Ο πρώτος με χρήση βάσης δεδομένων και ο δεύτερος μέσω αρχείων κειμένου κατάλληλης. txt. Έτσι, αναλόγως τον τρόπο υλοποίησης, ή καλύτερα αναλόγως τη βάση που θα ξεκίναγε η εφαρμογή, θα ακολουθούσε είτε δημιουργία βάσης δεδομένων και εν συνεχεία διασύνδεσή της με την εφαρμογή, είτε txt αρχείου ή txt αρχείων, διάβασμα αυτών μέσω της εφαρμογής και μέσου αυτού δημιουργία του γραφικού περιβάλλοντος.

Στο κεφάλαιο τέσσερα δείχθηκε μετά από τη σύγκριση που πραγματοποιήθηκε, ότι επιλέχθηκε το αρχείο κειμένου για διάβασμα μέσω της εφαρμογής. Σε αντίθετη περίπτωση, αν επιλεγόταν η βάση δεδομένων για την καταγραφή των καλλιιεργειών και των διάφορων ενεργειών σε κάθε καλλιέργεια θα έπρεπε να εγκατασταθεί η MySQL για χειρισμό της. Παρουσιάζεται ένα ενδεικτικό παράδειγμα του πώς θα ξεκίναγε η υλοποίηση της εφαρμογής σε αυτή την περίπτωση όπου τα δεδομένα θα μπορούσαν να καταγράφονται πιο εύκολα, να αλλάζουν πεδία εξίσου εύκολα και να συσχετίζονται τα δεδομένα των πινάκων.

```
CREATE DATABASE kipouros;  
USE kipouros;
```

```
CREATE TABLE proion (
```

```
id int not null auto-increment,  
perigrafi char(50) not null,  
sxolio varchar null,  
primary key (id)  
);
```

```
CREATE TABLE energeies(  
id int not null auto-increment,  
perigrafi char(50) not null,  
color_id int not null,  
primary key (id)  
);
```

```
CREATE TABLE xronodiagramma (  
id int not null auto-increment,  
proion_id int not null,  
energeia_id int not null,  
minas int not null,  
etos int not null,  
sxolia varchar null,  
primary key (id)  
);
```

Εφόσον τελικά δεν προτιμήθηκε ο τρόπος αυτός αλλά τα αρχεία κειμένου, η εφαρμογή που θα υλοποιούνταν σε τελικό στάδιο θα εκτελούνταν τοπικά. Αφού επιλέχθηκαν τα αρχεία κειμένου έπρεπε να βρεθούν τα δεδομένα που θα είναι αποθηκευμένα σε αυτά και να καθοριστεί η δομή τους.

7.2 Δράσεις-Παρεμβάσεις

Αναφέρθηκαν σε προηγούμενα κεφάλαια και ειδικότερα στο πέμπτο οι ενέργειες που κάποιος μπορεί να κάνει γενικότερα στις καλλιέργειες. Επίσης, έχει αναφερθεί ότι για τις ανάγκες του προγράμματος είχαμε χωρίσει τις ενέργειες σε δράσεις και παρεμβάσεις για τους σκοπούς της εφαρμογής. Σε αυτή την παράγραφο γίνεται ξανά μία αναφορά στις δράσεις και τις παρεμβάσεις μιάς και αποτέλεσαν τη βάση της εφαρμογής. Το πρόγραμμα αφορά διαχείριση καλλιεργειών μικρής κλίμακας. Ο κάθε χρήστης θα μπορεί να κάνει κάποιες ενέργειες μέσω αυτής, ενέργειες που αντικατοπτρίζουν τις πραγματικές.

Δράσεις αποτελούν την κύρια ενέργεια που κάποιος έχει κάνει στην καλλιέργεια την τελευταία φορά, με άλλα λόγια είναι η φάση στην κάθε καλλιέργεια που βρισκόμαστε ενώ παρεμβάσεις όπως

υποννοεί η ίδια η λέξη, μία παρέμβαση σε κάποια καλλιέργεια που δεν αλλάζει όμως την κύρια φάση της.

Στην κατηγορία των δράσεων ανήκουν ενέργειες όπως όργωμα, σπορά, πότισμα, συγκομιδή, νεκρή περίοδος μέχρι το επόμενο όργωμα, ακόμη και καταστροφή σοδιάς από κάποια φυσικά αίτια όπως χαλάζι, παγετός, φωτιά, ξηρασία. Στη δεύτερη κατηγορία ανήκουν οι παρεμβάσεις, ενέργειες όπως λίπασμα, προσθήκη φυτοφαρμάκων για την προστασία του καρπού, πότισμα καλλιέργειας.

Έχοντας επιλέξει τα αρχεία ως τρόπο αποθήκευσης, όλα τα δεδομένα θα έπρεπε να είναι αποθηκευμένα σε αυτά και την διαχείρισή τους θα την αναλαμβάνει εξολοκλήρου η εφαρμογή. Τα δεδομένα τα οποία θα ήταν αποθηκευμένα στα αρχεία κειμένου, ήταν από μιάς πλευράς οι ενέργειες που κάποιος μπορεί να κάνει σε μία καλλιέργεια, ο τρόπος ο οποίος θα καθορίζεται σε ποιά καλλιέργεια αναφέρονται τα κάθε δεδομένα καθώς και ποιά χρονική περίοδο συνέβη αλλαγή στη φάση κάποιας καλλιέργειας ή κάποια παρέμβαση.

Για τις ενέργειες-δεδομένα που θα ήταν αποθηκευμένα στα αρχεία κειμένου επιλέχθηκαν αρχικά ως δράσεις συγκεκριμένες ενέργειες. Ποτηράκι, Χώμα, Μεταφύτευση, Συγκομιδή, Απευθείας συγκομιδή όπως αγορά γλάστρας, Καταστροφή, Τέλος, αποτελούσαν τις ενέργειες αυτές. Ως παρεμβάσεις επίσης επιλέχθηκαν συγκεκριμένες ενέργειες. Λίπασμα, Αραιώμα, Πότισμα, Φυτοπροστασία, Κλάδεμα αποτελούσαν τις ενέργειες αυτές.

7.3 Αρχικός καθορισμός του config. ini

Έχοντας τις ενέργειες που ορίστηκαν παραπάνω έπρεπε με κάποιο τρόπο να συνδεθούν τις ενέργειες με κάποια εικόνα και κάποιο χρώμα το οποίο στην πορεία θα αποσκοπούσε στην εμφάνιση του τελικού γραφικού περιβάλλοντος της εφαρμογής. Έτσι, θα υπάρχει ένα βασικό αρχείο κειμένου όπου θα γίνεται όλη η κύρια επεξεργασία του προγράμματος, ένα config. ini αρχείο. Σχήμα 7.1.

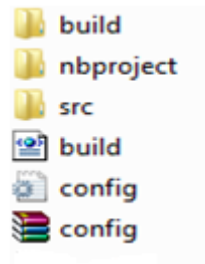
Το config. ini, θα βρίσκεται **στον μητρικό κατάλογο της εφαρμογής** και θα περιέχει όπως αναφέρθηκε πληροφορίες σχετικά με την κάθε ενέργεια και επίδραση. Σχήμα 7.2. Συγκεκριμένα θα συσχετίζει κάθε μία από αυτές με ένα αρχείο εικόνας και ένα γενικότερο χρώμα για τον φόντο. Σκοπός ήταν, όταν θα απεικονίζονται οι ενέργειες και επιδράσεις μιας καλλιέργειας αυτό να γινόταν με την εμφάνιση εικόνων και συγκεκριμένων χρωμάτων στο γραφικό περιβάλλον της εφαρμογής.



Σχήμα 7.1: config. ini

Και για την αρχική δομή του config. ini:

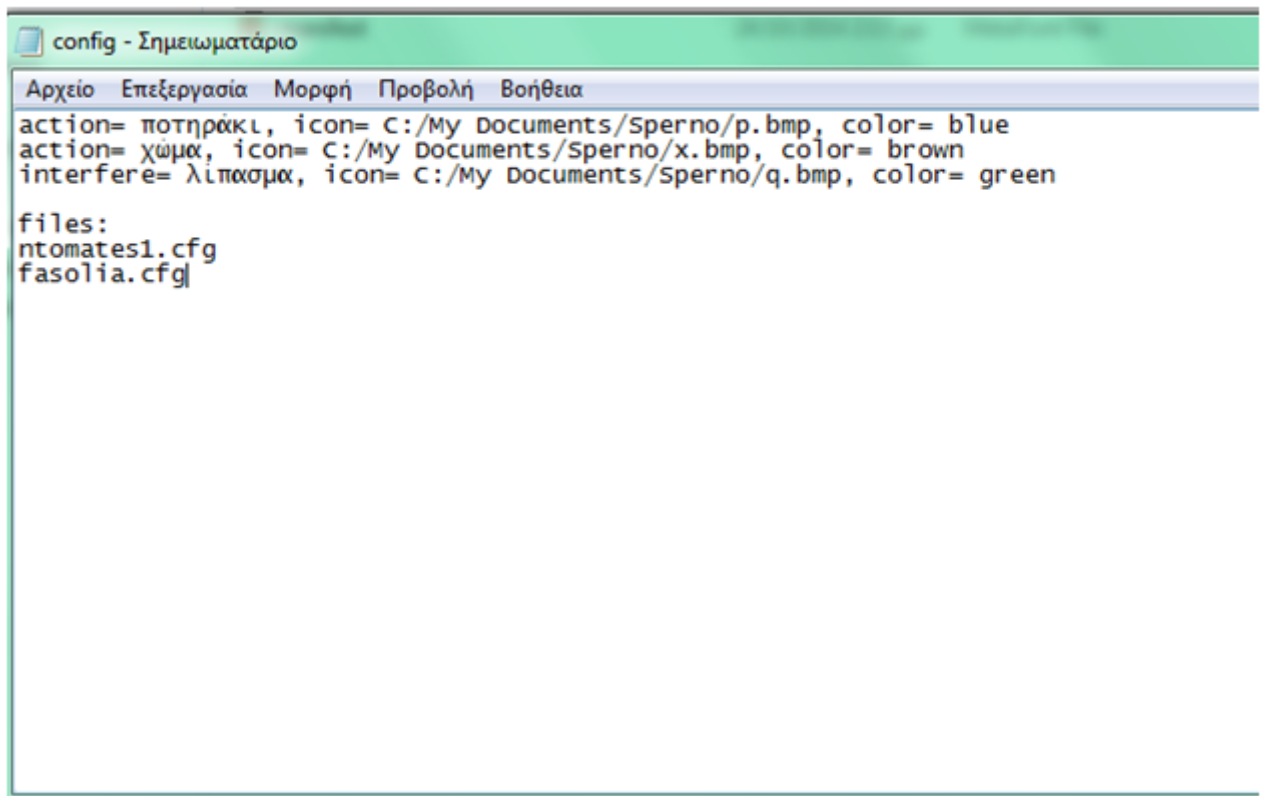
```
action= ποτηράκι, icon= C: /My Documents/Sperno/p. bmp, color= blue  
action= χώμα, icon= C: /My Documents/Sperno/x. bmp, color= brown  
interfere= λίπασμα, icon= C: /My Documents/Sperno/q. bmp, color= green
```



Σχήμα 7.2: Το config. ini στο μητρικό κατάλογο της εφαρμογής

files:
ntomates1. cfg
fasolia. cfg

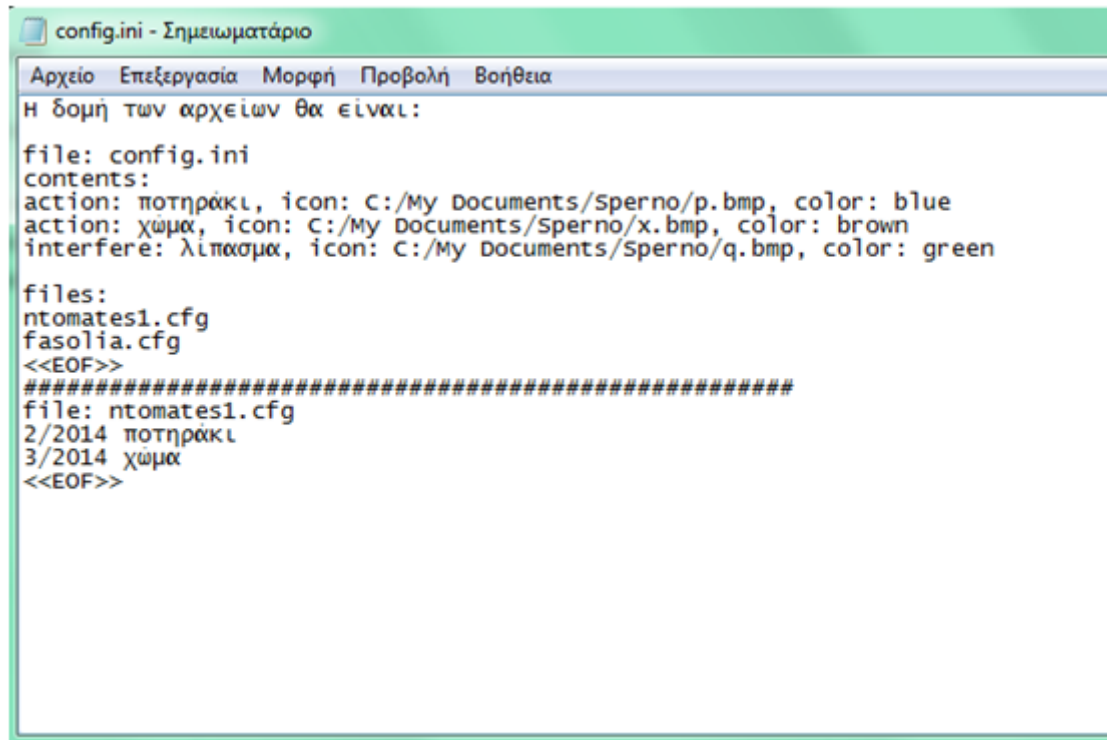
Η αρχική δομή του config. ini φαίνεται και στην εικόνα 7.3, ενώ στην εικόνα 7.4 φαίνεται η αρχική δομή του διαφοροποιημένη.



Σχήμα 7.3: Αρχική έκδοση του config. ini

Η αρχική δομή του **config. ini** προσαρμόστηκε σε μία μεταγενέστερη με σκοπό την ευκολία διαβάσματος και διασύνδεσης των δεδομένων που θα διαβάζονται από αυτό. Αρχικά δε τα χρώματα φόντου που καθορίστηκαν για την κάθε ενέργεια ήταν άγνωστα. Έτσι, η εσωτερική ημιτελής δομή του μετατράπηκε όπως φαίνεται παρακάτω και στην εικόνα 7.5.

action= ποτηράκι, icon= p. png, color= blue



Σχήμα 7.4: Δεύτερη έκδοση του config. ini text

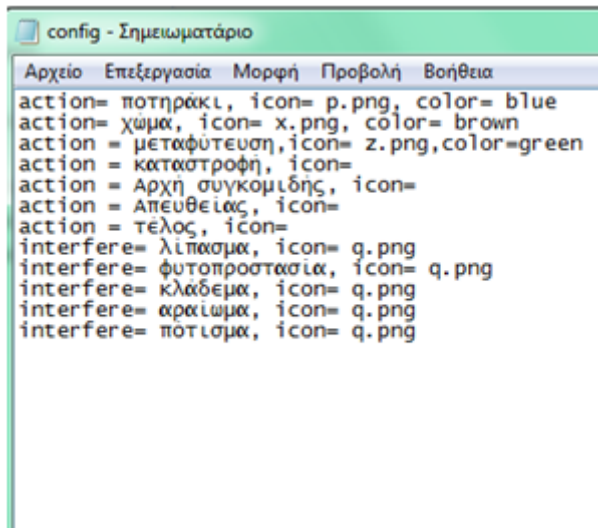
action= χώμα, icon= x. png, color= brown
 action = μεταφύτευση, icon= z. png, color= green
 action = καταστροφή, icon=
 action = Αρχή συγκομιδής, icon=
 action = Απευθείας, icon=
 action = τέλος, icon=
 interfere= λίπασμα, icon= q. png
 interfere= φυτοπροστασία, icon= q. png
 interfere= κλάδεμα, icon= q. png
 interfere= αραίωμα, icon= q. png
 interfere= πότισμα, icon= q. png

Έχοντας το σκελετο του αρχείου αυτού, αυτό στη συνέχεια διαμορφώθηκε με κάποια ενδεικτικά χρώματα, όπως παρουσιάζεται στην εικόνα 7.6.

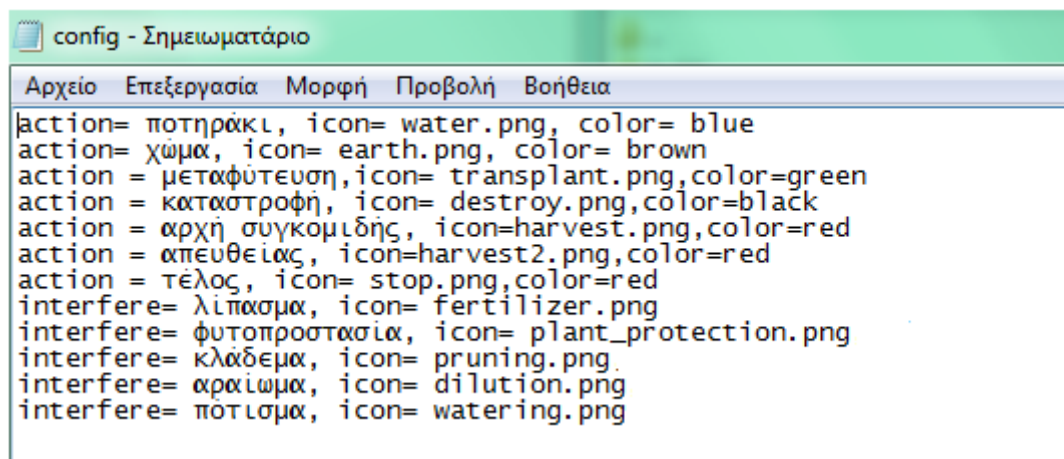
Το config. ini θα περιέχει τα actions, interferes και τα αντίστοιχα icons και colors για τις μπάρες χρόνου της κάθε καλλιέργειας. Επίσης ορίστηκε το σχόλιο, που θα μπορεί να κάνει ο χρήστης στο πρόγραμμα ως παρέμβαση.

7.4 Φάκελοι user, recommended και δομή αρχείων κειμένου

Στον μητρικό κατάλογο της εφαρμογής δημιουργήθηκαν δύο φάκελοι. Ο πρώτος, ονομάστηκε user και ο δεύτερος recommended.



Σχήμα 7.5: config. ini διαμορφωμένο



Σχήμα 7.6: config. ini με κάποια ενδεικτικά χρώματα

Έτσι, ο φάκελος user περιέχει αρχεία txt με πληροφορίες για κάθε καλλιέργεια. Κάθε αρχείο έχει μια συγκεκριμένη δομή. Στην πρώτη γραμμή αναφέρεται το όνομα της καλλιέργειας. Σε κάθε επόμενη γραμμή αναφέρεται η ενέργεια ή η επίδραση που έλαβε χώρα καθώς και η ημερομηνία. Σε κάθε text, φαίνεται κάθε καλλιέργεια με την ίδια δομή αλλά διαφορετική αλληλουχία γεγονότων. Κάποιες καλλιέργειες θα είναι ενεργές και κάποιες θα έχουν τελειώσει. Έτσι, για παράδειγμα ένα τέτοιο αρχείο θα μπορούσε να έχει την δομή:

```

type = φασόλια
10/09/2014, action = ποτηράκι
11/09/2014, action = χώμα
20/09/2014, interfere = λίπάσμα
20/09/2014, interfere = πότισμα
25/09/2014, interfere = το πρώτο λουλούδι

```

30/09/2014, action = μεταφύτευση

03/10/2014, action = τέλος

Στον φάκελο recommended υπάρχουν αρχεία με παρόμοια δομή με σχέση με τη δομή των αρχείων στο φάκελο user. Τα αρχεία αυτά λειτουργούν ως πρότυπα για να δίνουν συμβουλές ή κατευθύνσεις για μελλοντικές ενέργειες. Εφόσον δεν είναι δυνατόν να υπάρχει συγκεκριμένος χρόνος, απλώς αναγράφεται η σειρά εκτέλεσης των ενεργειών αυτών. Έτσι για παράδειγμα ένα τέτοιο αρχείο θα μπορούσε να έχει την παρακάτω δομή. Επίσης στην εικόνα 7.7, δείχνεται η τοποθεσία του φακέλου που περιέχει τις πρότυπες καλλιέργειες.

type = φασόλια

1, action = ποτηράκι

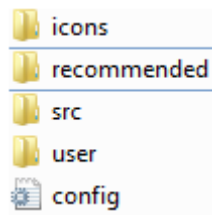
2, action = χώμα

8, interfere = λίπασμα

8, interfere = πότισμα

12, action = μεταφύτευση

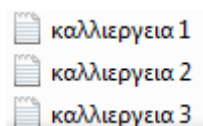
13, action = τέλος



Σχήμα 7.7: Ο φάκελος recommended στο μητρικό κατάλογο της εφαρμογής

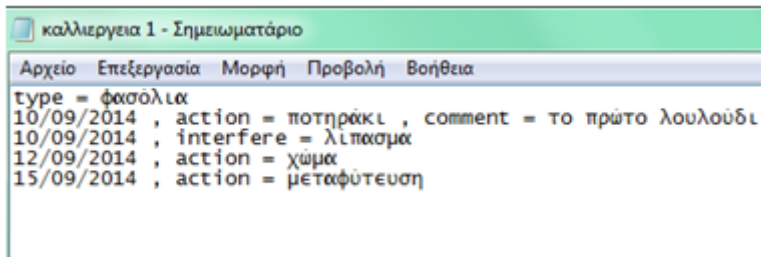
Στον φάκελο user δημιουργήθηκαν τρία παραδείγματα τριών αντίστοιχων καλλιεργειών, για να καθοριστεί η δομή αυτών των αρχείων κειμένου, στα οποία αποθηκεύονται πληροφορίες για την κάθε καλλιέργεια.

Δείχνονται αρχικά εξωτερικά τα τρία παραδείγματα των καλλιεργειών στην εικόνα 7.8 και στη συνέχεια στις εικόνες 7.9, 7.10 και 7.11 η εσωτερική δομή των τριών παραδειγμάτων αντίστοιχα.

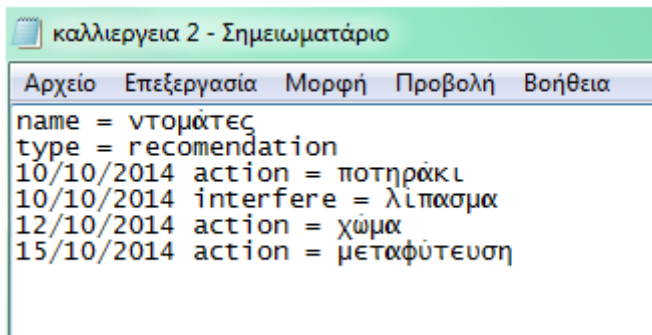


Σχήμα 7.8: Τρία παραδείγματα ενεργών καλλιεργειών εξωτερικά

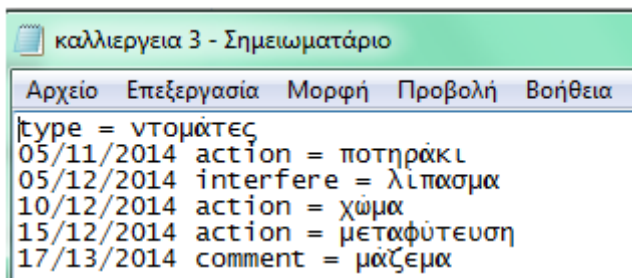
Η κάθε καλλιέργεια λοιπόν, θα καθορίζεται από το τι τύπου είναι με τη λέξη type, από κάτω η χρονική αλληλουχία ή καλύτερα οι ημερομηνίες και δίπλα τους τι action ή interfere αντιστοιχεί.



Σχήμα 7.9: Εσωτερική δομή πρώτης καλλιέργειας



Σχήμα 7.10: Εσωτερική δομή δεύτερης καλλιέργειας



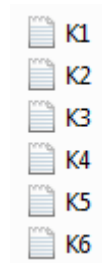
Σχήμα 7.11: Εσωτερική δομή τρίτης καλλιέργειας

Τα ονόματα, που θα αντιπροσωπεύουν τα αρχεία κειμένου και συνεπώς την κάθε καλλιέργεια, θα ακολουθούν συγκεκριμένο τρόπο ονομασίας. Έτσι μεταγενέστερα άλλαξε ο τρόπος ονομασίας των αρχείων. Το δεύτερο πρόθεμα ξεχωρίζει τις καλλιέργειες και αυτό φαίνεται στο σχήμα 7.12. Αυτό ισχύει για τις ενεργές και τις παλιές καλλιέργειες, όπου ενεργές αυτές που βρίσκονται σε εξέλιξη και παλιές αυτές που έχουν ήδη τελειώσει.

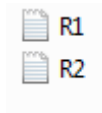
Το πρόγραμμα, εκτός των άλλων θα παρουσιάζει στο χρήστη και κάποιες ενδεικτικές καλλιέργειες με κύριο αυτοσκοπό να τον υποβοηθούν, παραθέτοντάς του πρότυπα παραδείγματα καλλιεργειών.

Για τις recommended, θα ξεκινάνε από το αρχικό πρόθεμα R. Το δεύτερο πρόθεμα θα ξεχωρίζει τις καλλιέργειες. Σχήμα 7.13.

Ο λόγος για τον οποίο βάλαμε με αυτό τον τρόπο τα ονόματα των text files, είναι για να



Σχήμα 7.12: Νέος τρόπος ονομασίας για τα text files τύπου ενεργές και παλιές



Σχήμα 7.13: Νέος τρόπος ονομασίας για τα text files τύπου recommended

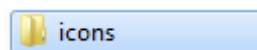
αποφευχθεί overwrite κατά την εκτέλεση του προγράμματος.

7.5 Εικόνες

Τα icons του προγράμματος αποτελούν εικόνες κατάλληλης του τύπου. png. Το πρόγραμμα που υλοποιούμε αφορά καλλιέργειες. Για την κάθε καλλιέργεια θα αντιστοιχεί κάποιο εικονίδιο. Επίσης στο πρόγραμμα οι ενέργειές μας όπως αναφέρθηκε θα είναι είτε Actions είτε interfere. Κάθε δράση και κάθε παρέμβαση θα την αποτελεί επίσης κάποιο εικονίδιο. Από αυτά, τρεις φάκελοι θα χρησιμοποιούνται για αποθήκευση των εικόνων.

7.5.1 Φάκελος icons

Ο φάκελος icons, θα περιέχει τρεις υποφακέλους. Ένα φάκελο actions, έναν άλλο interfere και έναν ακόμη που θα ονομάζεται plants. Σχήμα 7.14.

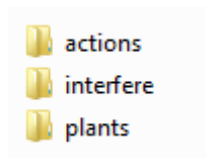


Σχήμα 7.14: Φάκελος icons

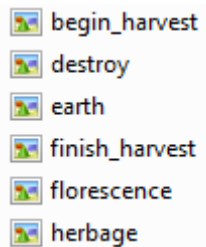
Κάθε ένας υποφάκελος περιέχει αρχεία εικόνας της μορφής png. Τοποθετήθηκαν και οι τρεις φάκελοι μαζί για χάρη ευκολίας. Έτσι, η εξωτερική δομή των φακέλων που ανήκουν στο φάκελο icons, το οποίο βρίσκεται **στο μητρικό κατάλογο της εφαρμογής** φαίνεται στην εικόνα 7.15 και στις εικόνες 7.16, 7.17 και 7.18, δίνονται παραδείγματα περιεχομένων των τριών φακέλων αντίστοιχα.

7.6 Προσχέδιο στο χαρτί του γραφικού περιβάλλοντος

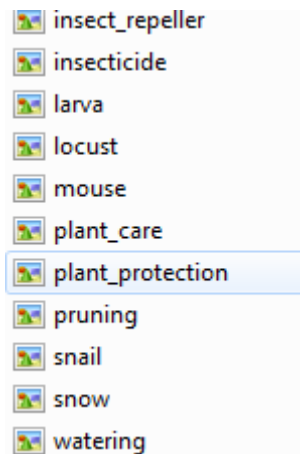
Έχοντας τα αρχεία κειμένου και τις εικόνες που χρειάστηκαν για τη μετέπειτα υλοποίηση του προγράμματος, σχεδιάστηκε ένα προσχέδιο του γραφικού περιβάλλοντος, το οποίο έμελε να υλο-



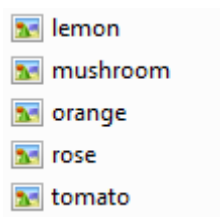
Σχήμα 7.15: Περιεχόμενα φακέλου icons



Σχήμα 7.16: Περιεχόμενα φακέλου actions



Σχήμα 7.17: Περιεχόμενα φακέλου interfere



Σχήμα 7.18: Περιεχόμενα φακέλου plants

ποιηθεί στη συνέχεια.

Στο σχήμα 7.19 φαίνονται οι ενεργές και οι παλιές καλλιέργειες όπως σχεδιάστηκαν, όπου κάθε γραμμή αντιπροσωπεύει και διαφορετική καλλιέργεια. Στο σχήμα 7.20 φαίνονται προτεινόμενες καλλιέργειες. Ομοίως, κάθε γραμμή αντιπροσωπεύει και κάποια άλλη καλλιέργεια. Στο σχήμα 7.21, δείχνονται οι ερμηνίες αυτών των σχεδίων στο χαρτί.



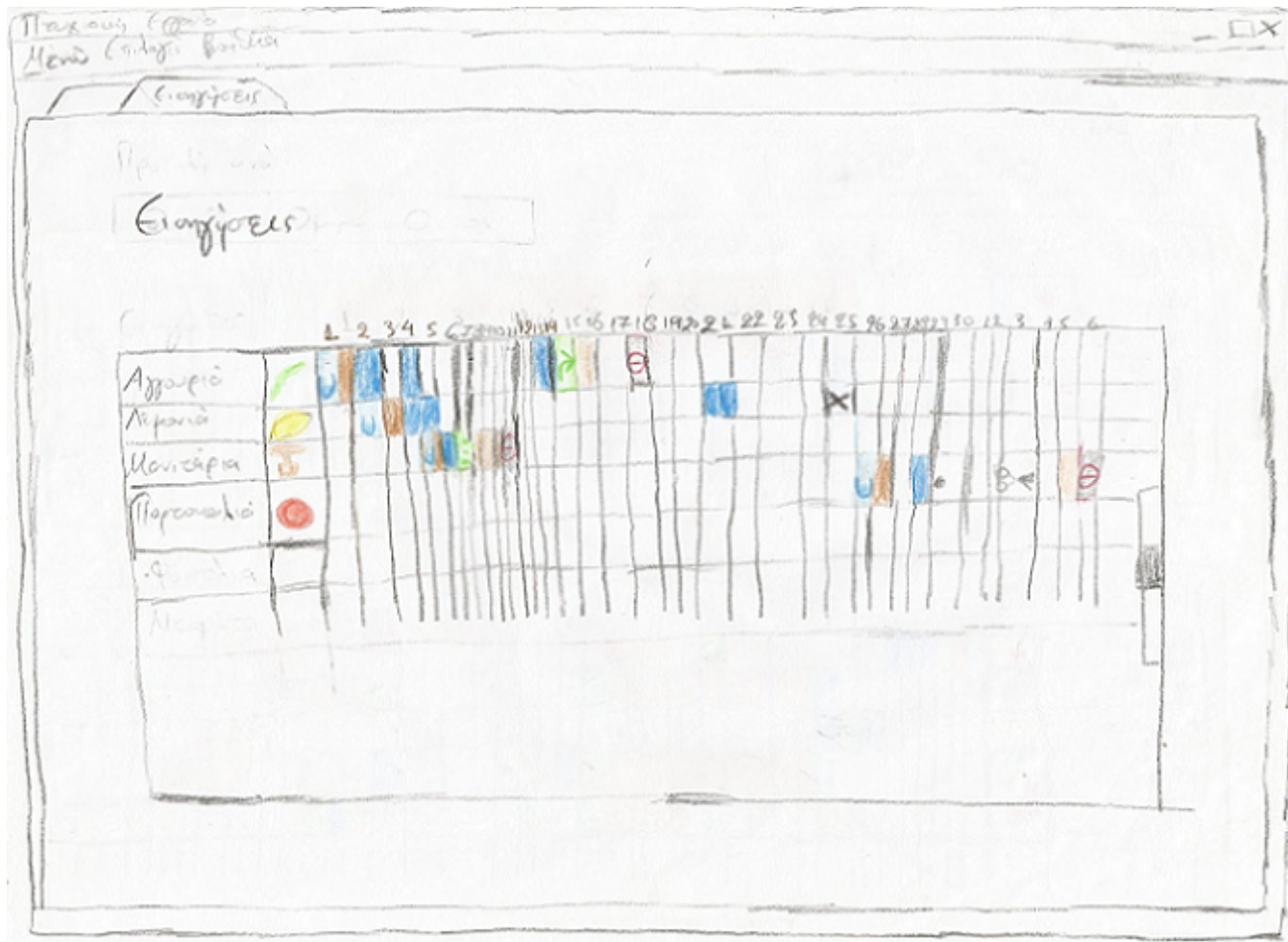
Σχήμα 7.19: Ενεργές και παλιές καλλιέργειες

7.7 Γραφικό περιβάλλον διεπαφής χρήστη - GUI

Μετά το προσχέδιο στο χαρτί της εφαρμογής, επόμενο βήμα ήταν η υλοποίηση σε γραφικό περιβάλλον διεπαφής μέσω Java Swing, με χρήση των εργαλείων της. Αναλύθηκε σε προηγούμενο κεφάλαιο η Java και στη συνέχεια στο Graphical User Interface. Στη Java Swing μπορούν να χρησιμοποιηθούν τα components που παρέχονται για σχεδιασμό της εφαρμογής. Δύο οι τρόποι υλοποίησής της. Ο πρώτος υλοποίηση με JButtons και ο δεύτερος με Rectangles. Αρχικά έγινε υλοποίηση της εφαρμογής με JButtons και αυτό φαίνεται στο σχήμα 7.22, αλλά τελικά προτιμήθηκε ο δεύτερος τρόπος και αυτό φαίνεται στο σχήμα 7.23. Προτιμήθηκε ο δεύτερος γιατί αυτοσκοπός στην υλοποίηση της εφαρμογής ήταν η απλότητα στο χρήστη και ένα μεγάλο μέρος αυτού αποτελεί το οπτικό μέρος της, να είναι όσο το δυνατόν πιο εύχρηστο και απλοϊκό σε κάθε χρήση.

7.7.1 Υλοποίηση με JButtons

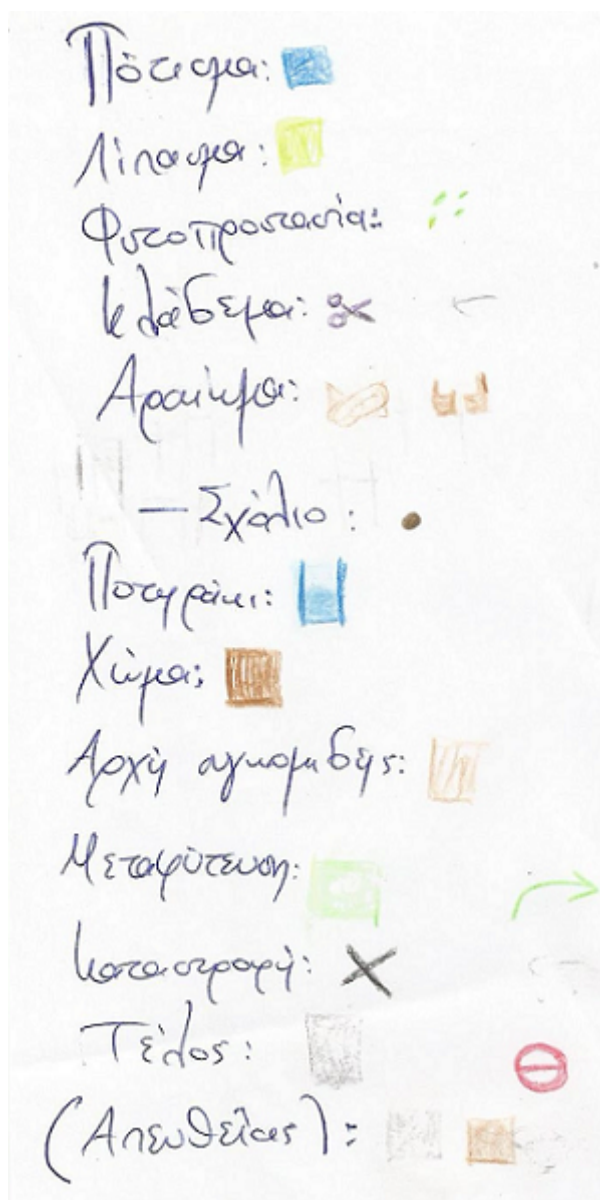
Στην υλοποίηση του προγράμματος με JButtons, οι χρωματισμοί υλοποιήθηκαν σε κουμπιά της java, τα JButtons. Στοιχισμένα κουμπιά σε σειρές και στοιβές αποτυπώνουν τις ενεργές, παλιές



Σχήμα 7.20: Προτεινόμενες καλλιέργειες

και πρότυπες καλλιέργειες, κάθε μια να βρίσκεται σε ξεχωριστή σειρά. Δεξιά κάθε καλλιέργειας με χρωματισμούς παρουσιάζεται η φάση της καλλιέργειας. Εικονίδια βρίσκονται πάνω από κάθε φάση καλλιέργειας. Τα εικονίδια αντιπροσωπεύουν τις δράσεις και τις παρεμβάσεις. Κάτω από εικονίδιο δράσης αλλάζει η φάση της καλλιέργειας, δηλαδή το χρώμα φόντου της ενώ κάτω από εικονίδιο παρέμβασης δεν αλλάζει το χρώμα φόντου και άρα ούτε και η φάση. Αριστερά του προγράμματος φαίνονται οι καλλιέργειες με τα αντίστοιχα εικονίδια και αριστερά κάθε εικονιδίου καλλιέργειας, βρίσκεται η περιγραφή της. Για παράδειγμα για καλλιέργεια ντομάτας φαίνεται το εικονίδιο της ντομάτας του εικονιδίου, φαίνεται η επιγραφή ντομάτα ως JLabel. Πάνω των καλλιεργειών φαίνονται τρεις επιλογές. Εκτός ενεργές, Εκτός παλλιές, Εκτός recommended. Πατώντας πάνω σε κάποιο από αυτά στη διεπαφή δε θα φαίνονται οι καλλιέργειες οι οποίες επιλέχθηκαν. Για παράδειγμα πατώντας στην επιλογή Εκτός ενεργές, στη διεπαφή θα φαίνονται οι παλλιές και recommended καλλιέργειες. Στη java, τα κουμπιά αυτά ονομάζονται radio buttons. Τέλος, παρατηρείται ένα menu, στη Java αναφέρεται ως JMenu και πάνω σε αυτό να έχει τις επιλογές Μενού, Επιλογές και Βοήθεια.

Η επιλογή των JButtons δεν προτιμήθηκε γιατί τα JButtons δυσχεραίνουν το οπτικό πεδίο της εφαρμογής. Πρωταρχικός στόχος της εφαρμογής, είναι να είναι απλοϊκή και ξεκάθαρη σε κάθε χρήστη. Έτσι η διεπαφή αυτή αναπτύχθηκε, αλλά έμμελε να αναπτυχθεί κάτι πιο λειτουργικό και



Σχήμα 7.21: Ερμηνίες σχεδίων

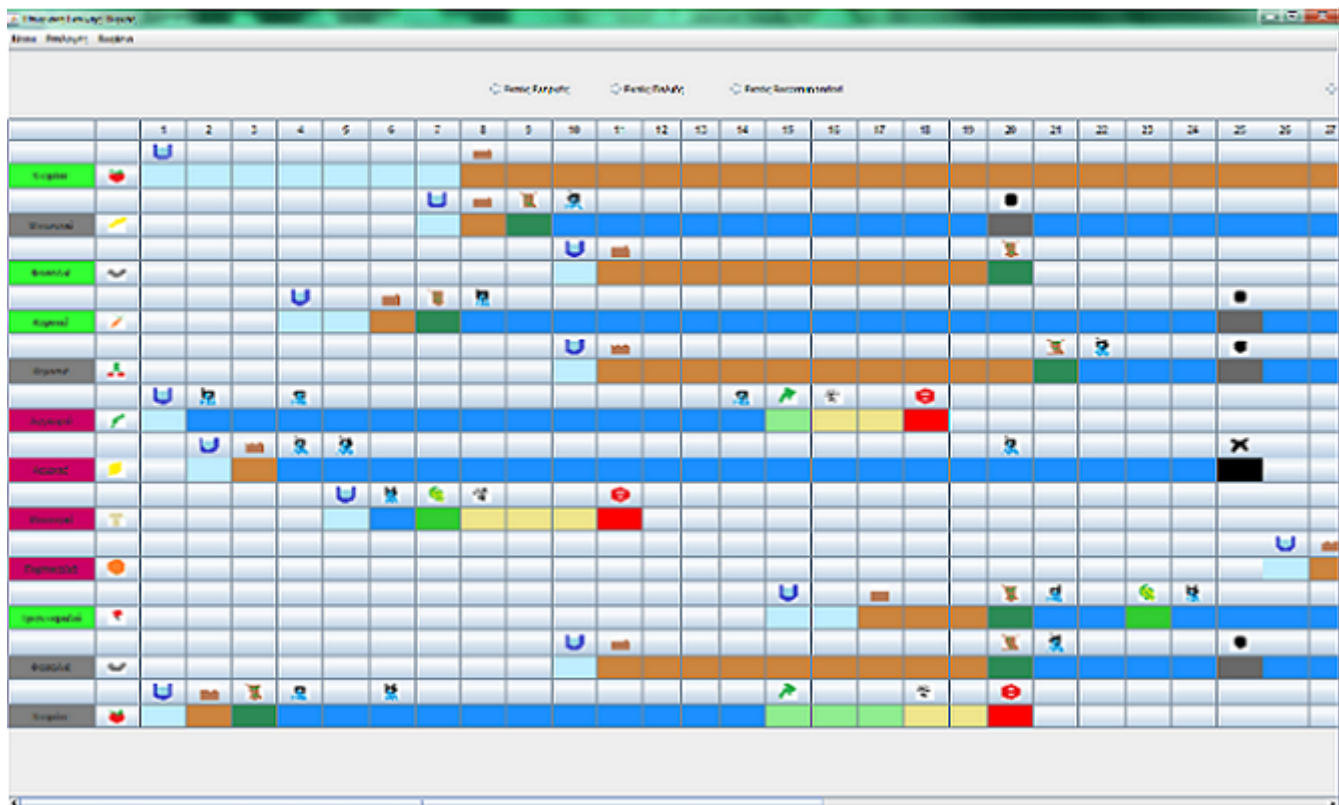
αποδοτικό.

7.7.2 Υλοποίηση με Rectangles

Η επιλογή που προτιμήθηκε είναι το Rectangle. Όπως υποδεικνύει και το όνομά του, είναι η Αγγλική ερμηνεία των ορθογωνίων. Σκοπός ήταν να αποφευχθούν οι γραμμές των JButtons που παρεμβάλλονται.

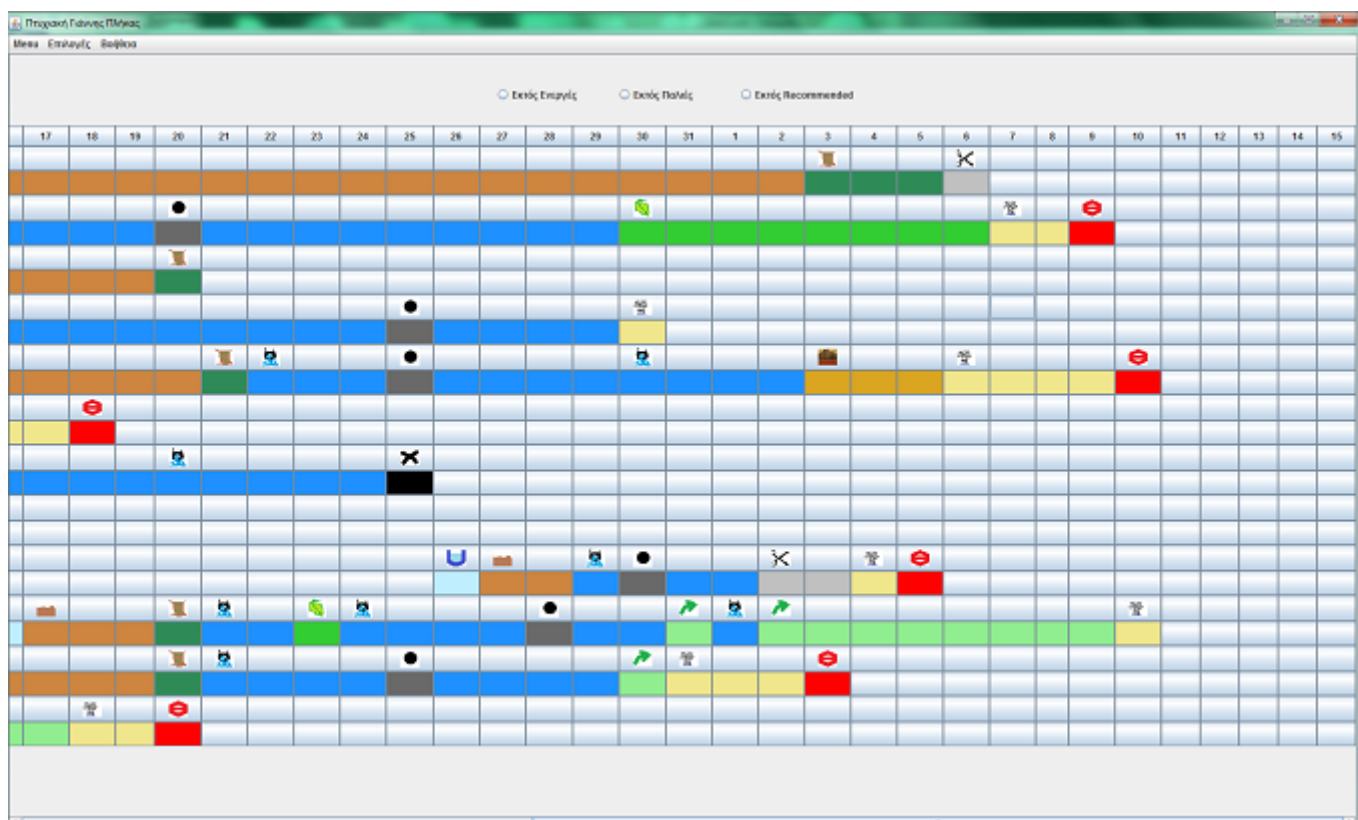
Παρακάτω φαίνεται με βηματικό τρόπο η υλοποίηση του γραφικού περιβάλλοντος με Rectangles.

Σε πρώτο στάδιο δημιουργήθηκαν rectangles με τυχαίους χρωματισμούς. Διαβάζονται οι ενεργές και οι καλλιέργειες που έχουν τελειώσει, μέσα από το φάκελο user. Αρχικά τέσσερα αρχεία



Σχήμα 7.22: Διεπαφή προγράμματος με JButtons, όπως τη βλέπει ο χρήστης

κειμένους ήταν αποθηκευμένα σε αυτόν και άρα τέσσερις καλλιέργειες διαβάστηκαν. Όμως δεν υλοποιήθηκε να εμφανίζονται αρκετά Rectangles στον οριζόντιο άξονα. Χρησιμοποιήθηκαν λοιπόν JLabels για εμφάνιση ονομάτων των καλλιεργειών και αντίστοιχα της ημερομηνίας όπως αυτά διαβάστηκαν από τα αρχεία. Σχήμα 7.24. Στη συνέχεια, διαβάστηκαν τα ονόματα των καλλιεργειών και αντιστοιχίστηκαν με τις εικόνες τους. Αυτό πραγματοποιήθηκε διαμέσου του αρχείου info.txt. Μέσω του αρχείου config.ini αντιστοιχίστηκαν οι δράσεις και οι παρεμβάσεις με κάποιο εικονίδιο. Σχήμα 7.25. Το config.ini υπέστη επεξεργασία στη συνέχεια για αντιστοίχιση χρωμάτων φόντου των δράσεων και παρεμβάσεων. Σχήμα 7.26. Αυτοσκοπός όμως ήταν μόνο οι δράσεις να αλλάζουν το χρώμα φόντου. Στην πορεία υπήρξε προβληματισμός γιατί τα Rectangles εμφανιζόταν με διαφορετικό πλάτος κι έτσι έπρεπε να βρεθεί κάποιος τρόπος να εμφανίζονται με το ίδιο πλάτος. Έτσι διαβάστηκε ένα εικονίδιο, λεπτής άσπρης γραμμής και αποθηκεύτηκε στα Rectangles. Με αυτό τον τρόπο όλα τα Rectangles έχοντας το ίδιο object, όπως συνηθίζεται να λέμε σε γλώσσα JAVA, απέκτησαν ίδιο πλάτος. Σχήμα 7.27. Ταυτόχρονα το σχόλιο προστέθηκε ως τύπος interfere και οι καλλιέργειες ευθυγραμμίστηκαν. Σχήμα 7.28. Παρ' όλα αυτά όμως, με το να παρουσιάζονται στην οθόνη Rectangles με μία λεπτή άσπρη γραμμή στο εσωτερικό τους, δεν έφερνε το επιθυμητό αποτέλεσμα. Έτσι μετατράπηκαν σε transparent εικόνες για να απαλοιφθεί το λευκό χρώμα. Ταυτόχρονα διαβάστηκαν όλες οι ενεργές και παλιές καλλιέργειες του φακέλου user. Τα Vector εικονίδια πάντα αποτελούνε μία πιο καλαίσθητη επιλογή. Για απεικόνιση του εικονιδίου καλλιέργειας χρησιμοποιήθηκαν Vectorized images. Σχήμα 7.29. Κατόπιν διαβάσματος των ενεργών και τελειωμένων καλλιεργειών, έπρεπε να διαβαστούν και οι πρότυπες καλλιέργειες του φακέλου recommended. Η αρχική απόπειρα δεν ήταν επιτυχής καθώς όλα τα εικονίδια έπεφταν το ένα πάνω στο άλλο. Σχήμα 7.30. Προστέθηκε στη συνέχεια η μπάρα χρόνου άνω των καλλιεργ-



Σχήμα 7.23: Διεπαφή προγράμματος με JButtons, έχοντας κάνει scroll δεξιά

γειών με κέντρο της τη σημερινή μέρα. Κάθε καλλιέργεια ξεκινάει ποιά από δική της ημερομηνία. Προσθήκη scroll-bar για κύλιση κρίθηκε αναγκαία. Σχήμα 7.31. Η εφαρμογή σε αυτό το σημείο εκτελούσε τη βασική της λειτουργία. Έπρεπε όμως ο χρήστης να έχει τη δυνατότητα να δημιουργεί ο ίδιος νέες καλλιέργειες και να προσθέτει νέες ενέργειες-συμβάντα όποτε αυτός το επιθυμεί. Έτσι δημιουργήθηκε μενού με επιλογές Δημιουργία για δημιουργία νέας καλλιέργειας, Αποθήκευση για αποθήκευση πιθανών αλλαγών και παράθυρο εισαγωγής νέου συμβάντος. Οποιαδήποτε αποθήκευση θα αλλάζει πλέον αυτομάτως τα αρχεία κειμένου των καλλιεργειών. Σχήμα 7.32. Οι Vectorized images αναφέραμε πίο πάνω ότι πάντα αποτελούνε την πιο καλαίσθητη επιλογή. Έτσι, αντικαταστάθηκαν όλες οι εικόνες της εφαρμογής με Vector εικονίδια. Προστέθηκαν ακόμη οι επιλογές Μεγένθυνη και Σμίκρυνση στο κύριο μενού για να μπορεί κάποιος να παρατηρήσει διαφορετικά επίπεδα λεπτομέρειας των καλλιεργειών. Επιπλέον, προστέθηκε η επιλογή Βοήθεια σε αυτό. Οι recommended καλλιέργειες σχεδιάστηκαν να διαχωρίζονται με το σύνολο ενεργών και παλιών καλλιεργειών, όπως τις έχουμε κατηγοριοποιήσει, μέσω μιας μπάρας κύλισης. Κανένα κλικ δεν είναι δυνατό πάνω τους για εισαγωγή συμβάντος. Ακόμη το κουμπί κλείσιμο του παραθύρου θα έχει επιλογές διαθέσιμες προς το χρήστη. Ο ίδιος θα μπορεί να επιλέξει και κατά την έξοδο, αποθήκευση. Όλα αυτά αντικατοπτρίζονται στο σχήμα 7.33. Στην εφαρμογή, κύριος αυτοσκοπός όμως είναι η ευκολία χρήσης του. Με αυτό το γνώμονα, οι recommended καλλιέργειες τοποθετήθηκαν στο ίδιο παράθυρο με τις υπόλοιπες καλλιέργειες. Για να διαχωρίζονται αυτές με τις ενεργές και παλιές καλλιέργειες, σχεδιάστηκαν τα panels που τα περιέχουν με μπορντό χρώμα φόντου. Στο τελικό στάδιο της εφαρμογής, επιλέχθηκαν να μη διαβάζονται τα χρόνια από τις καλλιέργειες αλλά ημέρες και μήνες. Δεν παραθέτεται επιπλέον εξήγηση σε αυτό το σχήμα, μιάς και το τελικό στάδιο της εφαρμογής όπως παρατηρείται στην εικόνα 7.34, αναλύθηκε στο έκτο κεφάλαιο.

φασόλια	χώμα 11/09/2014	λίπασμα 20/09/2014
ντομάτα	χώμα 01/09/2014	λίπασμα 03/09/2014
φασόλια	χώμα 11/09/2014	λίπασμα 20/09/2014
ντομάτα	χώμα 01/11/2014	λίπασμα 03/01/2014

Σχήμα 7.24: Αρχικό διάβασμα αρχείων και απόπειρα αποτύπωσης δεδομένων καλλιεργειών.

	 ποτηράκι 11/09/2014	 χώμα 20/09/2014
	 ποτηράκι 01/09/2014	 χώμα 03/09/2014
	 ποτηράκι 11/09/2014	 χώμα 20/09/2014
	 ποτηράκι 01/11/2014	 χώμα 03/01/2014

Σχήμα 7.25: Διασύνδεση ονόματος καλλιέργειας, δράσεων και παρεμβάσεων με τα εικονίδια τους, μέσω των αρχείων info. txt και config. ini.

	 10/09/2014	 11/09/2014	 20/09/2014  20/09/2014									
 01/09/2014	 01/09/2014	 03/09/2014	 03/09/2014	 06/09/2014								
	 10/09/2014	 11/09/2014	 20/09/2014  20/09/2014									
	 01/11/2014	 01/11/2014	 03/11/2014	2	2	2	2	2	2	2	2	

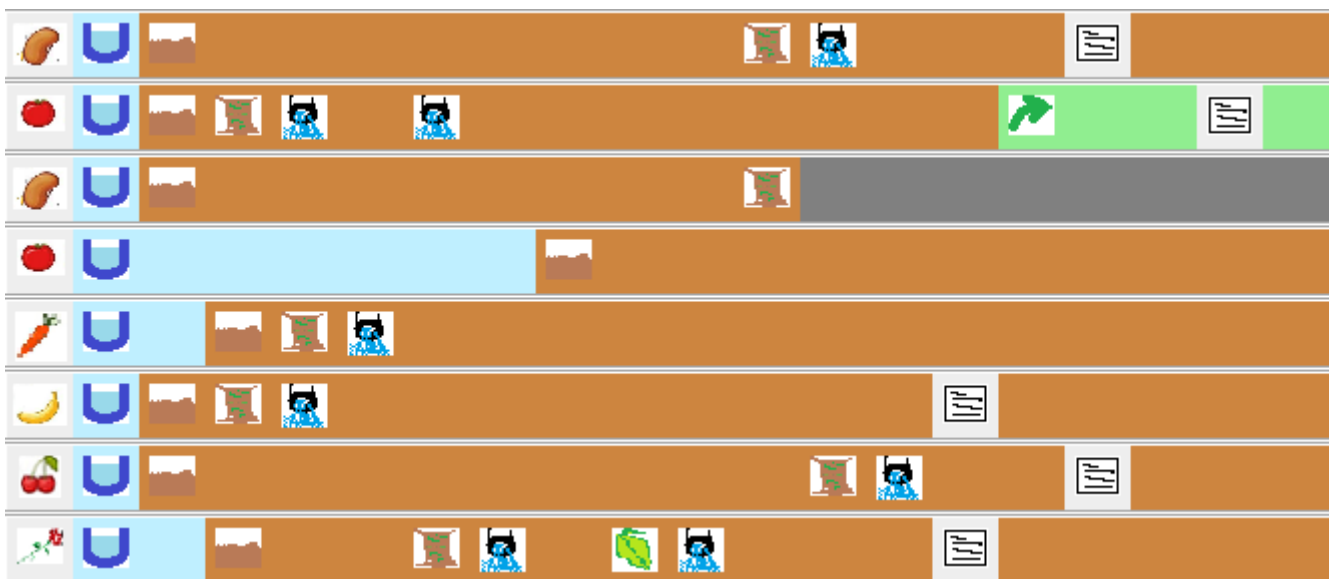
Σχήμα 7.26: Χρώμα φόντου για δράσεις και παρεμβάσεις.



Σχήμα 7.27: Προσθήκη λευκής γραμμής για ίδιο πλάτος Rectangle.



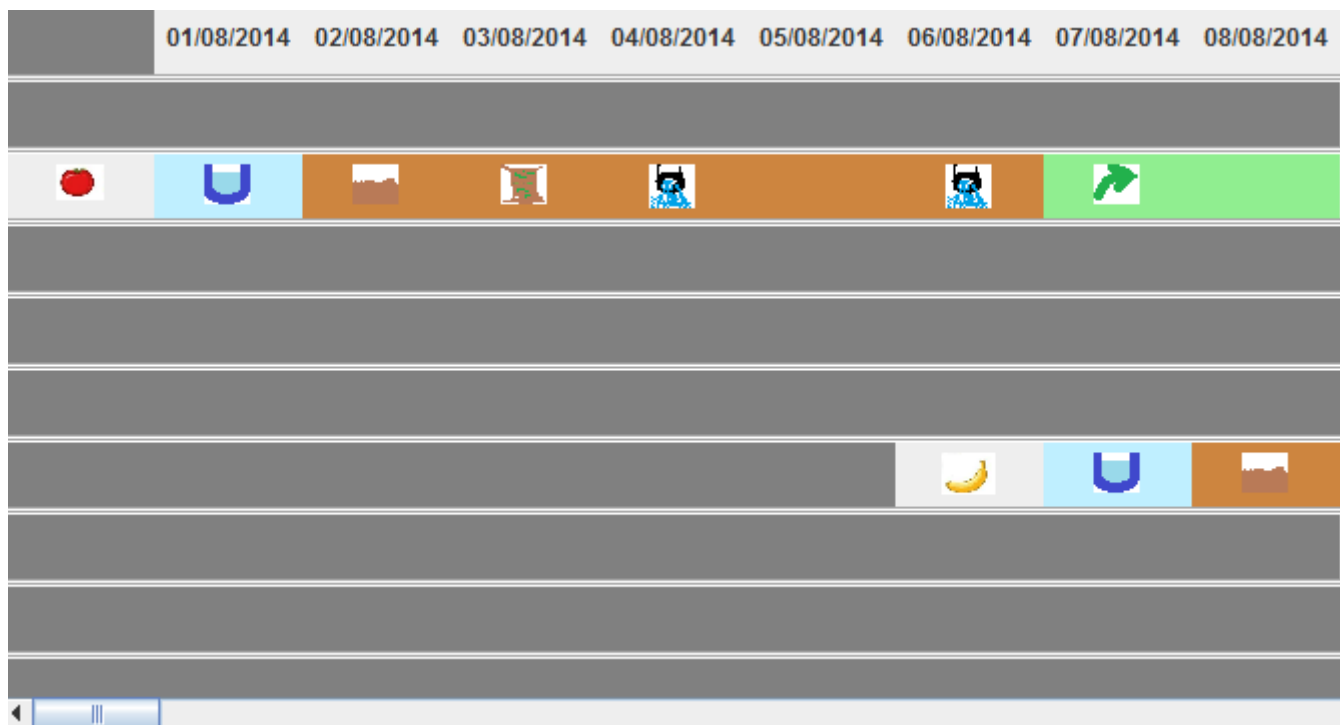
Σχήμα 7.28: Καλλιέργειες ευθυγραμμίστηκαν, σχόλιο ως interfere.



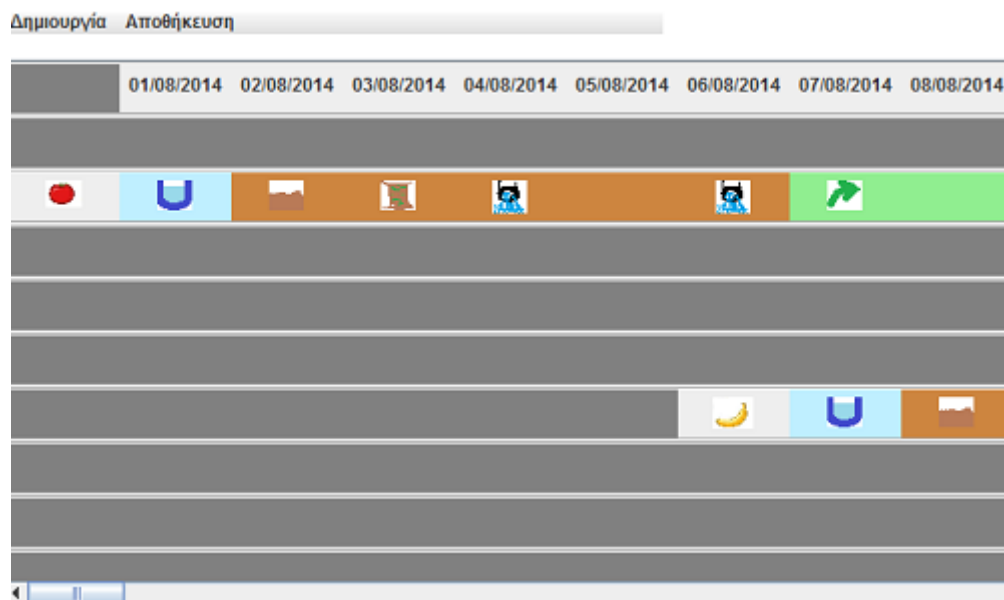
Σχήμα 7.29: Vectorized images για είδη καλλιέργειας, διάβασμα όλων των ενεργών και παλιών καλλιεργειών.



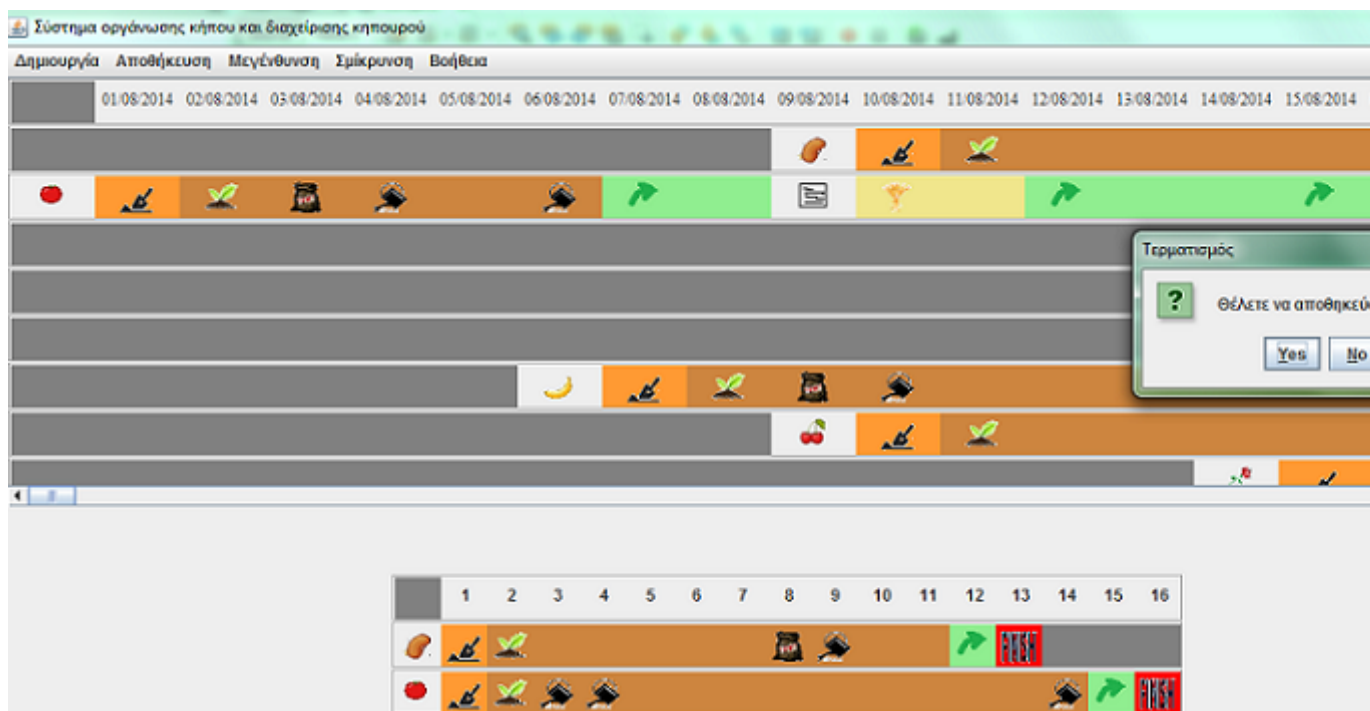
Σχήμα 7.30: Διάβασμα recommended τύπου καλλιεργειών, εικονίδια το ένα πάνω στο άλλο.



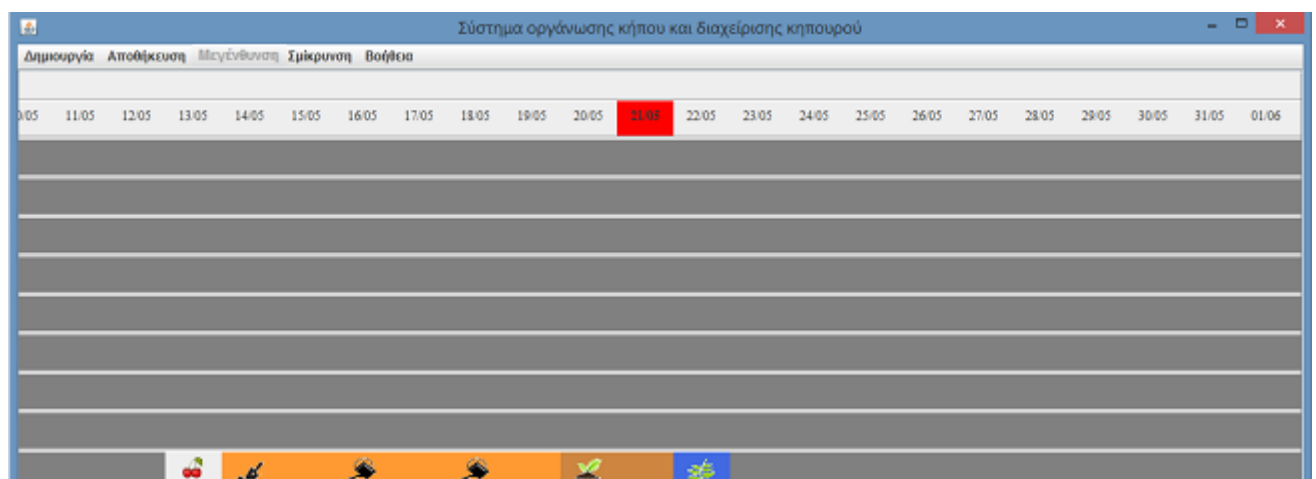
Σχήμα 7.31: Προσθήκη μπάρας χρόνου με κάθε καλλιέργεια να ξεκινάει από τη δική της ημερομηνία, μπάρα κύλισης για ευχέρεια.



Σχήμα 7.32: Μενού επιλογών με λειτουργίες Δημιουργία και Αποθήκευση, προσθήκη παραθύρου εισαγωγής νέου συμβάντος.



Σχήμα 7.33: Μεγένθυση, Σμίκρυνση, Βοήθεια οι επιπλέον λειτουργίες του μενού, επιλογή αποθήκευσης κατά το κλείσιμο της εφαρμογής, σωστή εμφάνιση recommended καλλιεργειών καθώς και Vectorized images για όλη την εφαρμογή.



Σχήμα 7.34: Κεντρικό παράθυρο με κέντρο τη σημερινή ημερομηνία η οποία εμφανίζει μέρες και μήνες όχι όμως χρόνια, recommended καλλιεργείες στο ίδιο παράθυρο με τις ενεργές με μπορντό χρώμα φόντου, υλοποίηση λειτουργίας αλλαγής κλίμακας.

Κεφάλαιο 8

Αρχιτεκτονική

Στο παραπάνω κεφάλαιο εξηγήσαμε τον τρόπο με τον οποίο έγινε η υλοποίηση της εφαρμογής με βηματικό τρόπο. Δεν παρουσιάσαμε όμως την τελική δομή του προγράμματος. Με το όρο αρχιτεκτονική, εννοούμε το πώς είναι αρχιτεκτονικά δομημένο το πρόγραμμα δηλαδή ποιος είναι ο σκελετός του προγράμματος. Η αρχιτεκτονική του προγράμματος, αφορά το πώς υλοποιήθηκε σε τελικό στάδιο το κομμάτι της εγκατάστασης, τα αρχεία του προγράμματος και εν γένει η JAVA.

8.1 Αρχεία

Αρχείο config.ini

Στο προηγούμενο κεφάλαιο παρουσιάστηκε το αρχείο config.ini. Το αρχείο αυτό βρίσκεται στον μητρικό κατάλογο της εφαρμογής.

Ξαναεπισημαίνεται ότι το αρχείο αυτό περιέχει πληροφορίες σχετικά με την κάθε ενέργεια και επίδραση συσχετίζοντας κάθε μία από αυτές με ένα αρχείο εικόνας και ένα γενικότερο χρώμα για τον φόντο, με κύριο αυτοσκοπό, να απεικονίζονται οι ενέργειες και επιδράσεις μιας καλλιέργειας και αυτό να γίνεται με την εμφάνιση εικόνων και συγκεκριμένων χρωμάτων στο γραφικό περιβάλλον της εφαρμογής.

Η τελική δομή του config.ini, είναι η εξής:

```
action= ποτηράκι, icon= water. png, color= 191 239 255  
action= χώμα, icon= earth. png, color= 205 133 63  
action = μεταφύτευση,icon= transplant. png,color=144 238 144  
action = καταστροφή, icon= destroy. png,color=255 255 255  
action = συγκομιδή, icon=harvest. png,color=240 230 140  
action = απευθείας, icon=harvest2. png,color=255 0 0  
action = τέλος, icon= stop. png,color=255 0 0  
interfere= λίπασμα, icon= fertilizer. png, color=46 139 87  
interfere= φυτοπροστασία, icon= plant_protection. png,color=50 205 50  
interfere= κλάδεμα, icon= pruning. png,color=192 192 192  
interfere= αραίωση, icon= dilution. png,color= 218 165 32  
interfere= πότισμα, icon= watering. png,color=30 144 255
```

interfere= σχόλιο, icon= comment. png, color=105 105 105

Αντιστοιχίζεται κάθε δράση, που απεικονίζεται ως action, σε μία εικόνα και ένα χρώμα. Αντιστοιχίζεται επίσης, κάθε παρέμβαση με μία εικόνα και ένα χρώμα.

Στο πρόγραμμα κύρια σημασία έχει να αλλάζει το χρώμα των δράσεων και όχι των παρεμβάσεων γιατί στο γραφικό περιβάλλον αλλάζει το χρώμα φόντου των δράσεων ενώ οι παρεμβάσεις δεν έχουν χρώμα φόντου.

Έτσι στην τελική δομή του config.ini, χρησιμοποιήθηκε διαφορετικός τρόπος για καθορισμό του χρώματος φόντου του κάθε action.

Ο τρόπος αυτός αφορά το μοντέλο χρώματος RGB, όπου αποτελεί ένα προσθετικό μοντέλο που χρησιμοποιεί τα χρώματα κόκκινο, πράσινο και μπλέ προσθέτοντάς τα με διαφορετικούς τρόπους ώστε να παράγει διαφορετικό χρωματισμό. Το όνομα RGB προέρχεται από τα αρχικά Red, Green, Blue. Η λογική του μοντέλου αυτού αφορά το γεγονός ότι κάθε απόχρωση των τριών χρωμάτων που αναφέρθηκαν μπορεί να απεικονιστεί με διαφορετικό αριθμό, με κάθε αριθμό να αντιστοιχεί σε συγκεκριμένο χρώμα του φάσματος χρωμάτων. Συνδυασμός και των τριών αποχρώσεων με την προσθετική λογική του RGB εξάγει κάποιο συγκεκριμένο χρώμα. Ως παράδειγμα αναφέρεται το χρώμα από το ποτηράκι, όπως φαίνεται στο config.ini και είναι color= 191 239 255, το οποίο αποτελεί μία απόχρωση του γαλάζιου χρώματος.

Με τα RGB colors ορίζονται δηλαδή τα χρώματα. Βέβαια, σχεδιάστηκε το config.ini να **λειτουργεί με δυναμικό τρόπο**. Αυτό σημαίνει ότι αλλάζοντας τα RGB colors σε κάποιο action έχει κάποιος τη δυνατότητα, να καθορίσει το επιθυμητό χρώμα του φόντου της κάθε δράσης στο γραφικό περιβάλλον.

Δεύτερο παράδειγμα τελικής έκδοσης του config.ini:

action= φύτευση καλλιέργειας σε χώμα, icon= plantation_in_earth. png, color= 255 154 51
 action= προσθήκη χώματος, icon= earth. png, color= 205 133 63
 action = μεταφύτευση, icon= transplant. png, color=144 238 144
 action = καταστροφή σποράς, icon= destroy. png, color=255 255 255
 action = αρχή συγκομιδής, icon= begin_harvest. png, color=240 230 140
 action = τέλος συγκομιδής, icon= finish_harvest. png, color=255 0 0
 action = φύτευμα καλλιέργειας σε φυτόριο, icon= plantation_seedbed. png, color=175 238 238
 action = αγορά φυτορίου σε χώμα, icon= new_plant_dirt. png, color=255 0 255
 action = αγορά έτοιμου φυτορίου, icon= new_plant_ready. png, color=199 21 133
 action = βλάστηση φυτού, icon= herbage. png, color=65 105 225
 action = ανθοφορία καλλιέργειας, icon= florescence. png, color=138 43 226
 interfere= προσθήκη λιπάσματος, icon= fertilizer. png, color=46 139 87
 interfere= φυτοπροστασία, icon= plant_protection. png, color=50 205 50
 interfere= κλάδεμα, icon= pruning. png, color=192 192 192
 interfere= αραιώση πυκνότητας καλλιέργειας, icon= dilution. png, color= 218 165 32
 interfere= πότισμα καλλιέργειας, icon= watering. png, color=30 144 255
 interfere= σχόλιο, icon= comment. png, color=105 105 105
 interfere= μελίγκρα, icon= aphid. png, color=105 105 105

interfere= κάμπια, icon= larva. png, color=105 105 105
 interfere= ακρίδα, icon= locust. png, color=105 105 105
 interfere= ποντίκι, icon= mouse. png, color=105 105 105
 interfere= σκουλήκι, icon= worm. png, color=105 105 105
 interfere= σαλιγκάρι, icon= snail. png, color=105 105 105
 interfere= υγρασία, icon= humidity. png, color=105 105 105
 interfere= χαλάζι, icon= hail. png, color=105 105 105
 interfere= ξηρασία, icon= drought. png, color=105 105 105
 interfere= χιόνι, icon= snow. png, color=105 105 105
 interfere= πλημμύρα, icon= flood. png, color=105 105 105
 interfere= ψύχος, icon= cold. png, color=105 105 105
 interfere= χρήση εντομοαπωθητικού, icon= insect_repeller. png, color=105 105 105
 interfere= χρήση εντομοκτόνου, icon= insecticide. png, color=105 105 105
 interfere= τοποθέτηση φράχτη γύρω από καλλιέργεια, icon= fence. png, color=105 105 105

Αρχεία για καταγραφή καλλιεργειών

Αναφέραμε ότι τα αρχεία που θα χρησιμοποιούνται για καταγραφή των καλλιεργειών από το πρόγραμμα είναι αυτά τα οποία θα αφορούν τις ενεργές και παλιές καλλιέργειες και τα αρχεία που θα αφορούν τις πρότυπες καλλιέργειες.

Για τις ενεργές και παλιές καλλιέργειες:

Στον μητρικό κατάλογο της εφαρμογής βρίσκεται ο φάκελος **user**. Μέσα στο φάκελο αυτό αποθηκεύτηκαν τα αρχεία κειμένου τα οποία θα χρησιμοποιούνται για διάβασμα των ενεργών και παλιών καλλιεργειών, όπως αναφέρθηκε στο προηγούμενο κεφάλαιο. Επίσης, αναφέρθηκε ότι τα ονόματα των αρχείων θα ακολουθούν αριθμητική αύξουσα σειρά με αρχικό πρόθεμα το K. Η τελική εσωτερική δομή των αρχείων αυτών, ακολουθεί τα εξής δύο παραδείγματα:

K1:

type = φασόλια
 10/08/2014, action = ποτηράκι
 11/08/2014, action = χώμα
 20/08/2014, interfere = λίπασμα
 21/08/2014, interfere = πότισμα
 25/08/2014, interfere = το πρώτο λουλούδι
 30/08/2014, action = μεταφύτευση
 31/08/2014, interfere = αρχή συγκομιδής
 03/09/2014, action = τέλος

K2:

type = ντομάτα
 01/08/2014, action = ποτηράκι

02/08/2014, action = χώμα
 03/08/2014, interfere = πότισμα
 04/08/2014, action = μεταφύτευση
 06/08/2014, action = μεταφύτευση
 07/08/2014, action = μεταφύτευση
 18/08/2014, interfere = αρχή συγκομιδής
 20/08/2014, action = τέλος

Στον ίδιο μητρικό κατάλογο, βρίσκεται και ο φάκελος **recommended**, ο οποίος όπως υποδεικνύει και το όνομά του εκεί αποθηκεύονται τα αρχεία κειμένου τα οποία θα χρησιμοποιούνται για διάβασμα των προτύπων καλλιεργειών. Αντίστοιχα τα ονόματα των αρχείων θα ακολουθούν αριθμητική αύξουσα σειρά αλλά με αρχικό πρόθεμα το R, όπως επίσης ειπώθηκε στο προηγούμενο κεφάλαιο. Για παράδειγμα R1,R2. Η τελική εσωτερική δομή των αρχείων αυτών ακολουθεί τα εξής δύο παραδείγματα:

R1:

type = φασόλια
 01/08, action = ποτηράκι
 02/08, action = χώμα
 08/08, interfere = λίπασμα
 09/08, interfere = πότισμα
 12/09, action = μεταφύτευση
 13/09, action = τέλος

R2:

type = ντομάτα
 01/09, action = ποτηράκι
 02/09, action = χώμα
 03/09, interfere = πότισμα
 04/09, interfere = πότισμα
 14/09, interfere = πότισμα
 15/09, action = μεταφύτευση
 16/09, action = τέλος

Αρχείο info.txt

Στον τελικό καθορισμό της δομής των αρχείων, δημιουργήθηκε ένα επιπλέον αρχείο, το **info.txt** με κύριο αυτοσκοπό όταν αυτό διαβάζεται διαμέσου της εφαρμογής να αντιστοιχίζονται τα ονόματα των καλλιεργειών με την εικόνα τους.

Για να γίνει μετάβαση στο αρχείο αυτό, διατρέχεται μέσω του μητρικού καταλόγου της εφαρμογής ο φάκελος icons και στη συνέχεια ο φάκελος plants. Μέσα στο φάκελο plants βρίσκεται και

το αρχείο info.txt. Στον ίδιο φάκελο βρίσκονται και οι εικόνες των καλλιέργειών.

Εσωτερική δομή του αρχείου info.txt :

```
descr: ντομάτα, file: icons\plants\tomato. png
descr: φασόλια, file: icons\plants\bean. png
descr: καροτιά, file: icons\plants\carot. png
descr: μπανάνα, file: icons\plants\banana. png
descr: κεράσι, file: icons\plants\cherry. png
descr: τριαντάφυλλο, file: icons\plants\rose. png
descr: αγγούρι, file: icons\plants\cucumber. png
descr: πορτοκάλι, file: icons\plants\orange. png
descr: μανιτάρι, file: icons\plants\mushroom. png
descr: λεμόνι, file: icons\plants\lemon. png
```

Κάθε descr, ισοδυναμεί με το όνομα της κάθε καλλιέργειας και στην ίδια σειρά αντιστοιχίζεται και η εικόνα της.

Το info.txt, είναι όπως και το config.ini, επίσης **δυναμικό**. Υπάρχει δηλαδή η δυνατότητα διαμέσω αυτού του αρχείου να προστεθεί νέα καλλιέργεια, να ονομαστεί κατά τον επιθυμητό τρόπο και να την αντιστοιχεί σε μία εικόνα που θα επιλεγεί από το χρήστη. Κάθε νέα εικόνα που θα επιλέγεται θα πρέπει όμως να βρίσκεται στον ίδιο φάκελο με το info.txt, δηλαδή στο φάκελο plants και με κατάλληλη .png, όπως τα έχουμε ορίσει.

Ένα παράδειγμα προσθήκης νέας καλλιέργειας με νέα εικόνα αντίστοιχα αποτελεί η τελευταία σειρά του επόμενου παραδείγματος:

```
descr: ντομάτα, file: icons\plants\tomato. png
descr: φασόλια, file: icons\plants\bean. png
descr: καροτιά, file: icons\plants\carot. png
descr: μπανάνα, file: icons\plants\banana. png
descr: κεράσι, file: icons\plants\cherry. png
descr: τριαντάφυλλο, file: icons\plants\rose. png
descr: αγγούρι, file: icons\plants\cucumber. png
descr: πορτοκάλι, file: icons\plants\orange. png
descr: μανιτάρι, file: icons\plants\mushroom. png
descr: λεμόνι, file: icons\plants\lemon. png
descr: πιπεριά, file: icons\plants\pepper. png
```

8.2 JAVA

Οι βασικές κλάσεις που έχουν υλοποιηθεί στη JAVA για τις ανάγκες της εφαρμογής είναι:

Agros:

Στόχος της κλάσης αυτής, είναι η αναζήτηση των αρχείων των φακέλων user και recommended καθώς και του αρχείου config.ini και στη συνέχεια η επεξεργασία των περιεχομένων τους. Στην κλάση αυτή βρίσκεται και η main κλάση του προγράμματος. Τα πεδία που περιέχει είναι: Ένα αλφαριθμητικό USER_DIRECTORY για καθορισμό του φακέλου user, ένα αλφαριθμητικό RECOMMENDED_DIRECTORY για καθορισμό του φακέλου recommended.

Μέθοδοι κλάσης Agros:

private static Scanner openFile(String fileName, boolean showErrorMessage): Χρησιμοποιείται για διάβασμα των δεδομένων των αρχείων. Στο σώμα της και στην εντολή catch καλεί την createNewFile(String new FileName) . Επιστρέφει τα νέα αρχεία.

private static Scanner createNewFile(String newFileName): Δημιουργία νέου αρχείου. Η Scanner openFile(String fileName, boolean showErrorMessage), καλεί αυτή τη συνάρτηση για αυτό το σκοπό. Επιστρέφει το νέο αρχείο στην openFile.

private static ArrayList<ConfigRecord> organize_config(): Σε αυτή την κλάση αρχικά καλείται η συνάρτηση openFile και πίο συγκεκριμένα η Scanner openFile(String fileName, boolean showErrorMessage), διαβάζεται το αρχείο config.ini, επεξεργάζονται τα δεδομένα του και προσθ  εται στο ArrayList τα δεδομένα κάθε σειράς. Επίσης ορίζεται ένα int id, και αυξάνεται για κάθε νέα καλλιέργεια που συναντάμε για να δηλωθεί μία ακέραιη τιμή σε κάθε εγγραφή του config που θα χρησιμοποιηθεί αργότερα για να τα συγκρίνουμε. Όλα αυτά γίνονται με την εντολή configRecords.add(new ConfigRecord(id++, type, name.trim(), icon.trim(), color.trim())) το οποίο αυξάνεται. Επιστρέφει τα δεδομένα του config.ini και πίο συγκεκριμένα τα configRecords που προστέθηκαν στο ArrayList.

private static ArrayList<String> search_directory_files(String dir_name): Με αυτή τη συνάρτηση βρίσκεται η τοποθεσία των αρχείων τα οποία στη συνέχεια τα ανοίγουμε και επεξεργαζόμαστε τα δεδομένα. Επειδή δε βρίσκονται στον μητρικό κατάλογο της εφαρμογής όπως το αρχείο config.ini, απαραίτητη είναι η συνάρτηση String get_working_directory(), η οποία και καλείται. Με την εντολή return fileContents επιστρέφονται τα περιεχόμενα των αρχείων.

private static String get_working_directory(): Η συνάρτηση που καλείται όπως αναφέρθηκε από την search_directory_files(String dir_name). Επιστρέφει το working_directory, δηλαδή την τοποθεσία των αρχείων, όπως υποδεικνύει και το όνομά της.

private static ArrayList<Kalliergeia> processFileContents(ArrayList<String> contents): Αυτή η συνάρτηση, επεξεργάζει τα δεδομένα των αρχείων. Με την εντολή kalliergies.add(kalliergeia), όπου kalliergeia = new Kalliergeia(kalliergeia_type, fileName, type) προσθέτουμε στο ArrayList όλες τις καλλιέργειες. Γίνεται επίσης επεξεργασία των δεδομένων των αρχείων και αναλύεται η κάθε σειρά. Έτσι αναλόγως αν η κάθε σειρά αφορά ACTION ή INTERFERE τύπο προσθέτουμε στην κάθε καλλιέργεια Records όπου ο αριθμός των Records αντιστοιχεί με τον αριθμό των σειρών του κάθε αρχείου που γίνεται επεξεργασία. Αυτό με τις εντολές kalliergeia.addRecord(new

`Record(fields.get(0), RecordType.ACTION, name.trim()))` και `kalliergeia.addRecord(new Record(fields.get(0), RecordType.INTERFERE, name.trim()))` για τύπο δράσης και τυπο παρέμβασης αντίστοιχα. Πριν το τέλος της συνάρτησης, ταξινομούμε τα αρχεία με τη συνάρτηση `sortRecord()` της κλάσης `Kalliergeia`. Η συνάρτηση επιστρέφει τις καλλιέργειες.

`public static void main(String[] args)`: Η `main` του προγράμματος. Εδώ αρχικά διαβάζεται το αρχείο `config` και φτιάχνεται λίστα με `interferes` και `actions`. Κάθε εγγραφή έχει χρώμα, όνομα και τύπο. Έπειτα γίνεται αναζήτηση του φακέλου `user` και στη συνέχεια επεξεργασία των αρχείων του. Εκεί φτιάχνεται λίστα με τις καλλιέργειες και στη συνέχεια λίστα με τα δεδομένα κάθε γραμμής αναλόγως τύπου `ACTION` και `INTERFERE`. Με τον ίδιο τρόπο γίνεται αναζήτηση του φακέλου `recommended` και επεξεργασία των αρχείων του. Στο πρόγραμμα ως `kalliergeies_fantastikes` εννοούνται οι πρότυπες καλλιέργειες. Εκεί φτιάχνεται λίστα με τις καλλιέργειες και στη συνέχεια λίστα με τα δεδομένα κάθε γραμμής αναλόγως τύπου `ACTION` και `INTERFERE`.

Στη συνέχεια όλα τα δεδομένα των καλλιεργειών, πραγματικών και προτύπων καλλιεργειών και της λίστας `configRecords` τα περνιούνται ως παράμετροι στο γραφικό περιβάλλον όπου γίνεται η μετέπειτα επεξεργασία.

ConfigRecord:

Η κλάση αυτή δίνει αντικείμενα που το καθένα αποθηκεύει μια καταχώρηση του αρχείου `config.ini`. Τα πεδία της κλάσης είναι: Ένα αλφαριθμητικό `name` για την ονομασία της ενέργειας ή της επίδρασης, ένα αντικείμενο `type` της `RecordType` με δυνατές τιμές `ACTION` και `INTERFERE`, ένα αντικείμενο της κλάσης `ImageIcon` για την αποθήκευση της εικόνας `png` και ένα αντικείμενο της κλάσης `Color`, για το χρώμα του φόντου.

Μέθοδοι κλάσης ConfigRecord:

`public ConfigRecord(int importance, String type, String name, String icon, String color)`: Η μέθοδος αυτή, δέχεται τις παραμέτρους της από την κλάση `Agros`. Συγκεκριμένα από τη σχέση `kalliergeia.addRecord(new Record(fields.get(0), RecordType.INTERFERE, name.trim()))` της συνάρτησης `private static ArrayList<Kalliergeia> processFileContents(ArrayList<String> contents)`. Έχοντας τα δεδομένα αυτά συνδέει την κάθε δράση με μία εικόνα και ένα χρώμα φόντου σύμφωνα με τα `RGB colors` και την κάθε παρέμβαση με ένα εικονίδιο. Ορίζεται επίσης ένα `importance`, σύμφωνα με το `id` της κλάσης `Agros`.

`public RecordType getRecordType()`: Επιστρέφει τον τύπο της καλλιέργειας. Ο τύπος της καλλιέργειας μπορεί να είναι είτε `ACTION` είτε `INTERFERE`. Η μέθοδος αυτή συνδέεται με την κλάση `RecordType`, όπου έχει οριστεί ο τύπος καλλιέργειας ως απαρίθμηση.

`public String getName()`: Επιστρέφει το όνομα της καλλιέργειας.

`public ImageIcon getImageIcon()`: Η μέθοδος αυτή επιστρέφει την εικόνα, δράσης ή παρέμβασης όπως έχουν οριστεί στο `config`.

`public Color getColor():` Επιστρέφει το χρώμα φόντου των τύπων δράσεων.

`public int getImportance():` Το `importance` που επιστρέφει, είναι αυτό που του δώθηκε με το `id` στην κλάση `Agros`. Το `id` αυξανόταν για κάθε εγγραφή προς τα κάτω του `config.ini`. Άρα μεγαλύτερο `importance` έχει η τελευταία εγγραφή και μειώνεται όσο πάμε προς την πρώτη. Επιστρέφει αυτό το `importance`.

`public boolean isMoreImportantTo(ConfigRecord cr):` Ακριβώς αυτό που αναφέρθηκε προηγουμένως. Συγκρίνει την πρώτη εγγραφή που διαβάζεται με την επόμενη και η επόμενη πάντα καθορίζεται πιά σημαντική. Επιστρέφει στην ουσία 1 λόγω αυτής της ανίσωσης-σύγκρισης.

GUI:

Γραφικό περιβάλλον διεπαφής χρήστη. Πεδία της κλάσης αυτής: `ArrayList` για τις ενεργές και τις παλιές καλλιέργειες οι οποίες αναφέρονται στο πρόγραμμα ως `kalliergies_pragmatikes`, `ArrayList` για τις πρότυπες καλλιέργειες οι οποίες αναφέρονται στο πρόγραμμα ως `kalliergies_fadastikes`, ένα `ArrayList` για το `configRecord` το οποίο αναφέρεται στο πρόγραμμα ως `configRecords`, ένα `ArrayList` για το κάθε `panel` των καλλιεργιών, στο πρόγραμμα γραμμένο `jPanel_list`, ένα `ArrayList` για αποθήκευση των χρωμάτων των δράσεων το `action_color_array_list` και τελευταίο είναι το `ArrayList<String> actions` για αποθήκευση των δράσεων. Ορίζεται ένα `Map` με το οποίο αντιστοιχίζεται η κάθε δράση και παρέμβαση με την αντίστοιχη εικόνα τους, το `image_map` και ένα `HashMap` για την αντιστοίχιση των δράσεων όπως διαβάζονται απ' το `config.ini` με τα χρώματά τους σε κλίμακα RGB. Στην άλλη ορίζονται δύο πίνακες αντικειμένων της κλάσης `JButton`, για τις ανάγκες της εφαρμογής. Ορίζονται δύο πίνακες `rectangle_array` και `dates_array` όπου στον πρώτο θα αποθηκευτούν τα `rectangles` και στο δεύτερο οι ημερομηνίες για τα `rectangles`. Τα `panels p`, `panelPragmatikes` και `panelFadastikes` είναι αντικείμενα της κλάσης `JPanel` και έχουν δηλωθεί με σκοπό το `p` να είναι το κεντρικό `panel` στο οποίο όλα τα άλλα `componets` και `panels` τοποθετούνται, το `panelPragmatikes` αυτό στο οποίο όλα τα `panel` των ενεργών και παλιών ενεργειών τοποθετούνται ενώ το `panelFadastikes` αυτό στο οποίο όλα τα `panel` των προτύπων καλλιεργιών τοποθετούνται. Ακόμη, δηλώνεται ένα αντικείμενο της κλάσης `JScrollPane` και ένα αλφαριθμητικό `msg`. Τέλος, τα `scaling_factor = 32`, `MIN_SCALING_FACTOR = 32` και `SCALING_FACTOR_STEP = 4` είναι ακέραιες μεταβλητές που χρησιμοποιούνται για εναλλαγή μεταξύ μεγένθυσης και σμίκρυνσης.

Μέθοδοι κλάσης GUI:

`private void setRectangleCreationImportance(Rectangle_creation Rc, int i, int record_index):` Για όλα τα `Rectangles` που αποτελούν το γραφικό περιβάλλον ορίζεται ένα `importance` το οποίο το περνάμε στην κλάση `Rectangle_creation`.

`private void saveData():` Η μέθοδος αυτή αφορά τη λειτουργία αποθήκευση. Όταν αποθηκεύεται η εφαρμογή αν έχει δημιουργηθεί νέα καλλιέργεια η μέθοδος αυτή τα αποθηκεύει στα αρχεία. Οτιδήποτε άλλες αλλαγές συμβούν στο πρόγραμμα η μέθοδος αυτή αναλαμβάνει την αποθήκευσή

τους επιφέροντας τις αντίστοιχες αλλαγές.

`private void arrange_image_map()`: Σε αυτή τη μέθοδο γίνεται διασύνδεση εικόνων και χρωμάτων με χρήση `map`. Τα χρώματα είναι όλα σε κλίμακα RGB, άρα μιλάμε για RGB χρώματα και καλύπτεται όλο το φάσμα χρωμάτων. Στη μέθοδο ελέγχεται αν η κάθε εγγραφή του `config.ini` είναι τύπου δράσης ή παρέμβασης. Αν είναι δράσης συνδέεται η αντίστοιχη εικόνα και χρώμα φόντου ενώ για παρέμβαση συνδέεται μόνο εικόνα. Βέβαια, σε αυτή τη μέθοδο καθορίζονται και άλλα χρώματα που χρησιμοποιούνται στο GUI, όπως για παράδειγμα χρώμα για τις ενεργές καλλιέργειες.

`void setMessage(String m)`: Ορίζεται ένα `msg` για τις λειτουργίες Εισαγωγή νέου συμβάντος, Δημιουργία νέας καλλιέργειας, το οποίο αντιπροσωπεύει το νέο αντικείμενο που επιλέχθηκε να προστεθεί στο GUI. Το `msg` το οποίο θα αντιπροσωπεύει τη δημιουργία νέας καλλιέργειας, το περνάμε πρώτα στο στη μέθοδο `startNewKalliergeia` και έπειτα στο γραφικό περιβάλλον ενώ το `msg` για εισαγωγή νέου συμβάντος, απ'ευθείας στην μέθοδο `GUI`. Επίσης αναλόγως το εικονίδιο που προστίθεται, αν είναι δράση ή παρέμβαση αλλάζουν τα χρώματα της μπάρας ή όχι αντίστοιχα.

`void startNewKalliergeia(String msg)`: Αφορά τη Δημιουργία νέας καλλιέργειας από το μενού επιλογών. Σε αυτή τη μέθοδο διαβάζεται η εικόνα νέας καλλιέργειας, η αντίστοιχη περιγραφή και ημερομηνία και τα δεδομένα αυτά στέλνονται στο γραφικό περιβάλλον. Σημαντικό ρόλο κατέχει το αρχείο `info.txt` από το οποίο διασυνδέεται η νέα εικόνα καλλιέργειας ή μία από τις παλιές και ο φάκελος `user` στον οποίο θα αποθηκευτεί το νέο αρχείο κειμένου με τα νέα δεδομένα. Τελικά δημιουργείται νέο πάνελ και καλείται η `repaint()` για εμφάνιση της νέας γραμμής καλλιέργειας στην εφαρμογή.

`private void repaint_panels()`: Κάθε φορά που εκτελείται η λειτουργία Μεγένθυνη ή Σμίκρυνση σβήνεται όλο το `panel` που περιέχει όλα τα γραφικά και ξανασχεδιάζεται με τα καινούργια δεδομένα. Αυτό το κάνει η μέθοδος `changePanel()`. Έπειτα, χρειάζεται αυτή η μέθοδος για να τα εμφανίσει. Επίσης μέσω του `Viewport()` της μεθόδου αυτής διατηρούμε την ίδια θέση του `scrollBar` πριν και μετά την αλλαγή κλίμακας.

`private void changePanel()`: Σε αυτή τη μέθοδο σβήνεται το πάνελ που περιέχει όλα τα γραφικά στοιχεία. Στην κλάση `Rectangle_creation` ορίστηκε ένα `importance`, από το οποίο δώθηκε ένα `importance` σε όλα τα `Rectangles`. Επίσης, στην κλάση `Agros` είχε δημιουργηθεί ένα `id` και στη συνέχεια δώθηκε ένα `importance` σε όλες τις εγγραφές του `config.ini` για σύγκριση. Όταν λοιπόν γίνεται σμίκρυνση πραγματοποιείται σύγκριση ανά τρία `Rectangles` για να υπερισχύσει το ένα έναντι των άλλων και να εμφανιστεί. Επειδή τα `Rectangles`, περιέχουν εικονίδια και τα εικονίδια ανήκουν στις εγγραφές του `config.ini` στις οποίες δώθηκε ένα `importance`, σύμφωνα με τη σύγκριση των εικόνων καθορίζεται πίο εκ των τριών `Rectangles` θα εμφανίζεται κάθε φορά κατά τη Σμίκρυνση.

`private void make_kalliergeies_pragmatikes_panel()`: Αυτή η μέθοδος αφορά τη δημιουργία ενεργών, παλιών καθώς και των προτύπων καλλιεργείων στο GUI. Σχεδιάζεται μέσω αυτής της μεθόδου η μπάρα χρόνου. Για να τα σχεδιαστούν όλα αυτά ορίζεται ένα βασικό `panel`. Στη συνέχεια διαβάζονται όλα τα δεδομένα των καλλιεργείων και προστίθενται σε νέα `Rectangles`. Έτσι

πολλά Rectangles σχεδιάζουν μία μπάρα εξέλιξης για κάθε καλλιέργεια, η κάθε μία να αποτελείται από πολλαπλά Rectangles. Η κάθε μπάρα της κάθε καλλιέργειας έπρεπε να προστεθεί στο δικό της panel. Έτσι κάθε μπάρα καλλιέργειας αποτελεί και δικό της panel. Σχεδιάζεται ακόμη ένα panel που αντικατοπτρίζει τη μπάρα χρόνου. Κάθε καλλιέργεια ορίζεται με τέτοια σειρά που να ξεκινά από το δικό της χρόνο. Τέλος, όλες οι γραμμές χρωμάτων λαμβάνουν υπόψη μόνο τα εικονίδια δράσης για αλλαγή χρώματος.

Εσωτερική κλάση του GUI MouseEvent1 implements MouseListener:

Αυτή η κλάση αποτελεί τον listener των λειτουργιών της εφαρμογής. Όλες οι λειτουργίες της εφαρμογής πρέπει να της ακούσει η ίδια η εφαρμογή για να μπορέσει να γίνει η λειτουργία, ή καλύτερα ένα γεγονός. Έτσι τις λειτουργίες Δημιουργία νέας καλλιέργειας, Αποθήκευση, Μεγέθυνση, Σμίκρυνση, Βοήθεια, αποθήκευση κατά την έξοδο αν επιλεγθεί καθώς και εισαγωγή νέου συμβάντος ακούγονται από την εφαρμογή μέσω αυτής της κλάσης. Ένας ActionCommand για κάθε λειτουργία υλοποιείται στη μέθοδο GUI και έπειτα το Command αποθηκεύεται στην κλάση αυτή.

`public GUI( ArrayList<Kalliergeia> kalliergies_pragmatikes, ArrayList<Kalliergeia> kalliergies_fadastikes, ArrayList<ConfigRecord> configRecords):` Η μέθοδος αυτή αναλύεται στη Λειτουργία διεπαφής χρήστη στο τέλος αυτού του κεφαλαίου, γιατί περιγράφει πλήρως τη λειτουργία της διεπαφής.

Kalliergeia:

Η κλάση αυτή χρησιμεύει για την αποθήκευση στοιχείων κάποιας καλλιέργειας στο info.txt. Τα πεδία που περιέχει είναι: Ένα αλφαριθμητικό fileName που αντιστοιχεί στην εικόνα μιας καλλιέργειας. Επίσης περιέχει ένα αντικείμενο Kalliergeia_type της KalliergeiaType με δυνατές τιμές: USER, RECOMMENDED, ένα αλφαριθμητικό type που θα κρατά το όνομα της καλλιέργειας, ένα ArrayList που κρατά τα Records των καλλιεργειών και ένα αντικείμενο της κλάσης ImageIcon, με σκοπό τη διασύνδεση κάθε ονόματος καλλιέργειας με την αντίστοιχη αντιπροσωπευτική εικόνα της.

Μέθοδοι κλάσης Kalliergeia:

`public boolean isActive():` Διαπερνάμε τις εγγραφές και είναι ενεργή καλλιέργεια επιστρέφει true αλλιώς επιστρέφει false.

`public void sortRecords():` Καλείται από την κλάση Agros και τη μέθοδο ArrayList<Kalliergeia> processFileContents(ArrayList<String> contents). Ταξινομούνται με αυτό τον τρόπο τα αρχεία με βάση την ημερομηνία. Πιο συγκεκριμένα αυτό το αναλαμβάνει η κλάση KalliergeiaRecordComparator.

`public int getDurationFromStartForRecord(int i):` Στο γραφικό περιβάλλον της διεπαφής οι διενέργειες των καλλιεργειών αναπαριστώνται με μία οριζόντια γραφική γραμμή στην οποία οι εικόνες δράσεων αλλάζουν την φάση της καλλιέργειας, δηλαδή χρώμα μπάρας ενώ των παρεμβάσεων όχι. Η οριζόντια γραμμή διαπερνάται με βήμα ημέρας. Για να αλλάξει χρώμα η μπάρα πρέπει να υπολογιστεί η μεταξύ τους ημερομηνία και να την επιστρέψει. Αυτό κάνει η συγκεκριμένη συ-

νάρτηση.

`public int getTotalDaysDuration()`: Σε αυτή τη μέθοδο υπολογίζονται οι συνολικές μέρες της μπάρας εξέλιξης της κάθε καλλιέργειας. Για να το κάνει αυτό εκτελείται μια αφαίρεση από την τελευταία μείον την αρχική ημερομηνία και τον αριθμό αυτόν τον επιστρέφει.

`public Kalliergeia(KalliergeiaType Kalliergeia_type, String fileName, String type)`: Μία πολύ σημαντική μέθοδος του προγράμματος. Διαβάζει ένα αρχείο κειμένου το οποίο ονομάζεται `info.txt` και βρίσκεται στο φάκελο `plants`, μέσα στο φάκελο `icons` του μητρικού φακέλου της εφαρμογής. Μέσω του αρχείου αυτού συνδέεται κάθε καλλιέργεια με μία εικόνα που την καθορίζει.

`public ImageIcon getIcon()`: Επιστρέφει την εικόνα της καλλιέργειας σύμφωνα με αυτές του αρχείου `info.txt`.

`public void addRecord(Record record)`: Όταν δημιουργείται νέα καλλιέργεια μέσω του γραφικού περιβάλλοντος, όνομα της νέας καλλιέργειας και αντίστοιχη εικόνα προστίθενται στο `info.txt`. Έτσι το `info.txt` το διακατέχει δυναμικότητα, όπως και του `config.ini`.

`public String getType()`: Επιστρέφει τον τύπο καλλιέργειας, όπως για παράδειγμα ντομάτα.

`public String getFileName()`: Επιστρέφει το όνομα της εικόνας καλλιέργειας.

`public KalliergeiaType getKalliergeia_type()`: Επιστρέφει τύπο καλλιέργειας, `USER` ή `RECOMMENDED`.

`public Record getRecord(int i)`: Επιστρέφει εγγραφή. Κάθε εγγραφή αντιστοιχίζεται σε μία σειρά του `info.txt`.

`public int getRecordsNumber()`: Επιστρέφει τον αριθμό των εγγραφών του `info.txt`.

KalliergeiaRecordComparator:

Η κλάση αυτή χρησιμεύει για την ταξινόμηση των εγγραφών των ενεργών και παλιών καλλιεργειών οι οποίες βρίσκονται στο φάκελο `user`.

Μέθοδοι κλάσης KalliergeiaRecordComparator: `public class KalliergeiaRecordComparator implements Comparator<Record>`: Για την ταξινόμηση, περνάμε τα δεδομένα του `ArrayList` που περιέχει τα `Record`. Για να ταξινομήσει τα αρχεία η συνάρτηση συγκρίνει την ημερομηνία της κάθε νέας εγγραφής που είναι να προστεθεί στις ήδη υπάρχουσες και αυτή τη συγκρίνει με την τελευταία εγγραφή. Επιστρέφει μία ακέραια τιμή που το πρόσημό της δείχνει το αποτέλεσμα της σύγκρισης των δύο εγγραφών.

KalliergeiaType:

Η κλάση αυτή χρησιμοποιείται για ορισμό του τύπου καλλιέργειας. Τα πεδία που περιέχει είναι: Μία απαρίθμηση `USER` για καθορισμό των ενεργών και παλαιών καλλιεργειών και μία απαρίθμηση `RECOMMENDED` για καθορισμό των προτύπων καλλιεργειών.

NewKalliergeiaPanel:

Η κλάση αυτή αφορά το panel δημιουργίας νέας καλλιέργειας. Τα πεδία που περιέχει είναι: Ένα αντικείμενο της κλάσης `TextField` για το text εισαγωγής ονόματος νέας καλλιέργειας, αντικείμενα της κλάσης `JLabel` για εμφάνιση των labels του παραθύρου, labels όπως Εισάγετε όνομα νέας καλλιέργειας. Ορίζεται ένα αντικείμενο της κλάσης `JDateChooser` για το ημερολόγιο επιλογής ημέρας, ένα αλφαριθμητικό `choice` για την καθορισμένη επιλογή, ένα αλφαριθμητικό `msg` το οποίο αφορά την εισαγωγή της κάθε επιλογής `choice` στο `gui`, ένα αντικείμενο του `GUI` γιατί εκεί θα προστίθεται το κάθε `msg`. Τέλος δηλώνεται ένα αντικείμενο της κλάσης `JButton` για εισαγωγή νέας εικόνας καλλιέργειας και ένα αλφαριθμητικό `iconFile` για την επιλογή της νέας εικόνας η οποία στη συνέχεια μέσω του `JButton` θα εισαχθεί.

Μέθοδοι κλάσης NewKalliergeiaPanel:

`public NewKalliergeiaPanel(GUI gui):` Κρατάμε μια λίστα από τις καλλιέργειες. Ανοίγεται αρχείο από ήδη υπάρχουσες καλλιέργειες, αντιστοιχίζεται η κάθε μία με ένα `JRadioButton` και ένα `JLabel` και έπειτα τοποθετούνται σε ένα `JPanel`, το `ListPane`. Σε κάθε `JRadioButton` αντιστοιχίζεται μία `ActionCommand` για να μπορεί ο χρήστης να κάνει επιλογή καλλιέργειας από τις ήδη υπάρχουσες. Σε ένα άλλο panel, το `datePanel` τοποθετείται ένας `JDateChooser` για επιλογή ημερομηνίας σε γραφικό επίπεδο. Εκτός αυτού, ο `JDateChooser` προτείνει στο χρήστη πάντοτε την τότε σημερινή ημερομηνία. Το `datePanel` τοποθετείται στο `ListPanel`. Ένα `JButton`, Τέλος χρησιμοποιείται για να εισαχθούν τα νέα δεδομένα στο `gui`.

`private String OpenFileChooser():` Ο χρήστης έχει την επιλογή να προσθέσει νέα καλλιέργεια με το όνομα που επιθυμεί. Για αυτό το σκοπό υπάρχει η επιλογή νέο είδος με ένα αντίστοιχο `MouseCommand`. Πατώντας νέο είδος ένα `textField` εμφανίζεται για πληκτρολόγηση του νέου ονόματος και η επιλογή εικονιδίου. Επιλογή εικονιδίου μπορεί να γίνει σε οποιονδήποτε από τους φακέλους του υπολογιστή. Η νέα καλλιέργεια που επιλέγει ο χρήστης προστίθεται στην ήδη υπάρχουσα λίστα και συνεπώς ως νέα καλλιέργεια στο παράθυρο για εισαγωγή. Η μέθοδος αυτή επιστρέφει το νέο αρχείο εικόνας που επιλέγεται.

`public void actionPerformed(ActionEvent e):` Αναφέρθηκε το `MouseCommand` προηγουμένως. Σε κάθε `MouseCommand` αντιστοιχίζεται κάποιο `ActionEvent`. Έτσι, για την επιλογή νέου είδους, όπως λόγου χάρη και την αποτύπωση αυτού στο `gui`. Τον σκοπό αυτό τον επιτελεί ο κώδικας της μεθόδου αυτής.

PopupInfoGetter:

Η κλάση αυτή δημιουργήθηκε για την εμφάνιση του παραθύρου εισαγωγής νέου συμβάντος. Τα πεδία που περιέχει είναι: Ένα αντικείμενο της κλάσης `JTextArea` για εισαγωγή σχολίου, ένα αλφαριθμητικό `choice` για την καθορισμένη επιλογή, ένα αλφαριθμητικό `msg` το οποίο αφορά την εισαγωγή της κάθε επιλογής `choice` στο `gui`, ένα αντικείμενο του `GUI` γιατί εκεί θα προστίθεται το κάθε `msg`. Στην κλάση αυτή χρησιμοποιήθηκε `ArrayList`, για αυτόματη αποθήκευση των δεδομένων του αρχείου `config.ini`. Ο πίνακας `images` που περιέχει αντικείμενα τύπου `ImageIcon` και ο

πίνακας `description` που περιέχει αντικείμενα αλφαριθμητικού τύπου ορίστηκαν στην κλάση αυτή με αυτοσκοπό εικόνα και περιγραφή αντίστοιχης ενέργειας να τοποθετείται στο ίδιο panel. Κάθε εικόνα και αντίστοιχη περιγραφή θα βρίσκονται στο ίδιο panel. Τέλος δημιουργήθηκε αντικείμενο της κλάσης `JComboBox` για προσθήκη των panels που δημιουργήθηκαν και ο πίνακας `intArray` που περιέχει μεταβλητές ακεραίου τύπου χρησιμοποιήθηκε για το μέγεθος του `combobox`.

Μέθοδοι κλάσης `PopupInfoGetter`:

`public PopupInfoGetter(GUI gui, String m, boolean showFinish, ArrayList<ConfigRecord> configRecords)`: Αναφέρθηκε πιο πάνω ότι στην κλάση αυτή χρησιμοποιήθηκε `ArrayList` για αυτόματη αποθήκευση των δεδομένων του αρχείου `config.ini` και ότι εικόνα και περιγραφή αντίστοιχης ενέργειας τοποθετείται στο ίδιο panel. Κάθε panel βρίσκεται σε ένα `ComboBox` με ένα `intArray` να ορίζει το μέγεθός του. Όλα αυτά γίνονται σε αυτή τη μέθοδο.

`public void actionPerformed(ActionEvent e)`: Με σκοπό το πρόγραμμα να καταλάβει την κάθε επιλογή και να τοποθετήσει αυτή στο `gui`, με κάποιο τρόπο το πρόγραμμα πρέπει να την καταλαβαίνει. Αυτό το κάνει η μέθοδος αυτή.

Εμφωλευμένη κλάση `ComboBoxRenderer` της `PopupInfoGetter`

`class ComboBoxRenderer extends JLabel implements ListCellRenderer`: Ο μόνος τρόπος για σύνδεση εικόνας και περιγραφής του πίνακα `images` και `description` σε ένα `ComboBox`. Κάθε αντικείμενο στη λίστα του `ComboBox` αντιστοιχίζεται σε έναν ακέραιο αριθμό.

Record:

Η κλάση αυτή παρέχει αντικείμενα που αντιπροσωπεύουν κάθε εγγραφή στα αρχεία του καταλόγου `user` και `recommended`. Η κλάση περιέχει τα πεδία: Ένα αλφαριθμητικό `description` που περιγράφει το όνομα της εγγραφής, ένα αντικείμενο `type`, της `RecordType` με δυνατές τιμές: `ACTION` και `INTERFERE`, ένα αντικείμενο `date` τύπου `Date` για να γνωρίζουμε πότε έγινε η συγκεκριμένη ενέργεια ή επίδραση. Επίσης δύο αντικείμενα της κλάσης `SimpleDateFormat`, για καθορισμό της μορφής της ημερομηνίας και μία μεταβλητή `recommended_day` που χρησιμοποιείται για αποτύπωση της ημερομηνίας των προτύπων καλλιεργειών. Για την ημερομηνία, έχοντας δύο αντικείμενα όπου το πρώτο, το `sdf` όπως ονομάστηκε χρησιμοποιείται για να εμφανίζεται ημερομηνίες της μορφής `Ημέρα/Μήνας/Χρόνος`. Το δεύτερο το `shdf` όπως ονομάστηκε, χρησιμοποιείται για να εμφανίζεται ημερομηνίες της μορφής `Ημέρα/Μήνας`, χωρίς δηλαδή να εμφανίζει έτη.

Μέθοδοι κλάσης `Record`:

`public Record(String date, RecordType type, String description)`: Αφορά την κάθε εγγραφή καλλιεργείας. Κάθε καλλιέργεια τη χαρακτηρίζει πλήρως το όνομά της, ο τύπος της για το αν είναι `USER` ή `RECOMMENDED` και ένα `description` στο οποίο θα περνάει κάποια εικόνα που θα αφορά

ACTION ή INTERFERE. Την εικόνα αυτή θα την περνάμε κυρίως από το gui όταν επρόκειτο να εισαγάγουμε νέο συμβάν.

public int getRecommendedDay(): Επιστροφή recommended_day για τις πρότυπες καλλιέργειες.

public String getDescription(): Επιστροφή του description που αναφέρθηκε παραπάνω.

public RecordType getType(): Ο τύπος της καλλιέργειας που επιστρέφεται αφορά το όνομα της καλλιέργειας.

public String getTypeAsString(): Επιστρέφεται ο τύπος της καλλιέργειας, ACTION ή INTERFERE, σε μορφή αλφαριθμητικού.

public Date getDate(): Επιστροφή της καθορισμένης ημερομηνίας.

public String getDateString(): Η μέθοδος αυτή αποσκοπεί στη μορφή της ημερομηνίας όπως έχει οριστεί. Δηλαδή με μορφή dd/MM/yyyy. Ημέρα/ Μήνας/Χρόνια.

public static String getShortDateString(Date d): Η μέθοδος αυτή αποσκοπεί στη μορφή της ημερομηνίας με μορφή dd/MM. Ημέρα/ Μήνας. Αυτό γιατί η μπάρα χρόνου εμφανίζεται τελικώς σε αυτή τη μορφοποίηση. Δηλαδή Ημέρα/ Μήνας και όχι χρόνια.

public static String getOtherDateString(Date d): Για επιστροφή ημερομηνίας με ακέραια μορφή.

RecordType:

Η κλάση αυτή χρησιμεύει για τον ορισμό του τύπου ενεργειών. Τα πεδία που περιέχει είναι: Μία απαρίθμηση ACTION, για καθορισμό των δράσεων οι οποίες αλλάζουν τη φάση της καλλιέργειας και μία απαρίθμηση INTERFERE για καθορισμό των παρεμβάσεων οι οποίες δεν αλλάζουν τη φάση της καλλιέργειας.

Rectangle_creation:

Κύριος σκοπός της εν λόγω κλάσης, είναι η δημιουργία των Rectangles που χρησιμοποιήθηκαν για την απεικόνιση των καλλιεργειών και του ανενεργού τμήματος του επιπέδου. Τα πεδία που περιέχει είναι: Μία μεταβλητή kalliergeia_id ακεραίου τύπου στην οποία θα αποθηκεύεται η σειρά της κάθε καλλιέργειας και αφού σε κάθε σειρά θα αποθηκεύεται διαφορετική καλλιέργεια, η μεταβλητή αυτή θα καθορίζει ποιά καλλιέργεια είναι ποιά. Η μεταβλητή imera ακεραίου τύπου ορίζεται ως πεδίο στην κλάση αυτή για αποθήκευση της ημέρας που θα εμφανίζεται το κάθε rectangle. Ένα recordImportance ορίζεται σε αυτή την κλάση και αρχικοποιείται με -1 στον constructor. Παρακάτω στις μεθόδους εξηγείται τι συμβαίνει.

Μέθοδοι κλάσης Rectangle_creation:

`public Rectangle_creation(int id, int d):` Constructor των Rectangle.

`public int get_kalliergeia_id():` Ένα id για κάθε καλλιέργεια. Τα Rectangles στο gui σχεδιάζονται με μία `for` με μεταβλητές `i` και `j`. Κάθετα το `i` και οριζόντια το `j`. Το `i` ορίζει την κάθε καλλιέργεια γιατί οι καλλιέργειες βρίσκονται κάθετες η μία με την άλλη ενώ το `j` αφορά την ημέρα που βρίσκεται το Rectangle.

`public int get_imera():` Εδώ όπως ακριβώς αναφέρθηκε από πάνω, επιστρέφεται η ημέρα που βρίσκεται το Rectangle στον πίνακα στον οποίο αυτό αναφέρεται ως `j`.

`public String toString():` Μας δίνει αλφαριθμητικό με στοιχεία την `kalliergeia_id`, `imera` και `getBackground` του Rectangle.

`public void setRecordImportance(int recordImportance):` Η μέθοδος αυτή και η επόμενη είναι πολύ σημαντική γιατί μέσω αυτών λειτουργεί η Μεγένθυνη και η Σμίκρυνση. Εδώ ορίζεται ένα `importance` για όλα τα rectangles. Λαμβάνει τα δεδομένα της από την κλάση GUI, γιατί τα Rectangles της διεπαφής μας ενδιαφέρουν να συγκρίνουμε μετέπειτα.

`public Rectangle_creation getMostImportantRectangle(Rectangle_creation previous, Rectangle_creation next):` Στην ουσία κάθε φορά που γίνεται σμίκρυνση γίνεται συνένωση τριών γειτονικών rectangles και το ένα από αυτά κυριαρχεί και εμφανίζεται. Όποιο rectangle, έχει μεγαλύτερο `importance` κυριαρχεί και εμφανίζεται κάθε φορά που γίνεται Σμίκρυνση. Έτσι μέσω του `importance` που ορίστηκε πιο πάνω στο `set`, γίνεται η σύγκριση που περιγράφηκε.

`public void paint(Graphics g):` Για να σχεδιαστεί το Rectangle στο gui χρειάζεται να καλέσουμε την `paint(Graphics g)`.

Λειτουργία διεπαφής χρήστη:

Ουσιαστικά περιγράφουμε ταυτόχρονα και τη μέθοδο GUI της κλάσης GUI.

Η κλάση GUI, είναι η κλάση της εφαρμογής στην οποία το γραφικό περιβάλλον διεπαφής χρήστη αναπτύσσεται λαμβάνοντας δεδομένα μέσω των αρχείων. Συγκεκριμένα, όταν εκκινείται η εφαρμογή διαβάζεται το `config.ini` και από κάθε εγγραφή σχηματίζεται ένα αντικείμενο της κλάσης `ConfigRecord`. Όλα τα αντικείμενα αποθηκεύονται σε ένα αντικείμενο της κλάσης `ArrayList`. Καλείται η `arrange_image_map()`, όπου καθορίζονται εικονίδια, δράσεις, παρεμβάσεις και αντιστοίχιση χρωμάτων των δράσεων. Σαρώνονται όλα τα αρχεία που βρίσκονται στον κατάλογο `user` και που περιέχουν πληροφορίες για κάθε καλλιέργεια. Οι πληροφορίες αποθηκεύονται σε ένα `ArrayList` με αντικείμενα της κλάσης `Kalliergeia`. Σαρώνονται όλα τα αρχεία που βρίσκονται στον κατάλογο `recommended` και που περιέχουν πληροφορίες για κάθε πρότυπη καλλιέργεια.

Οι πληροφορίες αποθηκεύονται σε ένα ArrayList με αντικείμενα της κλάσης Kalliergeia. Διαβάζεται επίσης ένα αρχείο info.txt και τα δεδομένα του αποθηκεύονται σε ένα ArrayList. Τα δεδομένα από τα αρχεία είναι γνωστά. Αρχικά δημιουργείται το μενού επιλογών με καθορισμένες τις επιλογές τους στη μέθοδο GUI και τα αντίστοιχα ActionCommands. Στη συνέχεια καλείται η μέθοδος make_kalliergeies_pragmatikes_panel() και σχεδιάζεται το βασικό πάνελ με όλα τα γραφικά στοιχεία. Έπειτα προστίθεται η μπάρα κύλισης στο βασικό panel. Τέλος καθορίζεται η θέση της μπάρας κύλισης και ορίζεται να εμφανίζεται στο κέντρο. Τελικά, εμφανίζεται η διεπαφή της εφαρμογής.

Κεφάλαιο 9

Παραδείγματα χρήσης

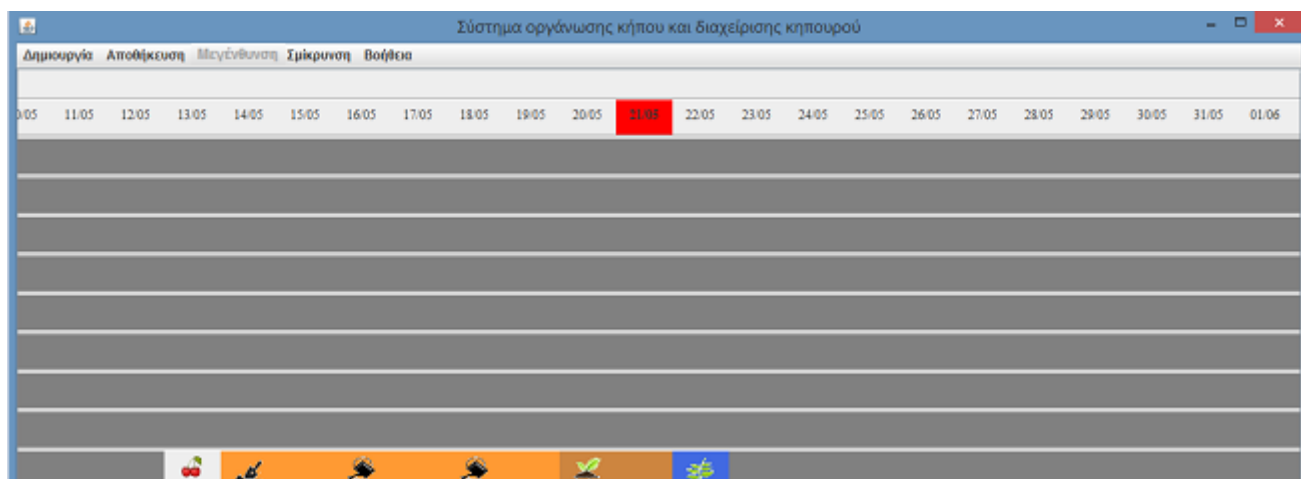
Στο κεφάλαιο αυτό παρουσιάζεται ένα παράδειγμα χρήσης της εφαρμογής, συμπεριλαμβάνοντας όλες τις λειτουργίες και δείχνονται εικόνες με παραδείγματα βημάτων για το πώς ο χρήστης μπορεί να επιτελέσει την κάθε μία από αυτές.

Εισαγωγή νέου συμβάντος

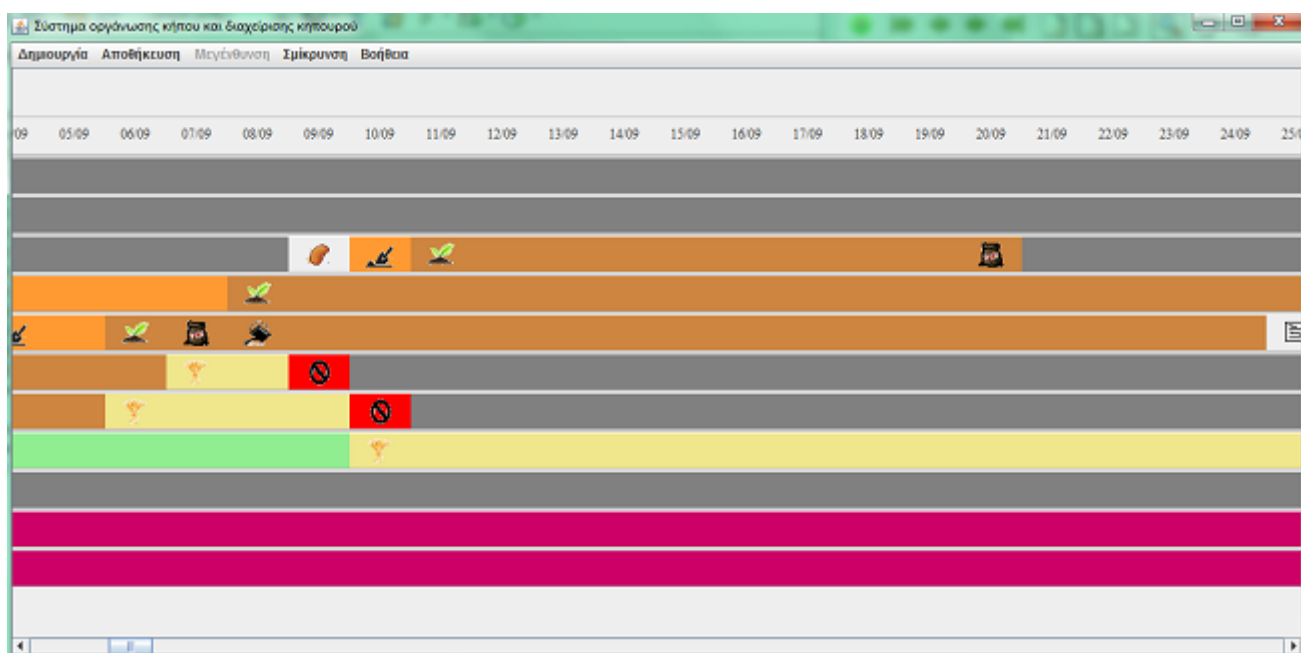
Ανοίγοντας κάποιος την εφαρμογή εμφανίζεται το κεντρικό παράθυρο, όπως φαίνεται στο σχήμα 9.1. Πατώντας στη μπάρα κύλισης και πηγαίνοντας αριστερά στους μήνες του προηγούμενου έτους παρατηρεί κάποιες καλλιέργειες που είχαν τελειώσει και κάποιες που ήταν σε εξέλιξη και δεν είχαν τελειώσει. Εικόνα 9.2. Κάνοντας κλικ σε μία ενεργή καλλιέργεια, λόγω χάρη στην καλλιέργεια φασολιάς, κάτω από την ημερομηνία 13/09 προκύπτει το παράθυρο εισαγωγής συμβάντος της εικόνας 9.3. Αρχικά αν επιλεγεί κάποια ενέργεια από αυτές που έχουμε καταχωρήσει στις δράσεις θα αλλάξει η φάση της καλλιέργειας. Επιλέγοντας λόγω χάρη μεταφύτευση στην ίδια ημερομηνία και το κουμπί τέλος, παρατηρείται ότι στις 13/09 αλλάζει η φάση της καλλιέργειας και προστίθεται αντίστοιχο εικονίδιο. Εικόνα 9.4. Πατώντας το κουμπί κλεισίματος κλείνει το παράθυρο εισαγωγής συμβάντος. Επιλέγοντας τώρα κάποιος κάποια ενέργεια τύπου παρέμβασης, δε θα αλλάξει η φάση της καλλιέργειας. Έστω για παράδειγμα στην ημερομηνία 15/09. Κάνοντας κλικ σε αυτή και επιλέγοντας από το παράθυρο εισαγωγής συμβάντος την επιλογή προσθήκη λιπάσματος, παρατηρείται ότι προστίθεται εικόνα χωρίς αλλαγή χρώματος φόντου. Εικόνα 9.5. Τέλος κάνοντας κλικ στην ημερομηνία 16/09 επιλέγοντας στο σχόλιο προκύπτει η νέα περιοχή κειμένου. Εκεί αν κάποιος αποτυπώσει κάτι ενδεικτικό και πατήσει τέλος, αντίστοιχο εικονίδιο σχολίου εμφανίζεται στη μπάρα, στην ημερομηνία που επιλέξαμε. Πηγαίνοντας πάνω στο εικονίδιο με το ποντίκι εμφανίζεται του εμφανίζεται το σχόλιο που αποτυπώθηκε.

Δημιουργία νέας καλλιέργειας

Ανοίγοντας κάποιος την εφαρμογή εμφανίζεται το κεντρικό παράθυρο της εικόνας 9.1. Επιλέγοντας τη Δημιουργία από το μενού επιλογών, στο ίδιο παράθυρο εμφανίζεται στην οθόνη το παράθυρο της εικόνας 9.7. Έστω ότι αρχικά κάποιος επιλέγει να δημιουργήσει μία καλλιέργεια από τις ήδη διαθέσιμες, για παράδειγμα καλλιέργεια καρότου. Έτσι επιλέγει την επιλογή καροτιά. Στη συνέχεια παρατηρεί την επιλογή Έναρξη. Εκεί προτείνεται η σημερινή μέρα. Έστω ότι η επιλογή αυτή παραμένει όπως είναι. Επιλέγοντας τέλος στη συνέχεια εμφανίζεται το σχήμα της εικόνας

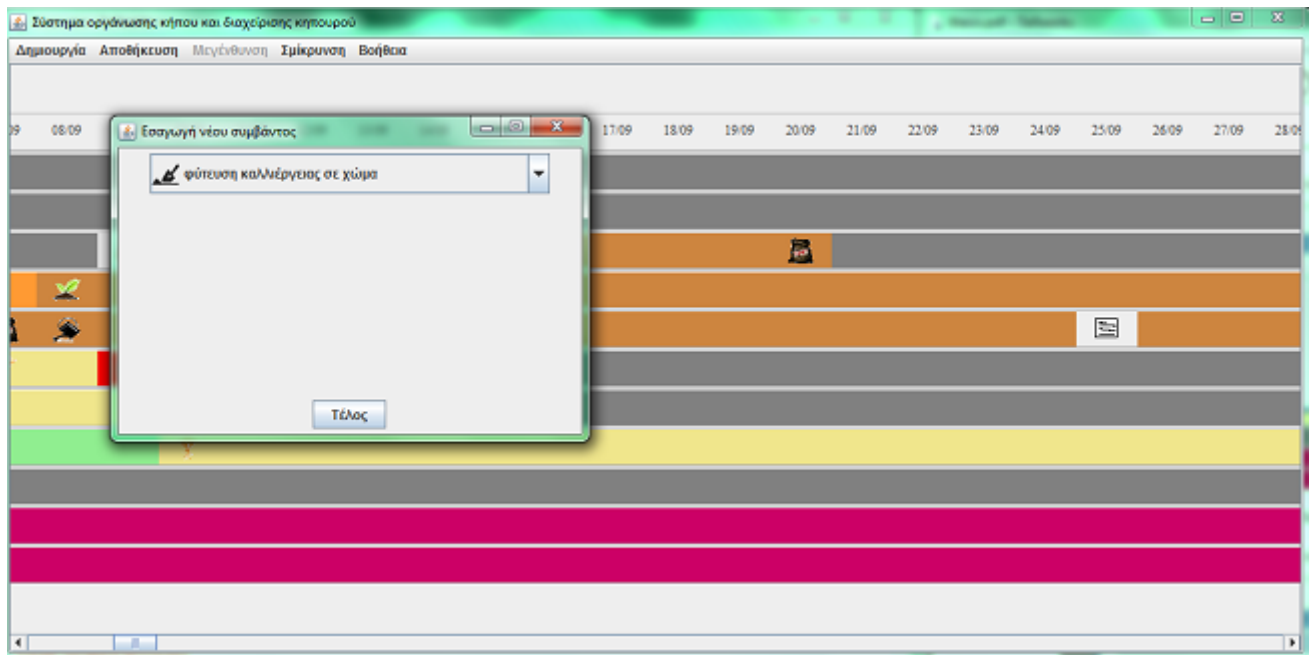


Σχήμα 9.1: Κεντρικό παράθυρο εφαρμογής

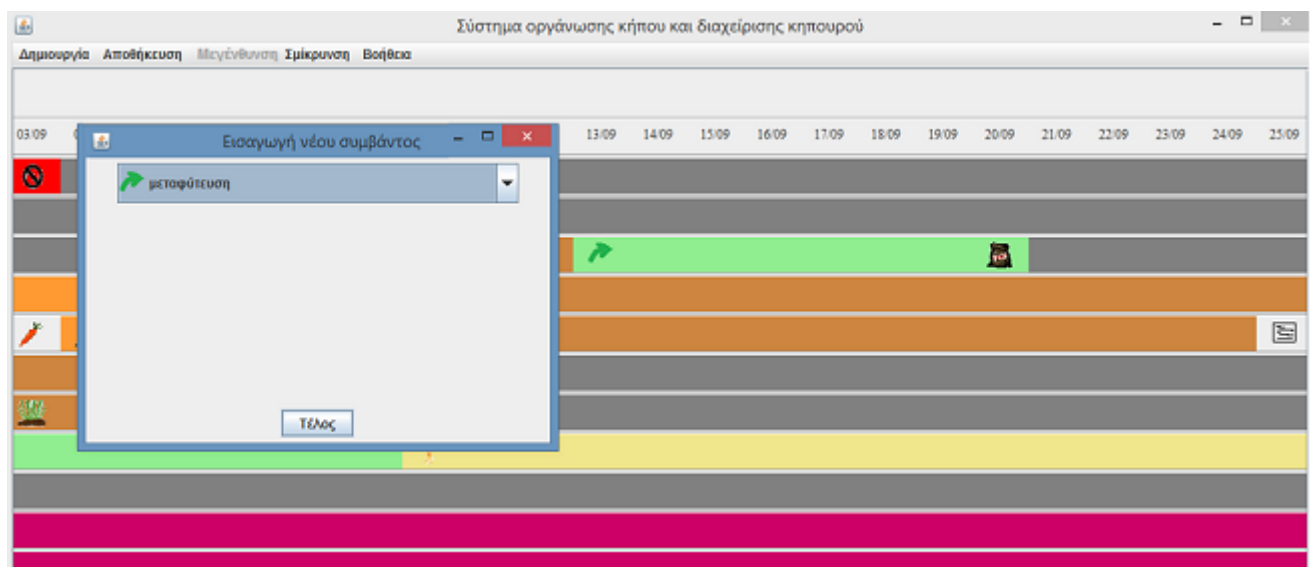


Σχήμα 9.2: Παλαιότερες καλλιέργειες όπου κάποιες βρίσκονται σε εξέλιξη και κάποιες έχουν τελειώσει

9.8. Μία νέα καλλιέργεια με νέα γραμμή εμφανίζεται με αρχή την ημερομηνία που επιλέχθηκε η τότε σημερινή. Επίσης παρατηρείται ότι εμφανίζεται αντίστοιχο εικονίδιο της καλλιέργειας και κάτω από την ημερομηνία που ζητήθηκε εμφανίζεται ως πρώτη ενέργεια η φύτευση καλλιέργειας σε χώμα. Στη συνέχεια κάποιος μπορεί να αφήσει την ενέργεια αυτή και να συνεχίσει προς τα δεξιά εισάγοντας νέες ενέργειες με το παράθυρο εισαγωγής συμβάντος, όπως φαίνεται στην εικόνα 9.9 ή ακόμη και να αλλάξει το εικονίδιο με άλλη δράση με χρήση παραθύρου εισαγωγής συμβάντος. Βέβαια, κάποιος μπορεί να εισάγει νέα καλλιέργεια σε άλλη ημερομηνία, έστω στις 24/05. Για να το κάνει αυτό στο παράθυρο εισαγωγής νέας καλλιέργειας θα επέλεγε νέο είδος. Στη συνέχεια θα επέλεγε Επιλογή εικονιδίου. Σχήμα 9.10. Πατώντας στη συνέχεια στο εικονίδιο της επιλογής Έναρξη θα προέκυπτε το σχήμα της εικόνας 9.11. Έπειτα από αυτές τις επιλογές η νέα καλλιέργεια θα εμφανιζόταν στην ημερομηνία, με την εικόνα που καθορίστηκε καθώς και με το



Σχήμα 9.3: Παράθυρο εισαγωγής συμβάντος



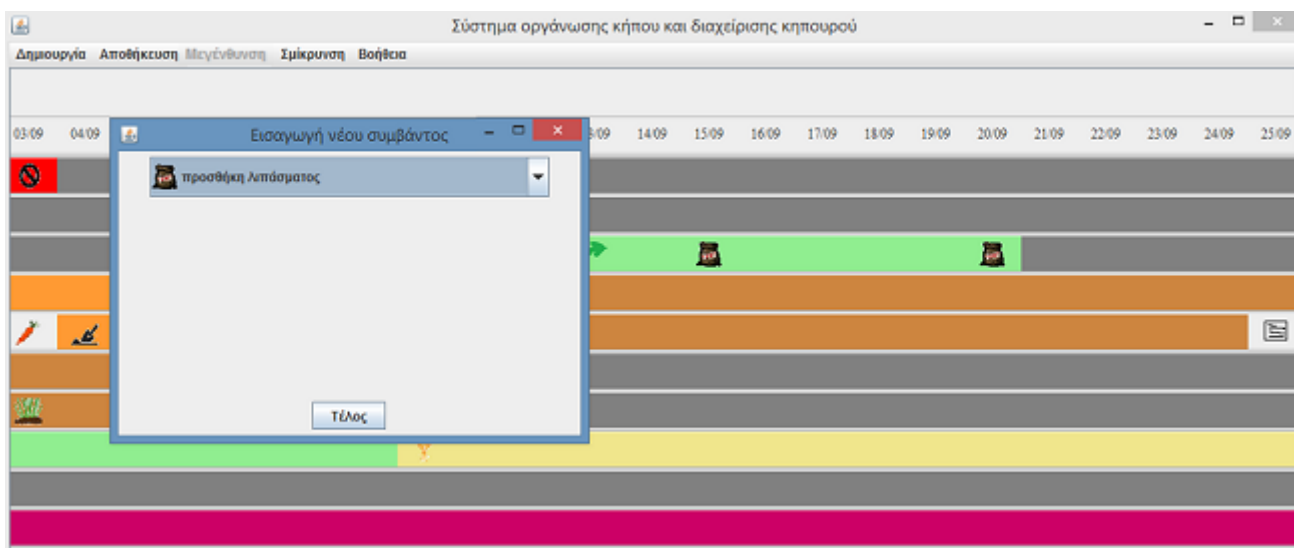
Σχήμα 9.4: Επιλογή ενέργειας που αλλάζει τη φάση

νέο όνομά της.

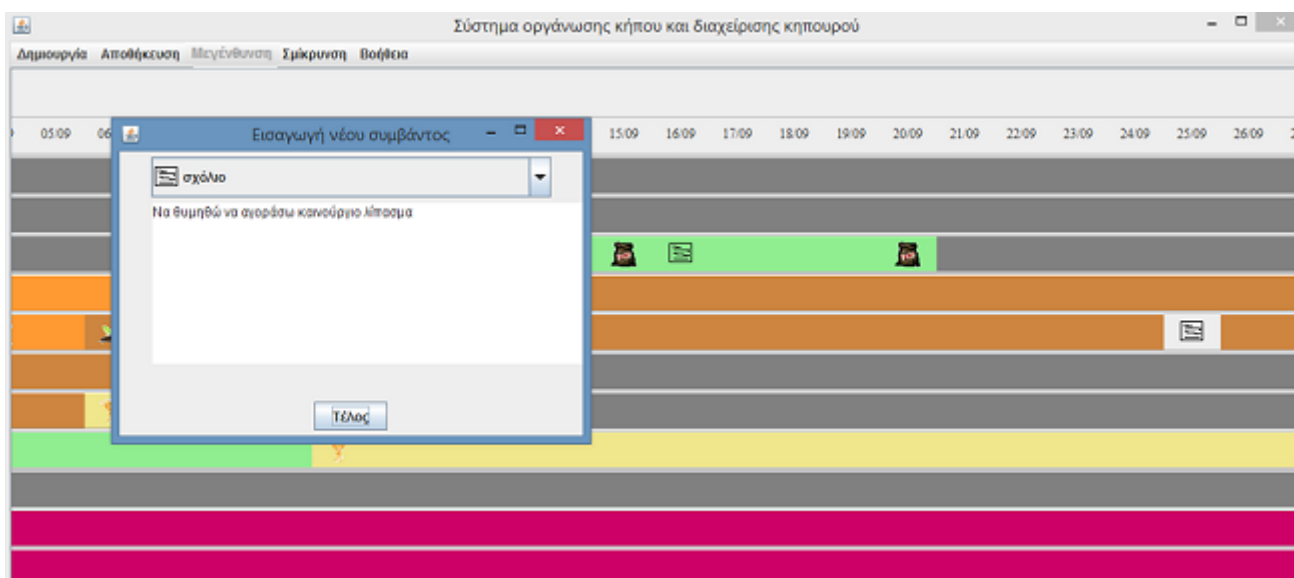
Αποθήκευση

Αποθήκευση μπορεί να γίνει είτε από το μενού επιλογών, είτε πατώντας το κουμπί έξοδος. Πατώντας κάποιος το κουμπί έξοδος εμφανίζεται το παράθυρο της εικόνας 9.12.

Βοήθεια



Σχήμα 9.5: Εισαγωγή ενέργειας που δεν αλλάζει τη φάση

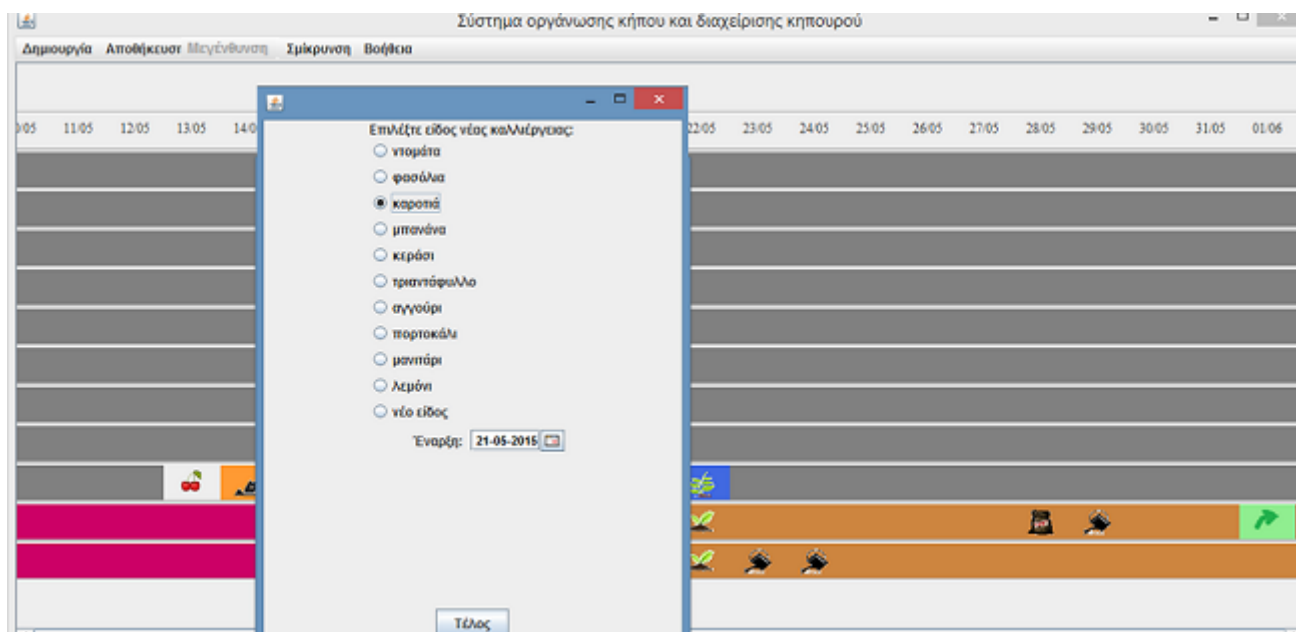


Σχήμα 9.6: Εισαγωγή σχολίου

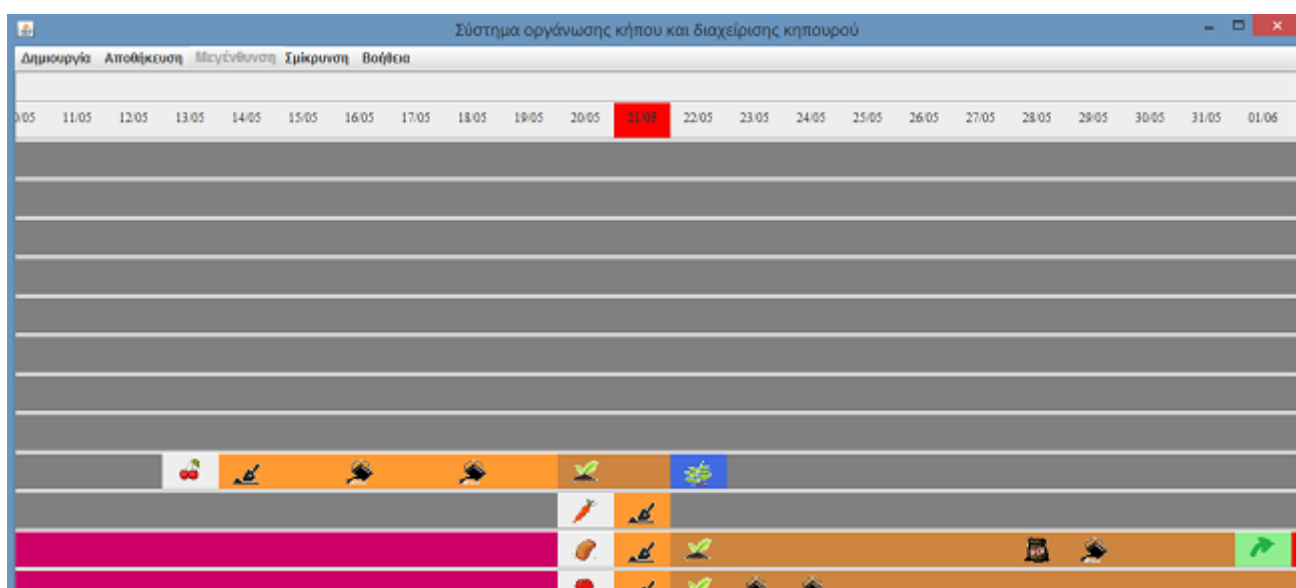
Επιλέγοντας τη λειτουργία Βοήθεια από το μενού επιλογών, εμφανίζεται το παράθυρο της εικόνας 9.13, καθώς και οι αντίστοιχες πληροφορίες σε αυτό.

Σμίκρυνση

Επιλέγοντας κάποιος Σμίκρυνση από το μενού επιλογών, έχει τη δυνατότητα να παρατηρεί μικρότερα επίπεδα λεπτομέρειας. Όσο περισσότερες φορές επιλέγει κάποιος Σμίκρυνση τόσο μικρότερα επίπεδα λεπτομέρειας θα παρατηρεί. Για παράδειγμα, επιλέγοντας Σμίκρυνση από τα αριστερά του παραθύρου της εφαρμογής μία φορά, εμφανίζεται το σχήμα της εικόνας 9.14. Πατώντας και δεύτερη φορά εμφανίζεται το σχήμα της εικόνας 9.15, ενώ πατώντας πολλές φορές μπορεί να φτάσει σε επίπεδο της εικόνας 9.16.



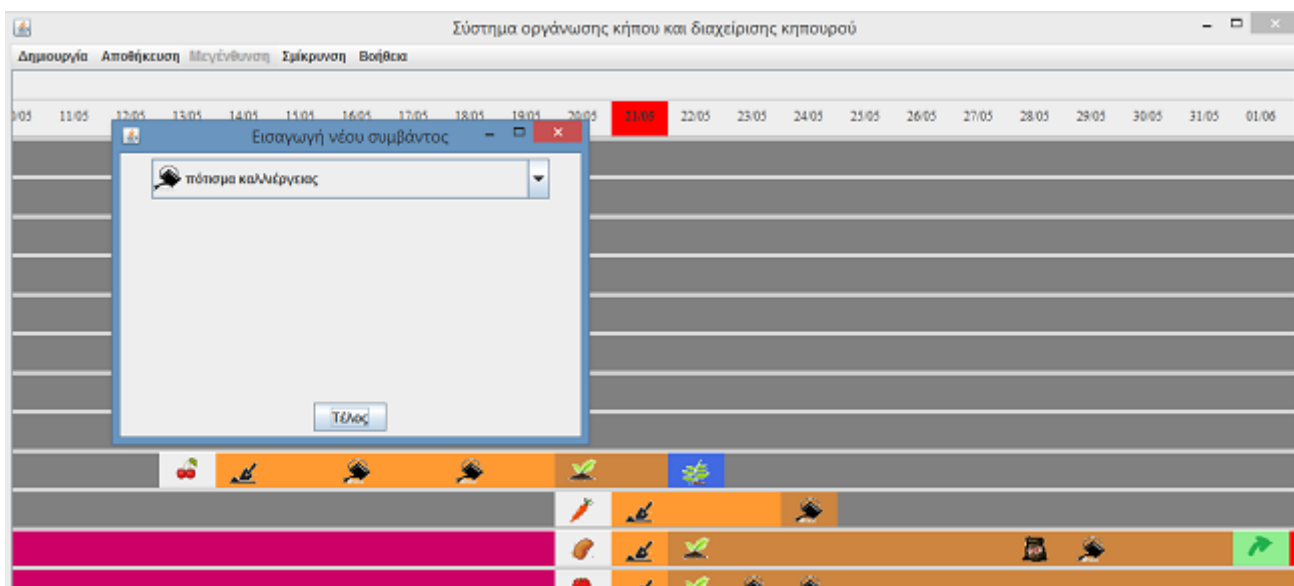
Σχήμα 9.7: Επιλογή νέας καλλιέργειας



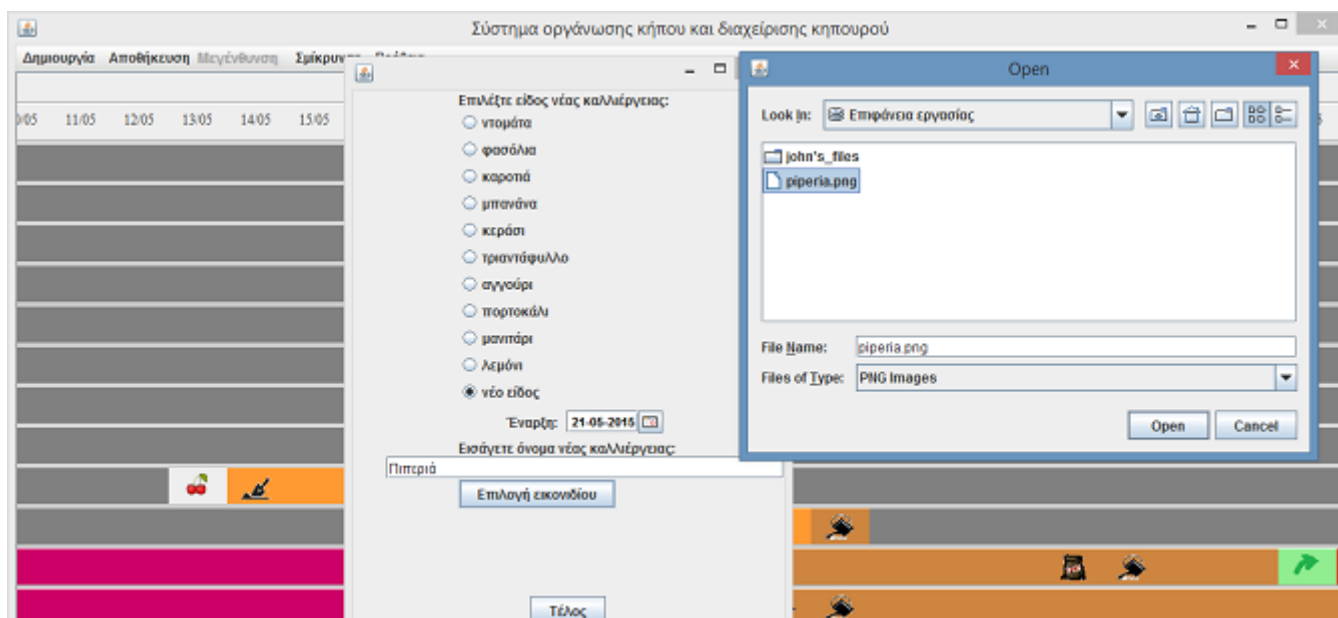
Σχήμα 9.8: Καλλιέργεια καροτιάς

Μεγένθυση

Αρχικά η επιλογή Μεγένθυση του μενού είναι γκριζαρισμένη. Έστω ότι κάποιος κάνει Σμίκρυνση μία φορά στην ίδια περιοχή της εφαρμογής όπως και προηγουμένως. Στη συνέχεια η επιλογή Μεγένθυση δεν είναι πιά γκριζαρισμένη και μπορεί να επιλεγθεί. Εικόνα 9.14. Έχοντας επιλεγθεί, η εφαρμογή τώρα θα επιστρέψει στο αρχικό της μέγεθος. Εικόνα 9.17. Αν κάποιος τώρα επιλέξει παραπάνω από μία φορές Σμίκρυνση, για παράδειγμα δύο φορές όπως φαίνεται και στο σχήμα 9. 15 και ο χρήστης επιλέξει Μεγένθυση μία φορά θα επιστρέψει σε μεγαλύτερο επίπεδο λεπτομέρειας όπως φαίνεται στο σχήμα της εικόνας 9.1.



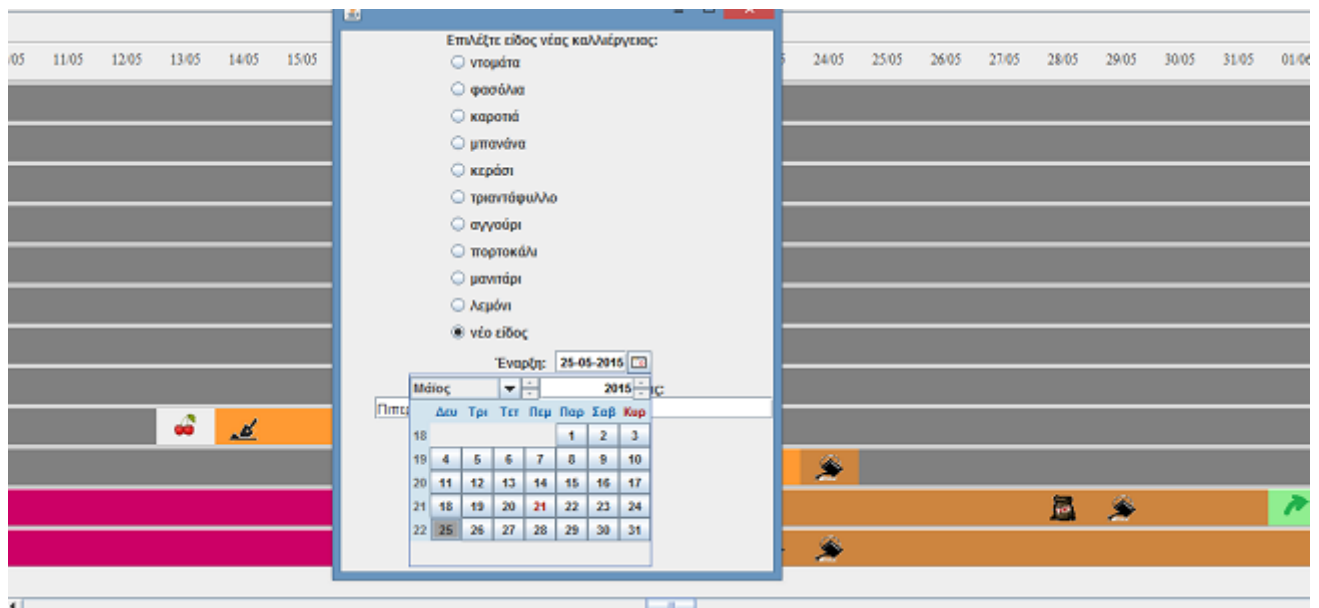
Σχήμα 9.9: Εισαγωγή νέας ενέργειας στην καλλιέργεια καροτιάς



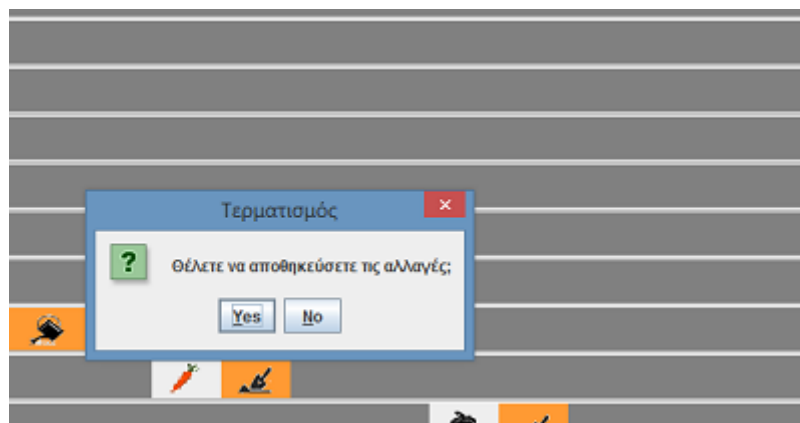
Σχήμα 9.10: Επιλογή νέου είδους καλλιέργειας

Κάθετα ονόματα καλλιεργειών

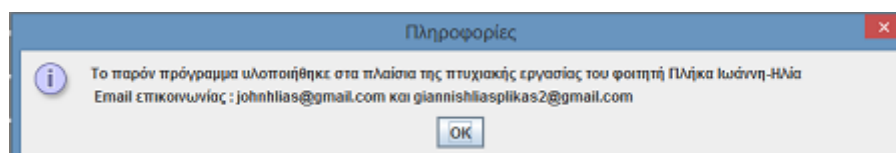
Αριστερά της εφαρμογής και κάθετα βρίσκονται τα ονόματα των καλλιεργειών, περιγράφοντας την καλλιέργεια κάθε σειράς όπως παρατηρήθηκαν και παραπάνω. Αυτά φαίνονται και αριστερά της εικόνας 9.18.



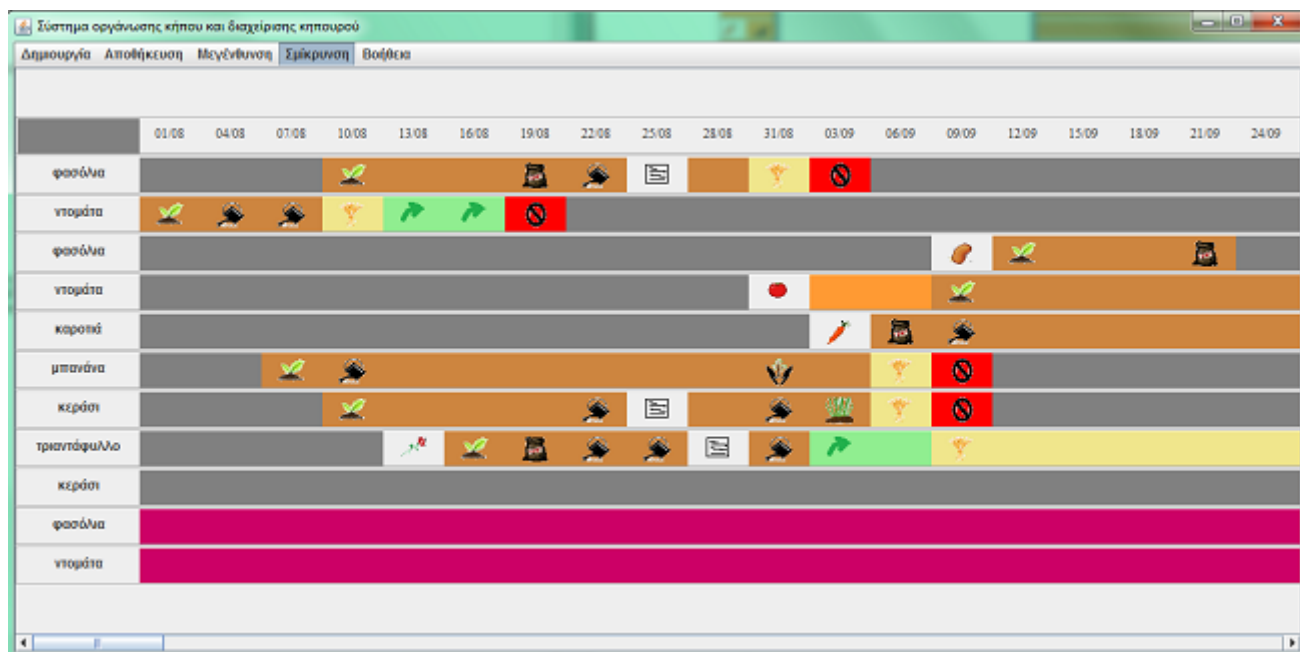
Σχήμα 9.11: Επιλογή νέας ημερομηνίας εισαγωγής



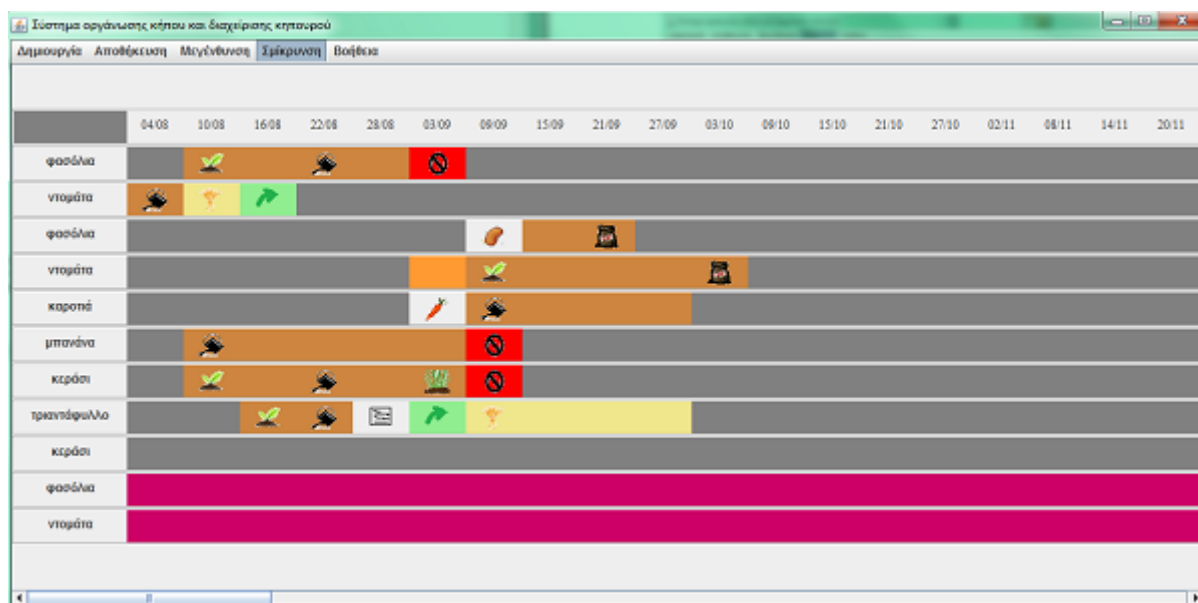
Σχήμα 9.12: Παράθυρο εξόδου



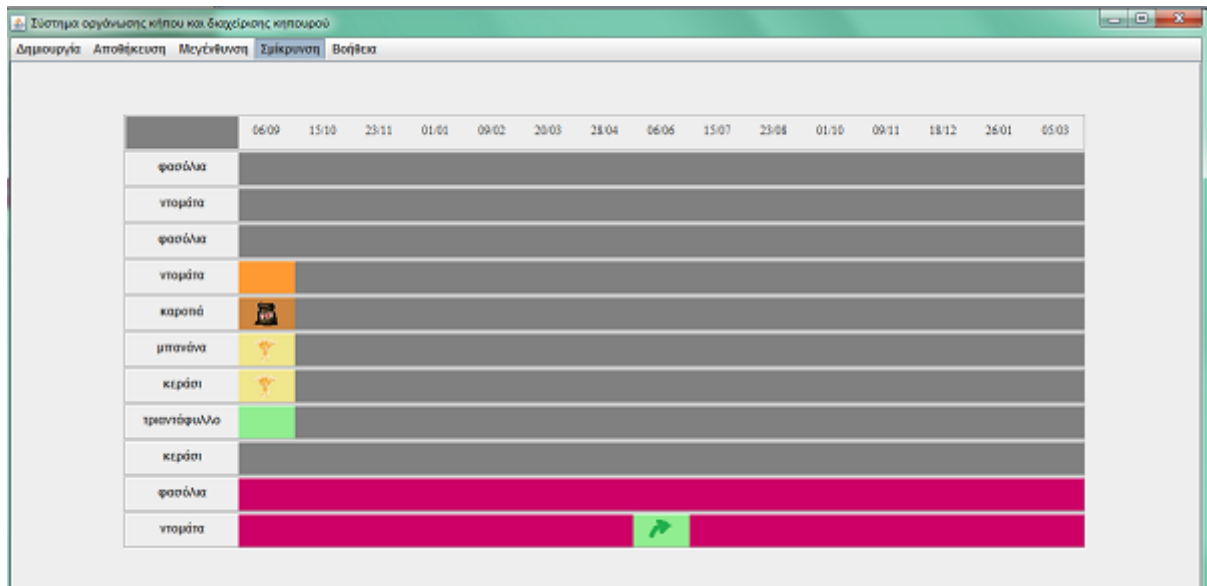
Σχήμα 9.13: Επιλογή Βοήθεια του μενού επιλογών



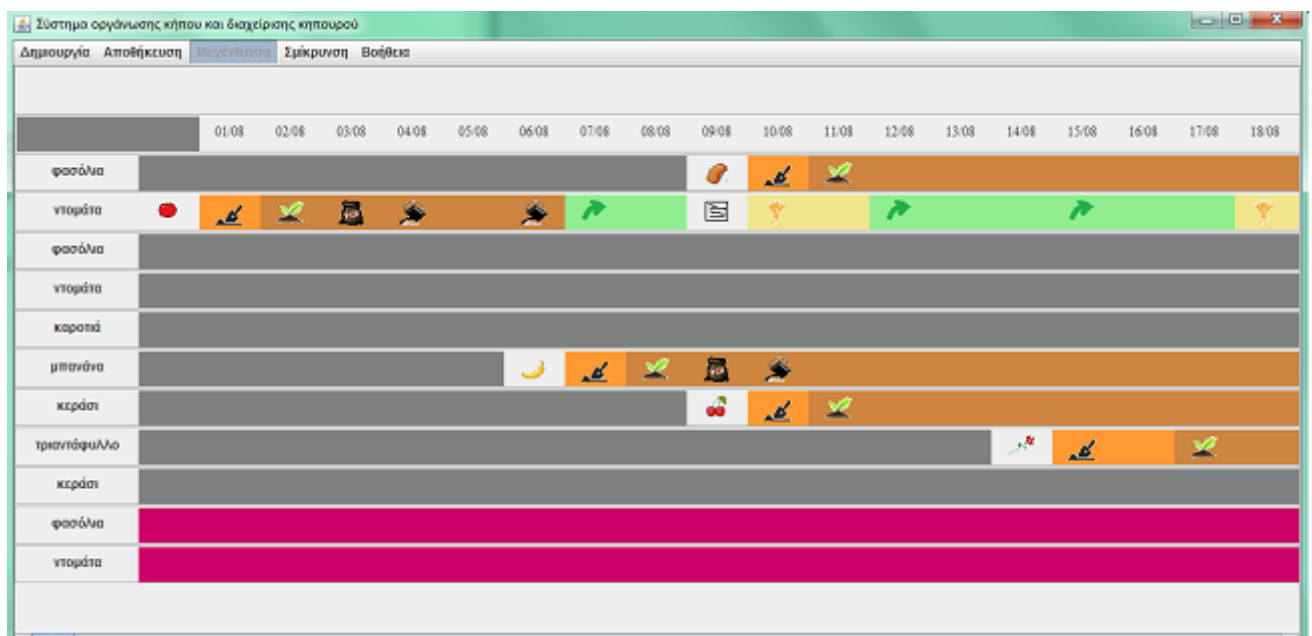
Σχήμα 9.14: Μία φορά Σμίκρυνση



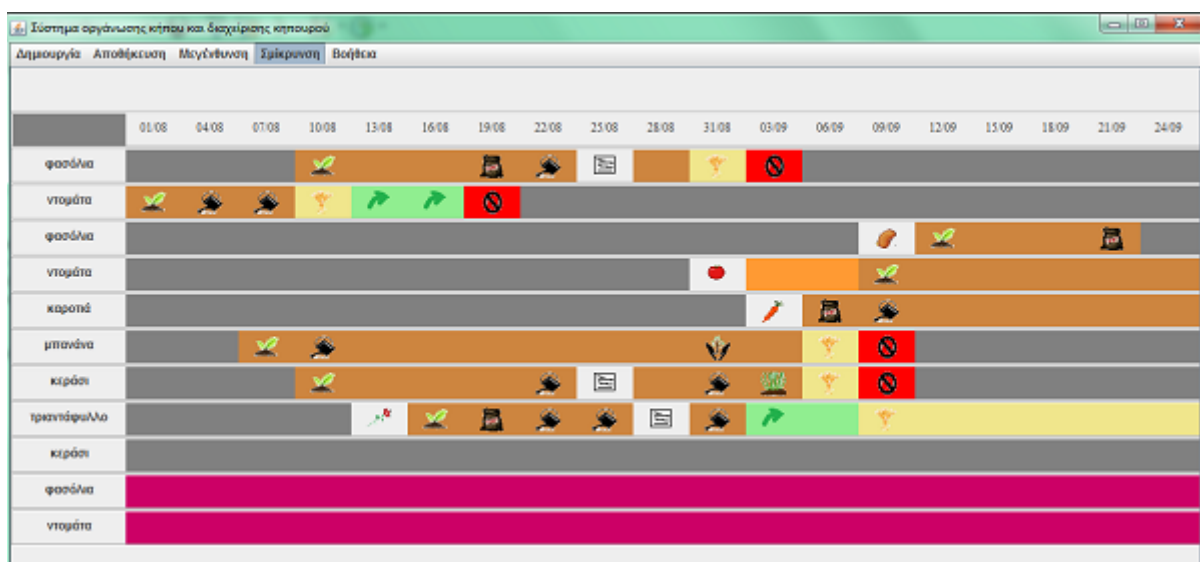
Σχήμα 9.15: Δεύτερη φορά Σμίκρυνση



Σχήμα 9.16: Σμίκρυνση πολλαπλές φορές



Σχήμα 9.17: Μεγένθυση αφού έχει γίνει Σμίκρυνση μία φορά



Σχήμα 9.18: Μεγένθυση αφού έχει γίνει Σμίκρυνση δύο φορές

Κεφάλαιο 10

Συμπεράσματα

Συμπεραίνουμε ότι η εφαρμογή που υλοποιήθηκε αποτελεί έναν ευχάριστο τρόπο να διαχειριστεί κάποιος τις καλλιέργειές του αλλά ταυτόχρονα και πολύ αποδοτικό, συνδυάζοντας εικόνες, χρώματα και χρήσιμες λειτουργίες. Έτσι ακόμη και ο πιο αρχάριος χρήστης θα μπορεί να τη χρησιμοποιήσει χωρίς κάποιους ενδοιασμούς, προγραμματίζοντας άριστα τις ενέργειές του πάνω στις καλλιέργειες καθώς και να προβλέπει μέσω των παρελθοντικών του ενεργειών, μελλοντικές ενέργειες στα φυτά που καλλιεργεί.

Η εφαρμογή, διακατέχεται από χρήσιμες λειτουργίες και δύσκολα παρατηρείται κάποια αδυναμία σε αυτή. Παρ'όλα αυτά, ως αδυναμία της θα μπορούσε να θεωρηθεί κατά την εισαγωγή νέου συμβάντος, όταν επιλέγεται να προστεθεί εικονίδιο καταστροφής εντός της μπάρας εξέλιξης κάποιας καλλιέργειας και όχι έξω από αυτή. Εκεί, θα μπορούσαν τα εικονίδια δεξιά της και τα αντίστοιχα χρώματα να εξαφανίζονται μιάς και κάποια καταστροφή έλαβε χώρα. Επίσης κατά τη δημιουργία νέας καλλιέργειας, δημιουργείται νέο εικονίδιο που την αντιπροσωπεύει και ως προτεινόμενη πρώτη ενέργεια η εισαγωγή νέας καλλιέργειας σε χώμα, στην επιλέξιμη ημερομηνία. Αν κάποιος χρήστης επιλέξει να εισάγει διαφορετικό εικονίδιο ενέργειας που αλλάζει τη φάση, στη θέση του προτεινόμενου εικονιδίου, το ένα επικαλύπτει το άλλο και το πρώτο εμφανίζεται με το χρώμα φόντου του δευτέρου. Ακόμη και αυτές οι αδυναμίες δε θεωρούνται αδυναμίες, μιας και θα μπορούσε κάποιος να τις εξελίξει μελλοντικά.

Η εφαρμογή, θα μπορούσε μελλοντικά να εξελιχθεί με διάφορους τρόπους. Ο ένας αναφέρθηκε προηγουμένως με την καταστροφή. Άλλη λειτουργία που κάποιος θα μπορούσε να προσθέσει σε αυτή, είναι οι λειτουργίες Εκτός ενεργές, Εκτός παλιές και Εκτός recommended. Αναλόγως την επιλογή του ο χρήστης θα μπορεί να βλέπει στο παράθυρο της εφαρμογής όλες τις καλλιέργειες εκτός αυτών που επέλεξε να μη βλέπει. Αυτές οι λειτουργίες θα μπορούσαν να υλοποιηθούν, είτε με RadioButtons, είτε μέσου του μενού επιλογών, είτε με JButtons. Αυτό εξαρτάται από το χρήστη. Ακόμη, αναφέρθηκε στις αδυναμίες η επικάλυψη του ενός εικονιδίου με το άλλο. Στην εφαρμογή, κατά τη δημιουργία νέας καλλιέργειας και στη θέση του προτεινόμενου εικονιδίου, θα μπορούσε να εμφανίζεται το εικονίδιο της πρόσθεσης το οποίο αν ο χρήστης το επέλεγε θα του επέτρεπε να εισάγει κάποια ενέργεια. Ως διαθέσιμες ενέργειες εκεί θα ήταν είτε φύτευμα καλλιέργειας σε φυτώριο, είτε αγορά φυτωρίου σε χώμα, είτε αγορά έτοιμου φυτωρίου, είτε τέλος φύτευση καλλιέργειας σε χώμα. Επίσης, στην εφαρμογή θα μπορούσαν να προστεθούν χρώματα γύρω από τις περσινές, γύρω από τις φετινές καλλιέργειες με σκοπό να τις ξεχωρίζει απευθείας οπτικά. Βέβαια, μία λειτουργία που θα μπορούσε κάποιος να προσθέσει στο μενού, είναι η Διαγραφή. Σε αυτή θα μπορεί κάποιος να διαγράψει όποια καλλιέργεια επιθυμεί και ταυτοχρόνως να διαγράφεται και το αντίστοιχο αρχείο. Πολύ χρήσιμη λειτουργία, θα ήταν κάποια επιλογή, μέσω της οποίας ο χρήστης

θα μπορεί να μεταβεί απευθείας σε συγκεκριμένη περίοδο, είτε αφορά αυτό συγκεκριμένη μέρα, είτε μήνα, χρόνο ή ακόμη και σε κάποια καλλιέργεια σε δεδομένη στιγμή. Αποδοτικό θα ήταν επίσης να μπορεί κάποιος να προσθέσει σε καλλιέργεια την τοποθεσία στην οποία αυτή βρίσκεται για να μη τη συγχέει με άλλη. MouseWheel listener, δεν αποτελεί μειονεκτική επιλογή καθώς θα μπορούσε με αυτό τον τρόπο να υλοποιηθεί η λειτουργία να μπορεί κάποιος να βλέπει μεγαλύτερα ή μικρότερα εικονίδια. Αυτό θα μπορούσε να γίνει και με χρήση κάποιων κουμπιών. Τελευταία αλλά όχι λιγότερο χρήσιμη λειτουργία θα μπορούσε να είναι το drag and drop, δηλαδή να μπορεί κάποιος απλά να σύρει το panel το οποίο περιέχει μία καλλιέργεια στη θέση κάποιου άλλου panel που περιέχει άλλη, αλλάζοντας με εκείνο θέσεις.

Κεφάλαιο 11

Βιβλιογραφία

11.1 Βιβλιογραφικές Αναφορές

1. Professional Java Fundamentals, Shy Cohen, Tom Mitchell, Adres Gonzalez, Lary Rodrigues, Kerry Hammil, Εκδόσεις WROX PRESS Ltd, 1996, ISBN 1-861000-38-3.
2. Εγχειρίδιο της Java, Laura Lemay, Charles L Perkins, Michael Morrison, Εκδόσεις Μ. Γκιούρδας, 1996, ISBN 960-512-083-6
3. Java ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ όγδοη έκδοση, Paul Deitel, Harvey Deitel, Εκδόσεις Μ. Γκιούρδας, 1996, ISBN 978-960-512-592-9
4. ΑΠΟΛΥΤΗ JAVA, πρώτη Ελληνική έκδοση, Δρ. Δημήτρης Ιακωβίδης, ISBN 978-960-411-645-4

11.2 Ηλεκτρονικές διευθύνσεις

1. <http://cs.nyu.edu/~yap/classes/visual/03s/lect/17/>
2. <https://docs.oracle.com/en/>
3. <http://stackoverflow.com/>
4. [http://en.wikipedia.org/wiki/Delphi_\(programming_language\)](http://en.wikipedia.org/wiki/Delphi_(programming_language))
5. <http://www.javacoffeebreak.com/tutorials/index.html>
6. <http://www.freewarejava.com/>
7. https://www3.ntu.edu.sg/home/ehchua/programming/java/J4a_GUI.html
8. <https://netbeans.org/kb/docs/java/gui-functionality.html>
9. <http://www.codeproject.com/Articles/33536/An-Introduction-to-Java-GUI-Programming>
10. <http://zetcode.com/tutorials/javaswingtutorial/>
11. http://www.tutorialspoint.com/javaexamples/java_simple_gui.htm
12. [https://msdn.microsoft.com/library/xk24xdbe\(v=vs.90\).aspx](https://msdn.microsoft.com/library/xk24xdbe(v=vs.90).aspx)
13. <https://msdn.microsoft.com/en-us/library/z1zx9t92.aspx>
14. <http://www.codingunit.com/php-tutorial-language-introduction>
15. <https://www.ruby-lang.org/en/>
16. <http://www.skilledup.com/articles/4-reasons-learn-ruby-first-programming-language>
17. <http://ebooks.edu.gr/modules/ebook/show.php/DSGL-C113/77/631,2289/>
18. <http://www.agrosales.gr/tips/generaladvise/tyyg>
19. <http://www.newsbeast.gr/technology>
20. <http://3dim-kater.pie.sch.gr/fitofarmaka.htm>

21. <http://lntalis.wix.com/tecnologia#!-/cg6n>
22. <http://www.viologika.gr/viologiki-kalliergeia.php>