



**ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ**

**ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ**

**ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ**

**ΜΟΥΣΙΚΟΣΥΝΘΕΤΗΣ**

**σύστημα ημιαυτόματης σύνθεσης Βυζαντινής μουσικής**

**ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ**

του

**ΘΕΟΔΩΡΟΥ ΚΑΡΒΟΥΝΗ**

**Επιβλέπων :** Ιωάννης Βασιλείου  
Καθηγητής Ε.Μ.Π.

Αθήνα, Ιούλιος 2008





ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ  
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ  
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ  
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

# ΜΟΥΣΙΚΟΣΥΝΘΕΤΗΣ

## σύστημα ημιαυτόματης σύνθεσης Βυζαντινής μουσικής

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

**ΘΕΟΔΩΡΟΥ ΚΑΡΒΟΥΝΗ**

**Επιβλέπων :** Ιωάννης Βασιλείου  
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 10<sup>η</sup> Ιουλίου 2008.

(Υπογραφή)

.....  
Ιωάννης Βασιλείου  
Καθηγητής Ε.Μ.Π.

(Υπογραφή)

.....  
Τίμος Σελλής  
Καθηγητής Ε.Μ.Π.

(Υπογραφή)

.....  
Βασίλειος Λούμος  
Καθηγητής Ε.Μ.Π.

Αθήνα, Ιούλιος 2008

*(Υπογραφή)*

.....

**ΘΕΟΔΩΡΟΣ ΚΑΡΒΟΥΝΗΣ**

Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

© 2008 – All rights reserved

## Περίληψη

Η παρούσα διπλωματική υλοποιεί ένα σύστημα ημιαυτόματης σύνθεσης Βυζαντινής μουσικής. Η κύρια λειτουργία του είναι η εύρεση μια κατάλληλης και σύμφωνης με το Βυζαντινό ύφος μελοποίησης για το ποιητικό κείμενο που εισάγει ο χρήστης. Αυτό το καταφέρνει εξετάζοντας τους τονισμούς των λέξεων και χρησιμοποιώντας βάσεις δεδομένων, οι οποίες περιέχουν πάρα πολλά μελωδικά μοτίβα της Βυζαντινής μουσικής. Δεν είναι όμως ένα εντελώς αυτόματο σύστημα καθώς παρέχει στον χρήστη τη δυνατότητα τροποποίησης επί της λύσης καθώς και πλήθος άλλων δυνατοτήτων οι οποίες σχηματίζουν ένα ολοκληρωμένο βοήθημα σύνθεσης. Είναι ένα πολύ χρήσιμο εργαλείο γιατί στη Βυζαντινή μουσική ο συνθέτης οφείλει να σεβαστεί την παράδοση και τα υπάρχοντα μελωδικά σχήματα αλλά δυστυχώς είναι πολύ εύκολο να ξεφύγει επηρεασμένος από άλλα ακούσματα (π.χ. Ευρωπαϊκή μουσική).

**Λέξεις Κλειδιά:** <<ημιαυτόματη σύνθεση, μουσικά μοτίβα, Βυζαντινή Μουσική >>



## **Abstract**

This diploma thesis implements a semi-automatic system for composition on Byzantine music. Its main function is to find an appropriate, and consistent with the Byzantine style, setting to music for the poetic text that the user inserts. This is feasible by examining the accentuations of the words and using databases, which contain too many melodic patterns of Byzantine music. However it isn't a completely automated system as it gives the user the ability to modify the solution as well as numerous other features which form an integrated device composition. It is a very useful tool because the Byzantine music composer has to respect the tradition and existing melodic shapes, but unfortunately it's very easy to be affected influenced by other sounds (e.g. European music).

**Keywords:** <<melodic forms, semi-automatic composition, Byzantine Music>>





## Πίνακας περιεχομένων

<b>1</b>	<b>Εισαγωγή.....</b>	<b>1</b>
1.1	Αντικείμενο Διπλωματικής.....	2
1.2	Οργάνωση Κειμένου.....	3
<b>2</b>	<b>Βυζαντινή Μουσική και Σύνθεση.....</b>	<b>4</b>
2.1	Βασικά Στοιχεία Ιστορίας της Βυζαντινής Μουσικής.....	4
2.1.1	<i>Προέλευση και Πρώιμη Χριστιανική Περίοδος.....</i>	<i>4</i>
2.1.2	<i>Ακμή της Βυζαντινής Περιόδου.....</i>	<i>6</i>
2.1.3	<i>Υστερη Βυζαντινή και Μεταβυζαντινή Περίοδος.....</i>	<i>10</i>
2.2	Βασικά Στοιχεία Θεωρίας Βυζαντινής Μουσικής.....	10
2.2.1	<i>Φθόγγοι και Κλίμακες.....</i>	<i>10</i>
2.2.2	<i>Μουσικοί Χαρακτήρες.....</i>	<i>11</i>
2.2.3	<i>Υποστατικά Σημεία.....</i>	<i>12</i>
2.2.4	<i>Μουσικά Γένη.....</i>	<i>12</i>
2.2.5	<i>Ήχοι.....</i>	<i>13</i>
2.3	Συγγραφή Βυζαντινής Μουσικής.....	13
2.3.1	<i>Το πακέτο EZ.....</i>	<i>13</i>
2.3.2	<i>Το Πακέτο BZ.....</i>	<i>13</i>
2.3.3	<i>Το Πρόγραμμα «Μελωδός».....</i>	<i>15</i>
2.3.4	<i>Βυζαντινογράφος.....</i>	<i>15</i>
2.3.5	<i>Βυζαντινή Κάλαμος.....</i>	<i>16</i>
2.4	Σύνθεση Βυζαντινής Μουσικής.....	17
2.4.1	<i>Βάση Μελωδικών Μοτίβων Πατρός Εφραίμ.....</i>	<i>17</i>
<b>3</b>	<b>Ανάλυση και Σχεδίαση.....</b>	<b>18</b>
3.1	Αρχιτεκτονική.....	18
3.2	Περιγραφή Λειτουργιών.....	20
3.2.1	<i>Το Πρόγραμμα.....</i>	<i>20</i>
3.2.2	<i>Η Διαπροσωπία.....</i>	<i>21</i>

3.2.3	<i>Η Βάση Δεδομένων</i> .....	23
3.3	Αρχιτεκτονική .....	23
3.4	Περιγραφή Βάσης Δεδομένων .....	25
<b>4</b>	<b>Υλοποίηση</b> .....	<b>27</b>
4.1	Τεχνολογικά Πρότυπα .....	27
4.1.1	<i>Java SDK 1.6</i> .....	27
4.1.2	<i>NetBeans IDE 6.0</i> .....	28
4.1.3	<i>iText library</i> .....	28
4.1.4	<i>Rich Text Format (.rtf) τύπος αρχείων</i> .....	29
4.2	Προγραμματιστικά εργαλεία .....	29
4.2.1	<i>Netbeans 6.0</i> .....	29
4.2.2	<i>Βιβλιοθήκες</i> .....	30
4.3	Αναλυτική Περιγραφή Κλάσεων .....	30
4.3.1	<i>wordGen.java</i> .....	30
4.3.2	<i>Word.java</i> .....	31
4.3.3	<i>wordEng.java</i> .....	32
4.3.4	<i>GenPhrase.java</i> .....	32
4.3.5	<i>CleanString.java</i> .....	32
4.3.6	<i>Elements.java</i> .....	32
4.3.7	<i>enterFrame.java</i> .....	33
4.3.8	<i>martFrame.java</i> .....	33
4.3.9	<i>FAQframe.java</i> .....	33
4.3.10	<i>Newframe.java</i> .....	33
4.4	Λεπτομέρειες Υλοποίησης .....	35
4.4.1	<i>Αυτόματος Συλλαβισμός του Κειμένου</i> .....	35
4.4.2	<i>Εύρεση Μελοποίησης του Κειμένου</i> .....	37
4.4.3	<i>Recalculation</i> .....	38
4.4.4	<i>Αλλαγή της Μελοποίησης Κάποιων Φράσεων</i> .....	39
4.5	Χρήστες .....	40
<b>5</b>	<b>Έλεγχος</b> .....	<b>42</b>
5.1	Μεθοδολογία Ελέγχου .....	42

5.2	Αναλυτική Παρουσίαση Ελέγχου .....	43
5.2.1	Ένα Βασικό Σενάριο Χρήσης .....	43
5.2.2	Μηνύματα.....	49
<b>6</b>	<b>Επίλογος .....</b>	<b>50</b>
6.1	Σύνοψη και Συμπεράσματα .....	50
6.2	Μελλοντικές Επεκτάσεις .....	51
6.2.1	Προσθήκη άλλων βάσεων .....	51
6.2.2	Αυτόματος Συλλαβισμός σε Άλλες Γλώσσες.....	51
6.2.3	Παρουσίαση Μελοποίησης .....	51
<b>7</b>	<b>Βιβλιογραφία.....</b>	<b>52</b>

# 1

## *Εισαγωγή*

Ο Μουσικοσυνθέτης είναι ένα σύστημα ημιαυτόματης σύνθεσης Βυζαντινής μουσικής. Η σημασία της ημιαυτόματης σύνθεσης στη Βυζαντινή μουσική έγκειται στο γεγονός ότι ο συνθέτης οφείλει να σεβαστεί την παράδοση και τα υπάρχοντα μελωδικά σχήματα αλλά δυστυχώς είναι πολύ εύκολο να ξεφύγει επηρεασμένος από άλλα ακούσματα (π.χ. Ευρωπαϊκή μουσική).

Μέχρι στιγμής δεν υπήρχαν αυτά τα μουσικά σχήματα καταγεγραμμένα αλλά τελευταία έχουν γίνει κάποιες προσπάθειες για αυτό. Για παράδειγμα ο ιερομόναχος Εφραίμ από την Αριζόνα εξετάζοντας αρκετά κείμενα Βυζαντινής μουσικής κατέγραψε πολλά από αυτά σε αρχαία κείμενα. Στο μέλλον ενδεχομένως να γίνουν και άλλες τέτοιες προσπάθειες. Έτσι προκύπτει η ανάγκη δημιουργίας ενός προγράμματος το οποίο να εξετάζει μια βάση μελωδιών και να προτείνει για κάθε είδος λέξης (σε σχέση με το πλήθος των συλλαβών και τον τονισμό) κατάλληλα μελωδικά σχήματα.

Μέχρι στιγμής δεν υπήρχε κάποια εφαρμογή η οποία να δίνει την δυνατότητα για σύνθεση ενός βυζαντινού μέλους με κάποιον τέτοιο αυτοματοποιημένο τρόπο. Αν κάποιος ήθελε να συνθέσει ένα μέλος κάπως αυτόματα το μόνο που μπορούσε να κάνει ήταν να βρει ο ίδιος για κάθε φράση ποιος τονικός κωδικός αντιστοιχεί και κατόπιν να ψάξει σε βάσεις δεδομένων και να διαλέξει μία από τις μελοποιήσεις της που ταιριάζουν. Είναι προφανές ότι η παραπάνω διαδικασία είναι πολύ κοπιαστική ιδίως δε αν το προς μελοποίηση κείμενο είναι μεγάλο. Εδώ έρχεται λοιπόν η εφαρμογή Μουσικοσυνθέτης η οποία μέσα από μια εύχρηστη διεπαφή δέχεται τους στοίχους από τον χρήστη τους επεξεργάζεται και του εμφανίζει μια πιθανή μελοποίηση. Ακόμη δίνεται η δυνατότητα στο χρήστη να επέμβει στο αποτέλεσμα με

διαφόρους τρόπους ώστε να επιτευχθεί το επιθυμητό αποτέλεσμα. Επίσης εκτός των άλλων υπάρχει η δυνατότητα εξαγωγής του αποτελέσματος σε έγγραφο του Microsoft Word.

Τα τελευταία χρόνια υπάρχει έντονο ενδιαφέρον για τη δημιουργία προγραμμάτων σχετικών με τη Βυζαντινή μουσική για αυτό και πιστεύουμε ότι ο Μουσικοσυνθέτης θα έχει μεγάλη απήχηση, Ο Μουσικοσυνθέτης μάλιστα πρωτοπορεί καθώς ασχολείται με τη σύνθεση της Βυζαντινής μουσικής ενώ τα ήδη υπάρχοντα προγράμματα ασχολούνται μόνο με τη συγγραφή της Βυζαντινής μουσικής.

Ο Μουσικοσυνθέτης απευθύνεται σε δύο είδη χρηστών. Κατ' αρχάς στοχεύει στο να καταστεί ένα πολύτιμο βοήθημα για κάποιον έμπειρο ψάλτη ο οποίος θέλει να συνθέσει ένα μέλος. Το σύστημα δεν βγάζει σαν έξοδο απλώς μια μελοποίηση αλλά δίνει παράλληλα σε έναν έμπειρο χρήστη τη δυνατότητα να επέμβει τροποποιώντας με διάφορους τρόπους την έξοδο και παραμετροποιώντας την διαδικασία ή ακόμα και να προσθέσει επιπλέον μελωδικά μοτίβα στη βάση.

Παρόλα αυτά το σύστημα στοχεύει στο να μπορεί ακόμα και κάποιος όχι ιδιαίτερα σχετικός με την Βυζαντινή μουσική χρήστης να πάρει από το σύστημα μια μελοποίηση για ένα κείμενο που θα εισάγει.

Για την αυτοματοποίηση δημιουργήθηκε το κυρίως πρόγραμμα του συστήματος σε Java το οποίο περιλαμβάνει και ένα πολύ εύχρηστο και λειτουργικό interface. Επιπλέον υπάρχει η δυνατότητα της εξαγωγής της μελοποίησης σε αρχείο .mf το οποίο μπορεί να αναγνωστεί και να τροποποιηθεί μέσω του Microsoft Word.

Η βάση μελωδιών που χρησιμοποιήθηκε είναι αυτή του π.Εφραίμ αλλά δεν χρησιμοποιήθηκε αυτούσια καθώς χρειάστηκε να γίνει η μεταγραφή της σε αρχεία τα οποία να είναι αναγνώσιμα από την Java.

## ***1.1 Αντικείμενο Διπλωματικής***

Το σύστημα «Μουσικοσυνθέτης» έχει κύριο σκοπό να βοηθήσει τον χρήστη στην σύνθεση Βυζαντινής μουσικής κατ' αρχάς απαλλάσσοντας τον από το συνεχές ψάξιμο σε τεράστια αρχεία για την εύρεση της μελοποίησης της κάθε φράσης του ποιητικού κειμένου αλλά παράλληλα αξιοποιώντας τις δυνατότητες της σύγχρονης τεχνολογίας ώστε να του παρέχει και αρκετές επιπλέον βολικές λειτουργίες. Συγκεκριμένα η εφαρμογή κάνει τα εξής:

- Διατηρεί πληροφορίες για τους διάφορους χρήστες οι οποίοι χρησιμοποιούν το πρόγραμμα.
- Συλλαβίζει αυτόματα το κείμενο που έδωσε ο χρήστης ώστε να παραχθούν οι κωδικοί οι οποίοι έχουν καταγραφεί στη βάση

-Βρίσκει μελωδίες που ταιριάζουν με αυτούς (μελοποιεί δηλαδή το κείμενο το οποίο παρέθεσε ο χρήστης)

-Επιτρέπει στον χρήστη να καθορίσει διάφορες παραμέτρους προτού ζητήσει μελοποίηση

-Επιτρέπει την τροποποίηση επιμέρους φράσεων της λύσης

-Δίνει την δυνατότητα για εξαγωγή της λύσης σε αρχείο .rtf και την παρουσίασή και τροποποίησή της σε προγράμματα επεξεργασίας κείμενο όπως το Microsoft Word

-Υπάρχει η δυνατότητα της επαναφοράς στη διαπροσωπία προηγούμενων ή και επόμενων μελοποιήσεων (undo & redo)

-Επιτρέπει στον χρήστη να εισάγει νέα μοτίβα στην βάση

## ***1.2 Οργάνωση Κειμένου***

Στο **πρώτο κεφάλαιο** γίνεται μια εισαγωγή και περιγράφεται το αντικείμενο της διπλωματικής καθώς και η οργάνωση του τόμου.

Στο **δεύτερο κεφάλαιο** αναφέρουμε όλα τα απαραίτητα στοιχεία σχετικά με την σύνθεση και τη συγγραφή στη Βυζαντινή μουσική και τα προγράμματα που υπάρχουν μέχρι τώρα.

Στο **τρίτο κεφάλαιο** γίνεται μια περιγραφή της αρχιτεκτονικής και της σχεδίασης του συστήματος καθώς και η ανάλυση των απαιτήσεων.

Στο **τέταρτο κεφάλαιο** με την υλοποίηση του συστήματος και αναλυτική περιγραφή της λειτουργίας των κυριότερων αλγορίθμων της εφαρμογής

Στο **πέμπτο** ασχολούμαστε με τον έλεγχο και την Εγκατάσταση της εφαρμογής

Στο **έκτο** αναφέρουμε τα γενικά συμπεράσματα στα οποία καταλήξαμε και περιγράφουμε κάποιες ιδέες για μελλοντικές επεκτάσεις της διπλωματικής

Τέλος στο **έβδομο** κεφάλαιο παραθέτουμε την βιβλιογραφία

# 2

## ***Βυζαντινή Μουσική και Σύνθεση***

### ***2.1 Βασικά Στοιχεία Ιστορίας της Βυζαντινής Μουσικής***

**Βυζαντινή μουσική** είναι κοινός όρος που χρησιμοποιείται προκειμένου να χαρακτηρίσει το μεσαιωνικό ιερό μέλος των χριστιανικών εκκλησιών που ακολούθησαν το ορθόδοξο λειτουργικό τυπικό.

Είναι πιθανό στα αρχικά της στάδια να υπήρξε μία περαιτέρω ανάπτυξη της αρχαίας ελληνικής μουσικής αν και πλέον η βυζαντινή μουσική θεωρείται ως ανεξάρτητο μουσικό είδος, με συγκερασμό στοιχείων που προέρχονται από συριακές, εβραϊκές καθώς επίσης και ελληνικές πηγές. Οι αρχές της χρονολογούνται από ορισμένους μελετητές στον 4ο αιώνα μ.Χ, λίγο μετά τη μεταφορά της πρωτεύουσας της Ρωμαϊκής Αυτοκρατορίας στην Κωνσταντινούπολη από το Μέγα Κωνσταντίνο.

Η βυζαντινή μουσική που διασώζεται είναι στο σύνολο της εκκλησιαστική, με εξαίρεση κάποιους αυτοκρατορικούς ύμνους, που και αυτοί έχουν θρησκευτικά στοιχεία. Το βυζαντινό άσμα ήταν μονωδικό, σε ελεύθερο ρυθμό, και προσπάθησε συχνά να απεικονίσει μελωδικά την έννοια των λέξεων. Η γλώσσα που χρησιμοποιήθηκε ήταν η ελληνική. Ο βυζαντινός ύμνος, του οποίου υπήρξαν τρεις τύποι, ήταν η μέγιστη έκφραση αυτού του μουσικού είδους.

#### ***2.1.1 Προέλευση και Πρώιμη Χριστιανική Περίοδος***

Η παράδοση του ανατολικού λειτουργικού άσματος, που καλύπτει τον ελληνόφωνο χώρο, αναπτύχθηκε στη Βυζαντινή Αυτοκρατορία από την καθιέρωση ως πρωτεύουσας της Κωνσταντινούπολης, το 330 μέχρι την πτώση της το 1453. Είναι αναντίρρητα σύνθετης

προέλευσης, προσεγγίζοντας στις καλλιτεχνικές και τεχνικές παραγωγές της κλασικής αρχαιότητας, στην εβραϊκή μουσική και εμπνευσμένη από τη μονοφωνική μελωδία που άνθισε στις πρώιμες χριστιανικές κοινότητες της Αλεξάνδρειας, της Αντιόχειας και της Εφέσου.

Βλέπουμε ότι υπάρχει μια βασική σύνδεση μεταξύ της μουσικής της Συναγωγής και της πρώιμης χριστιανικής μουσικής. Σχέση μεταξύ των δύο παραδόσεων υφίσταται υπό μορφή ομοιοτήτων της Ψαλμωδίας και των Ύμνων. Εν συντομία, Ψαλμωδία είναι η μελωδική απόδοση των Ψαλμών του Δαβίδ από την εβραϊκή κοινότητα, η οποία μεταφέρθηκε στη χριστιανική μουσική παράδοση και διαμόρφωσε τον τρόπο που διάφορες μορφές βυζαντινών μουσικών κομματιών αποδίδονταν (οι χριστιανικές δοξολογίες είναι το καλύτερο παράδειγμα της συντήρησης της εβραϊκής ψαλμωδίας). Οι ύμνοι αφ' ετέρου, είναι παραφράσεις του βιβλικού κειμένου, οι οποίες γράφονται με τέτοιο τρόπο ώστε να μπορούν να προσαρμοστούν σε έναν παραδοσιακό τύπο άσματος. Αυτή η πρακτική βασίστηκε σταθερά στην εβραϊκή παράδοση και συναντάται στις εβραϊκές τελετές. Οι πρώιμες χριστιανικές προσπάθειες υμνογραφίας καταδικάστηκαν αμέσως επειδή δεν βασίστηκαν αποκλειστικά στις λέξεις της Γραφής.

Έτσι μπορούμε να δούμε ότι η μεταφορά της εβραϊκής παράδοσης ήταν πρώτιστα πρακτικής φύσης. Αυτό σημαίνει ότι η προέλευση αυτού που σήμερα καλείται βυζαντινή μουσική βασίστηκε στις καθιερωμένες πρακτικές των προσήλυτων Εβραίων οι οποίοι μιμήθηκαν τη λατρεία των συναγωγών από τις οποίες προέρχονταν: κράτησαν απλά τις πρακτικές που έμαθαν κατά τα έτη που είχαν αφιερώσει στην ψαλμωδία και τη λατρεία στις Συναγωγές τους και εφάρμοσαν αυτές τις πρακτικές στη νέα τους λατρεία, που ήταν για αυτούς, μια συνέχεια της θρησκείας τους.

Παραδοσιακά, ο Πυθαγόρας αναφέρεται ως θεμελιωτής του μουσικού είδους που μετέπειτα εξελίχτηκε στη βυζαντινή μουσική. Αυτό ισχύει μέχρι ένα σημείο. Εκεί που οι Εβραίοι συνέβαλαν με την παράδοση και την πρακτική ο Πυθαγόρας συνέβαλε με τη θεωρία. Ήταν ο πρώτος που συνέδεσε τη μουσική με τα μαθηματικά και καινοτόμησε με τη μελέτη της ακουστικής. Ήταν επίσης ο πρώτος που δημιούργησε τους μουσικούς "ήχους" και απέδωσε τις αναλογίες τους με νότες. Αυτό δημιούργησε τις κλίμακες που είναι η βάση της Οκτώηχου του κέντρου δηλαδή της βυζαντινής μουσικής θεωρίας.

Οι αρχαιοελληνικοί μουσικοί ήχοι είναι απλά διαφορετικές ταξινομήσεις των φθόγγων των εναλλασσόμενων φωνητικών τόνων. Αυτές οι ταξινομήσεις δημιουργούν τις κλίμακες που συσχετίζονται η μία με την άλλη αλλά χαρακτηρίζονται από διαφορετικές διαθέσεις, όπως μία μείζων κλίμακα συγκρινόμενη με μια ελάσσονα στη δυτική μουσική. Κατά συνέπεια, οι ήχοι ταξινομήθηκαν με την απόδοση ονομάτων σύμφωνα με τη διάθεση που ομοίαζαν. Οι



οκτώ ήχοι που περιέχονται στη βυζαντινή μουσική είναι χωρισμένοι σε τρία γένη διαθέσεων. Αυτό είναι άμεσος απότοκος τη αρχαίας ελληνικής πρακτικής, γιατί και στα συστήματα, ο αριθμός και τα ονόματα των γενών είναι τα ίδια.

Τα βυζαντινά ασματικά χειρόγραφα χρονολογούνται από τον 9ο αιώνα, ενώ τα βιβλία Εκφωνητικής σημειογραφίας (ένα απλοϊκό γραφικό σύστημα με σκοπό να δείξει τον τρόπο ανάγνωσης των Γραφών) ξεκινούν έναν αιώνα νωρίτερα και εξακολουθούν να βρίσκονται σε χρήση έως το 12ο ή 13ο αιώνα. Η γνώση μας για την παλαιότερη περίοδο προέρχεται από τα "Τυπικά" (διατάξεις εκκλησιαστικών μυστηρίων και τελετών), τα έργα των εκκλησιαστικών Πατέρων και τις μεσαιωνικές διηγήσεις. Διεσπαρμένα δείγματα κειμένων ύμνων από τους πρώτους αιώνες του ελληνικού χριστιανισμού διατηρούνται μέχρι σήμερα. Μερικά από αυτά υιοθετούν τα μετρικά σχήματα της κλασσικής ελληνικής ποίησης αλλά η αλλαγή της προφοράς είχε καταστήσει εκείνα τα μέτρα κατά μεγάλο μέρος χωρίς έννοια, και, εκτός από τη λήψη ως πρότυπου των κλασσικών φορμών, οι βυζαντινοί ύμνοι των επόμενων αιώνων είναι πεζή ποίηση, στίχοι δίχως ρίμα και τονικό πρότυπο.

Ο κοινός όρος για έναν σύντομο ύμνο μιας στροφής, ή μιας σειράς στροφών, είναι Τροπάριο (αυτό μπορεί να φέρει την περαιτέρω συνεκδοχή ενός ύμνου που παρεμβάλλεται μεταξύ στίχων του ψαλτηρίου). Ένα γνωστό παράδειγμα, του οποίου η ύπαρξη βεβαιώνεται από τον 4ο αιώνα, είναι ο εσπερινός ύμνος, "Φως Ιλαρόν", ή ακόμα ο, αποδιδόμενος στον Αυτοκράτορα Ιουστινιανό Α' (527 - 565) ύμνος "Ο Μονογενής Υιός και Λόγος του Θεού", που ακούγεται σήμερα στην εισαγωγή της Θείας Λειτουργίας. Ίσως η γνωστότερη συλλογή τροπαρίων με καλλιτεχνική πατρότητα είναι αυτή του μοναχού Αυξεντίου (πρώτο μισό του 5ου αιώνα), που ιστορείται στη βιογραφία του αλλά δε διατηρήθηκε σε καμία μεταγενέστερη τυπική λατρευτική διάταξη.

### **2.1.2 Ακμή της Βυζαντινής Περιόδου**

Ήταν, στην πραγματικότητα, ο μοναστικός πληθυσμός που παρήγαγε τους πρώτους και εξαιρετικότερους υμνογράφους και μουσικούς — Ρωμανός ο Μελωδός, Ιωάννης Δαμασκηνός, Ανδρέας ο Κρης, και Θεόδωρος Στουδίτης. Και ήταν ο μοναστικός πληθυσμός που παρήγαγε επίσης τους εφευρέτες μιας περίπλοκης μουσικής σημειογραφίας που επέτρεψε στους γραφείς να συντηρήσουν σε χειρόγραφους κώδικες, τις μουσικές πρακτικές της μεσαιωνικής Ανατολής. Υπήρξε, φυσικά, κάποια πρώιμη μοναστική αντίθεση στη μουσική. Αλλά αυτό δεν σημαίνει ότι οι μοναχοί απέρριπταν το μέλος. Η απόρριψή τους ενέκειτο στην κοσμική μουσική, στη μουσική επίδειξη και τα εξωεκκλησιαστικά ασματικά μέλη και επωδούς.

Μιλώντας γενικά, εντούτοις, υπάρχει μια ευδιάκριτη αδιαφορία για την εκκλησιαστική μουσική στη βυζαντινή λογοτεχνία πριν το 10ο αιώνα. Υπήρχαν ελάχιστοι που παρατηρούσαν και κατέγραφαν αυτό το θέμα. Εκείνη τη περίοδο κανένας δεν πήγαινε στην εκκλησία με το σκεπτικό να ακούσει μια καλή χορωδία ή τις πρόσφατες μουσικές διασκευές ύμνων και ψαλμών. Ο πιστός γνώριζε ότι ο ίδιος θα περιλαμβανόταν σε κάποιο είδος μουσικής δραστηριότητας — ένα είδος σχετικά απλό και ως προς την εκμάθηση, εύκολο να ακολουθηθεί και να κατευθυνθεί. Δεν υπήρξε καμία προσπάθεια να συμβληθεί η μουσική με κάποια ιδιαίτερη ατμόσφαιρα ή θεατρικότητα.

Δύο έννοιες πρέπει να γίνουν κατανοητές για να εκτιμήσουμε πλήρως τη λειτουργία της μουσικής στη βυζαντινή λατρεία. Ο πρώτος, που διατήρησε τη βάση στην ελληνική θεολογική και μυστική σκέψη μέχρι την πτώση της αυτοκρατορίας, ήταν η πίστη στην αγγελική μετάδοση του ιερού άσματος: η υπόθεση ότι η εκκλησία ένωσε τους ανθρώπους σε κοινή προσευχή με τα αγγελικά τάγματα. Αυτή η αντίληψη είναι βέβαια παλαιότερη από την αφήγηση της Αποκάλυψης (Αποκ. 4.8-11), για το υμνητικό λειτούργημα των αγγέλων όπως συλλαμβάνεται στην Παλαιά Διαθήκη και τονίστηκε έντονα από τον Ησαΐα (6.14) και τον Ιεζεκιήλ (3.12). Σημαντικότερο είναι το γεγονός, που περιγράφεται στην Έξοδο 25, ότι το πρότυπο για την επίγεια λατρεία του Ισραήλ προήλθε από τον ουρανό. Η νύξη διαιωνίζεται στις γραφές των πρώτων Πατέρων, όπως ο Κλήμης Ρώμης, Ιουστίνος, Ιγνάτιος Αντιοχείας, Αθηναγόρας και Ψευδοδιονύσιος. Αναγνωρίζεται αργότερα στις λειτουργικές πραγματείες του Νικολάου Καβάσιλα και του Συμεών Αρχιεπισκόπου Θεσσαλονίκης.

Η επίδραση που αυτή η έννοια είχε στην εκκλησιαστική μουσική ήταν τριπλή: κατ' αρχάς, αναπαρήγαγε μια ιδιαίτερα συντηρητική στάση απέναντι στη μουσική σύνθεση αφετέρου, σταθεροποίησε τη μελωδική παράδοση ορισμένων ύμνων και τρίτον, συνέχισε, για κάποιο διάστημα, την ανωνυμία του συνθέτη. Γιατί εάν ένα άσμα είναι θεϊκής προέλευσης, κατόπιν η αναγνώριση που λαμβάνει ο «μεταφορέας» μελλοντικά οφείλει να είναι ελάχιστη. Αυτό ισχύει ιδιαίτερα όταν εξετάζουμε τους ύμνους που φέρονταν να ψάλησαν για πρώτη φορά από αγγελικές χορωδίες - όπως το Αμήν, Αλληλουία, το Τρισάγιο και η Δοξολογία. Συνεπώς, μέχρι τους χρόνους των Παλαιολόγων, ήταν ασύλληπτο για έναν συνθέτη να τοποθετήσει το όνομά του κάτω από ένα χειρόγραφο μουσικό κείμενο.

Οι ιδέες της πρωτοτυπίας και της ελεύθερης σύνθεσης παρόμοιες με εκείνες που εμφανίστηκαν στην πιο πρόσφατη μουσική πιθανώς δεν υπήρξαν ποτέ στους πρώιμους βυζαντινούς χρόνους. Η ίδια η έννοια της χρησιμοποίησης παραδοσιακών μεθόδων (ή μελωδικών τύπων) ως συνθετική τεχνική παρουσιάζει αρχαϊζουσα αντίληψη στο λειτουργικό άσμα, και είναι μερικώς αντίθετη της ελεύθερης, αυθεντικής δημιουργίας. Μοιάζει καταφανές ότι τα άσματα του βυζαντινού ρεπερτορίου που βρίσκεται στα μουσικά

χειρόγραφα από το δέκατο αιώνα μέχρι την εποχή της τέταρτης σταυροφορίας (1204 - 1261), αντιπροσωπεύουν το τελικό και μόνο επιζών στάδιο μίας εξέλιξης, οι αρχές της οποίας επιστρέφουν τουλάχιστον στον έκτο αιώνα και ενδεχομένως ακόμη και στο άσμα της Συναγωγής. Ποιες ακριβώς αλλαγές πραγματοποιήθηκαν στη μουσική κατά τη διάρκεια του διαμορφωτικού σταδίου είναι δύσκολο να ειπωθούν αλλά ορισμένα άσματα χρησιμοποιούμενα ακόμη και σήμερα ίσως να ρίχνουν φως στο θέμα. Αυτά περιλαμβάνουν τύπους απαγγελίας, μελωδικούς τύπους, και τις τυποποιημένες φράσεις που είναι εμφανείς στη λαϊκή μουσική και την παραδοσιακή μουσική διαφόρων πολιτισμών της ανατολής, συμπεριλαμβανομένης της μουσικής των Εβραίων.

Η δεύτερη, λιγότερο ισχυρή, αντίληψη ήταν αυτή της "κοινωνίας". Ο λόγος που φάνηκε πιο εύαλωτη ήταν ότι, μετά τον τέταρτο αιώνα, όταν αναλύθηκε και εισήχθη σε ένα θεολογικό σύστημα, ο δεσμός και η "ενότητα" του ιερατείου και των πιστών στη θεία λατρεία έγινε λιγότερο ισχυρός. Είναι, εντούτοις, μια από τις βασικές ιδέες κατανόησης διαφόρων δεδομένων για τα οποία έχουμε θεσπίσει σήμερα διαφορετικά ονόματα. Όσον αφορά τη μουσική απόδοση, η έννοια της "κοινωνίας" μπορεί να εφαρμοστεί στην αρχική χρήση της λέξης "χορός". Αναφέρονταν, όχι σε μια χωριστή ομάδα μέσα στην κοινότητα που επιφορτίστηκε με την ευθύνη της μουσικής, αλλά στην κοινότητα συνολικά.

Χαρακτηριστικό γνώρισμα της τελετής της Θείας Λειτουργίας ήταν η ενεργός δράση των πιστών στην απόδοσή, ιδιαίτερα στην απαγγελία των ύμνων, των αποκρίσεων και των ψαλμών. Οι όροι χορός, κοινωνία, Εκκλησία χρησιμοποιούνταν ως ταυτόσημοι στην πρώιμη βυζαντινή Εκκλησία. Στους ψαλμούς 149 και 150, οι Εβδομήκοντα μετέφρασαν την εβραϊκή λέξη *machol* στα ελληνικά ως χορός. Κατά συνέπεια, η πρώιμη εκκλησία δανείστηκε αυτήν την λέξη από την κλασσική αρχαιότητα ως προσδιορισμό για την κοινότητα, τη λατρεία και τον αίνο στον ουρανό και τη γη. Σε λίγο, εντούτοις, μια κληρικαλιστική τάση άρχισε να φανερώνεται στη γλωσσική πρακτική, ιδιαίτερα μετά τη Σύνοδο της Λαοδικείας, της οποίας ο δέκατος πέμπτος κανόνας επέτρεπε μόνο στους κανονικούς ψάλτες να συμμετέχουν σε χορωδίες εκκλησιών. Η λέξη "χορός" πλέον αναφέρονταν στην ειδική τάξη του κατώτερου κλήρου που έψαλλε στους ναούς- ακριβώς όπως, μιλώντας αρχιτεκτονικά, η χορωδία έγινε μια ιδιαίτερη περιοχή κοντά στο Ιερό - και ο χορός έγινε τελικά αντίστοιχος με τη λέξη κλήρος.

Η ανάπτυξη υμνογραφικών μορφών μεγάλης κλίμακας αρχίζει τον πέμπτο αιώνα με την άνοδο του Κοντακίου, ενός εκτενούς και περίτεχνου μετρικού κηρύγματος, προερχόμενου κατά πάσα πιθανότητα από την περιοχή της Συρίας, το οποίο βρίσκεται το αποκορύφωμά του στο έργο του Αγίου Ρωμανού του Μελωδού (έκτος αιώνας). Αυτό το ποιητικό κήρυγμα, που παραφράζει συνήθως κάποια βιβλική αφήγηση, περιλαμβάνει περίπου 20 έως 30 στροφές και

ψάλλεται κατά τη διάρκεια του Όρθρου σε απλό και άμεσο συλλαβικό ύφος (μία νότα ανά συλλαβή). Οι προγενέστερες μουσικές εκδοχές, εντούτοις, είναι "μελισματικές" (δηλαδή πολλές νότες ανά συλλαβή του κειμένου), και ανήκουν στην περίοδο του ένατου αιώνα και μεταγενέστερα όταν μειώθηκαν τα κοντάκια στο προοίμιο και τον πρώτο οίκο. Στο δεύτερο μισό του έβδομου αιώνα, το κοντάκιο αντικαταστάθηκε από έναν νέο τύπο ύμνου, τον Κανόνα, που εγκαινιάστηκε από τον Άγιο Ανδρέα Κρήτης(660 – 740 περίπου) και αναπτύχθηκε από τους Αγίους Ιωάννη Δαμασκηνό και Κοσμά Ιεροσολυμίτη (και οι δύο τον όγδοο αιώνα).

Ουσιαστικά, ο Κανόνας είναι ένας σύνθετος ύμνος που περιλαμβάνει εννέα ωδές, οι οποίες αρχικά αντιστοιχούσαν στις εννέα βιβλικές ωδές και συνδέονταν με αυτές δια μέσου της παράλληλης ποιητικής αναφοράς ή του Γραφικού εδαφίου.

Οι εννέα ωδές είναι:

- 1-2 Οι δύο ύμνοι του Μωυσέως (Εξοδος 15.1-19 και Δευτερονόμιο 32.1-43);
- 3-7 Οι προσευχές της Άννας, του Αβακούμ του Ησαΐου, του Ιωνά και των Τριών Παίδων (Α' Βασιλειών 2.1-10, Αβακούμ 3.1-19, Ησαΐας 26.9-20, Ιωνάς 2.3-10, Δανιήλ 3.26-56)
- 8. Ο ύμνος των Τριών Παίδων (Δανιήλ 3.57-88)
- 9. Ο ύμνος της Θεοτόκου και η ωδή του Ζαχαρίου (Λουκάς 1.46-55 και 68-79).

Οι Κανόνες έχουν έναν ειρμό, μια στροφή πρότυπο για κάθε ωδή, ακολουθούμενο από τρία, τέσσερα ή περισσότερο τροπάρια που είναι οι ακριβείς μετρικές αναπαραγωγές του ειρμού, με αυτόν τον τρόπο εφαρμόζεται το ίδιο μέλος σε όλα τα τροπάρια εξίσου καλά. Οι εννέα ειρμοί, ωστόσο, είναι μετρικά ανόμοιοι, συνεπώς, ένας ολόκληρος Κανόνας περιλαμβάνει εννέα ανεξάρτητες μελωδίες (συνηθέστερα όμως οκτώ, καθώς η δεύτερη ωδή ψάλλεται μόνον κατά την περίοδο της Μεγάλης Τεσσαρακοστής, λόγω του αυστηρά προτρεπτικού σε μετάνοια ύφους της), οι οποίες ενώνονται μουσικά με τον ίδιο τρόπο και νοηματικά από τις αναφορές στο γενικό θέμα της λειτουργικής περίπτωσης, και μερικές φορές από την ακροστιχίδα. Οι ειρμοί με συλλαβικό ύφος συγκεντρώθηκαν στο Ειρμολόγιο, έναν ογκώδη τόμο που εμφανίστηκε αρχικά στο μέσο του δέκατου αιώνα και περιέχει πέραν των χιλίων πρότυπων τροπαρίων ταξινομημένων κατά την Οκτώηχο.

Ένα άλλο είδος ύμνου, σημαντικό και για το σύνολό του και για την ποικιλία της λειτουργικής χρήσης του, είναι το Στιχηρό. Τα εορταστικά στιχηρά, που συνοδεύουν και τους πάγιους ψαλμούς στην αρχή και στο τέλος του Εσπερινού και των Αίνων του Όρθρου,

υφίστανται για όλες τις εορτάσιμες ημέρες του έτους, τις Κυριακές και τις καθημερινές, ταξινομημένα ώστε να καλύπτουν έναν επαναλαμβανόμενο κύκλο οκτώ εβδομάδων με βάση τους Ήχους αρχίζοντας από την ημέρα του Πάσχα. Οι μελωδίες τους που βρίσκονται στο Στιχηράριον, είναι αρκετά πιο επιμελημένες και ποικίλες απ' ότι στην παράδοση του Ειρμολογίου.

### ***2.1.3 Ύστερη Βυζαντινή και Μεταβυζαντινή Περίοδος***

Με το τέλος της δημιουργικής ποιητικής σύνθεσης, το βυζαντινό άσμα εισήλθε στην τελική περιόδό του, που αφιερώθηκε κατά ένα μεγάλο μέρος στην παραγωγή των πιο επιμελημένων μουσικών διασκευών των παραδοσιακών κειμένων: είτε καλλωπισμοί των προηγούμενων απλούστερων μελωδιών, είτε διατήρηση της αρχικής μουσικής με αλλαγή του ύφους σε πιο εκλεπτυσμένο και διακοσμημένο. Αυτή ήταν η εργασία των αποκαλούμενων Μαϊστόρων, εκ των οποίων ο πλέον διάσημος υπήρξε ο Άγιος Ιωάννης ο Κουκουζέλης, συγκρινόμενος στη βυζαντινή λογοτεχνία με τον Άγιο Ιωάννη το Δαμασκηνό, ως καινοτόμος στην ανάπτυξη του άσματος. Με τον πολλαπλασιασμό των νέων διατάξεων και την επεξεργασία των παλαιών που συνεχίστηκε στους αιώνες μετά την πτώση της Κωνσταντινούπολης, μέχρι και το τέλος του δέκατου όγδοου αιώνα το αρχικό ρεπερτόριο των μεσαιωνικών μουσικών χειρογράφων είχε υποκατασταθεί σε μεγάλο μέρος από πιο πρόσφατες συνθέσεις, και ακόμη και το βασικό πρότυπο σύστημα είχε υποβληθεί σε βαθιά τροποποίηση.

Ο Χρυσάνθος Μαδύτου, ο Γρηγόριος Λευίτης και ο Χουρμούζιος Χαρτοφύλακας ήταν αρμόδιοι για την αναγκαία μεταρρύθμιση της σημειογραφίας της ελληνικής εκκλησιαστικής μουσικής. Ουσιαστικά, αυτή η εργασία απέδωσε την απλοποίηση των βυζαντινών μουσικών σύμβολων, τα οποία, από τις αρχές του 19ου αιώνα, είχαν γίνει τόσο σύνθετα και τεχνικά ώστε μόνο πολύ καλά καταρτισμένοι ψάλτες ήταν σε θέση να τα ερμηνεύσουν σωστά. Παρά τα πολυάριθμα μειονεκτήματά του, το έργο των τριών μεταρρυθμιστών αποτελεί ορόσημο στη μουσική ιστορία της ελληνορθόδοξης εκκλησίας, δεδομένου ότι εισήγαγαν το σύστημα της νεο-βυζαντινής μουσικής επάνω στο οποίο βασίζεται το σημερινό ελληνορθόδοξο εκκλησιαστικό μέλος.

## ***2.2 Βασικά Στοιχεία Θεωρίας Βυζαντινής Μουσικής***

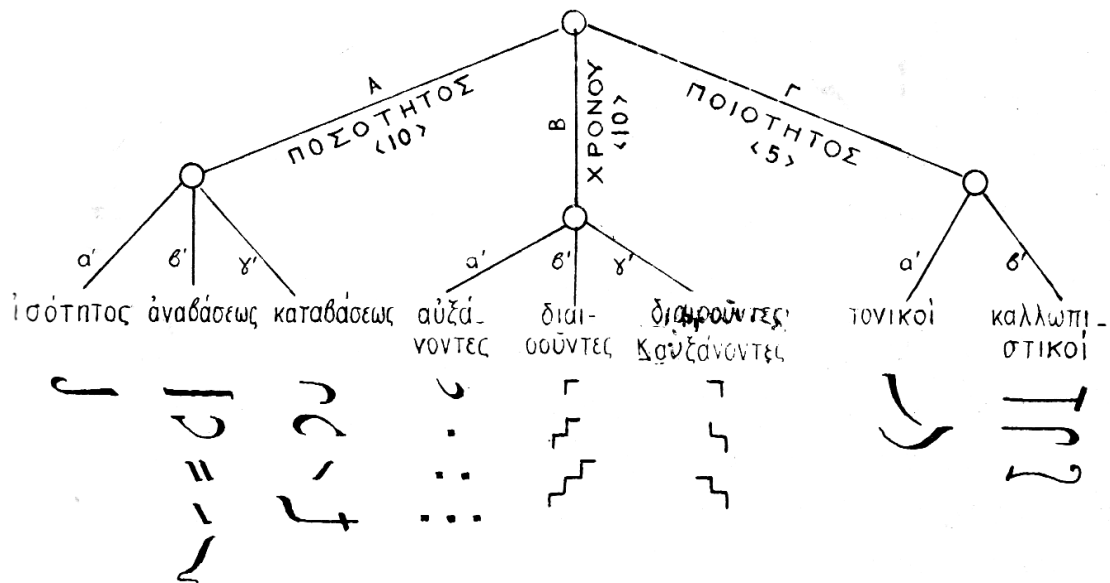
### ***2.2.1 Φθόγγοι και Κλίμακες***

Οι φθόγγοι ή τόνοι στη βυζαντινή μουσική διακρίνονται σε επτά και ονομάζονται: Πα, Βου, Γα, Δι, Κε, Ζω και Νη. Αυτοί οι τόνοι εκφωνούνται κατέχοντας ο καθένας μία βαθμίδα.

Ανεβαίνοντας από την πρώτη βαθμίδα έως την έβδομη (άνοδος ή επίτασις ή οξύτης) και κατεβαίνοντας από την έβδομη μέχρι την πρώτη (κάθοδος ή άνεσις ή βαρύτης) σχηματίζουμε μία κλίμακα. Η Βυζαντινή μουσική μεταχειρίζεται τρεις τέτοιες κλίμακες από τις οποίες η πρώτη ως κατώτατη ονομάζεται Υπάτη ή Βαρεία διά πασών, η δεύτερη Μέση διά πασών και η τρίτη ως ανώτατη Νήτη ή Οξεία διά πασών. Η χρήση τους ακολουθεί τρεις τρόπους α) σε συνεχή ανάβαση ή κατάβαση, β) σε υπερβατή ανάβαση ή κατάβαση και γ) σε εναλλασσόμενη συνεχή και υπερβατή ανάβαση ή κατάβαση. Όταν ένας τόνος χωρίζεται σε δύο άνισα μέρη και χρησιμοποιείται το ένα από αυτά τα διαστήματα, αυτό το διάστημα ονομάζεται ημιτόνιο.

### 2.2.2 Μουσικοί Χαρακτήρες

Προκειμένου να εγγραφεί και να μεταδοθεί η ποσότητα της μελωδίας στη βυζαντινή μουσική δημιουργήθηκε ένα ιδιαίτερο σύστημα δέκα χαρακτήρων. Από αυτούς οι έξι είναι ανιόντες και οι τέσσερις κατιόντες. Τα ονόματά τους είναι Ίσον, Ολίγον, Πεταστή, Κεντήματα, Κέντημα, Υψηλή, Απόστροφος, Ελαφρόν, Υπορροή, Χαμηλή. Οι δέκα χαρακτήρες της ποσότητας διαιρούνται σε τρεις τάξεις, στα Σώματα, τα Πνεύματα και τους Ουδέτερους. Άλλες ονομασίες των χαρακτήρων είναι μουσικά γράμματα και φθογγόσημα. Αν και εκφράζουν την ανάβαση και την κατάβαση των τόνων δεν έχει έκαστο ξεχωριστό τόνο αλλά τους ορίζουν όταν προηγείται κάποιος τόνος ως βάση. Αν δεν υπάρχει βάση δεν μπορούν να εκφράσουν κάποιο μουσικό νόημα. Όταν συμπλέκονται μεταξύ τους εκφράζουν όλους τους μουσικούς τόνους όλων των κλιμάκων.



Σχήμα 2.1

Οι χαρακτήρες της Βυζαντινής Μουσικής

### 2.2.3 Υποστατικά Σημεία

Τα υποστατικά σημεία, τα οποία διαιρούνται σε έγχρονα και άχρονα, είναι έντεκα. Κλάσμα, Απλή, Γοργόν, Αργόν, Βαρεία, Ομαλόν, Αντικένωμα, Ψηφιστόν, Έτερον ή Σύνδεσμος, Ενδόφωνον και Σταυρός. Τα έγχρονα ή έγχρονες υποστάσεις ονομάστηκαν έτσι επειδή φανερώνουν χρονική ποσότητα ενώ τα άχρονα ή άχρονες υποστάσεις δε δαπανούν χρόνο.

### 2.2.4 Μουσικά Γένη

Γένος στη βυζαντινή μουσική ονομάζεται η διαίρεση της τετράχορδης τάξης των φθόγγων κατά το Διατεσσάρων σύστημα. Υπάρχουν τρία γένη : Διατονικό, Χρωματικό και Εναρμόνιο.

#### Διατονικό γένος

Η κλίμακα του Διατονικού γένους σύγκεται από δύο τετράχορδα χωρισμένα όμοια. Η ομοιότητά τους υφίσταται από τα αναλόγως ίσα διαστήματα των τόνων που περιέχουν.

#### Χρωματικό γένος

Χρώμα λέγεται στη μουσική εκείνο το οποίο μπορεί να βάψει την ποιότητα που παράγεται από τους φθόγγους της διατονικής κλίμακας και να παράσχει ποιότητα που έχει διαφορετικό ύφος. Αυτό μπορούν να το κάνουν οι υφέσεις και οι δίεςεις. Χρωματικό γένος είναι λοιπόν εκείνο στου οποίου την κλίμακα βρίσκονται ημίτονα είτε σε ύφεση είτε σε δίεση ή και σε ύφεση και σε δίεση.

#### Εναρμόνιο γένος

Εναρμόνιο ονομάζεται το γένος το οποίο έχει στην κλίμακά του τεταρτημόριο του μείζονος τόνου. Αυτό το διάστημα λέγεται ύφεση ή δίεση εναρμόνιος.

Τα Εκκλησιαστικά μέλη διακρίνονται επίσης σε τρεις κατηγορίες: στα *ειρμολογικά*, στα *στιχηραρικά* και στα *παπαδικά* είδη.

- Το **ειρμολογικό μέλος**, όταν κάθε συλλαβή του κειμένου αντιστοιχεί σε έναν φθόγγο, λέγεται **σύντομο**, ενώ όταν κάθε συλλαβή του κειμένου αντιστοιχεί σε διπλάσιο συνήθως χρόνο, λέγεται αργό **ειρμολογικό**.
- Το **στιχηραρικό μέλος** είναι μελισματικότερο του πρώτου και διακρίνεται και αυτό σε σύντομο και αργό **στιχηραρικό**.

- Το **παπαδική** είναι **μέλος** κατ' εξοχήν αργό, αυτό σημαίνει ότι σε κάθε συλλαβή αντιστοιχούν πολλοί μουσικοί φθόγγοι. Στο είδος αυτό περιλαμβάνονται και τα λεγόμενα «μαθήματα» (μέλη) της ελευθέρας μελοποιίας.

### **2.2.5 Ήχοι**

Η Βυζαντινή Μουσική ακολουθεί παραλλαγμένη σε κάποια σημεία την Πυθαγορική Οκτάχορδο. Οι οκτώ ήχοι ή τρόποι της είναι : *Πρώτος, Δεύτερος, Τρίτος, Τέταρτος, Πλάγιος του Πρώτου, Πλάγιος του Δευτέρου, Βαρύς και Πλάγιος του Τετάρτου*. Οι ήχοι Πρώτος, Τέταρτος, Πλάγιος του Πρώτου και Πλάγιος του Τετάρτου ανήκουν στο Διατονικό γένος. Οι ήχοι Δεύτερος και Πλάγιος του Δευτέρου ανήκουν στο Χρωματικό γένος και οι ήχοι Τρίτος και Βαρύς στο Εναρμόνιο.

## **2.3 Συγγραφή Βυζαντινής Μουσικής**

Εδώ παραθέτουμε τις επιλογές που υπάρχουν αυτή τη στιγμή για κάποιον ο οποίος επιθυμεί να γράψει σε ηλεκτρονική μορφή κάποιο βυζαντινό μέλος.

### **2.3.1 Το πακέτο EZ**

Το πακέτο συγγραφής Βυζαντινής Μουσικής EZ δημιουργήθηκε στο ελληνικό μοναστήρι του Αγίου Αντωνίου στην Αριζόνα το 2005. Αποτελεί βελτίωση της γραμματοσειράς Psaltica, που δημιουργήθηκε από την Elie Daoun. Περιλαμβάνει μια σειρά μακροεντολών που το καθιστούν ιδιαίτερα εύχρηστο και προσδίδουν επαγγελματική ποιότητα στο αποτέλεσμα. Οι μακροεντολές αυτές επανατοποθετούν με ακρίβεια τους χαρακτήρες για άψογη εμφάνιση και αλλάζουν τα χρώματα κάποιων συμβόλων.

### **2.3.2 Το Πακέτο BZ**

Στο πακέτο BZ (το οποίο αναπτύχθηκε από τον Στέφανο Σουλδάτο) έχουν γίνει αρκετές βελτιώσεις σε σχέση με το EZ, στο οποίο βασίστηκε. Ανακατατάχθηκαν οι αντιστοιχίσεις των συμβόλων με το πληκτρολόγιο για ευκολία στον εντοπισμό των σημαδιών, στην πληκτρολόγηση και στην απομνημόνευση. Αναπτύχθηκαν επιπλέον μακροεντολές που αυτοματοποιούν χρονοβόρες και επιρρεπείς σε λάθη διαδικασίες, όπως η εισαγωγή κενών όπου ακριβώς χρειάζεται ανάμεσα στα σημάδια. Δημιουργήθηκε μια γραμμή εργαλείων με κουμπιά που βοηθούν το χρήστη στο χειρισμό του πακέτου. Προστέθηκαν ορισμένα σύμβολα όπως τα διπλά ισοκρατήματα.



A) **Μελέτη και βελτίωση θέσεων συμβόλων:** Κατά τη δημιουργία του BZ δόθηκε μεγάλη σημασία στη θέση των συμβόλων έτσι ώστε η συγγραφή να διευκολύνεται και να επιταχύνεται. Μετακινήθηκαν τα πιο συνηθισμένα σύμβολα στην πρώτη από τις πέντε γραμματοσειρές (BZ Byzantina). Μελετήθηκαν οι θέσεις των συμβόλων σε κάθε γραμματοσειρά, ώστε όχι μόνο να είναι εύκολη η απομνημόνευση, αλλά να είναι και ξεκούραστη για τους καρπούς η πληκτρολόγηση. Συγκεκριμένα, τα συνήθη σύμβολα μεταφέρθηκαν στη μεσαία γραμμή του πληκτρολογίου για να απαιτούνται μικρότερες κινήσεις κατά την πληκτρολόγηση, ιδιαίτερα για όσους χρησιμοποιούν τυφλό σύστημα πληκτρολόγησης. Τέλος, πολλά από τα σύμβολα ομαδοποιήθηκαν και αντιστοιχήθηκαν σε γειτονικές θέσεις του πληκτρολογίου για να είναι πιο εύκολος ο εντοπισμός και η απομνημόνευση.

B) **Προσθήκη λειτουργιών:** Στο πακέτο BZ έχουν προστεθεί μακροεντολές που σχετίζονται με τη μορφοποίηση του μουσικού και του ποιητικού κειμένου. Λειτουργίες σχετικές με το μέγεθος γραμματοσειράς, τα διάκενα και τη διαμόρφωση του πλαισίου κειμένου δεν απασχολούν πλέον το χρήστη, αλλά γίνονται αυτόματα με χρήση μακροεντολών. Πλέον, δεν απαιτείται από το χρήστη να αφήνει κενά μεταξύ των συμβόλων, καθώς κενά εισάγονται αυτόματα μόνο εκεί που χρειάζονται. Οι μακροεντολές PolishNotes και MakeNotesRed βελτιώθηκαν ώστε να λειτουργούν ανεξάρτητα και να μπορούν να κληθούν περισσότερες από μια φορές. Προστέθηκαν οι μακροεντολές BZ\_MakeMartyriesRed και MakeIsonRed που χρωματίζουν κόκκινα μόνο τις μαρτυρίες και το ίσον αντίστοιχα. Επιπλέον, έχει δημιουργηθεί μια πλήρης γραμμή εργαλείων που διευκολύνει τόσο τη συγγραφή όσο και τη χρήση των μακροεντολών. Τέλος, έχουν προστεθεί μακροεντολές που μετατρέπουν κείμενα γραμμένα με άλλες γραμματοσειρές (ED-97, EZ) σε BZ, ώστε να μπορεί κάποιος εύκολα να χρησιμοποιήσει το πακέτο BZ.

Γ) **Προσθήκη συμβόλων:** Στο BZ έχουν προστεθεί κάποια σύμβολα που δεν υπήρχαν στο EZ. Συγκεκριμένα, έχουν προστεθεί διπλά ισοκρατήματα, π.χ. «(Π-K)», το σύμβολο του Α΄ δίφωνου χρωματικού ήχου (ήχος νάος), μια πλήρης σειρά από μικρές διαστολές και μέτρα, μια νέα θέση για τον σταυρό (πάνω και ανάμεσα στους φθόγγους). Ειδικά για τα μέτρα 5-6-7-

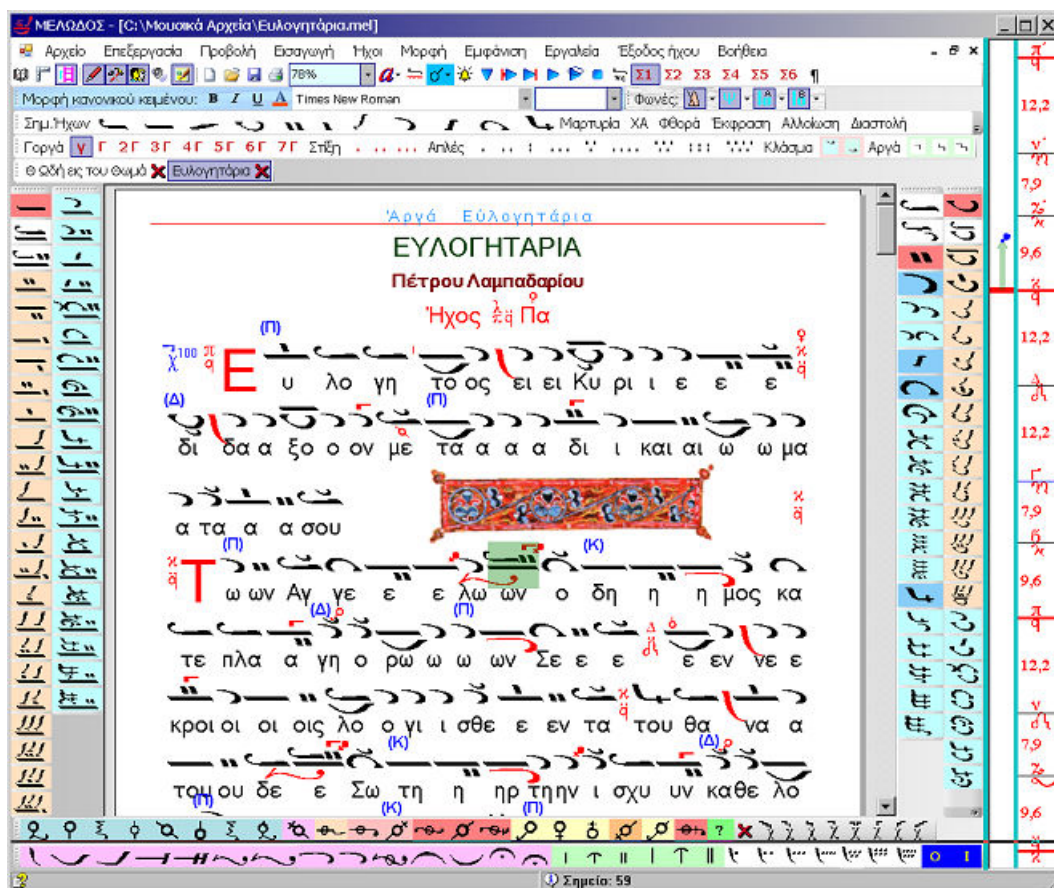


8, πλέον πληκτρολογούνται αμέσως μετά τη διπλή διαστολή, ώστε να στοιχίζονται σύμφωνα με τη διαστολή ανεξάρτητα από το πλάτος του φθόγγου.

Σχήμα 2.2 Η εργαλιογραμμή του BZ

### 2.3.3 Το Πρόγραμμα «Μελωδός»

Το πρόγραμμα Μελωδός είναι ένα ανεξάρτητο πρόγραμμα το οποίο έχει το δικό του επεξεργαστή κειμένου και δεν ενσωματώνεται στο MS Word. Ο χρήστης γράφει το σύμβολα είτε επιλέγοντας τα από το interface είτε πληκτρολογώντας. Έχει επίσης κατορθώσει κάτι μοναδικό ανάμεσα στα άλλα προγράμματα Βυζαντινής μουσικής. Μπορεί κανείς να ακούσει τα κείμενα που γράφει και να τα σώσει μάλιστα και σε αρχεία midi. Έτσι ο χρήστης μπορεί εύκολα να καταλάβει αν έγραψε κάτι λάθος από το άκουσμα. Έχει επίσης πάρα πολλές δυνατότητες για τροποποίηση της εμφάνισης των συμβόλων και του κειμένου.



Σχήμα 2.3 Η διαπροσωπία του προγράμματος «Μελωδός»

### 2.3.4 Βυζαντινογράφος

Ο Βυζαντινογράφος είναι ένα νέο σύστημα γραφής της Βυζαντινής Μουσικής Σημειογραφίας στον Η/Υ. Έχει δημιουργηθεί από τον Δρ. Γκεζερλή Γ. Βελισάριο, στο Πανεπιστήμιο Αθηνών, στο Τμήμα Πληροφορικής και Τηλεπικοινωνιών. Ενσωματώνεται στο περιβάλλον του Microsoft Word και παρέχει μία εργαλειογραμμή με 25 εικονίδια (όπως παρουσιάζεται παρακάτω), τα οποία βοηθούν τον χρήστη να συγγράμει όλα τα βυζαντινά μουσικά σύμβολα της Βυζαντινής Παρασημαντικής, σε οποιοδήποτε μέγεθος, με χρήση

χρωμάτων (3 χρώματα), ενώ μπορεί πολύ εύκολα να γράψει και διαμορφώσει τους δικούς του ψαλμούς και να τους εκτυπώσει στον εκτυπωτή του.

Όλα τα σύμβολα της Βυζαντινής Παρασημαντικής έχουν αντιστοιχηθεί στα πλήκτρα του πληκτρολογίου με έναν βέλτιστο τρόπο, ώστε η συγγραφή να γίνεται με μεγάλη ταχύτητα. Επίσης στην νέα έκδοση έχουν συμπεριληφθεί σύμβολα της παλαιάς παρασημαντικής (π.χ. τρομικό, στρεπτό, ισάκι, λύγισμα, τζάκισμα, πίεσμα, παρακλητική κλπ.), καθώς επίσης και σύμβολα απαραίτητα για την συγγραφή των παραδοσιακών δημοτικών τραγουδιών (π.χ. πεντάγοργο κ.λπ).



Σχήμα 2.4 Η εργαλιογραμμή του «Βυζαντινογράφου»

### 2.3.5 Βυζαντινή Κάλαμος

Εμπνευστής του προγράμματος ο κ. Παναγιώτης Μπακάλης, Επ. Καθηγητής στο Τμήμα Υπολογιστών και Δικτύων του Πανεπιστημίου του Greenwich Λονδίνου.

Πρόκειται για ένα πρόγραμμα αυτοματοποιημένης μεθόδου γραφής και επεξεργασίας κειμένου βυζαντινής μουσικής με ηλεκτρονικό Υπολογιστή. Συνεργάζεται και απεικονίζει κείμενα βυζαντινής μουσικής στον επεξεργαστή κειμένου Word της Microsoft.

Αποτελείται από τρεις πίνακες που περιέχουν εικονίδια με όλους τους χαρακτήρες της Βυζαντινής μουσικής, οι οποίοι εμφανίζονται στην οθόνη μαζί με τον επεξεργαστή κειμένου. Ακόμη, από μία ολοκληρωμένη συμβολοσειρά Βυζαντινών χαρακτήρων και μία Ελληνική πολυτονική γραμματοσειρά (Unicode).

Η Βυζαντινή Κάλαμος χρησιμοποιεί λογισμικό που αυτοματοποιεί τον τρόπο γραφής της μουσικής, καταργώντας την πληκτρολόγηση και την χρονοβόρα απομνημόνευση.

Παρέχει έναν πολύ εύκολο τρόπο συστηματοποιημένης γραφής της μουσικής, δεν είναι απλώς μια συμβολοσειρά. Το λογισμικό της λύνει πολλά προβλήματα που μόνες τους οι συμβολοσειρές με την πληκτρολόγηση δύσκολα τα λύνουν.

Κάνει τέλος εύκολη στον χρήστη την πολλαπλή αλλαγή μουσικών θέσεων στα μουσικά κείμενά του σύμφωνα με τις προτιμήσεις του.

## 2.4 Σύνθεση Βυζαντινής Μουσικής

Το πρόγραμμα μουσικοσυνθέτης πρωτοπορεί καθώς αυτοματοποιεί την διαδικασία της σύνθεσης. Φυσικά μπορεί κάποιος να συνθέσει ένα βυζαντινό μέλος μη αυτόματα. Πέρα όμως από την εμπειρία που θα πρέπει να υπάρχει καθώς και τους δύσκολους κανόνες ορθογραφίας της βυζαντινής μουσικής στις μέρες μας όμως είναι πολύ πιθανό να ξεφύγει από το παραδοσιακό ύφος κυρίως λόγω του επηρεασμού από άλλα στυλ μουσικής. Επιπλέον σε χώρες όπου η Ορθοδοξία εισήλθε σχετικά πρόσφατα δεν υπάρχει και η ανάλογη εμπειρία. Υπάρχει λοιπόν η ανάγκη για ένα σύστημα το οποίο θα μελοποιεί αυτοματοποιώντας τη διαδικασία και επιπλέον θα ακολουθεί το παραδοσιακό ύφος της Βυζαντινής μουσικής. Αυτό είναι ο Μουσικοσυνθέτης.

### 2.4.1 Βάση Μελωδικών Μοτίβων Πατρός Εφραίμ

Ο Μουσικοσυνθέτης χρησιμοποιεί την βάση μελωδικών μοτίβων του πατρός Εφραίμ από την Αριζόνα των Η.Π.Α ο οποίος εξετάζοντας αναλυτικά πάρα πολλά Βυζαντινά κομμάτια συγκέντρωσε σχεδόν όλα τα υπάρχοντα μελωδικά σχήματα της Βυζαντινής μουσικής σε μια βάση δεδομένων. Ως μοτίβο θεωρείται η μελοποίηση μιας λέξης ή φράσης. Τα μοτίβα αυτά είναι κατηγοριοποιημένα ανά ήχο, ανά τονικό ύψος και ανάλογα με τους τονισμούς της λέξης (φράσης). Για παράδειγμα η λέξη ανεδείχθης έχει δύο συλλαβές χωρίς τόνο (α, νε) μετά μία τονισμένη (δείχ) και τέλος μία άτονη (θεις). Άρα έχει τον τονικό κωδικό 0010 (άτονη-άτονη-τονισμένη-άτονη). Αν βρισκόμαστε σε πρώτο ήχο και η προηγούμενη λέξη-φράση τελείωσε σε φθόγγο (ύψος) πα, μία πιθανή μελοποίηση είναι η παρακάτω:

$$\begin{array}{ccccccc} \pi & \text{—} & \text{—} & \text{—} & \text{—} & \text{—} & \pi \\ \eta & 0 & 0 & 1 & 0 & \eta & \eta \end{array}$$

Αυτή η βάση μελωδιών ήταν το πρώτο βοήθημα για τη σύνθεση Βυζαντινής μουσικής. Αξίζει πάντως να σημειωθεί εδώ ότι η βάση του π.Εφραίμ δεν έχει εφαρμογή στα αργά μέλη της Βυζαντινής μουσικής καθώς σε αυτά οι φράσεις διαρκούν πάρα πολύ και οι μελοποιήσεις τους δεν βασίζονται γενικά σε κάποια πρότυπα.

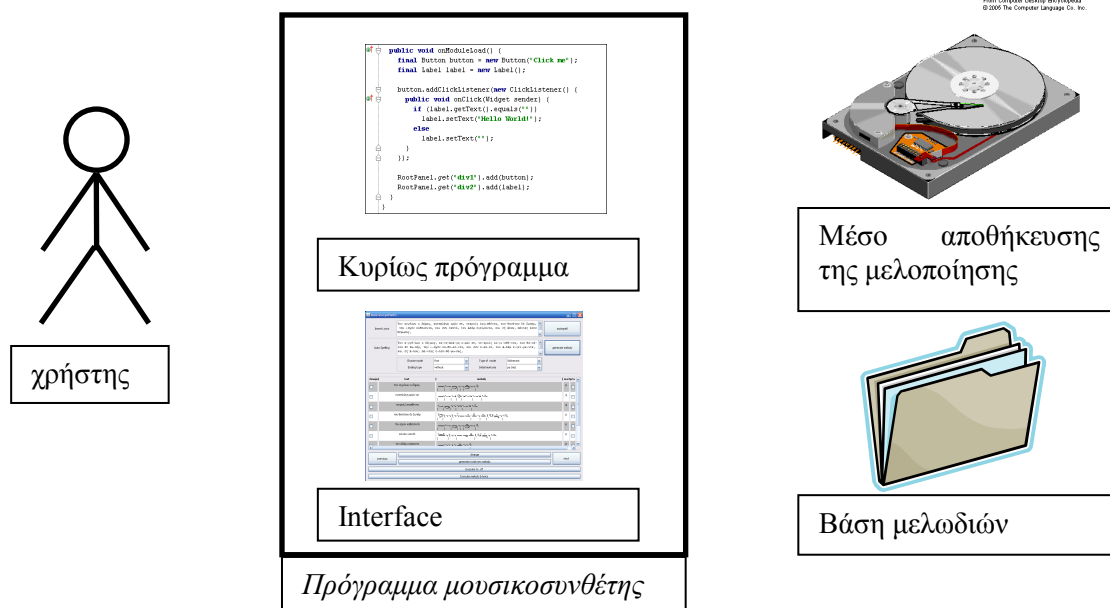
# 3

## Ανάλυση και Σχεδίαση

Στο κεφάλαιο αυτό θα ακολουθήσει η περιγραφή της αρχιτεκτονικής του συστήματος και θα γίνει η ανάλυση απαιτήσεων καθώς και η περιγραφή της λειτουργίας που το κάθε υποσύστημα θα πρέπει να επιτελεί.

### 3.1 Αρχιτεκτονική

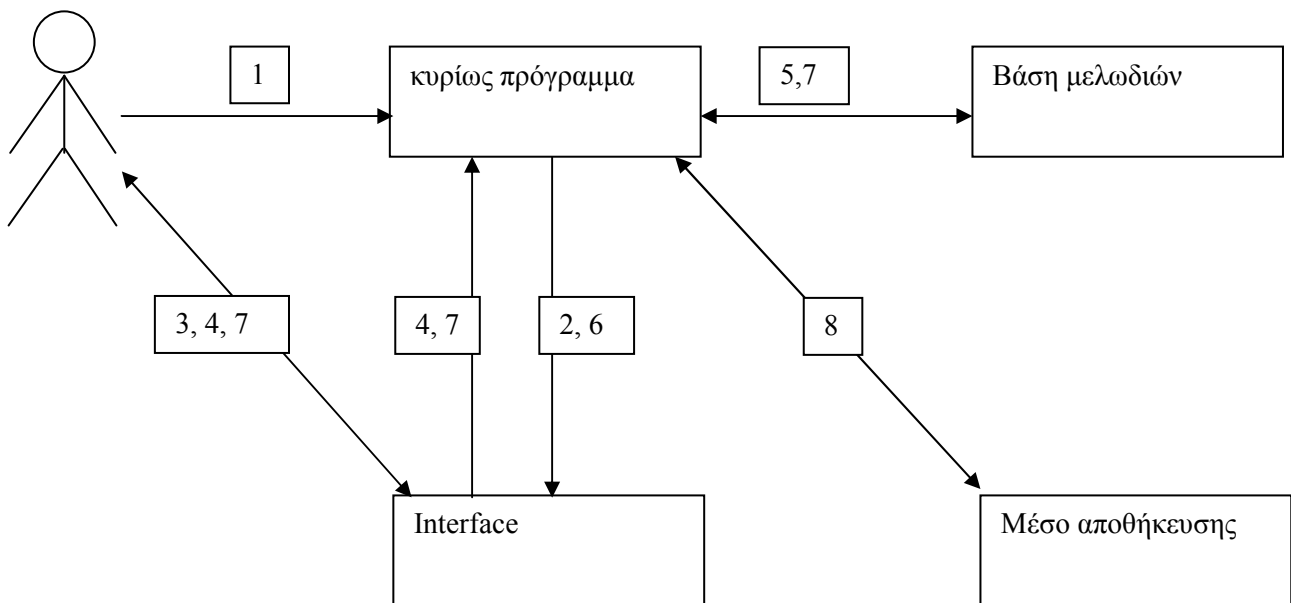
Η αρχιτεκτονική του συστήματος είναι σχετικά απλή. Ο Μουσικοσυνθέτης αποτελείται από τέσσερα υποσυστήματα. Είναι το κυρίως πρόγραμμα, το interface, η βάση με τις έτοιμες μελωδίες και τέλος ο χώρος αποθήκευσης της τελικής μελοποίησης.



Σχήμα 3.1 Η αρχιτεκτονική του συστήματος

Η ακολουθία των διαδικασιών την οποία πρέπει να υλοποιεί το σύστημα είναι η παρακάτω:

- 1) Ο χρήστης τρέχει το πρόγραμμα.
- 2) Εμφανίζεται το interface.
- 3) Ο χρήστης επικοινωνεί με αυτό.
- 4) Κατά καιρούς ζητά μελοποίηση για όλο το κείμενο ή για επιμέρους φράσεις.
- 5) Το πρόγραμμα παίρνει την εντολή για μελοποίηση και αφού κάνει τις απαραίτητες προεργασίες επικοινωνεί με τη βάση και βρίσκει μία μελοποίηση.
- 6) Αυτή η μελοποίηση στέλνεται στο interface για παρουσίαση.
- 7) Ο χρήστης μπορεί να αλλάξει κάποιες επιλογές και να ζητήσει νέου μελοποίηση όλου ή μέρους του κειμένου.
- 8) Μπορεί επίσης να ζητήσει την εξαγωγή της σε έγγραφο ttf. Σε αυτήν την περίπτωση το πρόγραμμα επικοινωνεί με τον σκληρό δίσκο (ή με όποιο άλλο μέσο αποθήκευσης έχει επιλέξει ο χρήστης) για τη δημιουργία του.



Σχήμα 3.2 Το επικοινωνιακό διάγραμμα του συστήματος

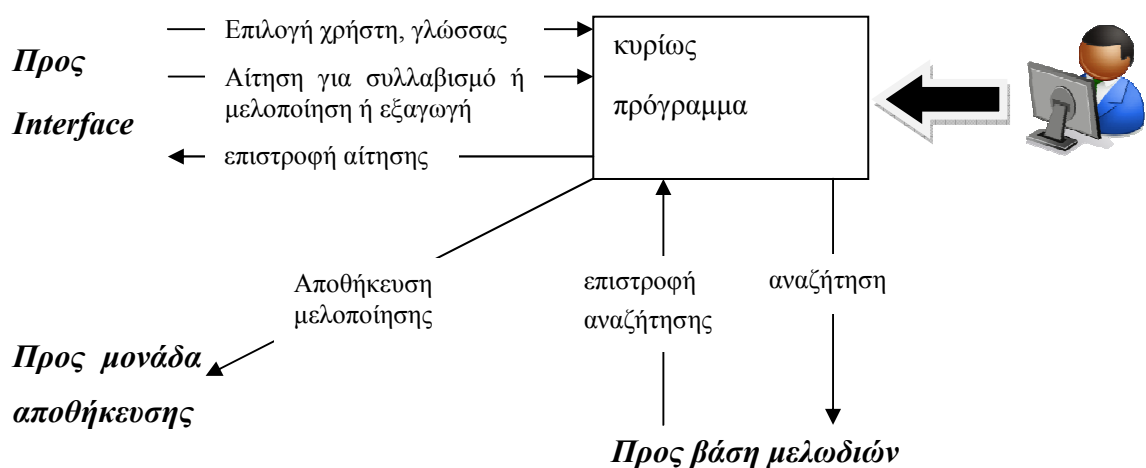
## 3.2 Περιγραφή Λειτουργιών

Εδώ, παρουσιάζονται οι απαιτήσεις και οι λειτουργίες που θα πρέπει αυτό να επιτελεί το κάθε υποσύστημα.

### 3.2.1 Το Πρόγραμμα

- Το πρόγραμμα θα πρέπει να τρέχει αποδοτικά δίνοντας γρήγορα απαντήσεις χωρίς μεγάλες καθυστερήσεις
- Το πρόγραμμα θα πρέπει να είναι συμβατό τουλάχιστον με τα Windows XP
- Θα πρέπει να μην έχει μεγάλες απαιτήσεις υλικού ώστε να μπορεί να εκτελεστεί και σε παλαιότερους υπολογιστές
- Θα πρέπει επίσης να αξιοποιεί τις δυνατότητες της Java ως αντικειμενοστραφούς γλώσσας
- Ως εφαρμογή ανοικτού κώδικα θα πρέπει επίσης να είναι σχεδιασμένο έτσι ώστε να καθιστά εύκολη την τροποποίηση του από άλλους χρήστες

Ακολουθεί το διάγραμμα που παρουσιάζει αναλυτικά τις λειτουργίες του υποσυστήματος και περιγράφεται αναλυτικά.



Σχήμα 3.3 Διάγραμμα ροής δεδομένων του προγράμματος

Όπως φαίνεται το πρόγραμμα από τη στιγμή που ο χρήστης θα το εκκινήσει αλληλεπιδρά με το interface από το οποίο αρχικά γίνεται η επιλογή του χρήστη και της γλώσσας που θα χρησιμοποιηθεί για το κείμενο. Κατόπιν παίρνει αιτήσεις για εύρεση συλλαβισμού του κειμένου που εισήγαγε ο χρήστης ή μελοποίησης του κειμένου (ή κάποιας φράσης του).

Για να στείλει στο interface την απάντηση σε κάποιο από αυτά τα αιτήματα πρέπει το πρόγραμμα να επικοινωνήσει με τα αρχεία της βάσης των μελωδιών. Όταν τελειώσει την επικοινωνία αυτή μπορεί να επιστρέψει τη μελοποίηση η οποία ζητήθηκε.

Μπορεί επίσης το να λάβει αίτηση για εξαγωγή του κειμένου σε αρχείο οπότε και θα επικοινωνήσει με το αποθηκευτικό μέσον (σκληρό δίσκο ή αφαιρούμενο δίσκο κ.τ.λ.) για να γίνουν οι κατάλληλες κινήσεις.

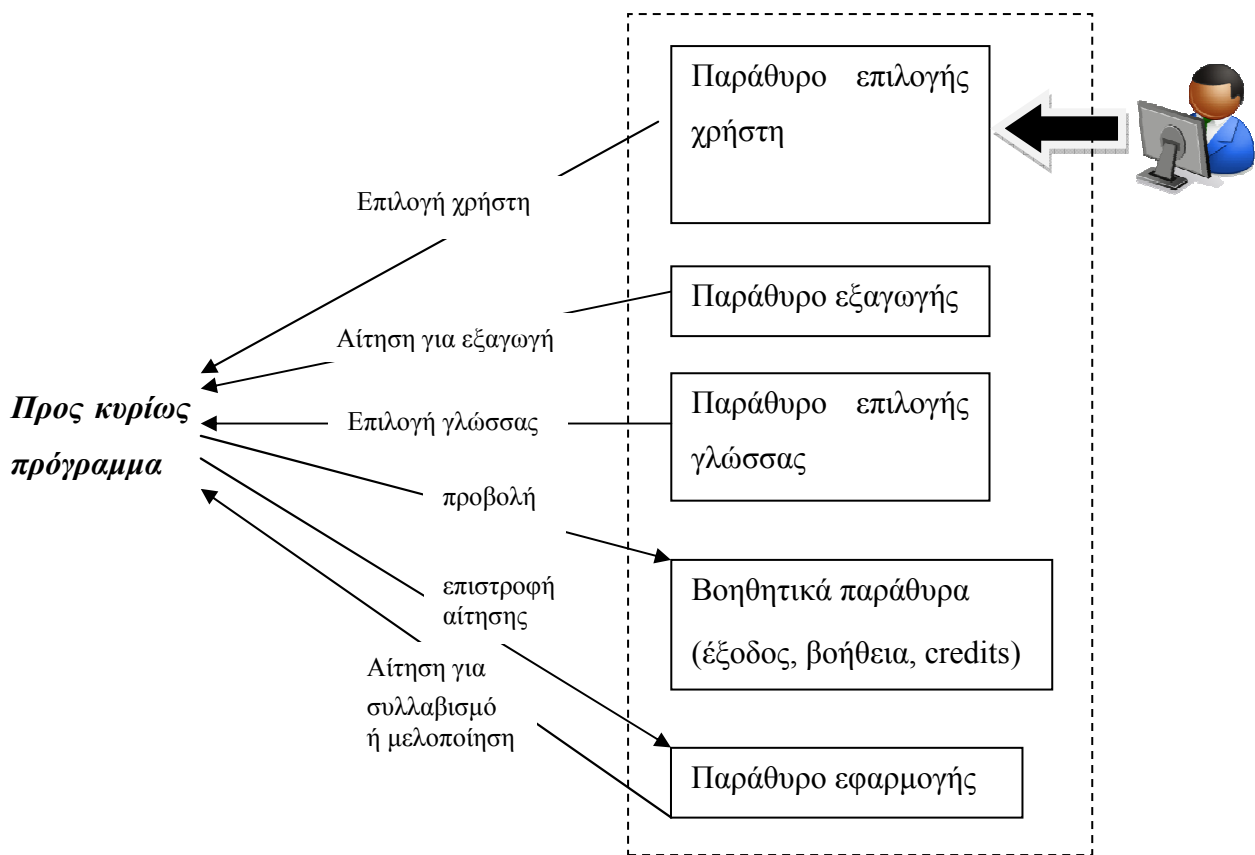
### ***3.2.2 Η Διαπροσωπία***

Οι απαιτήσεις από το υποσύστημα της διαπροσωπίας είναι οι ακόλουθες:

- Το σύστημα θα πρέπει να είναι φιλικό προς το χρήστη.
- Η εκμάθηση χρήσης του συστήματος θα πρέπει να μην απαιτεί ούτε χρόνο ούτε το διάβασμα κάποιου εγχειριδίου χρήσης, αλλά να γίνεται μόνο με την ανάγνωση απλών οδηγιών που θα παρέχονται από το μενού της εφαρμογής είτε με μόνιμα μηνύματα είτε με την εμφάνιση εκτάκτων μηνυμάτων κάθε φορά που ο χρήστης αλληλεπιδρά με το σύστημα.
- Το περιβάλλον διαπροσωπίας θα πρέπει να βοηθά το χρήστη να αποφεύγει τα σφάλματα. Στην περίπτωση σφαλμάτων θα πρέπει να τον κατευθύνει και να του προτείνει ενέργειες προς αποφυγή τους.
- Θα πρέπει να υποστηρίζει Αγγλικά και Ελληνικά.

Ακολουθεί το διάγραμμα που παρουσιάζει αναλυτικά τις λειτουργίες του υποσυστήματος και περιγράφεται αναλυτικά.





Σχήμα 3.4 Διάγραμμα ροής δεδομένων της διαπροσωπίας

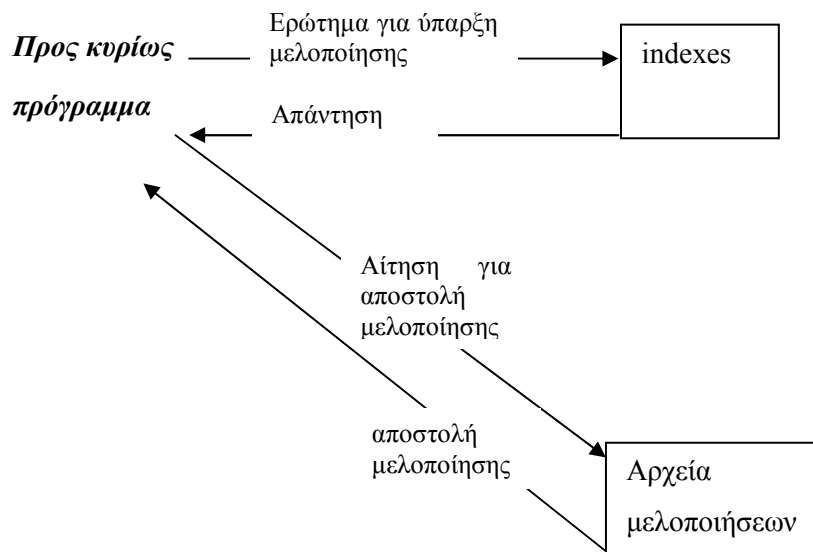
Η διαπροσωπία όπως φαίνεται μεταφέρει της εντολές του χρήστη προς το κυρίως πρόγραμμα. Αρχικά γίνεται η επιλογή του χρήστη και κατόπιν η επιλογή της γλώσσας. Αυτές οι δύο επιλογές καθορίζουν την μορφή του παραθύρου εφαρμογής. Από το παράθυρο αυτό (το οποίο προφανώς είναι το κυρίως παράθυρο του προγράμματος) εισάγει το κείμενο, λαμβάνει τον συλλαβισμό και την μελοποίηση και κάνει τροποποιήσεις πάνω σε αυτή. Αν κατόπιν επιλέξει να εξάγει την μελοποίηση σε ένα αρχείο θα του εμφανιστεί το παράθυρο εξαγωγής από όπου μπορεί να κάνει κάποιες επιλογές για την μορφή που θα έχει το αρχείο. Ακόμη από το μενού μπορεί ο χρήστης να επιλέξει να τερματίσει την λειτουργία του προγράμματος. Σε αυτήν την περίπτωση θα του εμφανιστεί ένα παράθυρο το οποίο ζητά από τον χρήστη να επιβεβαιώσει ότι θέλει να τερματιστεί το πρόγραμμα. Τέλος από το μενού πάλι, ο χρήστης μπορεί να επιλέξει να δει την βοήθεια η τα credits. Πάλι το πρόγραμμα θα του προβάλλει τα αντίστοιχα παράθυρα.

### 3.2.3 Η Βάση Δεδομένων

-Η βάση των μελωδιών θα πρέπει να διευκολύνει την προσθήκη νέων δεδομένων από το χρήστη.

-Θα πρέπει να οργανωθεί έτσι ώστε η αναζήτηση να δίνει γρήγορα απάντηση.

Ακολουθεί το διάγραμμα που παρουσιάζει αναλυτικά τις λειτουργίες του υποσυστήματος και περιγράφεται αναλυτικά.



Σχήμα 3.5 Διάγραμμα ροής δεδομένων της βάσης δεδομένων

Όπως φαίνεται από το παραπάνω σχήμα όταν το πρόγραμμα χρειαστεί να βρει για κάποια φράση μια μελοποίηση θα ψάξει πρώτα στο κατάλληλο index file. Αν δει ότι υπάρχουν εκεί μελοποιήσεις για την συγκεκριμένη φράση θα επιλέξει μία από αυτές και θα προσπελάσει το αντίστοιχο αρχείο από τη βάση των μελοποιήσεων για να την φέρει στο πρόγραμμα.

### 3.3 Αρχιτεκτονική

Παραθέτουμε τις κλάσεις του συστήματος με μία σύντομη περιγραφή.

Κλάση wordGeneral: Υλοποιεί την έννοια της λέξης. Περιέχει πληροφορίες για τις συλλαβές, τους τονισμούς, τη μελοποίηση κ.α. Επειδή ο χρήστης μπορεί να επιλέξει να γράψει το κείμενο στα Αγγλικά έχουμε δημιουργήσει δύο υποκλάσεις, μία για Ελληνικές λέξεις (word) και μία για Αγγλικές ή άλλες γλώσσες (wordEng).

Κλάση GenPhrase: Υλοποιεί όπως ίσως είναι προφανές την έννοια της φράσης. Φυσικά κάθε φράση αποτελείται από λέξεις και αυτό επίσης υλοποιείται με την εισαγωγή των κατάλληλων αντικειμένων στην κλάση. Κρατά πληροφορίες για τις λέξεις, τους τονισμούς, τη μελοποίηση της φράσης κ.α.

Κλάση Elements: Είναι η κλάση η οποία υλοποιεί την δυνατότητα της αναίρεσης (undo). Διατηρεί τα διάφορα στιγμιότυπα των ρυθμίσεων του προγράμματος (όπως το κείμενο εισαγωγής, το είδος και τον ήχο του μέλους, τις επιλογές του χρήστη) ενώ αυτό τρέχει ώστε να μπορεί να τις επαναφέρει στην περίπτωση που ο χρήστης κάνει αναίρεση.

Κλάση cleanString: Είναι η κλάση που αναλαμβάνει να κάνει τις απαραίτητες διεργασίες για την εξαγωγή του αποτελέσματος σε αρχείο .rtf το οποίο θα αποθηκευτεί στον δίσκο. Μπορούμε να την φανταστούμε σαν έναν μεσολαβητή μεταξύ κυρίως προγράμματος και μέσω αποθήκευσης (π.χ. σκληρού δίσκου) του υπολογιστή.

Κλάση rtfFilter: Είναι μια κλάση η οποία απλώς καθορίζει ότι ο τύπος των αρχείων στα οποία θα αποθηκευτούν τα αποτελέσματα της εφαρμογής θα είναι .rtf.

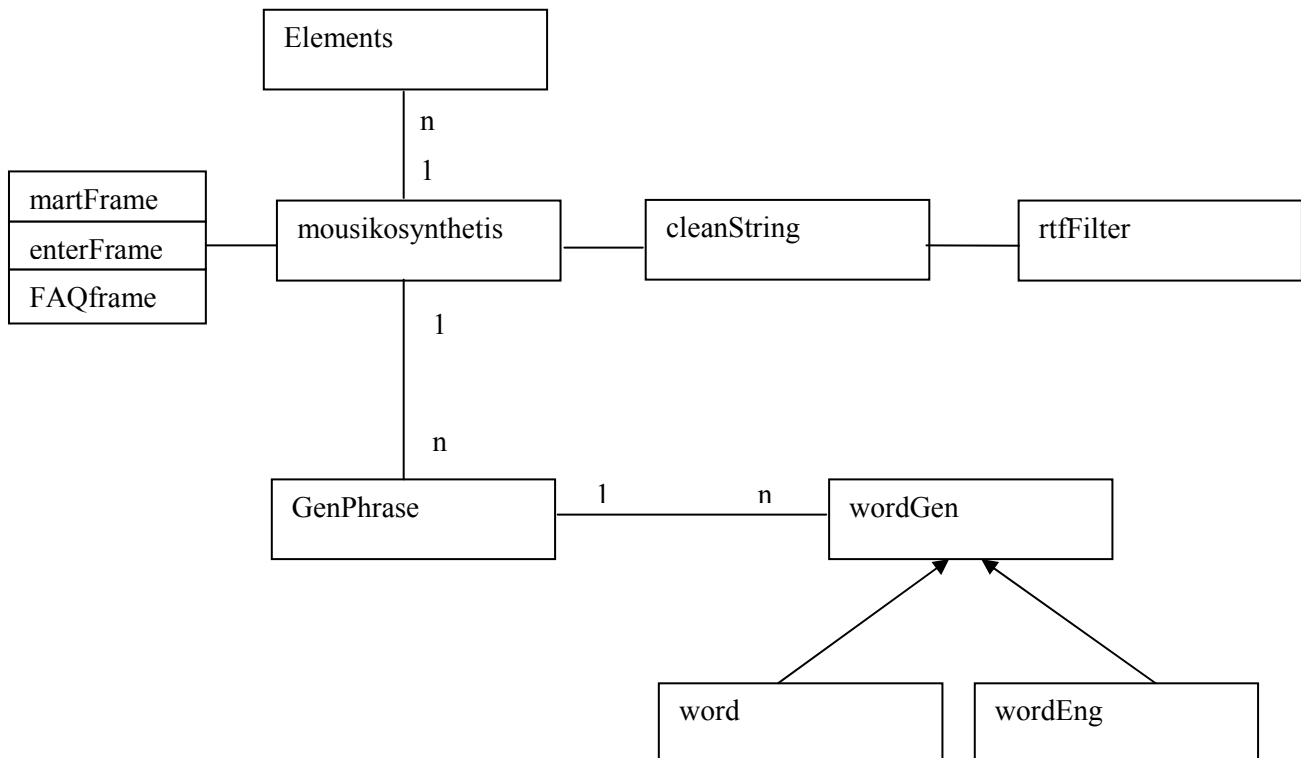
Κλάση mousikosynthetis: Υλοποιεί το κυρίως πρόγραμμα. Η λειτουργία του θα αναλυθεί εκτενέστατα παρακάτω.

Επίσης υπάρχουν και τρεις άλλες κλάσεις για τα βοηθητικά παράθυρα του συστήματος:

Κλάση martFrame: Είναι η κλάση του παραθύρου το οποίο εμφανίζεται όταν ο χρήστης θελήσει να εξάγει την μελοποίηση σε αρχείο. Στο παράθυρο από ο χρήστης κάνει διάφορες επιλογές σχετικά με την μορφή που θα έχει το εξαγόμενο αρχείο. Συγκεκριμένα ο χρήστης επιλέγει ποιες μαρτυρίες θα εξαχθούν καθώς και αν το κείμενο θα μπει χωριστά η παράλληλα με το κείμενο.

Κλάση enterFrame: Είναι η κλάση του εισαγωγικού παραθύρου του προγράμματος στο οποίο γίνεται η επιλογή ή η δημιουργία ενός χρήστη..

Κλάση FAQframe: Είναι η κλάση του παραθύρου το οποίο εμφανίζει τις συνήθεις ερωτήσεις (F.A.Q) στην γλώσσα την οποία έχει επιλέξει ο χρήστης.

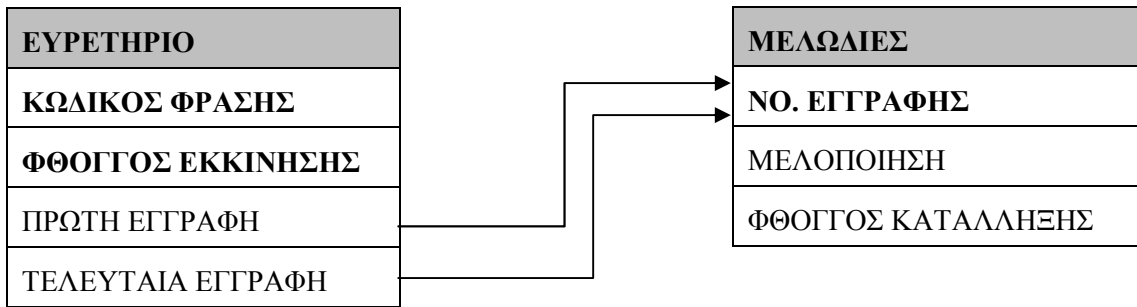


Σχήμα 3.6 Απλό διάγραμμα κλάσεων UML

### 3.4 Περιγραφή Βάσης Δεδομένων

Οι πίνακες της βάσης δεδομένων χωρίζονται σε δύο κατηγορίες, στα ευρετήρια (indexes) και στα αρχεία των μελωδιών. Υπάρχουν πολλά τέτοια ζευγάρια πινάκων. Όταν ζητείται μελοποίηση για μία φράση εξετάζουμε διάφορες παραμέτρους όπως το είδος του μέλους, τον ήχο στον οποίο βρισκόμαστε, το είδος της κατάληξης που έχει επιλέξει ο χρήστης (αν είμαστε σε καταληκτική φράση) και κατόπιν πάμε στο αντίστοιχο ζευγάρι πινάκων για την αναζήτηση.

Κάθε ζευγάρι πινάκων ακολουθεί το παρακάτω διάγραμμα του σχεσιακού μοντέλου:



Σχήμα 3.7 Απλοποιημένο σχεσιακό διάγραμμα βάσης

Ένα παράδειγμα. Ας πούμε ότι είμαστε σε πρώτο ήχο και ότι συνθέτουμε ένα στοιχηρατικό μέλος και κάποια στιγμή το πρόγραμμα θέλει να βρει μελωδία για την φράση «άγιος αθάνατος». Που να ξεκινά από τη νότα πα Το πρόγραμμα έχοντας βρει τον τονικό κωδικό της λέξης (εδώ είναι 1000100) θα πάει στο κατάλληλο index της βάσης και εκεί θα ψάξει για τον κωδικό. Το ευρετήριο αυτό έχει πάρα πολλές εγγραφές μέσα στις οποίες και οι παρακάτω

...

1000001 g 369 375

1000001 p 376 376

1000010 g 377 383

1000010 p 384 387

1000011 g 388 394

**1000100 p 395 398**

1001000 g 399 405

1001001 g 406 412

...

Μόλις βρει την γραμμή με τον κωδικό (αν υπάρχει) ή μόλις περάσει σε γραμμές με κωδικούς μεγαλύτερου μεγέθους σταματά το ψάξιμο μιας και το index είναι ταξινομημένο. Εδώ βρήκε πως υπάρχουν μελοποιήσεις. Οι δύο αριθμοί μας δείχνουν σε ποιες εγγραφές του αρχείου με τη βάση των μελωδιών βρίσκονται οι μελοποιήσεις αυτές. Το πρόγραμμα διαλέγει στην τύχη μία από αυτές τις εγγραφές (έστω την 396η) και πάει στον πίνακα των μελωδιών από όπου και παίρνει την μελωδία. Στην εγγραφή αυτή συναντά το παρακάτω. Το πρώτο γράμμα (p) δείχνει σε ποια νότα καταλήγει η μελοποίηση (στη νότα πα) και το δεύτερο είναι η μελοποίηση.

p |[b]d|[b]||[b]j|[b]j|[b]a|[b]u|[b]\_|[b]r|[b]\|[b]a|[b]h|[b]x|[b]\_|[b]r|[b]\|[b]a|[b]j|[b]r|[b]a|[b]u

Την παίρνει και την παρουσιάζει στο interface με τα Βυζαντινά μουσικά σύμβολα.

# 4

## *Υλοποίηση*

Εδώ θα αναλύσουμε λεπτομερώς θέματα υλοποίησης του συστήματος.

### *4.1 Τεχνολογικά Πρότυπα*

#### *4.1.1 Java SDK 1.6*

Ένα από τα βασικά πλεονεκτήματα της Java έναντι των περισσότερων άλλων γλωσσών είναι η ανεξαρτησία του λειτουργικού συστήματος και πλατφόρμας. Τα προγράμματα που είναι γραμμένα σε *Java* τρέχουν ακριβώς το ίδιο σε Windows, Linux, Unix και Macintosh (σύντομα θα τρέχουν και σε Playstation καθώς και σε άλλες κονσόλες παιχνιδιών) χωρίς να χρειαστεί να ξαναγίνει μεταγλώττιση (compiling) ή να αλλάξει ο πηγαίος κώδικας για κάθε διαφορετικό λειτουργικό σύστημα. Για να επιτευχθεί όμως αυτό χρειαζόταν κάποιος τρόπος έτσι ώστε τα προγράμματα γραμμένα σε Java να μπορούν να είναι «κατανοητά» από κάθε υπολογιστή ανεξάρτητα του είδους επεξεργαστή (Intel x86, IBM, Sun SPARC, Motorola) αλλά και λειτουργικού συστήματος (Windows, Unix, Linux, Unix, MacOS). Ο λόγος είναι ότι κάθε κεντρική μονάδα επεξεργασίας κατανοεί διαφορετικό κώδικα μηχανής. Ο συμβολικός (*assembly*) κώδικας που εκτελείται σε Windows είναι διαφορετικός από αυτόν που εκτελείται σε έναν υπολογιστή Macintosh. Η λύση δόθηκε με την ανάπτυξη της *Εικονικής Μηχανής (Virtual Machine* ή VM ή EM στα ελληνικά).

### 4.1.2 *NetBeans IDE 6.0*

Για να γράψει κάποιος κώδικα Java δε χρειάζεται τίποτα άλλο παρά έναν επεξεργαστή κειμένου, όπως το Σημειωματάριο (Notepad) των Windows ή ο vi (γνωστός στο χώρο του Unix). Παρ' όλα αυτά, ένα ολοκληρωμένο περιβάλλον ανάπτυξης (*IDE*) βοηθάει πολύ, ιδιαίτερα στον εντοπισμό σφαλμάτων (εκσφαλμάτωση - debugging). Για την ανάπτυξη της παρούσας διπλωματικής χρησιμοποιήθηκε το NetBeans IDE 6.0 που μαζί με το Eclipse είναι τα πιο διαδεδομένα.

Το **NetBeans IDE** είναι ένα ανοικτού κώδικα ολοκληρωμένο περιβάλλον ανάπτυξης το οποίο γράφτηκε εξ ολοκλήρου στην Java χρησιμοποιώντας την πλατφόρμα NetBeans. Το NetBeans IDE υποστηρίζει την ανάπτυξη όλων των τύπων εφαρμογής της Java ( J2SE, WEB, EJB και κινητές εφαρμογές)

Η τρέχουσα έκδοση είναι η NetBeans IDE 5.5.1, η οποία βγήκε το Μάιο του 2007. Το NetBeans IDE 5.5 επεκτείνει τις υπάρχουσες Java EE λειτουργίες . Επιπλέον το NetBeans Enterprise Pack υποστηρίζει την ανάπτυξη Java EE 5 enterprise εφαρμογών, συμπεριλαμβανομένων των SOA visual design tools, XML schema tools, web services orchestration (για BPEL), και UML modeling. Το NetBeans C/C++ Pack υποστηρίζει C/C++ projects. Το NetBeans 5.5.1 έχει όλη την λειτουργικότητα του NetBeans 5.5 και επιπλέον έχουν διορθωθεί αρκετά bugs.

### 4.1.3 *iText library*

Το **iText** είναι μία ανοικτού κώδικα βιβλιοθήκη για τη δημιουργία και το χειρισμό PDF, RTF, και HTML αρχείων μέσα στη Java. Γράφτηκε από τους Bruno Lowagie, Paulo Soares, και άλλους και διανέμεται κάτω από τη Mozilla Public License με το LGPL ως εναλλακτική άδεια.

Το ίδιο έγγραφο μπορεί να εξαχθεί σε διάφορους τύπους αρχείων, ή και σε διάφορες περιπτώσεις του ίδιου τύπου. Τα στοιχεία μπορούν να γραφτούν σε ένα αρχείο ή, παραδείγματος χάριν, από ένα servlet σε έναν περιηγητή ιστού

Πιο πρόσφατα, έχει επεκταθεί σε μια βιβλιοθήκη PDF γενικού σκοπού, ικανή να συμπληρώνει φόρμες, να μετακινεί σελίδες από ένα PDF σε άλλο και τα λοιπά. Αυτές οι επεκτάσεις είναι συχνά αμοιβαία αποκλειστικές. Μια κατηγορία του επιτρέπει να συμπληρώσει τις μορφές, ενώ μια διαφορετική και ασυμβίβαστη κλάση το καθιστά ικανό να αντιγράψει σελίδες από ένα PDF σε άλλο.

η υποστήριξη iText's PDF είναι, εν τούτοις, αρκετά εκτενής. Υποστηρίζει PDF's PKI-based signatures, 40-bit και 128-bit encryption, color correction, PDF/X, color management via ICC profiles, και άλλα....

#### **4.1.4 Rich Text Format (.rtf) τύπος αρχείων**

Το **Rich Text Format** (συχνά χρησιμοποιείται σαν συντόμευση το **RTF**) είναι ένας αποκλειστικής εκμετάλλευσης τύπος αρχείου κειμένου ο οποίος αναπτύχθηκε από τη Microsoft το 1987 για ανταλλαγή αρχείων σε διαφορετικές πλατφόρμες. Οι περισσότεροι επεξεργαστές κειμένου (και φυσικά το Microsoft Word) έχουν πλέον τη δυνατότητα να αναγνωρίζουν και να γράφουν τέτοιου τύπου αρχεία.

Το RTF είναι ένας 8-bit τύπος αρχείου. Αυτό θα μπορούσε να το περιορίσει σε χαρακτήρες ASCII, αλλά το RTF μπορεί να περιγράψει παραπάνω από τους ASCII με τη χρήση ακολουθιών διαφυγής οι οποίες επιτρέπουν τον ορισμό και άλλων συμβόλων (όπως για παράδειγμα Unicode χαρακτήρων)

Αντίθετα με τους περισσότερους τύπους αρχείων κειμένου, τα .rtf αρχεία είναι αναγνώσιμα από τον άνθρωπο. Σήμερα οι αναγνώσιμοι βασισμένοι σε XML τύποι αρχείων είναι κάτι το αρκετά σύνηθες. Η ικανότητα αυτή όμως ήταν κάτι σπουδαίο όταν πρωτοεμφανίστηκε το RTF .

## **4.2 Προγραμματιστικά εργαλεία**

### **4.2.1 Netbeans 6.0**

Το NetBeans είναι το ενοποιημένο περιβάλλον ανάπτυξης εφαρμογών Java που αναπτύσσεται από την Sun Microsystems. Ξεκίνησε από μια φοιτητική εργασία και αγοράστηκε από την Sun η οποία άνοιξε τον κώδικά του.

Για να γράψει κάποιος κώδικα Java δε χρειάζεται τίποτα άλλο παρά έναν επεξεργαστή κειμένου, όπως το Σημειωματάριο (Notepad) των Windows ή τον vi (γνωστός στο χώρο του Unix). Παρ' όλα αυτά, ένα ολοκληρωμένο περιβάλλον ανάπτυξης (*IDE*) βοηθάει πολύ, ιδιαίτερα στον εντοπισμό σφαλμάτων (εκσφαλμάτωση - debugging). Για την ανάπτυξη της παρούσας διπλωματικής χρησιμοποιήθηκε το NetBeans IDE 6.0 που μαζί με το Eclipse είναι τα πιο διαδεδομένα περιβάλλοντα ανάπτυξης.



## 4.2.2 Βιβλιοθήκες

Στις διάφορες κλάσεις της εφαρμογής έγινε χρήση των παρακάτω πακέτων της Java.

- 1) `import java.io.*;`
- 2) `import java.util.*;`
- 3) `import java.awt.*;`
- 4) `import java.io.FileNotFoundException;`
- 5) `import java.io.FileOutputStream;`
- 6) `import java.io.IOException;`
- 7) `import javax.swing.*;`
- 8) `import javax.swing.text.*;`
- 9) `import javax.swing.JFileChooser;`
- 10) `import javax.swing.JOptionPane;`
- 11) `import com.lowagie.text.BadElementException;`
- 12) `import com.lowagie.text.DocumentException;`
- 13) `import com.lowagie.text.Document;`
- 14) `import com.lowagie.text.DocumentException;`
- 15) `import com.lowagie.text.Paragraph;`
- 16) `import com.lowagie.text.Phrase;`
- 17) `import com.lowagie.text.rtf.RtfWriter2;`
- 18) `import com.lowagie.text.Font;`
- 19) `import com.lowagie.text.pdf.BaseFont;`
- 20) `import com.lowagie.text.*;`

Να σημειωθεί ότι κάποια από τα παραπάνω πακέτα δεν υπάρχουν στο sdk αλλά ανήκουν στην βιβλιοθήκη του iText.

## 4.3 Αναλυτική Περιγραφή Κλάσεων

### 4.3.1 *wordGen.java*

Δεν περιέχει μεθόδους παρά μόνο κάποιες μεταβλητές που περιγράφουν μία γενικού τύπου λέξη, διάφορες πληροφορίες για την μελοποίησή της και τη νότα από την οποία ξεκινά

και αυτήν στην οποία τελειώνει καθώς και τη θέση(γραμμή) της μελοποίησης στη βάση δεδομένων. Συγκεκριμένα οι μεταβλητές είναι οι εξής:

```
String[] pieces; //κάθε κελί περιέχει μία συλλαβή
String[] pieces2; //κάθε κελί περιέχει τον κώδικα μιας συλλαβής(π.χ j)
String tone=""; //ο δυαδικός κωδικός των τόνων της λέξης (π.χ 100)
String toneCode=""; //όλος ο κωδικός της λέξης (π.χ Sja)
String toneCodeF=""; //όλος ο κωδικός της λέξης με κάθετες(π.χ ||S||j||a)
String extended=""; //η λέξη μαζί με τα -
String compressed=""; //η λέξη χωρίς τα -
String startLevel,endLevel;//το ύψος της αρχής και του τέλους της μελωδίας
int hitNumber=0; //η θέση της μελωδίας μέσα στη βάση δεδομένων
```

### 4.3.2 *Word.java*

Περιέχει διάφορες μεταβλητές καθώς και τις παρακάτω μεθόδους:

word(String w):

Είναι ο κατασκευαστής της κλάσης. Εκτελεί τις εξής λειτουργίες:

- 1) Παίρνει σαν είσοδο ένα String από το κυρίως πρόγραμμα. Το String αυτό περιέχει παύλες (-) οι οποίες το χωρίζουν σε συλλαβές.
- 2) Με τη βοήθεια ενός StringTokenizer το σπάει σε συλλαβές και τις φυλάσσει σε έναν πίνακα
- 3) Με τη βοήθεια της μεθόδου has tone (βλ. παρακάτω) φτιάχνει έναν πίνακα ο οποίος έχει στα κελιά του 0 ή 1 ανάλογα με το αν η αντίστοιχη συλλαβή τονίζεται ή όχι.

isVowel(char c) :

Παίρνει έναν χαρακτήρα-γράμμα και εξετάζει εάν είναι φωνήεν

hasTone(char c) :

Παίρνει έναν χαρακτήρα-γράμμα και εξετάζει εάν τονίζεται

diphthong(String duo) :

Παίρνει ένα String δύο χαρακτήρων και εξετάζει εάν είναι δίφθογγος.

### 4.3.3 *wordEng.java*

Περιέχει διάφορες μεταβλητές καθώς και την παρακάτω μέθοδο κατασκευαστή:  
word(String w):

Είναι ο κατασκευαστής της κλάσης. Εκτελεί τις έξης λειτουργίες:

- 1) Παίρνει σαν είσοδο ένα String από το κυρίως πρόγραμμα. Το String αυτό περιέχει παύλες (-) οι οποίες το χωρίζουν σε συλλαβές και τόνους (´) που δείχνουν αν η συλλαβή που ακολουθεί είναι τονισμένη .
- 2) Με τη βοήθεια ενός StringTokenizer το σπάει σε συλλαβές και τις φυλάσσει σε έναν πίνακα
- 3) Φτιάχνει έναν πίνακα ο οποίος έχει στα κελιά του 0 ή 1 ανάλογα με το αν η αντίστοιχη συλλαβή έχει τόνο ή όχι.

### 4.3.4 *GenPhrase.java*

Έχει μόνο μία μέθοδο, τον κατασκευαστή της GenPhrase(String w), η οποία απλώς παίρνει ένα String, το καθαρίζει από τις παύλες, και τους τόνους που έχει και το φυλάει καθαρό σε άλλη μία νέα μεταβλητή String. Έχει ακόμα διάφορες μεταβλητές οι οποίες κρατάν πληροφορίες για τη φράση και την μελοποίηση της

### 4.3.5 *CleanString.java*

Έχει μόνο μία συνάρτηση, τον κατασκευαστή της, η οποία :

- 1) Εμφανίζει στον χρήστη ένα παράθυρο για να επιλέξει πού θα αποθηκεύσει το αρχείο.
- 2) Δημιουργεί το αρχείο
- 3) Ενσωματώνει σε αυτό τις γραμματοσειρές των Βυζαντινών συμβόλων
- 4) Γράφει σε αυτό το κείμενο και την μελοποίηση του με τον τρόπο που ο χρήστης έχει επιλέξει

### 4.3.6 *Elements.java*

Κρατά πληροφορίες για την κατάσταση του συστήματος κατά τις διάφορες χρονικές στιγμές. Εκτός από τη φράση φυλάει επίσης τον ήχο, το αριθμό των αντικειμένων που έχουν φτιαχτεί από την αρχή του προγράμματος, τον μέγιστο αριθμό οπισθοδρομήσεων που μπορούν να γίνουν και άλλα.

Έχει μία μέθοδο κατασκευαστή η οποία απλώς δίνει στο στιγμιότυπο της κλάσης που δημιουργείται έναν χαρακτηριστικό αριθμό (id) και αυξάνει τη μεταβλητή η οποία κρατά το πλήθος των αντικειμένων Elements.

#### **4.3.7 enterFrame.java**

Έχει μία μέθοδο η οποία σχηματίζει τα στοιχεία της διαπροσωπίας του παραθύρου. Έχει ακόμα και μία μέθοδο κατασκευαστή η καλεί την παραπάνω μέθοδο και επιπλέον ρυθμίζει το παράθυρο έτσι ώστε αν ο χρήστης το κλείσει από το «χ» πάνω δεξιά να κλείσει και το πρόγραμμα.

#### **4.3.8 martFrame.java**

Έχει και αυτή μία μέθοδο η οποία σχηματίζει τα στοιχεία της διαπροσωπίας του παραθύρου. Έχει ακόμα και μία μέθοδο κατασκευαστή η καλεί την παραπάνω μέθοδο και επίσης καθορίζει τη γλώσσα με την οποία θα γραφτούν οι ετικέτες της διαπροσωπίας του παραθύρου ανάλογα με την γλώσσα που έχει επιλέξει ο χρήστης κατά την είσοδό του στην εφαρμογή.

#### **4.3.9 FAQframe.java**

Έχει φυσικά και αυτή μία μέθοδο η οποία σχηματίζει τα στοιχεία της διαπροσωπίας του παραθύρου καθώς και μία μέθοδο κατασκευαστή η καλεί την πρώτη μέθοδο. Η κλάση αυτή είναι ουσιαστικά ένα παράθυρο το οποίο εμφανίζει ένα κείμενο με τις «συνήθεις ερωτήσεις» (FAQ) όταν ο χρήστης πατήσει στο αντίστοιχο πεδίο του μενού της βοήθειας.

#### **4.3.10 Newframe.java**

Είναι το κυρίως πρόγραμμα το οποίο συνεργάζεται με τις υπόλοιπες κλάσεις για να φέρει εις πέρας τις λειτουργίες της εφαρμογής και της διαπροσωπίας. Οι σημαντικότερες συναρτήσεις είναι οι εξής:

- 1) errorHandle: Φροντίζει τα σχετικά με τα μηνύματα τα οποία βγάζει το πρόγραμμα στον χρήστη όταν δεν υπάρχει μελοποίηση για αυτό που ζήτησε.
- 2) Recalculate: Εκτελεί την λειτουργία της οπισθοδρόμησης όταν κατά την αναζήτηση μιας μελωδίας δεν βρεθεί φράση. Αυτή η συνάρτηση εξετάζει αν υπάρχει περίπτωση αλλάζοντας τις προηγούμενες φράσεις να βρεθεί λύση.
- 3) createTextPane: Σχεδιάζει στο interface τη μελοποίηση του κειμένου
- 4) addStylesToDocument: Ρυθμίζει τα πεδία κειμένου της διαπροσωπίας ώστε να δεχθούν τις γραμματοσειρές με τα Βυζαντινά μουσικά σύμβολα
- 5) search: Παίρνει τον τονικό κωδικό μιας φράσης και βρίσκει μελοποίηση για αυτόν

- 6) `NewJFrame`: Η συνάρτηση κατασκευαστής, εκτελεί τις απαραίτητες αρχικοποιήσεις και εισαγωγικές λειτουργίες του προγράμματος
- 7) `enterActionPerformed`: Συνάρτηση η οποία καλείται όταν γίνει η επιλογή χρήστη για να μπει ο χρήστης στο κυρίως interface.
- 8) `registerActionPerformed`: Καλείται όταν γίνεται εγγραφή (registration) νέου χρήστη
- 9) `initComponents`: Σχεδιάζει τα σταθερά στοιχεία της διαπροσωπίας.
- 10) `okButActionPerformed`: Καλείται όταν ο χρήστης ζητήσει μελοποίηση. Παίρνει το συλλαβισμένο κείμενο, βρίσκει τις φράσεις και τους κωδικούς τους και ψάχνει για μελοποίηση.
- 11) `spellActionPerformed`: Καλείται όταν ο χρήστης ζητήσει να γίνει αυτόματος συλλαβισμός στο κείμενο. Από τη θέση των φωνηέντων της κάθε λέξης καταλαβαίνει που χωρίζονται οι συλλαβές.
- 12) `chngActionPerformed`: Καλείται όταν ο χρήστης επιλέξει την αλλαγή της μελοποίησης μίας φράσης.
- 13) `createVariPanel`: Εμφανίζει στη διαπροσωπία τις πιθανές μελοποιήσεις μιας φράσης όταν ο χρήστης την επιλέξει.
- 14) `aboutMenuItemActionPerformed`: Εμφανίζει ένα παράθυρο με τα στοιχεία του προγράμματος όταν πατηθεί το αντίστοιχο πεδίο στο μενού.
- 15) `guideMenuItemActionPerformed`: Εμφανίζει το παράθυρο της βοήθειας προγράμματος όταν πατηθεί το αντίστοιχο πεδίο στο μενού.
- 16) `aqMenuItemActionPerformed`: Εμφανίζει ένα παράθυρο με τις συνήθεις ερωτήσεις (F.A.Q) όταν πατηθεί το αντίστοιχο πεδίο στο μενού.
- 17) `exitMenuItemActionPerformed`: Ρυθμίζει τα σχετικά με το κλείσιμο του προγράμματος.
- 18) `seperMenuItemActionPerformed`: Σχεδιάζει το παράθυρο για την εξαγωγή της μελωδίας οποίο ο χρήστης κάνει διάφορες επιλογές.
- 19) `undoMenuItemActionPerformed`: Υλοποιεί την λειτουργία της αναίρεσης (undo)
- 20) `redoMenuItemActionPerformed`: `undoMenuItemActionPerformed`: Υλοποιεί την λειτουργία της επανάληψης (redo)
- 21) `textsToGreek`: Αλλάζει στην Ελληνική γλώσσα της ετικέτες σε όλα τα στοιχεία της διαπροσωπίας για του χρήστες που επιλέγουν τα Ελληνικά ως γλώσσα του interface.

22) chlang2MenuItemActionPerformed: Κάνει τις απαραίτητες κινήσεις όταν ο χρήστης επιλέξει να αλλάξει την γλώσσα στην οποία εισάγει το κείμενο.

## 4.4 Λεπτομέρειες Υλοποίησης

Εδώ περιγράφουμε λεπτομερώς θέματα της διπλωματικής που έχουν τεχνικό ή αλγοριθμικό ενδιαφέρον.

### 4.4.1 Αυτόματος Συλλαβισμός του Κειμένου

Ο χρήστης πληκτρολογεί το κείμενό του σε ένα πεδίο κειμένου και πατώντας το κουμπί «spell» το πρόγραμμα κάνει τον συλλαβισμό και του παρουσιάζει σε ένα νέο πεδίο κειμένου το συλλαβισμένο κείμενο.

Ο κώδικας που κάνει αυτή την δουλειά είναι ο παρακάτω:

```
tokenizer = new StringTokenizer(inText.getText());
String[] pieces = new String[tokenizer.countTokens()]; //δημιουργία μεγέθους όσο και τα τόκενς
//Το tokenizer είχε τις λέξεις αλλά τώρα θα τις βάλω και στον πίνακα pieces
int i = 0;
while (tokenizer.hasMoreTokens()) { //Τις βάζω τυπώνοντας τις παράλληλά
    pieces[i] = tokenizer.nextToken();
    System.out.println(String.valueOf(i)+"word is "+pieces[i]);
    if(pieces[i].endsWith(", ")) xString=xString+extend(pieces[i])+", ";
    else xString=xString+extend(pieces[i])+" ";
    i++;
}
syllableText.setText(xString+" ");
```

Όπως φαίνεται η διαδικασία είναι η εξής:

- 1) Παίρνουμε με τη βοήθεια ενός tokenizer μία μία τις λέξεις από το κείμενο του χρήστη και τις καταχωρούμε σε έναν πίνακα.
- 2) Κάθε λέξη στέλνεται στην μέθοδο extend η οποία επιστρέφει τη λέξη με παύλες που χωρίζουν τις συλλαβές.
- 3) Οι λέξεις καταχωρούνται με την νέα τους (συλλαβισμένη) μορφή σε ένα String το οποίο και στέλνεται στο δεύτερο πεδίο κειμένου.

Για παράδειγμα, ας υποθέσουμε ότι ο χρήστης έχει εισάγει στο πρώτο πεδίο κειμένου τις φράσεις «αγγέλων ο δήμος, κατεπλάγη ορών σε, νεκροίς λογισθέντα». Ο βρόγχος θα ξεκινήσει παίρνοντας την πρώτη λέξη (αγγέλων), θα την βάλει στον πίνακα και θα την στείλει στην extend η οποία θα επιστρέψει το String α-γγέ-λων, την λέξη δηλαδή χωρισμένη με παύλες σε συλλαβές. Η μεταβλητή xString μαζεύει μία μία τις συλλαβισμένες λέξεις που βγαίνουν μέσα στο βρόγχο. Έτσι στο τέλος η τιμή του xString θα είναι «α-γγέ-λων ο δή-μος,

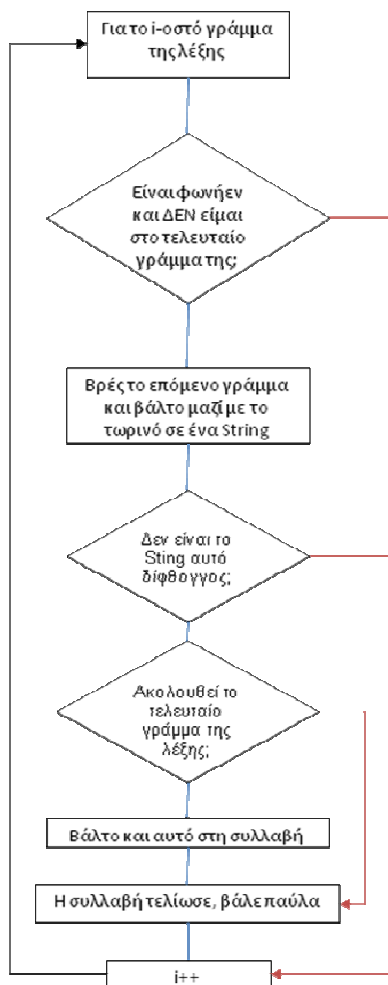
Τον α-γγέ-λων ο δή-μος, κα-τε-πλά-γη ο-ρών σε, νε-κροίς λο-γι-σθέ-ντα, του θα-νά-του δε Σω-τήρ, την ι-σχύν κα-θ  
ε-λό-ντα, και συν ε-αυ-τό, τον Α-δάμ ε-γεί-ρα-ντα, και εξ Α-δου, πά-ντας ε-λευ-θέ-ρω-σας

κα-τε-πλά-γη ο-ρών σε, νε-κροίς λο-γι-σθέ-ντα». Αυτό ακριβώς που θα παρουσιαστεί στον χρήστη στο δεύτερο πεδίο κειμένου.

Αυτό ακριβώς που θα παρουσιαστεί

Έχει ενδιαφέρον όμως να δούμε πώς ακριβώς η συνάρτηση extend καταλαβαίνει τις συλλαβές. Ας δείξουμε το σχετικό κομμάτι από τον κώδικά της:

```
private String extend(String w){
    boolean stop=false;
    w=w.replaceAll(",","");//μην υπολογίσεις τις τελίες
    int length = w.length();
    String duo="",next="",extended="";
    if(selectedValue.equals("Greek")) {
        for(int i=0; i<length; i++) { //ΑΥΤΟΜΑΤΟΣ ΣΥΛΛΑΒΙΣΜΟΣ
            if (!stop) extended=extended.concat(extended.valueOf(w.charAt(i)));
            if (word.isVowel(w.charAt(i)) { //αν έχουμε φωνήεν
                if(i+1<w.length()) {System.out.println("next"); next=""w.charAt(i+1); duo=""w.charAt(i)+next;
                    if(!word.diphthong(duo)){//και το τωρινό γρα'μμα δεν είναι το πρώτο γράμμα ενός δίφθογγου
                        if (length!=1&&!word.isVowel(next.charAt(0)) &&i==length-2){//αν ακολουθείτο τελευταίο γρ
                            extended=extended.concat(extended.valueOf(w.charAt(i+1))); //θα βάλλουμε και το τελ
                            stop=true;
                        }else extended=extended.concat("-"); //τέλος της συλλαβής
                    }
                }
            }
        }
    }
}
```



Σχήμα 5.1

Η λειτουργία του φαίνεται απλοποιημένη στο διάγραμμα ροής αριστερά. Για παράδειγμα όταν κληθεί η extend για την λέξη «άγγελος» που αναφέραμε προηγουμένως θα συμβεί το εξής:

- 1) Εξετάζεται ο πρώτος χαρακτήρας της λέξης ( το άλφα) αν είναι φωνήεν.
- 2) Αφού είναι φωνήεν φτιάχνουμε το String με το όνομα duo το οποίο αποτελείται από το άλφα και τον επόμενο γράμμα της λέξης. Άρα η τιμή του duo είναι «αγ»
- 3) Εξετάζεται αν το duo ΔΕΝ είναι δίφθογγος
- 4) Επειδή αυτό είναι αληθές εξετάζεται αν ακολουθεί το τελευταίο γράμμα της λέξης
- 5) Αυτό είναι ψευδές άρα έφτασα στο τέλος της συλλαβής οπότε βάζω παύλα στο Sting extend το οποίο είναι μια καθολική μεταβλητή η οποία φυλάει τις λέξεις με την νέα (συλλαβισμένη) μορφή.

σχεδιάγραμμα αυτόματου συλλαβισμού

#### 4.4.2 Εύρεση Μελοποίησης του Κειμένου

Όταν πατηθεί το σχετικό κουμπί (και αφού έχει σχηματιστεί η συλλαβισμένη μορφή του κειμένου), για κάθε φράση που εισήγαγε ο χρήστης καλούμε την search με όρισμα το κελί του πίνακα των φράσεων που την περιέχει. Η συνάρτηση θα ψάξει στο ευρετήριο για να δει αν υπάρχουν μελοποιήσεις για τη φράση και αν όντως υπάρχουν θα πάει στο αρχείο της βάσεις για να επιλέξει μία από αυτές. Η μελοποίηση θα καταχωρηθεί στο πεδίο symbols της φράσης. Αν όμως δεν υπάρχει απολύτως τίποτα για τον κωδικό αυτό θα πάρει την τιμή “nothingAtAll”. Έχει ενδιαφέρον να δούμε τη Τρίτη περίπτωση, κατά την οποία βρέθηκε μεν κάτι στο σωστό κωδικό αλλά όχι στο σωστό ύψος. Σε αυτήν την περίπτωση στο πεδίο symbols θα γραφτεί το String “nothingInThisLevel”. Σε αυτή την περίπτωση λοιπόν υπάρχουν ελπίδες να βρεθεί μελοποίηση αν καταφέρουμε να αλλάξουμε τις μελοποιήσεις των προηγούμενων φράσεων έτσι ώστε να καταλήγουν στο επιθυμητό ύψος. Αυτή την διαδικασία ονομάσαμε recalculation και την λειτουργία της θα την αναλύσουμε αργότερα, Εδώ παραθέτουμε τον κώδικα της συνάρτησης search.

```
private String search(GenPhrase current){//Παίρνει κωδικό (π.χ 101) και προσθέτει στο letters το αντίστοιχο
String input="",code,code0,code1,from,to,clear,toneCode,mode,endPad="",kind,ending;
int line=0,hits=0,randomMelody,indexS,indexE,thres;
int[] lineBase;
Boolean match=true,exists=false;
StringTokenizer tokenizer;
BufferedReader console,consoleInd;

mode=elems[Elements.cursor].mode;
kind=elems[Elements.cursor].type;
ending=elems[Elements.cursor].ending;
//αν είμαι στην προ-τελευταία φράση
if(ending.equals("3") &&phraseBase.length>1&&current.equals(phraseBase[phraseBase.length-2])){
    endPad="4";//πρώτη φράση εκτεταμένης κατάληξης
}
//ΕΛΕΓΧΟΣ ΚΑΤΑΛΗΞΗΣ αν είμαι στην τελευταία φράση
if(current.equals(phraseBase[phraseBase.length-1])){
    if(ending.equals("1")){
        endPad="1";//απλή
    }
    if(ending.equals("2")){
        endPad="2";//θροιστικά
    }
    if(ending.equals("3")){
        endPad="3";//δεύτερη φράση εκτεταμένης κατάληξης
    }
}
try{
    console = new BufferedReader(new FileReader("database/database"+mode+kind+endPad+".txt"));
    consoleInd = new BufferedReader(new FileReader("database/database"+mode+kind+endPad+"_index.txt"));
    do{
        try {input = consoleInd.readLine();} catch (IOException e) {}
        if(input==null) break;
        tokenizer = new StringTokenizer(input);
        code=tokenizer.nextToken();
        if(code.equals(current.dual)){//Υπάρχουν μελοποιήσεις για τον κωδικό της φράσης...
            exists=true;
            from=tokenizer.nextToken();
            if(level.equals(from)){//...που να ξεκινούν και από τον σωστό τονο
                indexS=Integer.parseInt(tokenizer.nextToken());
                indexE=Integer.parseInt(tokenizer.nextToken());
                thres=(int)(Math.random()*(indexE-indexS+1)+indexS);//επιλογή μίας τυχαίας γραμμής από τ
                try { for(int a=0;a<thres;a++) input = console.readLine();} catch (IOException e) {}
            }
        }
    }
}
```



```

tokenizer = new StringTokenizer(input);
current.startLevel=from;
current.endLevel=tokenizer.nextToken();
toneCode=tokenizer.nextToken();
level=current.endLevel;
clear=cleanString(toneCode,current);//Στέλνει το S/j/a και τη λέξη, στο clear επιστρέφει
current.indexS=indexS;      current.indexE=indexE;
return clear;
    }
}
}while(code.length()<=current.dual.length());
}catch(FileNotFoundException e){System.out.println("Error "+e.toString());}
if(exists) return "nothingInThisLevel"; else return "nothingAtAll";
}
}

```

### 4.4.3 Recalculation

Έχει ενδιαφέρον να δούμε πώς ακριβώς γίνεται αυτή η πολύ χρήσιμη λειτουργία.

Παραθέτουμε τον κώδικα της συνάρτησης recalculate.

```

private boolean recalculate (int i, int j){
    boolean exitFromREC=false,disaster=false;
    String temp="",newLevel="";
    int times,times2;

    do{
        times=0;
        do{//ψάξε μέχρι να βρείς κάτι που να καταλήγει αλλού μήπως και μπορεί να
            level=phraseBase[i-1].startLevel;
            phraseBase[i-1].symbols=search(phraseBase[i-1]);
            newLevel=phraseBase[i-1].endLevel;
            times++;
        }while(levels[i-1].indexOf(newLevel)>=0 && times<100 );
        //ΔΕΝ ΒΡΗΚΕ ΚΑΤΙ ΠΟΥ ΝΑ ΚΑΤΑΛΗΓΕΙ ΣΕ ΑΛΛΟ ΔΕΒΕΑ, ΙΣΩΣ ΠΡΕΠΕΙ ΝΑ ΑΛΛΑΞΟΥΜΕ
        if(times>=100){
            if(i>1+j){//ΑΝ ΥΠΑΡΧΟΥΝ ΠΡΟΗΓΟΥΜΕΝΕΣ ΦΡΑΣΕΙΣ...
                levels[i-2]=levels[i-2].concat(phraseBase[i-2].endLevel);
                if(recalculate(i-1,j)==false){
                    return false;
                }
            }
            else {//...ΑΛΛΑΙΩΣ ΤΟ RECALC ΕΠΙΣΤΡΕΦΕΙ ΛΑΘΟΣ ΓΙΑΤΙ ΔΕΝ ΒΡΗΚΕ ΑΥΤΟ
                return false;
            }
        }
        else { //ΟΛΑ ΟΚ
            phraseBase[i].startLevel=phraseBase[i-1].endLevel;
            return true;
        }
        //ΨΑΧΝΩ ΕΑΝΑ ΜΗΠΩΣ ΤΩΡΑ ΠΟΥ ΑΛΛΑΞΕ ΤΟ startLevel υπάρχει κάτι
    }while(true);
}
}

```

Όπως φαίνεται πρόκειται για μία αναδρομική συνάρτηση.

Αναλυτικά:

- 1) Υπάρχει ένας βρόγχος ο οποίος (λόγω της συνθήκης του η οποία είναι μονίμως true) εκτελείται συνεχώς μέχρι η συνάρτηση να επιστρέψει true ή false.
- 2) Μετά υπάρχει ένας δεύτερος βρόγχος στον οποίο κάνουμε προσπάθειες για να βρεθεί μία μελοποίηση η οποία να καταλήγει σε διαφορετικό ύψος
- 3) Αν μετά από 100 προσπάθειες δεν βρούμε τίποτα τότε αναδρομικά θα καλέσουμε την recalculate για την αμέσως προηγούμενη φράση (αν υπάρχει βέβαια). Αν έρθει false τότε και η τωρινή κλήση της recalculate επιστρέφει false
- 4) Αν βρέθηκε κάτι επιστέφει true

Για παράδειγμα ας πάρουμε πάλι το κείμενο «αγγέλων ο δήμος, κατεπλάγη ορών σε, νεκροίς λογισθέντα». Έχει χωριστεί από τον χρήστη σε τρεις φράσεις (με τα κόμματα). Κατόπιν ας υποθέσουμε ότι για την πρώτη φράση βρίσκει μία μελοποίηση η οποία ξεκινά από πα και καταλήγει σε δι, για την δεύτερη από δι σε βου και για την τρίτη δεν βρίσκει τίποτα που να ξεκινά από βου αλλά ξέρει ότι υπάρχουν μελοποιήσεις από άλλες νότες. Άρα πρέπει να κάνει recalculation.

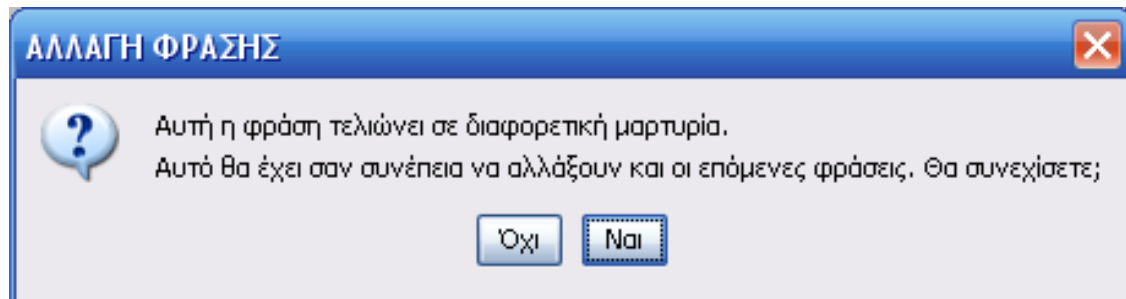
Καλείται λοιπόν η recalculate και, όπως περιγράψαμε προηγουμένως, ψάχνει μήπως για την δεύτερη φράση βρεθεί μελοποίηση που να λήγει σε κάτι διαφορετικό του βου. Αν δεν βρεθεί θα κληθεί ξανά για την πρώτη φράση (για να βρει κάτι διαφορετικό του δι). Αν τώρα βρεθεί κάτι (π.χ. που να καταλήγει σε γα) ξαναψάχνουμε για την δεύτερη φράση κάτι που να ξεκινά πλέον σε γα. Αν ούτε για το γα βρεθεί κάτι ξανακάνει recalculation κ.ο.κ

Αν τελικά δεν μπορέσει να βρεθεί λύση η αρχική κλήση της recalculate επιστρέφει false και όπως εξηγήσαμε στο 5.2.2 θα εμφανιστεί σχετικό μήνυμα στο χρήστη. Αν όμως βρεθεί λύση, αν δηλαδή συνεχίζει ψάχνοντας για μελοποιήσεις από αυτό το νέο επίπεδο όπως περιγράφηκε επίσης στο 5.2.2

#### ***4.4.4 Αλλαγή της Μελοποίησης Κάποιων Φράσεων***

Ο χρήστης αφού πάρει την μελοποίηση του κειμένου μπορεί να επιθυμεί να αλλάξει την μελοποίηση των φράσεων. Αφού επιλέξει από τα checkboxes της διαπροσωπίας την φράση την οποία επιθυμεί να αλλάξει θα του εμφανισθούν δεξιά όλες οι πιθανές μελοποιήσεις για αυτήν. Ο χρήστης αν θέλει επιλέγει μία από αυτές και αμέσως η αλλαγή που έκανε παρουσιάζεται στην διαπροσωπία. Στην περίπτωση όμως που η νέα μελοποίηση καταλήγει σε άλλη νότα από ότι η προηγούμενη το πρόγραμμα καταλαβαίνει ότι θα πρέπει να αλλάξει και

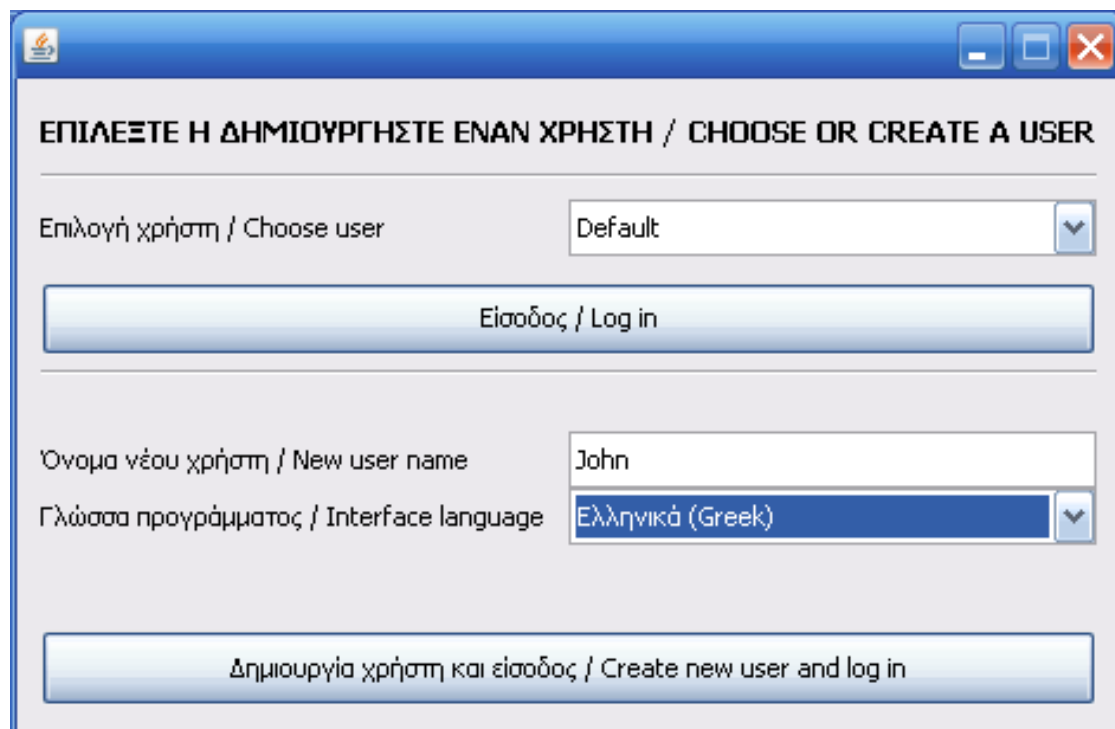
τις επόμενες μελοποιήσεις. Επειδή ο χρήστης μπορεί να μην το θέλει αυτό (γιατί ενδεχομένως κοπίασε πολύ για να βρει τις μελοποιήσεις των επόμενων φράσεων) το πρόγραμμα εμφανίζει ένα παράθυρο όπου ρωτά τον χρήστη αν συμφωνεί να αλλάξουν και οι επόμενες ή αν προτιμά να αφήσει τα πράγματα ως είχαν.



Αν δεν συμφωνεί τότε το πρόγραμμα βρίσκει νέες μελοποιήσεις για τις επόμενες φράσεις οι οποίες να ταιριάζουν με το νέο τονικό ύψος, όπως ακριβώς αναλύσαμε στην ενότητα 6.2.2

## 4.5 Χρήστες

Όταν ο χρήστης τρέχει το πρόγραμμα συναντά το παράθυρο επιλογής και δημιουργίας χρήστη.



Εδώ μπορεί να δημιουργήσει έναν νέο χρήστη για τον οποίο θα καθορίσει ένα ψευδώνυμο και τη γλώσσα για το interface της εφαρμογής. Αυτές οι δύο πληροφορίες θα αποθηκευτούν σε ένα αρχείο το οποίο κρατά αυτές τις πληροφορίες για κάθε χρήστη. Έτσι όταν θα ξαναοίξει το πρόγραμμα ο χρήστης θα δει το ψευδώνυμο το οποίο είχε εισάγει προηγουμένως να έχει εισαχθεί στην λίστα των χρηστών και επιλέγοντας να κάνει log in με αυτό θα παρατηρήσει ότι το interface είναι στην γλώσσα την οποία είχε επιλέξει και την πρώτη φορά.

# 5

## *Έλεγχος*

Παρακάτω παραθέτουμε τα σχετικά με τον έλεγχο της αξιοπιστίας του Μουσικοσυνθέτη.

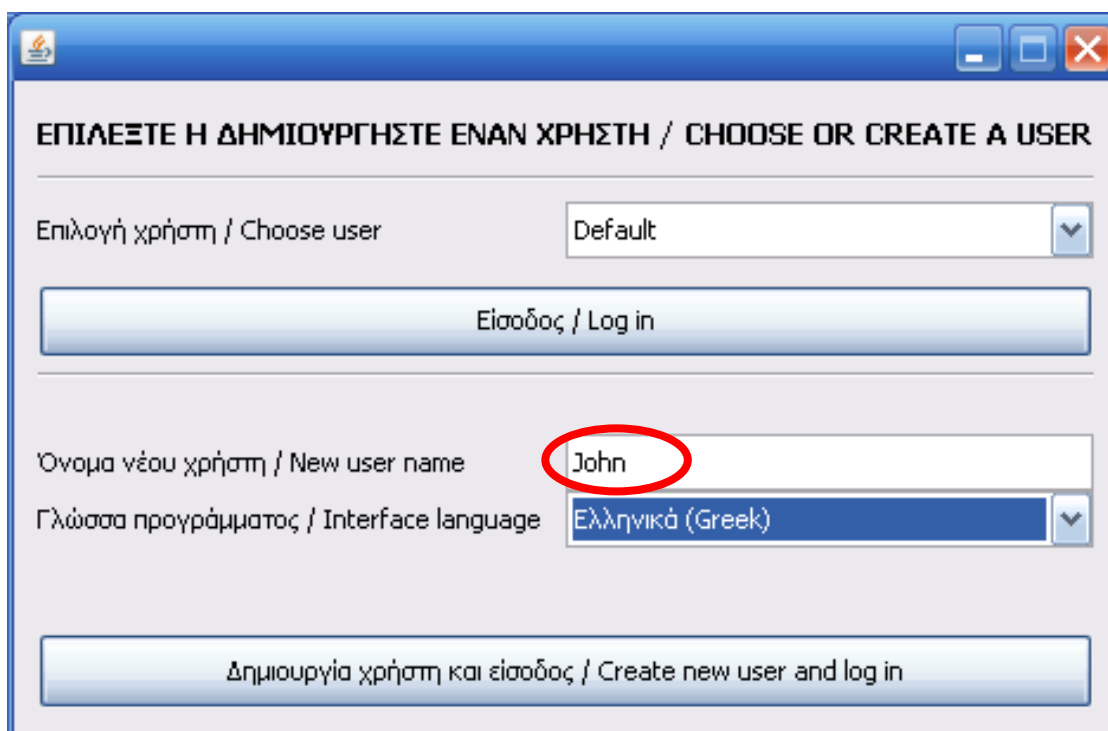
### *5.1 Μεθοδολογία Ελέγχου*

Έγινε διεξοδικός έλεγχος των σεναρίων λειτουργίας για να διαπιστωθεί η σωστή λειτουργία του προγράμματος. Επίσης ελέγχτηκαν τα αρχεία της βάσεις και η επικοινωνία του προγράμματος με τον υπολογιστή για την έξοδο της μελοποίησης στο Word. Τα σενάρια αυτά θα έπρεπε να καλύπτουν όλες τις δυνατές περιπτώσεις ελέγχου, κάτι που είναι, όμως, πρακτικά αδύνατο. Παρόλα αυτά, καλύπτουν όλο το φάσμα των δυνατών τρόπων χρήσης της εφαρμογής. Ένα σύντομο σενάριο χρήσης (demo) παρουσιάζεται στην επόμενη ενότητα.

## 5.2 Αναλυτική Παρουσίαση Ελέγχου

### 5.2.1 Ένα Βασικό Σενάριο Χρήσης

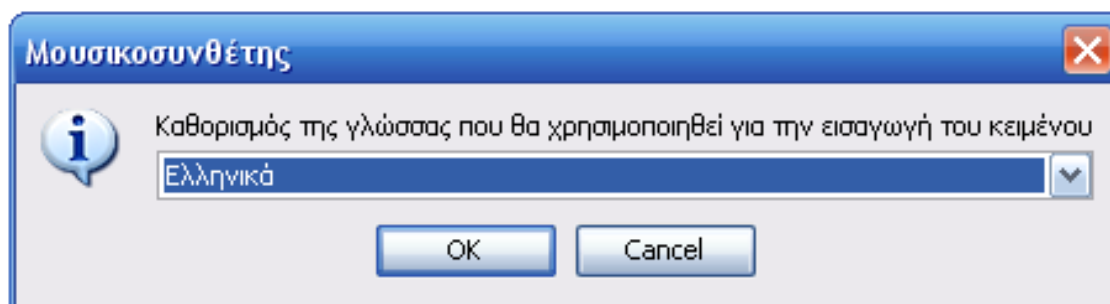
Αρχικά βγαίνει το παράθυρο των χρηστών. Εδώ ο χρήστης επιλέγει το ψευδώνυμο με το οποίο θα μπει στο πρόγραμμα. Για κάθε ψευδώνυμο-χρήστη το πρόγραμμα γνωρίζει τις επιλογές του και έτσι θα παραμετροποιήσει αναλόγως την λειτουργία του.



The screenshot shows a dialog box titled "ΕΠΙΛΕΞΤΕ Η ΔΗΜΙΟΥΡΓΗΣΤΕ ΕΝΑΝ ΧΡΗΣΤΗ / CHOOSE OR CREATE A USER". It contains a dropdown menu for "Επιλογή χρήστη / Choose user" with "Default" selected. Below it is a "Είσοδος / Log in" button. Further down is a text input field for "Όνομα νέου χρήστη / New user name" containing "John", which is circled in red. Below that is a dropdown menu for "Γλώσσα προγράμματος / Interface language" with "Ελληνικά (Greek)" selected. At the bottom is a "Δημιουργία χρήστη και είσοδος / Create new user and log in" button.

Εισαγωγή νέου χρήστη με το όνομα John και επιλογή Ελληνικής γλώσσας για την διαπροσωπία

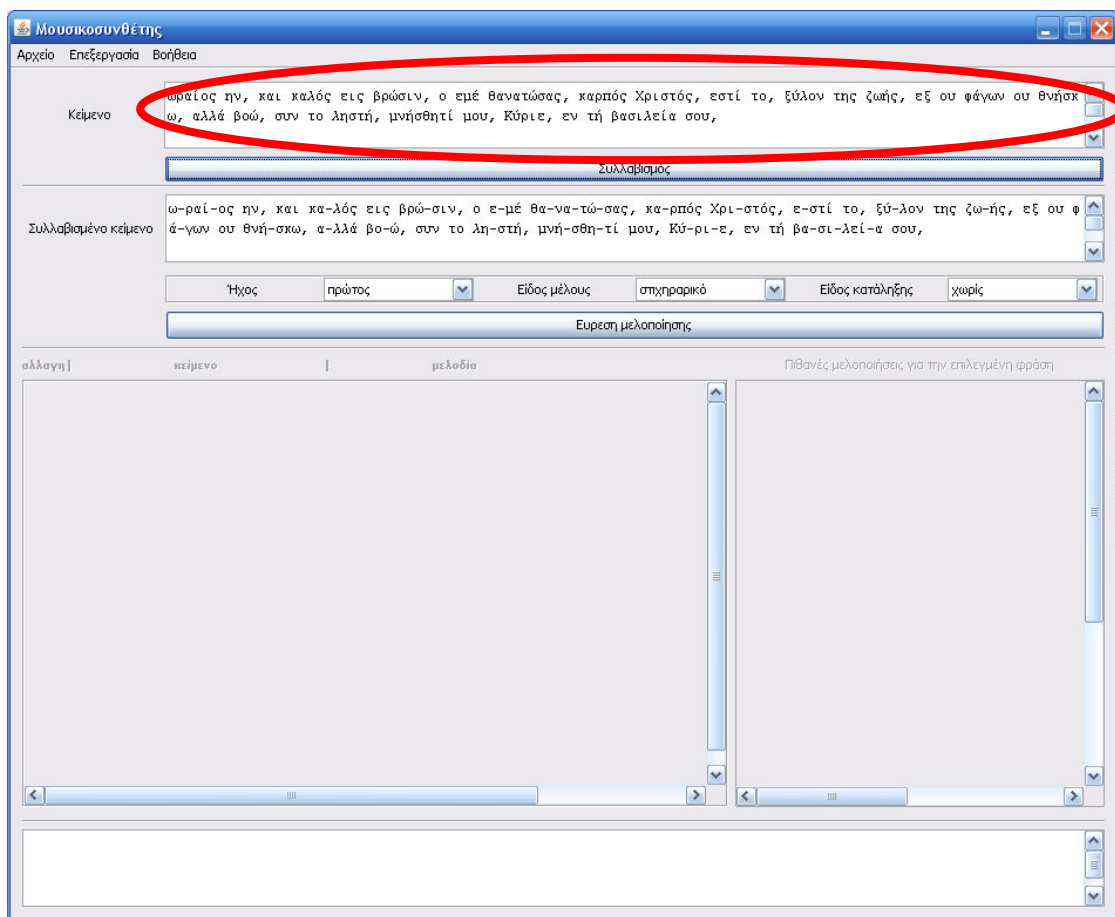
Το δεύτερο παράθυρο είναι αυτό στο οποίο ο χρήστης επιλέγει την γλώσσα εισαγωγής.



The screenshot shows a dialog box titled "Μουσικοσυνθέτης". It features an information icon on the left and a text label: "Καθορισμός της γλώσσας που θα χρησιμοποιηθεί για την εισαγωγή του κειμένου". Below the label is a dropdown menu with "Ελληνικά" selected. At the bottom are "OK" and "Cancel" buttons.

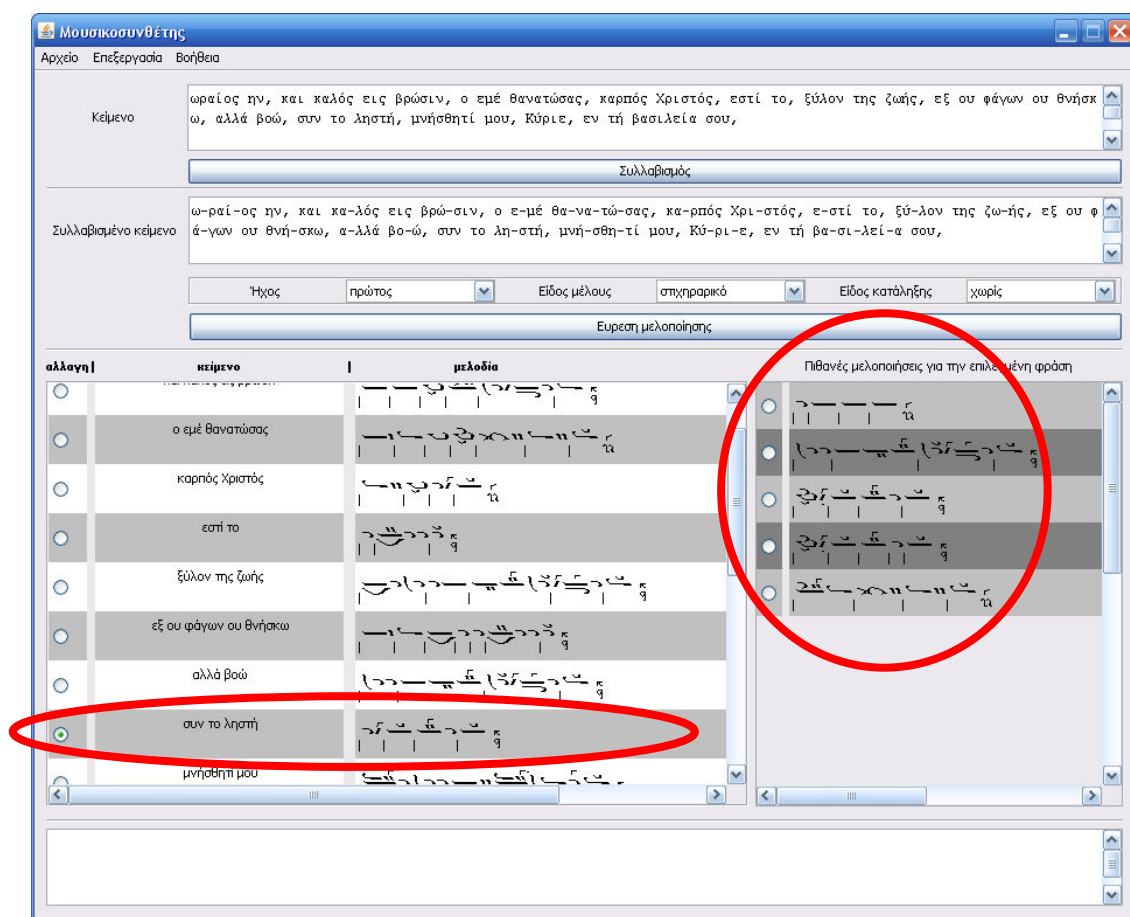
Επιλογή γλώσσας για το ποιητικό κείμενο

Κατόπιν ο χρήστης εισάγει το ποιητικό κείμενο στην γλώσσα την οποία προηγουμένως επέλεξε. Πατώντας το κουμπί του συλλαβισμού λαμβάνει στο επόμενο πεδίο κειμένου το συλλαβισμένο κείμενο. Αν θέλει μπορεί να επέμβει σε αυτό αλλάζοντας τον συλλαβισμό ή τους τονισμούς.



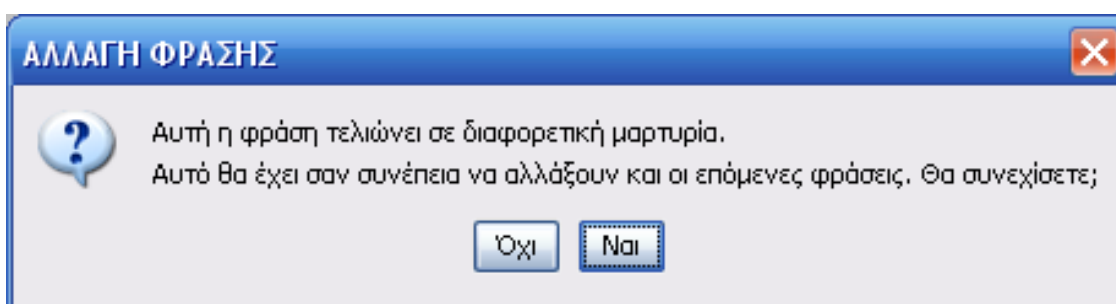
Εισαγωγή του κειμένου, εμφάνιση αυτόματου συλλαβισμού

Αφού ο χρήστης κάνει τις επιλογές του για τον ήχο, το είδος και την κατάληξη της μελοποίησης πατά το αντίστοιχο κουμπί για να πάρει την μελοποίηση. Αφού βρεθεί και παρουσιαστεί στην διαπροσωπία του προγράμματος ο χρήστης μπορεί επιλέγοντας μια φράση να δει στα δεξιά ποιές άλλες μελοποιήσεις ταιριάζουν στην συγκεκριμένη φράση και αν δει κάποια η οποία τον ικανοποιεί περισσότερο την επιλέγει.



Εμφάνιση μελοποίησης, επιλογή της φράσης «συν το ληστή» για αλλαγή

Αν η νέα μελοποίηση καταλήγει σε άλλο φθόγγο θα πρέπει φυσικά να αλλάξουν και οι επόμενες φράσεις. Επειδή όμως ο χρήστης μπορεί να μην θέλει κάτι τέτοιο εμφανίζεται το παράθυρο επιβεβαίωσης.



Επιλέγοντας την πρώτη μελοποίηση η οποία λήγει σε γα (αντί για πα που ήταν πριν) εμφανίζεται το παραπάνω παράθυρο για επιβεβαίωση



Αν ο χρήστης συμφωνήσει γίνεται η αλλαγή της φράσης.

Μουσικοσυνθέτης

Αρχείο Επεξεργασία Βοήθεια

Κείμενο  
ωραίος ην, και καλός εις βρώσιν, ο εμέ θανατώσας, καρπός Χριστός, εστί το, ξύλον της ζωής, εξ ου φάγων ου θνήσκω, αλλά βοώ, συν το ληστή, μνήσθητί μου, Κύριε, εν τή βασιλεία σου,

Συλλαβισμός  
ω-ραί-ος ην, και κα-λός εις βρώ-σιν, ο ε-μέ θα-να-τώ-σας, κα-ρπός Χρι-στός, ε-στί το, ξύ-λον της ζω-ής, εξ ου φά-γων ου θνή-σκω, α-λλά βο-ώ, συν το λη-στή, μνή-σθη-τί μου, Κύ-ρι-ε, εν τή βα-σι-λεί-α σου,

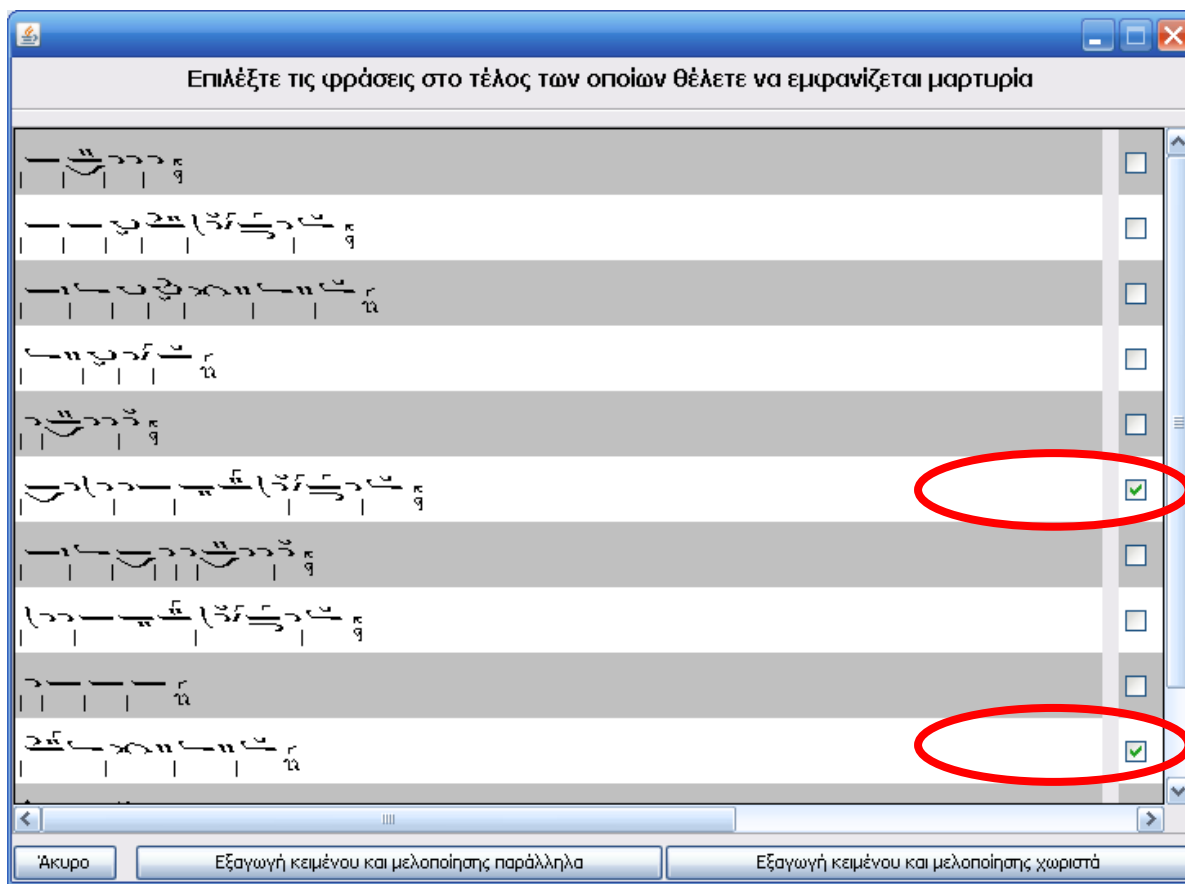
Ήχος πρώτος Είδος μέλους σπληνρικό Είδος κατάληξης χωρίς

Εύρεση μελοποίησης

αλλαγή	κείμενο	μελοδία	Πιθανές μελοποιήσεις για την επιλεγμένη φράση
<input type="radio"/>	καρπός Χριστός	...	...
<input type="radio"/>	εστί το	...	...
<input type="radio"/>	ξύλον της ζωής	...	...
<input type="radio"/>	εξ ου φάγων ου θνήσκω	...	...
<input type="radio"/>	αλλά βοώ	...	...
<input checked="" type="radio"/>	συν το ληστή	...	...
<input type="radio"/>	μνήσθητί μου	...	...
<input type="radio"/>	Κύριε	...	...

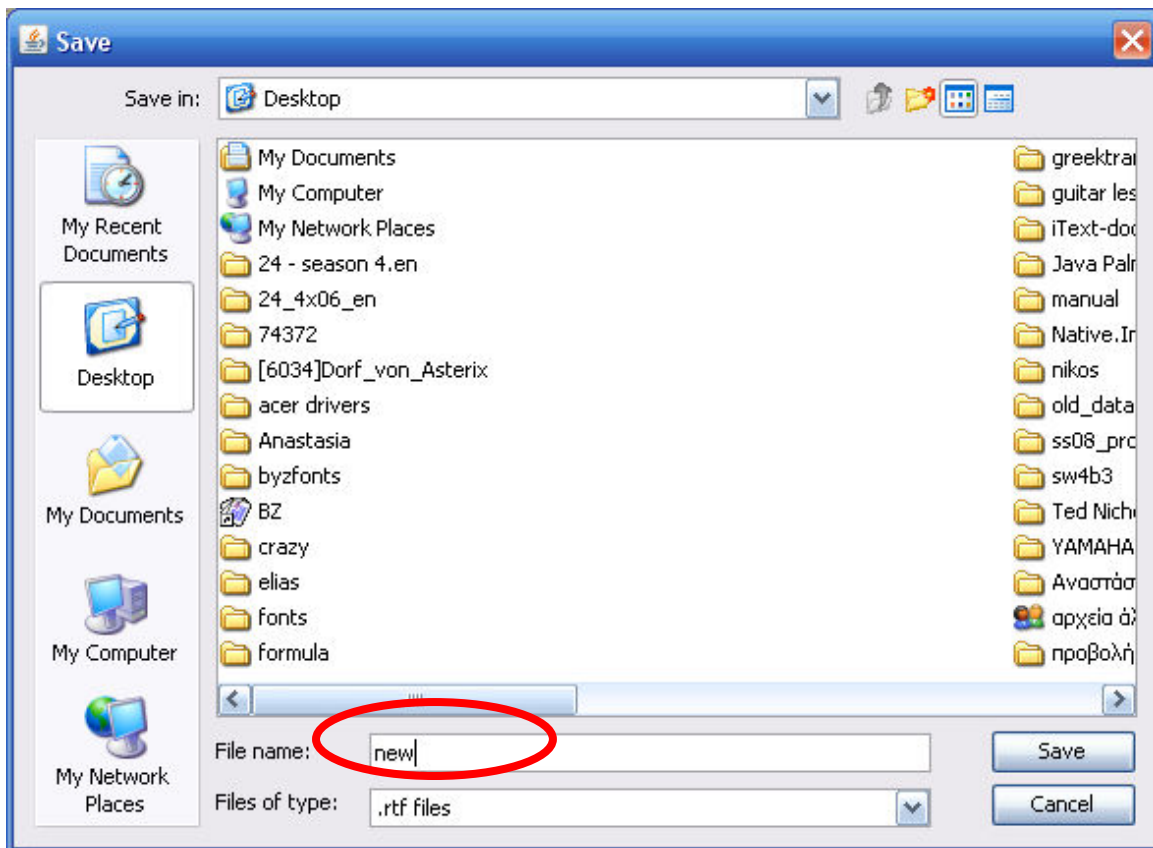
Πράγματι, η φράση άλλαξε, τώρα ο χρήστης αν δεν θέλει να κάνει άλλες αλλαγές θα κάνει εξαγωγή του κειμένου σε αρχείο .rtf

Αν θέλει ο χρήστης μπορεί οποιαδήποτε στιγμή να εξάγει τα αποτελέσματα σε αρχείο κειμένου. Επιλέγοντας «Εξαγωγή» από το μενού θα του παρουσιαστεί το παρακάτω παράθυρο. Εδώ κάνει κάποιες ρυθμίσεις σχετικά με την μορφή που θα έχει το εξαγόμενο κείμενο και πατά το ανάλογο κουμπί.



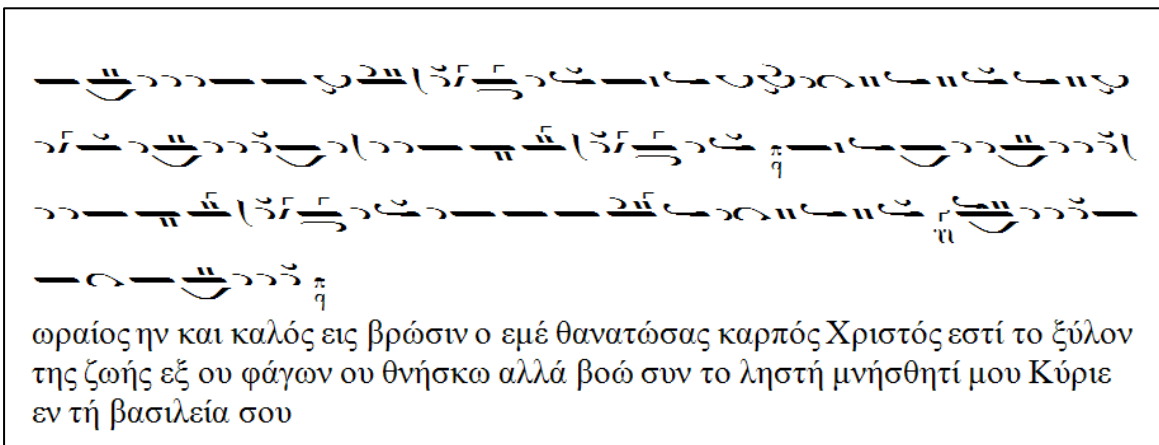
Στο παράθυρο εξαγωγής ο χρήστης επιλέγει να εμφανιστούν οι μαρτυρίες των φράσεων 6 και 10 και να είναι χωριστά; Το κείμενο απ' τη μελοποίηση

Του παρουσιάζεται ένα παράθυρο όπου μπορεί να αναζητήσει την κατάλληλη τοποθεσία για την αποθήκευση του αρχείου.



Επιλέγει να σώσει το αρχείο στην επιφάνεια εργασίας με το όνομα new

Τέλος μπορεί να ανοίξει το αρχείο με κάποιο πρόγραμμα (όπως το Microsoft Word) και να δει ή να εκτυπώσει τα αποτελέσματα.



Ανοίγουμε το new.rtf

### **5.2.2 Μηνύματα**

Πολλές φορές το πρόγραμμα δεν βρίσκει μελοποίηση για κάποια φράση. Αυτό μπορεί να συμβαίνει για διάφορους λόγους

- 1) Υπάρχουν μελοποιήσεις για τον κωδικό της φράσης στη βάση αλλά δεν ταιριάζουν με τα προηγούμενα. Ότι μελοποίηση και να μπει δηλαδή στις προηγούμενες φράσεις σε καμία περίπτωση δεν μπορέσουν να φτάσουν στα τονικά ύψη από τα οποία μπορεί να ξεκινήσει η συγκεκριμένη φράση.
- 2) Δεν υπάρχει τίποτα στη βάση για τον κωδικό της φράσης. Ασφαλώς οι κωδικοί των φράσεων είναι στην πραγματικότητα άπειροι και στη βάση δεν έχουν καταγραφεί τα πάντα. Αυτό μπορεί όμως και να συμβαίνει λόγω λάθους κατά την πληκτρολόγηση.

# 6

## *Επίλογος*

Στην πρώτη ενότητα του κεφαλαίου συνοψίζεται το θέμα της διπλωματικής εργασίας και παρουσιάζονται τα συμπεράσματα που προέκυψαν κατά την διαδικασία ανάλυσης, σχεδίασης και υλοποίησης του συστήματος. Στην δεύτερη ενότητα γίνεται αναφορά σε προτεινόμενους τρόπους επέκτασης της εφαρμογής.

### *6.1 Σύνοψη και Συμπεράσματα*

Η παρούσα διπλωματική δίνει λύσεις σε πολλά προβλήματα που συναντά ένας συνθέτης Βυζαντινής μουσικής. Το πιο βασικό προτέρημα του Μουσικοσυνθέτη είναι το ότι δίνει τη δυνατότητα να πάρει πολύ γρήγορα μια μελοποίηση για το κείμενο που εισήγαγε αλλά επιπλέον του δίνει την δυνατότητα να επέμβει εις βάθος σε αυτή τροποποιώντας την κάθε φράση ξεχωριστά.

Η διπλωματική αυτή και κυρίως το προγραμματιστικό κομμάτι κινήθηκε σε έναν χώρο ανεξερεύνητο. Για να μπορέσει η Java να διαβάσει τα αρχεία της βάσης χρειάστηκε να ξαναγραφτούν σε άλλη μορφή. Χρησιμοποιήθηκαν για αυτό τον σκοπό και μακροεντολές του Microsoft Word (τα αρχεία της βάσης ήταν σε έγγραφα του Word) αλλά η δουλειά έγινε κυρίως με το χέρι. Για να μπορέσει επίσης η Java να γράψει τις Βυζαντινές νότες σε έγγραφα του Word χρειάστηκε να χρησιμοποιηθεί η βιβλιοθήκη iText, σε συνδυασμό με τις Βυζαντινές γραμματοσειρές (οι οποίες επίσης έπρεπε να μετατραπούν σε Unicode μορφή για να λειτουργήσουν σωστά με την Java).

Για όλα τα παραπάνω δεν υπήρχε κάποια έτοιμη λύση και χρειάστηκε να καταβληθεί πολύ προσπάθεια για να δουλέψουν όλα αυτά μαζί χωρίς προβλήματα.

Παρόλα αυτά ο στόχος επετεύχθη, ο Μουσικοσυνθέτης είναι πλέον ένα πρόγραμμα πολύ χρήσιμο για όσους ασχολούνται με την Βυζαντινή μουσική.

## **6.2 Μελλοντικές Επεκτάσεις**

### **6.2.1 Προσθήκη άλλων βάσεων**

Μελλοντικά θα μπορούσαν προστεθούν και επιπλέον βάσεις μελωδιών, πιο πλήρεις και πιο εκτεταμένες. Ο χρήστης θα μπορούσε μάλιστα να επιλέγει (ενδεχομένως από μία λίστα η οποία θα υπάρχει στην διαπροσωπία της βάσης) την βάση στην οποία κάθε φορά θα θέλει να αναζητήσει την μελοποίηση.

### **6.2.2 Αυτόματος Συλλαβισμός σε Άλλες Γλώσσες**

Μέχρι στιγμής γίνεται αυτόματος συλλαβισμός όταν ο χρήστης εισάγει Ελληνικό κείμενο ενώ αν γράψει σε άλλη γλώσσα θα πρέπει να χωρίσει τις συλλαβές με παύλες (-) πράγμα το οποίο δεν είναι πολύ βολικό. Ο αυτόματος συλλαβισμός και για αυτές τις γλώσσες θα απαλλάξει τον χρήστη από αυτόν τον κόπο.

### **6.2.3 Παρουσίαση Μελοποίησης**

Ο τρόπος με τον οποίο παρουσιάζεται η λύση στον χρήστη (σε αρχείο του Microsoft Word) για την ώρα είναι συμβατός με τη μορφή του πακέτου BZ. Μελλοντικά θα μπορούσαν να συνδεθούν στην εφαρμογή και άλλα πακέτα και προγράμματα συγγραφής.

# 7

## *Βιβλιογραφία*

- [Χρυ32] Χρυσάνθου Αρχιεπισκόπου Διρραχίου του εκ Μαδύτων, "Θεωρητικόν Μέγα της Μουσικής", Τεργέστη, 1832
- [Τρε49] Παναγιώτου Ν. Τρεμπέλα, "Εκλογή Ελληνικής Ορθοδόξου Υμνογραφίας", Αθήναι, 1949
- [Ψαχ80] Ψάχου Κ., "Το Οκτάηχον σύστημα της Βυζαντινής Μουσικής", Εκδόσεις Μιχ. Ι. Πολυχρονάκης, Νεάπολις Κρήτης, 1980
- [Rea40] G. Reese, "Music in the Middle Ages", 1940
- [Αγα55] Παναγιώτου Αγαθοκλέους, "Θεωρητικόν της Εκκλησιαστικής Μουσικής", 1855
- [Ευθ97] Αβραάμ Χ. Ευθυμιάδη, "Μαθήματα Βυζαντινής Εκκλησιαστικής Μουσικής", εκδ. Δ', Θεσσαλονίκη, 1997
- [Καρ33] Σίμωνος Καρά, Η Βυζαντινή μουσική σημειογραφία, Αθήναι, 1933
- [Μαρ74] Μαργαζιώτου Ιωάννου, "Θεωρητικόν Βυζαντινής Εκκλησιαστικής Μουσικής", Αθήνα, 1974
- [Itx] <http://itextdocs.lowagie.com/tutorial/>
- [net] <http://www.netbeans.org/community/sources/hg.html>
- [jav] <http://java.sun.com/javase/>
- [ant] <http://www.stanthonysmonastery.org/music/Odigies.doc>

[mel] <http://www.melodos.com/index2.htm>  
[gbl] <http://cgi.di.uoa.gr/~gbelis/>  
[byz] <http://www.byzantinikalamos.com/>