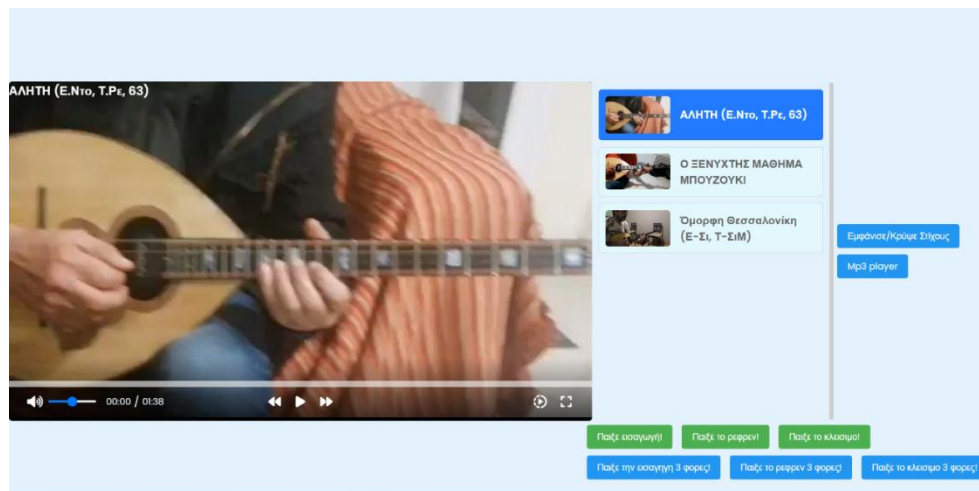


ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

«Εφαρμογή υποβοήθησης εκμάθησης μουσικών
έργων»



Του φοιτητή
Θεοδοσιάδη Σταμάτιου
Αρ. Μητρώου: 123919

Επιβλέπων
Δεληγιάννης Ιγνάτιος
Βαθμίδα Καθηγητής

Τίτλος Δ.Ε. Εφαρμογή υποβοήθησης εκμάθησης μουσικών έργων

Κωδικός Δ.Ε. ...

Όνοματεπώνυμο φοιτητή Θεοδοσιάδη Σταμάτιου

Όνοματεπώνυμο εισηγητή ...
Ημερομηνία ανάληψης Δ.Ε. ...
Ημερομηνία περάτωσης Δ.Ε. ...

Βεβαιώνω ότι είμαι ο συγγραφέας αυτής της εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, έχω καταγράψει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών, εικόνων και κειμένου, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επιπλέον, βεβαιώνω ότι αυτή η εργασία προετοιμάστηκε από εμένα προσωπικά, ειδικά ως διπλωματική εργασία, στο Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του ΔΙ.ΠΑ.Ε.

Η παρούσα εργασία αποτελεί πνευματική ιδιοκτησία του φοιτητή Θεοδοσιάδη Σταμάτιου που την εκπόνησε/αν. Στο πλαίσιο της πολιτικής ανοικτής πρόσβασης, ο συγγραφέας/δημιουργός εκχωρεί στο Διεθνές Πανεπιστήμιο της Ελλάδος άδεια χρήσης του δικαιώματος αναπαραγωγής, δανεισμού, παρουσίασης στο κοινό και ψηφιακής διάχυσης της εργασίας διεθνώς, σε ηλεκτρονική μορφή και σε οποιοδήποτε μέσο, για διδακτικούς και ερευνητικούς σκοπούς, άνευ ανταλλάγματος. Η ανοικτή πρόσβαση στο πλήρες κείμενο της εργασίας, δεν σημαίνει καθ' οιονδήποτε τρόπο παραχώρηση δικαιωμάτων διανοητικής ιδιοκτησίας του συγγραφέα/δημιουργού, ούτε επιτρέπει την αναπαραγωγή, αναδημοσίευση, αντιγραφή, πώληση, εμπορική χρήση, διανομή, έκδοση, μεταφόρτωση (downloading), ανάρτηση (uploading), μετάφραση, τροποποίηση με οποιονδήποτε τρόπο, τμηματικά ή περιληπτικά της εργασίας, χωρίς τη ρητή προηγούμενη έγγραφη συναίνεση του συγγραφέα/δημιουργού.

Η έγκριση της διπλωματικής εργασίας από το Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του Διεθνούς Πανεπιστημίου της Ελλάδος, δεν υποδηλώνει απαραίτητως και αποδοχή των απόψεων του συγγραφέα, εκ μέρους του Τμήματος.

«Αφιερώνω τη πτυχιακή εργασία στην οικογένεια μου, και στον υπεύθυνο καθηγητή μου που με στήριξαν, βοήθησαν, καθοδήγησαν και έδειξαν την μέγιστη κατανόηση σε οποιαδήποτε δυσκολία με τον δικό τους τρόπο στην ολοκλήρωση αυτής της εργασίας».

Πρόλογος

Από πολύ νωρίς στην εξελικτική πορεία του ανθρώπινου γένους παρατηρούμε μια διαρκή ανάπτυξη γνώσεων και ικανοτήτων σε πολλαπλούς τομείς. Τομείς όπως οι επιστήμες, τεχνολογία, φιλοσοφία και η γραφή, έχουν πλέον φτάσει σε ένα πολύ υψηλό επίπεδο που βελτιώνουν την καθημερινή μας ζωή. Μέσα σε αυτά δεν μπορούμε να εξαιρέσουμε και να μην συμπεριλάβουμε τις τέχνες και κυριότερα την τέχνη της μουσικής. Η μουσική ιστορικά αποτελούσε πάντα ένα αναπόσπαστο κομμάτι για κάθε πολιτισμό. Ακόμα και σήμερα είναι μία κοινή μορφή έκφρασης, κατανοητή από κάθε λαό ασχέτως των διαφορών ανάμεσα τους. Έτσι ακόμα και σήμερα είναι ένας διαρκώς αναπτυσσόμενος κλάδος τέχνης, όπου οι άνθρωποι αισθάνονται την ανάγκη είτε να την απολαμβάνουν ακούγοντάς την είτε προσπαθώντας να την κατανοήσουν και να την μάθουν.

Περίληψη

Η συγκεκριμένη πτυχιακή εργασία έχει ως βασικό στόχο να προσφέρει σε έμπειρους ή και άπειρους ανθρώπους ένα εργαλείο εκμάθησης μουσικών έργων. Υπάρχουν αρκετές εφαρμογές που αποσκοπούν στο ίδιο αποτέλεσμα, όπου πολλές από αυτές χρησιμοποιούν διαφορετικές προσεγγίσεις, ωστόσο είναι κατά βάση συνδρομητικές και συνήθως αξιοποιούν ανθρώπινο δυναμικό κάνοντας το κόστος αυτών αρκετά υψηλό. Έτσι, άτομα με οικονομικές δυσκολίες αδυνατούν να έχουν πρόσβαση σε αυτά, πράγμα που κάνει την εκμάθηση της μουσικής λιγότερο προσβάσιμη προς αυτά. Ο τρόπος που επιτυγχάνεται η εκμάθηση είναι απλός, δίνει στον εκάστοτε χρήστη την δυνατότητα να προσθέτει μόνος του τα κομμάτια που θέλει να μάθει είτε σε μορφή βίντεο είτε σε μορφή ήχου. Παράλληλα μέσω των διαφόρων λειτουργιών αναπαραγωγής του βίντεο ή ηχητικού αρχείου, το πρόγραμμα δίνει την επιλογή στον χρήστη να μάθει το μουσικό κομμάτι είτε οπτικά είτε ακουστικά.

Abstract

The main goal of this thesis is to offer experienced or inexperienced people a tool for learning musical works. There are several applications that aim for the same result, where many of them use different approaches, however they are basically subscription-based and usually use human resources making their costs quite high. Thus, people with financial difficulties are unable to access them, which makes learning music less accessible to them. The way learning is achieved is simple. It gives the user the ability to add the tracks he wants to learn either in video or audio format. At the same time, through the various playback functions of the video or audio file, the program gives the user the option to learn the piece of music either visually or audibly.

Ευχαριστίες

Με την σειρά μου θεωρώ απαραίτητο να ευχαριστήσω την κοινότητα του ανοιχτού κώδικα που έχει κάνει δυνατή την υλοποίηση οποιαδήποτε ιδέας. Χωρίς την καθημερινή συμβολή ανθρώπων τέτοιων κοινοτήτων η ιδέα μου και γενικότερα ο προγραμματισμος θα ήταν λιγότερο προσβάσιμος.

Περιεχόμενα

Πρόλογος	6
Περίληψη	7
Abstract	8
Ευχαριστίες	9
Περιεχόμενα	10
Κατάλογος Σχημάτων	13
Η μεθοδολογία προγραμματισμού Scrum	14
1.1 Εισαγωγή	14
1.2 Τι είναι το Scrum;	14
1.3 Το Scrum Framework	14
1.4 Scrum artifacts	15
1.5 Scrum Roles	16
Έννοιες Front-End και Back-End	17
2.1 Εισαγωγή	17
2.2 Front-End	18
2.3 Back-End	20
Επιλογή Εργαλείων και Γλωσσών	21
3.1 Εργαλείο σύνταξης κώδικα VS code	21
3.2 Γιατι HTML:	22
3.3 Πλεονεκτήματα HTML:	23
3.4 Πως λειτουργεί η HTML:	23
3.5 Πως χρησιμοποιείται η HTML:	24
3.6 Τι είναι το CSS	24
3.7 Πλεονεκτήματα του CSS	24
3.8 Πως λειτουργεί το CSS	25
3.9 Τι είναι η JavaScript	25
3.10 Πως λειτουργεί η JavaScript	26
3.11 Πλεονεκτήματα της JavaScript	26
Γραφικό περιβάλλον της εφαρμογής	29

4.1	Εισαγωγή	29
4.2	Ανάλυση του γραφικού περιβάλλοντος	29
4.3	Γραφικό περιβάλλον του Video player	30
4.3.1	Τα στοιχεία του Video player	30
4.3.2	Κουμπιά ρυθμίσεων και μπάρες του Video player	31
4.3.3	Λίστα αρχείων προς αναπαραγωγή	34
4.3.4	Κουμπιά Εμφάνισε/Κρύψε Στίχους και MP3 player	36
4.3.5	Κουμπιά Αναπαραγωγής από συγκεκριμένο σημείο του αρχείου	37
4.4	Γραφικό περιβάλλον του Audio player	40
4.4.1	Ανάλυση γραφικών στοιχείων Audio Player	41
4.4.2	Λειτουργία κουμπιών "Choose files" και "Video Player"	41
4.4.3	Λειτουργία κουμπιών ρυθμίσεων του Audio Player	42
4.4.4	Λειτουργίες μπάρας σημείου χρονικής ένδειξης και συρόμενης μπάρας για ρύθμιση έντασης ήχου	43
	Ανάλυση κώδικα	43
5.1	Εισαγωγή	43
5.2	Η δομή του project	43
5.3	Το index.html	45
5.3.1	Ανάλυση του index.html	45
5.3.2	Το <head> tag του index.html	45
5.3.3	Το <body> tag του index.html	46
5.4	Ανάλυση του style.css	47
5.5	Ανάλυση του script.js	48
5.5.1	Σταθερές και μεταβλητές του script.js	48
5.5.2	Η μέθοδος formatTime	49
5.5.3	Τα event Listeners και μέθοδοι για την ανανέωση του χρόνου	49
5.5.4	Τα event Listeners και μέθοδοι για τις ρυθμίσεις έντασης ήχου	49
5.5.5	Τα event Listeners και μέθοδοι για τα κουμπιά ταχύτητας αναπαραγωγής και πλήρους οθόνης	50
5.5.6	Τα event Listeners και μέθοδοι για τα κουμπιά play/pause, backward, forward	50
5.5.7	Τα event Listeners και μέθοδοι για τους στίχους	50
5.5.8	Μέθοδοι για λειτουργίες κουμπιών επανάληψης και αναπαραγωγής από συγκεκριμένο σημείο	51

5.6 Ανάλυση του mp3player.html	51
5.6.1 Το <body> tag του mp3player.html	51
5.7 Ανάλυση του scriptMp3.js	52
5.7.1 Μέθοδος formatTime()	52
5.7.2 Το event Listeners fileInput.addEventListener('change', (event) => {...});	53
5.7.3 Τα event Listeners και μέθοδοι για τα κουμπιά ρυθμίσεων και μπάρας ήχου	54

Κατάλογος Σχημάτων

Σχήμα 1: Scrum framework

Σχήμα 3: Visual Studio code sample

Σχήμα 4.1: Συνολικό GUI Video Player

Σχήμα 4.2: Στοιχεία Video Player

Σχήμα 4.3: Ρυθμίσεις Video Player

Σχήμα 4.4: Ρυθμίσεις Video Player

Σχήμα 4.5: Mute button

Σχήμα 4.6: Κεντρικά κουμπιά ρυθμίσεων

Σχήμα 4.7: Pause button

Σχήμα 4.8: Κουμπιά ρύθμισης ταχύτητας αναπαραγωγής και λειτουργίας πλήρους οθόνης

Σχήμα 4.9: Play speed settings

Σχήμα 4.10: Λίστα αρχείων προς αναπαραγωγή

Σχήμα 4.11: Hover on list feature

Σχήμα 4.12: Κουμπιά Εμφάνισε/Κρύψε Στίχους και MP3 player

Σχήμα 4.13: Πεδίο εμφάνισης στίχων

Σχήμα 4.14: Κουμπια αναπαραγωγής συγκεκριμένου σημείου και επαναλήψεων

Σχήμα 4.15: Αποτέλεσμα χρήσης κουμπιού "Παίξε την εισαγωγή!"

Σχήμα 4.16: Αποτέλεσμα χρήσης κουμπιού "Παίξε το ρεφρέν!"

Σχήμα 4.17: Αποτέλεσμα χρήσης κουμπιού "Παίξε το κλείσιμο!"

Σχήμα 4.18: Συνολικό GUI Audio Player

Σχήμα 4.19: Στάδιο επιλογής μουσικού έργου

Σχήμα 4.20: Αποτέλεσμα χρήσης κουμπιού Choose Files

Σχήμα 5.1: Δομή project

Η μεθοδολογία προγραμματισμού Scrum

1.1 Εισαγωγή

Για την ολοκλήρωση της συγκεκριμένης πτυχιακής χρησιμοποιήθηκε ένα πλαίσιο (framework) για την ανάπτυξη, παράδοση και την διατήρηση των λειτουργιών σε ένα σχετικά σύνθετο περιβάλλον με όνομα Scrum.

1.2 Τι είναι το Scrum;

Scrum είναι πλαίσιο επαναληπτικής και επαυξητικής ευέλικτης μεθοδολογίας ανάπτυξης λογισμικού για τη διαχείριση της ανάπτυξης ενός προϊόντος. Ορίζει μια ευέλικτη, ολιστική στρατηγική για την ανάπτυξη του προϊόντος, όπου μια ομάδα ανάπτυξης λειτουργεί ως μονάδα για την επίτευξη ενός κοινού στόχου και επιτρέπει στις ομάδες να αυτοοργανώνονται με την ενθάρρυνση της φυσικής συνεργασίας ή διαδικτυακής συνεργασίας όλων των μελών της ομάδας, καθώς και την καθημερινή πρόσωπο-με-πρόσωπο επικοινωνία μεταξύ όλων των μελών της ομάδας και κλάδων στο σχεδιασμό.

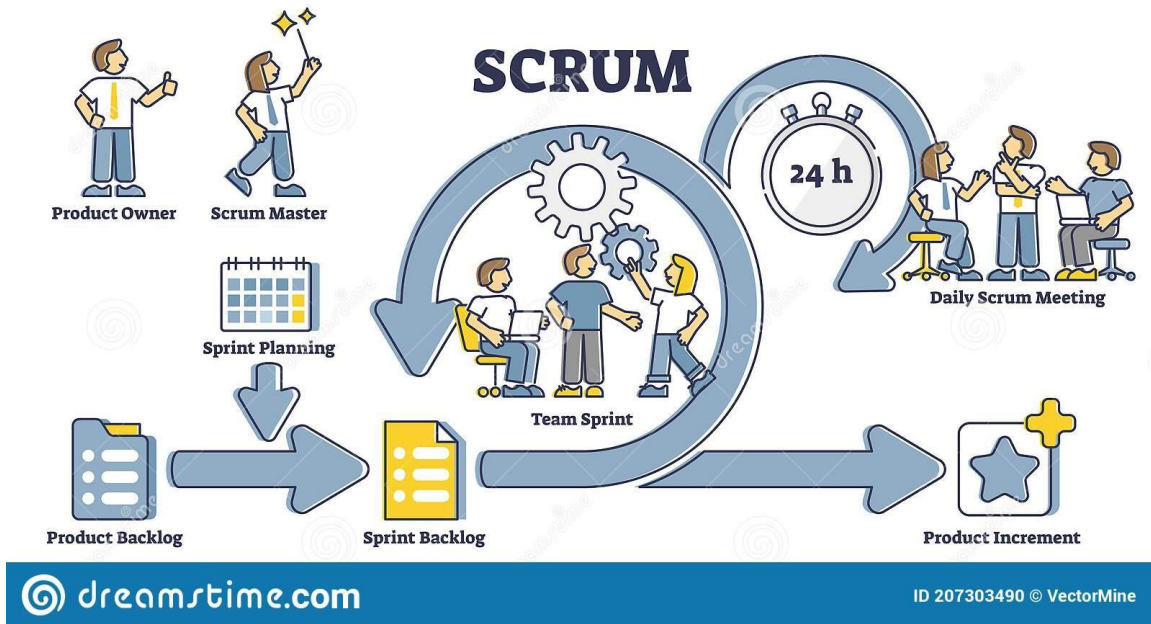
Η λέξη scrum προέρχεται από τη μέθοδο [*scrummage*](#) (στα Αγγλικά) του [Ράγκμπι](#).

Η αρχή-κλειδί του scrum είναι η αναγνώριση του ότι κατά τη διάρκεια της διαδικασίας παραγωγής, οι πελάτες μπορούν να αλλάξουν γνώμη σχετικά με το τι θέλουν και χρειάζονται (συχνά αυτό αναφέρεται ως μεταβλητότητα αναγκών), καθώς και ότι μη αναμενόμενες προκλήσεις δεν είναι εύκολα διαχειρίσιμες με έναν παραδοσιακό προγνωστικό ή σχεδιασμένο τρόπο. Το scrum υιοθετεί μία εμπειρική προσέγγιση αποδεχόμενο ότι το πρόβλημα δεν μπορεί να κατανοηθεί πλήρως ή να οριστεί, εστιάζοντας αντ' αυτού στην μεγιστοποίηση της ικανότητας της ομάδας να διανέμει γρήγορα, να αποκρίνεται σε αναδυόμενες απαιτήσεις και να προσαρμόζεται σε εξελισσόμενες τεχνολογίες και αλλαγές στην κατάσταση της αγοράς.

1.3 Το Scrum Framework

Το Scrum framework είναι ευρετικό στη φύση, σημαίνει ότι αναγνωρίζει το γεγονός ότι μια ομάδα δεν γνωρίζει τα πάντα στην αρχή ενός έργου και η μάθηση εξελίσσεται σταδιακά κατά τη διάρκεια ενός έργου. Είναι κατασκευασμένο με τρόπο που οι ομάδες συνεχίζουν να ανανεώνουν τις προτεραιότητές τους σύμφωνα με τις απαιτήσεις του τελικού χρήστη και να προσαρμόζονται συνεχώς στις κυμαινόμενες συνθήκες.

Το Scrum είναι δομημένο αλλά όχι άκαμπτο στην εφαρμογή του. Η εφαρμογή και η εκτέλεσή του μπορούν να προσαρμοστούν στις ανάγκες οποιουδήποτε οργανισμού.



Σχήμα 1: Scrum framework

Το πλαίσιο Scrum αποτελείται από:

- **Scrum Artifacts**
- **Scrum Roles**
- **Scrum Rules**
- **Scrum Events**

1.4 Scrum artifacts

Στο scrum μπορούμε να ξεχωρίσουμε τρία διαφορετικά artifacts. Τα artifacts είναι κάτι που φτιάχνουμε όπως ένα εργαλείο για την επίλυση ενός προβλήματος. Αυτά τα artifacts είναι τα εξής:

1. Το **Product Backlog**, είναι η κύρια λίστα που πρέπει να ολοκληρωθεί και διατηρείται κατά κύριο λόγο από τον product manager. Αυτή είναι μία δυναμική λίστα δυνατοτήτων, απαιτήσεων, βελτιστοποιησης και επιδιορθώσεων που λειτουργεί ως είσοδος για το ανεκτέλεστο sprint. Είναι ουσιαστικά η λίστα των «To-Do» της ομάδας. Το ανεκτέλεστο προϊόν επανεξετάζεται συνεχώς, ιεραρχείται εκ νέου και διατηρείται από τον ιδιοκτήτη του προϊόντος επειδή η αγορά αλλάζει συνεχώς, και πρέπει τα σχετικά προβλήματα που προκύπτουν να αλλάζουν συνεχώς.

2. Το **Sprint Backlog**, είναι η λίστα με στοιχεία, ιστορίες χρήστη (user stories), ή διορθώσεις σφαλμάτων που επιλέγονται από την ομάδα ανάπτυξης για εφαρμογή στον τρέχοντα κύκλο sprint. Πριν από κάθε sprint στην συνάντηση προγραμματισμού sprint η ομάδα ορίζει σε ποια αντικείμενα από το product backlog του sprint θα εργαστεί. Ένα sprint backlog μπορεί να είναι ευέλικτο και μπορεί να εξελιχθεί κατά τη διάρκεια ενός sprint.
3. Το **Increment** (ή αλλιώς sprint goal), είναι το χρήσιμο τελικό προϊόν ενός sprint. Κάθε Increment προστίθεται στις ήδη υπάρχουσες αυξήσεις και επαληθεύεται διεξοδικά. Με αυτό τον τρόπο διασφαλίζεται ότι όλες οι προσαυξήσεις λειτουργούν μαζί. Μπορούν να δημιουργηθούν πολλαπλές προσαυξήσεις σε ένα sprint. Το άθροισμα των προσαυξήσεων παρουσιάζεται στο sprint review υποστηρίζοντας έτσι τον εμπειρισμό. Ωστόσο μία προσαύξηση μπορεί να παραδοθεί στους ενδιαφερόμενους πριν το τέλος του sprint.

1.5 Scrum Roles

Κάθε Ομάδα Scrum αποτελείται από τρεις ηγετικούς ρόλους. Ανεξάρτητα από τον τίτλο της εργασίας σας, μπορείτε να παίξετε οποιονδήποτε από τους παρακάτω ρόλους στο Scrum. Κάθε ένας από αυτούς τους ρόλους έχει ένα καθορισμένο σύνολο ευθυνών και η επιτυχής ολοκλήρωση του έργου εξαρτάται από το πόσο στενά συνεργάζονται μεταξύ τους.

Οι ρόλοι είναι οι εξής:

- **Product Owner**
- **Scrum Team**
- **Scrum Master**

Product Owner

Ο Product Owner, γνωστός και ως PO στη συντομευμένη του μορφή, είναι ένας κεντρικός ρόλος στο πλαίσιο του Scrum. Εκπροσωπεί τον τελικό πελάτη ή τα ενδιαφερόμενα μέρη. Είναι ο κύριος χαρακτήρας στο agile testing, ο οποίος δημιουργεί ιστορίες (stories) και χαρακτηριστικά (features) από τις απαιτήσεις που παρέχονται από τους τελικούς χρήστες ή πελάτες (αναφέρονται επίσης ως Epics/User Stories). Υπάρχει πάντα ένας ιδιοκτήτης προϊόντος σε οποιαδήποτε ομάδα scrum.

Scrum Team

Η Scrum Team είναι μια ομάδα ατόμων όπως προγραμματιστές (developers), δοκιμαστές (testers) κ.λπ. που συνεργάζονται για να παραδώσουν τα ζητούμενα και δεσμευμένα προϊόντα που απαιτεί ο πελάτης ή ο τελικός χρήστης. Η ομάδα scrum δίνει προτεραιότητα στα στοιχεία από το Product Backlog στα οποία πιστεύουν ότι μπορούν να υλοποιηθούν και να παραδοθούν. Η ομάδα Scrum είναι υπεύθυνη για την παράδοση της εργασίας στον πελάτη.

Οι ομάδες Scrum, επίσης γνωστές ως Ομάδες Ανάπτυξης (Development Team), είναι δομημένες και εξουσιοδοτημένες να αυτο-οργανώνονται και να διαχειρίζονται την εργασία τους.

Scrum Master

Ένας Scrum Master, γνωστός και ως SM στη συντομογραφία του, βοηθά την ομάδα scrum να αποδώσει στο υψηλότερο επίπεδο. Προστατεύει την ομάδα τόσο από εσωτερικούς όσο και από εξωτερικούς περισπασμούς. Ο ρόλος του δεν είναι διαχειριστής έργου ή ιδιοκτήτης προϊόντος, αλλά συντονιστής που διασφαλίζει ότι δεν υπάρχουν εμπόδια που μπορούν να παρεμποδίσουν την ικανότητα της ομάδας να παραδίδει προϊόντα και υπηρεσίες.

Έννοιες Front-End και Back-End

2.1 Εισαγωγή

Οι ιστότοποι αποτελούνται από δύο μέρη: το μπροστινό μέρος (Front-End), το οποίο βιώνουν οι χρήστες, και το πίσω μέρος (Back-End), το οποίο περιλαμβάνει την αόρατη δομή που καθιστά δυνατή την πρόσοψη. Η διαφορά μεταξύ ανάπτυξης front-end και back-end μπορεί να φαίνεται μπερδεμένη, δεδομένων των ομοιοτήτων τους στη σωστή λειτουργία των ιστότοπων. Το Frontend και το Backend είναι οι δύο πιο δημοφιλείς όροι που χρησιμοποιούνται στην ανάπτυξη Ιστού. Το μπροστινό μέρος είναι αυτό που βλέπουν και αλληλεπιδρούν οι χρήστες και το backend είναι πώς λειτουργούν όλα. Κάθε πλευρά πρέπει να επικοινωνεί και να λειτουργεί αποτελεσματικά με την άλλη ως ενιαία μονάδα για τη βελτίωση της λειτουργικότητας του ιστότοπου.

Οι προγραμματιστές front-end σχεδιάζουν τις οπτικές πτυχές των ιστότοπων με τους οποίους μπορούν να αλληλεπιδράσουν οι χρήστες, συμπεριλαμβανομένων των χρωμάτων, της διάταξης και των γραμματοσειρών. Οι προγραμματιστές back-end δημιουργούν την αόρατη δομή που βοηθά τους ιστότοπους να λειτουργούν σωστά. Οι προγραμματιστές back-end έχουν

διαφορετικές τεχνικές δεξιότητες και συνήθως λαμβάνουν υψηλότερους μισθούς από τους προγραμματιστές front-end.

2.2 Front-End

Γενικές πληροφορίες για το Front-End

Το frontend ενός προγράμματος λογισμικού ή ενός ιστότοπου είναι οτιδήποτε αλληλεπιδρά ο χρήστης. Από την άποψη του χρήστη, το frontend είναι συνώνυμο με το περιβάλλον εργασίας χρήστη (user interface). Από την άποψη του προγραμματιστή, είναι ο σχεδιασμός της διεπαφής-ιστοσελίδας-λογισμικού και ο προγραμματισμός που κάνουν την ιστοσελίδα λειτουργική προς τον χρήστη.

Ένας από τους πρωταρχικούς στόχους της ανάπτυξης του frontend είναι να δημιουργήσει μια ομαλή εμπειρία χρήστη. Με άλλα λόγια, το μπροστινό μέρος μιας εφαρμογής ή ενός ιστότοπου θα πρέπει να είναι διαισθητικό και εύκολο στη χρήση. Αν και αυτό ακούγεται σαν ένας απλός στόχος, μπορεί να είναι εκπληκτικά περίπλοκος, καθώς δεν είναι όλοι οι χρήστες ή οι συσκευές ίδιοι. Για παράδειγμα, μια εφαρμογή που αναπτύχθηκε για μια φορητή συσκευή απαιτεί μια σημαντικά διαφορετική διεπαφή (interface) από μια εφαρμογή επιτραπέζιου υπολογιστή. Οι ιστότοποι πρέπει να λειτουργούν καλά σε πολλαπλές συσκευές ασχέτως το μέγεθος της οθόνης. Γι' αυτό η σύγχρονη ανάπτυξη ιστού συνήθως περιλαμβάνει δυναμικό σχεδιασμό του user interface ώστε να είναι όμορφο και λειτουργικό σε κάθε συσκευή που μπορεί να χρησιμοποιεί ο χρήστης.

Παραδείγματα στοιχείων του Front-End:

1. application or page layout (διάταξη σελίδας)
2. graphics (γραφικά)
3. audio and video elements (στοιχεία ήχου και βίντεο)
4. text content (περιεχόμενο κειμένου)
5. user interface elements (buttons, links, [toolbars](#), navigation bars, etc.)
6. input areas ([dialog boxes](#), form [fields](#), text areas, etc.)
7. user flow (how one interface leads to the next)

Γλώσσες προγραμματισμού front-end

Το τμήμα του front-end δημιουργείται χρησιμοποιώντας ορισμένες γλώσσες που αναλύονται παρακάτω:

HTML:

Η HTML σημαίνει Hypertext Markup Language (γλώσσα σήμανσης υπερκειμένου). Χρησιμοποιείται για το σχεδιασμό του μπροστινού τμήματος των ιστοσελίδων χρησιμοποιώντας μια γλώσσα σήμανσης. Η HTML είναι ένας συνδυασμός Hypertext (υπερκειμένου) και Markup language (γλώσσας σήμανσης). Γενικά υπάρχει η άποψη ότι η HTML δεν μπορεί να θεωρηθεί από μόνη σαν μία γλώσσα προγραμματισμού. Παρόλου που σχεδόν όλο το σύγχρονο διαδίκτυο και οι περισσότερες ιστοσελίδες βασίζονται σε αυτή.

CSS:

Τα Cascading Style Sheets (CSS) είναι μια style sheet γλώσσα που χρησιμοποιείται για την περιγραφή της παρουσίασης ενός εγγράφου γραμμένου σε μια γλώσσα σήμανσης όπως η HTML ή η XML (συμπεριλαμβανομένων των διαλέκτων XML όπως SVG, MathML ή XHTML). Το CSS είναι μια τεχνολογία ακρογωνιαίου λίθου του Παγκόσμιου Ιστού, μαζί με HTML και JavaScript.

Το CSS έχει σχεδιαστεί για να επιτρέπει τον διαχωρισμό περιεχομένου και παρουσίασης, συμπεριλαμβανομένων της διάταξης, των χρωμάτων και των γραμματοσειρών. Αυτός ο διαχωρισμός μπορεί να βελτιώσει την προσβασιμότητα στο περιεχόμενο της σελίδας, παρέχει μεγαλύτερη ευελιξία και έλεγχο στην προδιαγραφή των χαρακτηριστικών παρουσίασης. Επίσης δίνει την δυνατότητα για κοινή χρήση μορφοποίησης σε πολλαπλές σελίδες. Το σχετικό αρχείο CSS καθορίζεται σε ένα ξεχωριστό αρχείο .css, το οποίο μειώνει την πολυπλοκότητα και την επανάληψη στο δομικό περιεχόμενο.

JavaScript:

Η JavaScript, που συχνά συντομεύεται ως JS, είναι μια γλώσσα προγραμματισμού που αποτελεί μια από τις βασικές τεχνολογίες του Παγκόσμιου Ιστού, μαζί με την HTML και το CSS. Από το 2022, το 98% των ιστότοπων χρησιμοποιούν JavaScript στην πλευρά του πελάτη (client side) για τη συμπεριφορά της ιστοσελίδας, συχνά ενσωματώνοντας πάνω της βιβλιοθήκες τρίτων. Όλα τα μεγάλα προγράμματα περιήγησης ιστού διαθέτουν ειδική μηχανή JavaScript για την εκτέλεση του κώδικα στις συσκευές των χρηστών.

Η JavaScript είναι μια γλώσσα υψηλού επιπέδου, η οποία συμμορφώνεται με το πρότυπο ECMAScript. Διαθέτει δυναμική πληκτρολόγηση, αντικειμενοστραφή προσανατολισμό που βασίζεται σε πρωτότυπα και λειτουργίες πρώτης κατηγορίας. Είναι πολλαπλών παραδειγμάτων,

υποστηρίζοντας στυλ προγραμματισμού που βασίζονται σε γεγονότα, λειτουργικά και επιτακτικά. Διαθέτει διεπαφές προγραμματισμού εφαρμογών (API) για εργασία με κείμενο (text), ημερομηνίες (date), κανονικές εκφράσεις (regular expressions), τυπικές δομές δεδομένων και το μοντέλο αντικειμένου εγγράφου Document Object Model (DOM). Το πολύ μεγάλο πλεονέκτημα της αποτελεί ότι είναι εφαρμόσιμη τόσο στο front-end όσο και στο back-end.

Υπάρχουν πολλές άλλες γλώσσες μέσω των οποίων μπορεί κανείς να κάνει ανάπτυξη front-end κώδικα ανάλογα με το framework που επιλέγει, για παράδειγμα το Flutter χρησιμοποιεί Dart, η React χρησιμοποιεί JavaScript και το Django χρησιμοποιεί Python.

2.3 Back-End

Γενικές πληροφορίες για το Back-End

Το backend είναι η πλευρά του διακομιστή του ιστότοπου. Αποθηκεύει και τακτοποιεί δεδομένα και διασφαλίζει επίσης ότι όλα στην πλευρά του πελάτη του ιστότοπου λειτουργούν καλά. Είναι μέρος του ιστότοπου που δεν μπορούμε να δούμε και να αλληλεπιδράσουμε. Είναι το τμήμα του λογισμικού που δεν έρχεται σε άμεση επαφή με τους χρήστες. Τα εξαρτήματα και τα χαρακτηριστικά που αναπτύχθηκαν από σχεδιαστές backend είναι έμμεσα προσβάσιμα από τους χρήστες μέσω μιας εφαρμογής front-end. Δραστηριότητες, όπως η συγγραφή API, η δημιουργία βιβλιοθηκών και η εργασία με στοιχεία συστήματος χωρίς διεπαφές χρήστη ή ακόμα και συστήματα επιστημονικού προγραμματισμού, περιλαμβάνονται επίσης στο backend.

Οι προγραμματιστές back-end χρειάζονται προηγμένη εμπειρία σε γλώσσες προγραμματισμού από την πλευρά του διακομιστή, όπως Java, Python και Ruby για τη δημιουργία εφαρμογών. Οι προγραμματιστές back-end χρησιμοποιούν εργαλεία όπως ο SQL Server και η Oracle για την αποθήκευση, την οργάνωση και την αλλαγή δεδομένων. Οι προγραμματιστές back-end συνεργάζονται με προγραμματιστές front-end, για να κατανοήσουν και να επιτύχουν τους στόχους κάθε έργου.

Γλώσσες προγραμματισμού Back-End

Το τμήμα του front-end δημιουργείται χρησιμοποιώντας ορισμένες γλώσσες που αναλύονται παρακάτω:

PHP:

Η PHP είναι μια γλώσσα δέσμης ενεργειών από την πλευρά του διακομιστή που έχει σχεδιαστεί ειδικά για την ανάπτυξη Ιστού. Εφόσον ο κώδικας PHP εκτελείται από την πλευρά του διακομιστή, ονομάζεται server-side scripting language (γλώσσα από την πλευρά του διακομιστή).

C++:

Είναι μια γλώσσα προγραμματισμού γενικής χρήσης και χρησιμοποιείται ευρέως στις μέρες μας για ανταγωνιστικό προγραμματισμό. Χρησιμοποιείται επίσης ως γλώσσα back-end

Java:

Η Java είναι μια από τις πιο δημοφιλείς και ευρέως χρησιμοποιούμενες γλώσσες προγραμματισμού και πλατφόρμες.

Python:

Η Python είναι μια γλώσσα προγραμματισμού που μας επιτρέπει να εργαζόμαστε γρήγορα και να ενσωματώνουμε συστήματα πιο αποτελεσματικά. Είναι μια πολύ σημαντική γλώσσα για το back-end. Επίσης, είναι μια γλώσσα που στις μέρες μας αποκτά όλο και περισσότερη φήμη.

Επιλογή Εργαλείων και Γλωσσών

Οι 3 γλώσσες που επιλέχθηκαν για την υλοποίηση του λογισμικού είναι οι εξής: **HTML, CSS, JavaScript**. Το εργαλείο για την σύνταξη του κώδικα (code editor) που επιλέχθηκε είναι το **Visual Studio Code**. Η επιλογή των γλωσσών προγραμματισμού καθορίστηκε αρχικά διότι θεωρούνται εύκολες στην δομή τους, στο συντακτικό τους καθώς είναι 3 γλώσσες όπου κατά την διάρκεια των σπουδών μου διδάχτηκα σε έναν ικανοποιητικό βαθμό. Ακόμη είναι γλώσσες όπου μπορεί άνετα ο καθένας να βρεί υλικό εκμάθησης στο διαδίκτυο. Επίσης, παρέχεται μεγάλη βοήθεια και απαντήσεις σε πολλά ερωτήματα ή προβληματισμούς που εμφανίζονται κατά την διάρκεια σύνταξης του κώδικα από την κοινότητα του ανοιχτού κώδικα.

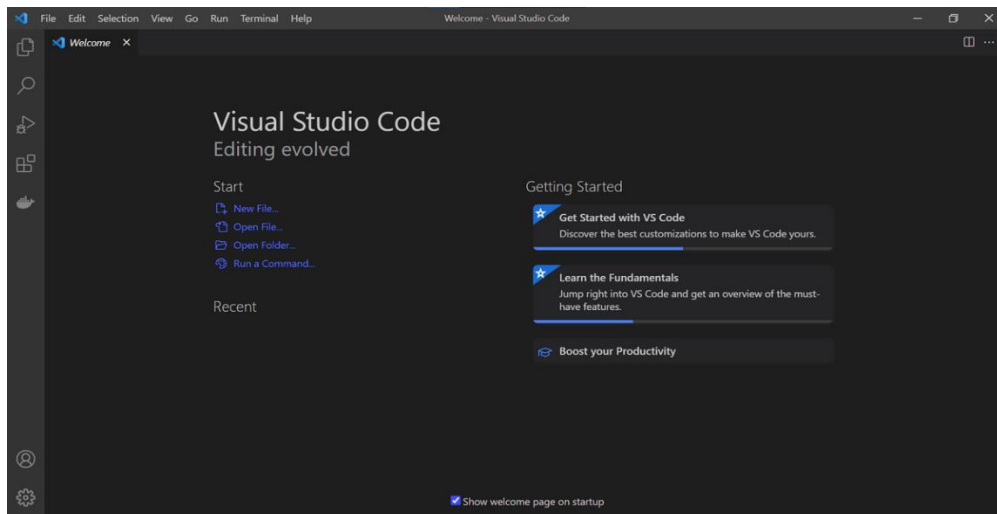
3.1 Εργαλείο σύνταξης κώδικα VS code

Το Visual Studio Code, που συνήθως αναφέρεται και ως VScode, είναι ένα πρόγραμμα επεξεργασίας πηγαίου κώδικα (code-editor) που δημιουργήθηκε από τη Microsoft με το Electron Framework, για Windows, Linux και mac OS. Περιλαμβάνει δυνατότητες υποστήριξης για εντοπισμό σφαλμάτων(debugging), επισήμανση σύνταξης (syntax highlighting), έξυπνη συμπλήρωση κώδικα, ανακατασκευή κώδικα (code refactoring) και ένα ακόμα πλεονέκτημα διαθέτει ενσωματωμένο Git. Οι χρήστες μπορούν να αλλάξουν το θέμα, τις συντομεύσεις πληκτρολογίου και να εγκαταστήσουν επεκτάσεις της αρέσκεια τους όπου επηρεάζουν είτε τη λειτουργικότητα είτε το εμφανισιακό κομμάτι του editor. Επίσης διανέμεται δωρεάν προς όλους

τους χρήστες χωρίς να επιφέρει κάποιο επιπλέον κόστος στην χρήση του. Οι χρήστες μπορούν με μία πολύ απλή αναζήτηση σε μια μηχανή αναζήτησης να το βρουν, να το κατεβάσουν και να το εγκαταστήσουν στον προσωπικό τους υπολογιστή.

Μέχρι πρόσφατα, δεν υπήρχε σχεδόν κανένα πρόγραμμα επεξεργασίας IDE ή κώδικα που να ήταν τόσο φιλικό προς το χρήστη ώστε ακόμη και οι νέοι χρήστες να μπορούν να χρησιμοποιήσουν κάθε δυνατότητα χωρίς καμία ταλαιπωρία. Η φιλική προς την κωδικοποίηση δυνατότητα και η αναγνώριση σφαλμάτων κώδικα βοηθά επίσης τους χρήστες να κάνουν τον κώδικα πιο αποτελεσματικό και χωρίς σφάλματα.

Το user interface του παρόλο που διαθέτει πολλαπλές επιλογές και λειτουργίες, παραμένει να είναι απλοϊκό και εύκολα κατανοητό σε άτομα είτε που διαθέτουν μια εξοικείωση με άλλα code editors είτε όχι. Αυτό επιτυγχάνεται διότι από την πρώτη στιγμή που ο χρήστης θα ανοίξει το VScode, εμφανίζεται η επιλογή εκμάθησης του UI βήμα προς βήμα.



Σημα 3: Visual Studio Code sample

3.2 Γιατί HTML:

Η HTML, ή HyperText Markup Language, είναι ο κώδικας που συνθέτει μια ιστοσελίδα. Είναι μια τυπική γλώσσα που καταλαβαίνουν όλα τα προγράμματα περιήγησης ιστού. Έτσι, χρησιμοποιώντας HTML, υπάρχει η δυνατότητα να δημιουργούνται σελίδες που θα έχουν την ίδια εμφάνιση σε οποιαδήποτε συσκευή ή πλατφόρμα. Εκτός από τις βασικές επιλογές μορφοποίησης κειμένου, το HTML σας επιτρέπει να προσθέσετε εικόνες, βίντεο και άλλα περιεχόμενα πολυμέσων στις σελίδες σας. Επιπλέον, παρέχει τρόπους ελέγχου προβολής και αλληλεπίδρασης του περιεχομένου.

3.3 Πλεονεκτήματα HTML:

Μερικά από τα πολλά πλεονεκτήματα που διαθέτει η επιλογή χρήσης της γλώσσας HTML είναι τα παρακάτω:

4. Κατά τη δημιουργία ενός ιστότοπου, η HTML είναι το θεμέλιο που συγκρατεί όλα τα στοιχεία περιεχομένου και σχεδίασης μαζί.
5. Δίνει δομή στην ιστοσελίδα, ορίζοντας πράγματα όπως επικεφαλίδες, παραγράφους και συνδέσμους. Χωρίς HTML, μια ιστοσελίδα θα ήταν απλώς ένα μπέρδεμα κειμένου και εικόνων
6. Ενώ άλλες τεχνολογίες, όπως οι γλώσσες προγραμματισμού CSS και JavaScript, προσθέτουν στην αισθητική και τη λειτουργικότητα ενός ιστότοπου, η HTML παραμένει αν όχι το πιο κρίσιμο ένα από τα πιο σημαντικά στοιχεία σε κάθε ιστοσελίδα
7. Με τη συμπερίληψη κατάλληλων ετικετών και τίτλων, η HTML βοηθά τις μηχανές αναζήτησης να ανιχνεύουν και να ευρετηριάζουν εύκολα έναν ιστότοπο, βελτιώνοντας τις πιθανότητές του να εμφανιστεί σε σχετικά αποτελέσματα αναζήτησης

Για τους προγραμματιστές ιστού και για τους επαγγελματίες του μάρκετινγκ, η κατανόηση και η χρήση του HTML μπορεί να οδηγήσει σε βελτιωμένη διαδικτυακή προβολή και αυξημένες δυνατότητες επιτυχίας. Εν ολίγοις, η HTML είναι απαραίτητη για τη δημιουργία ενός αποτελεσματικού ιστότοπου.

3.4 Πως λειτουργεί η HTML:

Όταν ένας χρήστης επισκέπτεται έναν ιστότοπο, το πρόγραμμα περιήγησης στέλνει ένα αίτημα στον διακομιστή που φιλοξενεί τον ιστότοπο. Στη συνέχεια, ο διακομιστής ανακτά το έγγραφο HTML και το στέλνει πίσω στο πρόγραμμα περιήγησης. Το πρόγραμμα περιήγησης διαβάζει τον κώδικα HTML και τον μεταφράζει σε μια οπτική διάταξη, επιτρέποντας στον χρήστη να αλληλεπιδράσει με τον ιστότοπο. Η HTML καθορίζει τη δομή και το περιεχόμενο ενός ιστότοπου. Μαζί με τις γλώσσες προγραμματισμού CSS και JavaScript, η HTML επιτρέπει στους προγραμματιστές να δημιουργούν οπτικά ελκυστικούς και λειτουργικούς ιστότοπους. Εκτός από το ότι είναι σημαντικό, η HTML είναι αρκετά εύκολη για αρχάριους να μάθουν και να τη χρησιμοποιήσουν.

Κατανοώντας τις βασικές ετικέτες (HTML tags) και τα χαρακτηριστικά (attributes), ακόμη και αρχάριοι προγραμματιστές μπορούν να δημιουργήσουν το δικό τους περιεχόμενο ιστού. Συνολικά, η HTML είναι αναπόσπαστο μέρος της δημιουργίας επιτυχημένων ιστοσελίδων. Τα περισσότερα προγράμματα περιήγησης ιστού, όπως το Microsoft Internet Explorer, το Mozilla Firefox ή το Google Chrome, μπορούν να αποδώσουν σελίδες HTML. Επιπλέον, υπάρχουν διαθέσιμα πολλά δωρεάν και εμπορικά πακέτα λογισμικού που θα σας επιτρέψουν να προβάλετε σελίδες HTML εκτός σύνδεσης.

3.5 Πως χρησιμοποιείται η HTML:

Ένας προγραμματιστής ιστού (web developer), είναι σημαντικό να έχει μια σταθερή κατανόηση της HTML. Η δημιουργία μιας σελίδας HTML είναι σχετικά απλή. Υπάρχουν πολλά διαθέσιμα δωρεάν και εμπορικά πακέτα λογισμικού που θα σας βοηθήσουν στη δημιουργία της σελίδας σας. Επιπλέον, πολλά διαδικτυακά σεμινάρια μπορούν να σας καθοδηγήσουν στη διαδικασία εκμάθησης της καθώς τα περισσότερα διανέμονται χωρίς κάποιο κόστος.

Ακολουθούν μερικά βασικά βήματα για τη δημιουργία μιας σελίδας HTML:

1. Πρώτα δημιουργήστε ένα κενό έγγραφο με ένα πρόγραμμα επεξεργασίας κειμένου και αποθηκεύστε το με την επέκταση αρχείου ".html".
2. Στη συνέχεια, το έγγραφο μπορεί να προβληθεί σε οποιοδήποτε πρόγραμμα περιήγησης ιστού, όπου θα δείτε απλώς τον ακατέργαστο κώδικα HTML
3. Μέσω της χρήσης ετικετών όπως <p> για την παράγραφο και για την εικόνα, μπορείτε να προσθέσετε περιεχόμενο στη σελίδα.
4. Είναι επίσης δυνατό να βελτιώσετε τη σχεδίαση και τη διάταξη μέσω της χρήσης CSS

Συμπεριλαμβάνοντας αυτά τα στοιχεία και προσθέτοντας μερικά ακόμα στον κώδικα HTML, μπορεί να δημιουργηθεί μια καλά σχεδιασμένη και λειτουργική ιστοσελίδα.

3.6 Τι είναι το CSS

Το CSS σημαίνει Cascading Style Sheets. Είναι η γλώσσα κωδικοποίησης που δίνει σε έναν ιστότοπο την εμφάνιση και τη διάταξή του. Μαζί με την HTML, το CSS είναι θεμελιώδες για το σχεδιασμό ιστοσελίδων. Χωρίς αυτό, οι ιστότοποι θα εξακολουθούσαν να είναι απλό κείμενο σε λευκό φόντο.

Πριν από την ανάπτυξη του CSS το 1996 από την World Wide Web Consortium (Κοινοπραξία του Παγκόσμιου Ιστού), οι ιστοσελίδες ήταν εξαιρετικά περιορισμένες τόσο σε μορφή όσο και σε λειτουργία. Τα πρώτα προγράμματα περιήγησης παρουσίαζαν μια σελίδα ως υπερκείμενο - απλό κείμενο, εικόνες και συνδέσμους προς άλλες σελίδες υπερκειμένου.

3.7 Πλεονεκτήματα του CSS

Το CSS έχει πολλά πλεονεκτήματα που επιτρέπουν στους προγραμματιστές να σχεδιάζουν έναν ιστότοπο. Μερικά από αυτά τα πλεονεκτήματα είναι:

Καλύτερη ταχύτητα ιστοσελίδας: Για να λειτουργεί αποτελεσματικά ένας ιστότοπος, θα πρέπει να έχει μικρό χρόνο φόρτωσης. Στη σύγχρονη εποχή, οι άνθρωποι συνήθως έχουν την υπομονή να περιμένουν μόνο μερικά δευτερόλεπτα για να φορτώσει ένας ιστότοπος. Επομένως, είναι σημαντικό να εξασφαλίσετε μεγαλύτερη ταχύτητα. Για τις εταιρείες που θέλουν να διασφαλίσουν μια ταχύτερη και ομαλή εμπειρία ιστότοπου, το CSS είναι πρωταρχικής σημασίας για την επιτυχία τους.

Ευκολία στη συντήρηση: Το CSS είναι εύκολο στη συντήρηση. Αυτό οφείλεται στο γεγονός ότι μια αλλαγή σε μια γραμμή κώδικα μπορεί να επηρεάζει ολόκληρη την ιστοσελίδα ή πολλαπλά σημεία της. Επίσης, εάν απαιτούνται βελτιώσεις, τότε απαιτείται λιγότερη προσπάθεια για να επηρεαστούν οι αλλαγές στον κώδικα της ιστοσελίδας.

Συνεπής Σχεδιασμός: Θα έχετε δει πολλούς ιστότοπους που είναι κομψοί και φιλικοί προς το χρήστη. Ένα κοινό πράγμα σε όλους αυτούς τους ιστότοπους είναι η συνέπεια στο σχεδιασμό. Το CSS επιτρέπει στους προγραμματιστές να διασφαλίζουν ότι τα στοιχεία στυλ εφαρμόζονται με συνέπεια σε πολλές ιστοσελίδες του ιστότοπου.

Καλύτερη συμβατότητα συσκευών: Οι άνθρωποι χρησιμοποιούν διαφορετικές έξυπνες συσκευές για να δουν έναν συγκεκριμένο ιστότοπο. Μπορεί να είναι smartphone, υπολογιστής ή φορητός υπολογιστής. Για το σκοπό αυτό, οι ιστότοποι απαιτείται να είναι συμβατοί με κάθε συσκευή. Το CSS διασφαλίζει ότι η εργασία εκτελείται ομαλά παρέχοντας καλύτερη συμβατότητα.

3.8 Πως λειτουργεί το CSS

Για να κατανοήσουμε τα βασικά του τρόπου λειτουργίας του CSS, πρέπει πρώτα να κατανοήσουμε λίγο τη σύγχρονη HTML. Οι προγραμματιστές ιστού σχεδιάζουν τις σελίδες σύμφωνα με το "μοντέλο πλαισίου"(box model). Μια ιστοσελίδα είναι μια σειρά πλαισίων, το καθένα από αυτά περιέχει ένα διακριτό στοιχείο. Αυτά τα κουτιά είναι ένθετα, συνήθως το ένα μέσα στο άλλο.

Για παράδειγμα, η κεφαλίδα μιας σελίδας είναι ένα πλαίσιο και περιέχει πολλά μικρότερα πλαίσια που περιλαμβάνουν όλα τα στοιχεία που συνθέτουν μια κεφαλίδα: λογότυπο, πλοήγηση, κουμπιά μέσω κοινωνικής δικτύωσης, κουμπιά καλαθιού αγορών κ.λπ. Χρησιμοποιώντας το CSS, ένας προγραμματιστής εκχωρεί στυλ στο " πλαίσιο κεφαλίδας. Σε αυτό το παράδειγμα, ως υποθέσουμε ότι ο προγραμματιστής κάνει το κείμενο μέσα στην κεφαλίδα μωβ, γραμματοσειρά Arial και ύψος δεκαπέντε πόντους.

Εδώ μπαίνει στο παιχνίδι το "cascading" κομμάτι των cascading style sheets. Τα στυλ γραμματοσειράς που εφαρμόζονται στην κεφαλίδα κατατάσσονται σε όλα τα στοιχεία που περιέχονται μέσα στην κεφαλίδα. Τα στοιχεία που περιέχουν κείμενο όπως η πλοήγηση, οι σύνδεσμοι ή οι παροτρύνσεις για δράση θα είναι όλα μωβ, Arial και ύψος δεκαπέντε πόντων.

3.9 Τι είναι η JavaScript

Η JavaScript είναι μια γλώσσα προγραμματισμού από την πλευρά του πελάτη που χρησιμοποιείται για τη δημιουργία δυναμικών αλληλεπιδράσεων στην ανάπτυξη εφαρμογών, παιχνιδιών και ιστού. Συνήθως αντιπροσωπεύεται ως JS, αυτή η γλώσσα θεωρείται μία από τις βασικές τεχνολογίες του Ιστού σε συνδυασμό με HTML και CSS.

Οι προγραμματιστές χρησιμοποιούν JavaScript για να ζωντανέψουν μια ιστοσελίδα, καθώς μετατρέπει μια κατά τα άλλα στατική σελίδα σε μια που ανταποκρίνεται στις ενέργειες των χρηστών, ζωντανεύει τη σελίδα με κίνηση και εμφανίζει περιεχόμενο σε πραγματικό χρόνο, μεταβλητό και εξατομικευμένο περιεχόμενο.

3.10 Πως λειτουργεί η JavaScript

Η JavaScript συνέχισε να αναπτύσσεται μαζί με νέα προγράμματα περιήγησης όπως το Mozilla Firefox και το Google Chrome από τότε. Η τελευταία μάλιστα άρχισε να αναπτύσσει την πρώτη σύγχρονη μηχανή JavaScript, που ονομάζεται V8, η οποία μεταγλωττίζει τον bytecode σε εγγενή κώδικα μηχανής

Σήμερα, η JavaScript διαθέτει πολλά πλαίσια(frameworks) και βιβλιοθήκες για την απλοποίηση σύνθετων έργων, όπως το AngularJS, το jQuery και το ReactJS. Αρχικά εκτελείται από την πλευρά του πελάτη, η υλοποίηση JavaScript έχει διακλαδωθεί στην πλευρά του διακομιστή μετά την ανάπτυξη του Node.js - ένα περιβάλλον διακομιστή πολλαπλών πλατφορμών που έχει δημιουργηθεί στη μηχανή JavaScript V8 του Google Chrome.

Η ανάπτυξη πλαισίων JavaScript, που αποτελούνται από βιβλιοθήκες κώδικα JavaScript, επιτρέπει στους προγραμματιστές να χρησιμοποιούν προγραμμένο κώδικα JavaScript στα έργα τους. Τους εξοικονομεί χρόνο και προσπάθεια από το να χρειαστεί να κωδικοποιήσουν τις λειτουργίες προγραμματισμού από την αρχή. Κάθε JavaScript framework έχει χαρακτηριστικά που στοχεύουν στην απλοποίηση της διαδικασίας ανάπτυξης και εντοπισμού σφαλμάτων.

Για παράδειγμα, στα πλαίσια του JavaScript front-end όπως το jQuery και το ReactJS βελτιώνουν την αποδοτικότητα του σχεδιασμού. Επιτρέπουν στους προγραμματιστές να επαναχρησιμοποιούν και να ενημερώνουν στοιχεία κώδικα χωρίς να επηρεάζουν το ένα το άλλο, τη λειτουργία ή την αξία. Εν τω μεταξύ, στα πλαίσια ανάπτυξης εφαρμογών για κινητά όπως το Cordova και το Titanium καθιστούν δυνατή τη δημιουργία υβριδικών εφαρμογών.

Επιπλέον, η JavaScript σάς επιτρέπει να αλλάξετε περιεχόμενο HTML και τιμές χαρακτηριστικών χωρίς να φορτώσετε ξανά την ιστοσελίδα πρώτα. Αυτό συμβαίνει επειδή η JavaScript υποστηρίζει τους ακόλουθους τύπους δεδομένων:

1. String: αποτελείται από δεδομένα κειμένου γραμμένα μέσα σε εισαγωγικά. Για παράδειγμα, "Hello world"
2. Number: καλύπτει ακέραιους και αριθμούς κινητής υποδιαστολής
3. Boolean – ένας λογικός τύπος δεδομένων με αληθείς και ψευδείς τιμές.
4. Null – περιέχει μια μηδενική τιμή.
5. Object (Αντικείμενο) – για σύνθετες δομές δεδομένων

3.11 Πλεονεκτήματα της JavaScript

Ο λόγος για τον οποίο η JavaScript είναι μια από τις πιο δημοφιλείς γλώσσες προγραμματισμού είναι το γεγονός ότι είναι εξαιρετικά ευέλικτη. Πολλοί προγραμματιστές τη θεωρούν την κύρια επιλογή τους εκτός και αν χρειάζονται μια συγκεκριμένη λειτουργία που δεν παρέχει η JavaScript.

Δεν υπάρχει τέλεια γλώσσα προγραμματισμού. Επομένως, μια νέα γλώσσα κερδίζει δημοτικότητα κάθε λίγα χρόνια για να ικανοποιήσει τις ανάγκες της εποχής. Για παράδειγμα, η γλώσσα προγραμματισμού C κυριάρχησε τη δεκαετία του 1990 και είχε μικρό ανταγωνισμό για λίγο. Αργότερα, καθώς ο χρόνος και η ζήτηση της κοινότητας άλλαξαν, η Python κέρδισε δημοτικότητα λόγω των τεράστιων πλεονεκτημάτων της σε σχέση με τη C, και τελικά, η JavaScript ανέβηκε στη σκηνή.

Παρακάτω παρουσιάζονται αριθμημένα και στην συνέχεια θα αναλυθούν κάποια βασικά πλεονεκτήματα της JavaScript που την καθιστούν ανώτερη από άλλες γλώσσες προγραμματισμού:

8. Ταχύτητα(Speed)
9. Μειώνει το φόρτο στον διακομιστή (Reduces load on the server)
10. Ευκολία στη χρήση (Ease of use)
11. Πλούσια διεπαφή (Rich Interface)
12. Ευστροφία (Versatility)
13. Εκτεταμένη λειτουργικότητα (Extended functionality)
14. Διαλειτουργικότητα (Interoperability)
15. Δημοτικότητα (Popularity)
16. Ανεξαρτησία πλατφόρμας (Platform independence)
17. Ισχυρές βιβλιοθήκες (Powerful frameworks)
18. Χαρακτηριστικά διαδικαστικού προγραμματισμού (Procedural programming features)
19. Απόκριση στη δραστηριότητα των χρηστών (Response to user activity)
20. Ενημερώσεις (Updates)

Ταχύτητα(Speed):

Οι ερμηνευμένες γλώσσες (όπως και η JavaScript) είναι γλώσσες προγραμματισμού οι οποίες δεν έχουν προ μεταγλωττιστεί για το μηχάνημα-στόχο σε αναγνώσιμη από μηχανή μορφή. Οι ερμηνευμένες γλώσσες ήταν κάποτε πολύ πιο αργές από τις μεταγλωττισμένες γλώσσες. Αλλά, με την ανάπτυξη της just-in-time compilation, αυτό το χάσμα συρρικνώνεται. Παραδείγματα κοινών γλωσσών διερμηνείας είναι η PHP, η Ruby, η Python και η JavaScript.

Μειώνει το φόρτο στον διακομιστή (Reduces load on the server):

Η γλώσσα εκτελείται στην πλευρά του πελάτη και όχι στον διακομιστή. Έτσι, ο διακομιστής δεν χρειάζεται να αντιμετωπίσει το φόρτο της εκτέλεσης JavaScript. Μόλις μειωθεί αυτή η επιβάρυνση, ο διακομιστής θα λειτουργεί πιο γρήγορα και θα επικεντρωθεί σε άλλες εργασίες, όπως η διαχείριση δεδομένων.

Ευκολία στη χρήση (Ease of use):

Η JavaScript είναι μια από τις απλούστερες γλώσσες για εκμάθηση, ιδιαίτερα για προγραμματισμό Ιστού. Έχει σχεδιαστεί με σκοπό να είναι απλή στην χρήση και κατανόηση για τους προγραμματιστές ιστού.

Πλούσια διεπαφή (Rich Interface):

Η JavaScript προσφέρει στους προγραμματιστές μια μεγάλη ποικιλία διεπαφών(interfaces) για τη δημιουργία ελκυστικών ιστότοπων.

Ευστροφία (Versatility):

Η ευελιξία της JavaScript είναι το πιο πολύτιμο πλεονέκτημα της. Υπάρχουν πολλοί διαφορετικοί τρόποι για να την ενσωματώσετε στον ιστότοπό σας. Δεδομένου ότι το Node.js ενσωματώνεται αποτελεσματικά με το MongoDB και τη MySQL, μπορεί όχι μόνο να δημιουργήσει και να ολοκληρώσει το μπροστινό μέρος του ιστότοπου(Front-End) αλλά και να χειριστεί το πίσω άκρο του(Back-End).

Εκτεταμένη λειτουργικότητα (Extended functionality):

Για εξοικονόμηση χρόνου και χρημάτων, πρόσθετα τρίτων(add-ons), όπως το Greasemonkey (επέκταση του Mozilla Firefox) επιτρέπουν στους προγραμματιστές να ενσωματώνουν μικρά τμήματα προκατασκευασμένου κώδικα στον κώδικά τους. Αυτά τα πρόσθετα καθιστούν ευκολότερο και ταχύτερο για τους προγραμματιστές τη δημιουργία εφαρμογών JavaScript από ό,τι θα μπορούσαν με άλλες γλώσσες κωδικοποίησης.

Διαλειτουργικότητα (Interoperability):

Οποιαδήποτε ιστοσελίδα ή σενάριο άλλης γλώσσας υπολογιστή μπορεί να την ενσωματώσει.

Δημοτικότητα (Popularity):

Η JavaScript είναι μια από τις πιο δημοφιλείς γλώσσες για ανάπτυξη ιστού. Είναι ένας σημαντικός παράγοντας για κάθε ιστότοπο που λειτουργεί. Ακόμη και οι πιο δημοφιλείς ιστότοποι παγκοσμίως, συμπεριλαμβανομένου του Amazon και της Google, χρησιμοποιούν JavaScript καθώς θεωρείται μια πολύ ισχυρή τεχνολογία. Λόγω της σταθερά αυξανόμενης δημοτικότητάς της, είναι πλέον πιο απλό από ποτέ η εκμάθηση αυτής της γλώσσας στο διαδίκτυο μέσω διαφόρων μαθημάτων.

Ανεξαρτησία πλατφόρμας (Platform independence):

Τα περισσότερα προγράμματα περιήγησης υποστηρίζουν JavaScript, καθιστώντας απλό για οποιοδήποτε πρόγραμμα περιήγησης να κατανοήσει και να αναγνωρίσει τον κώδικα JavaScript. Δεν χρειάζεται να περάσετε από καμία διαδικασία εγκατάστασης, επειδή είναι μια εύκολα διαθέσιμη τεχνολογία. Απλώς χρησιμοποιήστε το πρόγραμμα περιήγησής σας (browser)

Ισχυρές βιβλιοθήκες (Powerful frameworks):

Τριγύρω από την JavaScript, υπάρχουν πολλά frameworks με έτοιμα προς χρήση κομμάτια κώδικα. Αυτά τα κομμάτια κώδικα είναι εύκολα ως προς την αποσφαλμάτωση τους αλλά και στην διατήρησή τους. Ανάλογα με την εν λόγω δομή, παρέχει πρόσβαση σε πολλές πρόσθετες δυνατότητες που θα αυξήσουν εκθετικά την απόδοσή του προγράμματος ή ιστοσελίδας.

Χαρακτηριστικά διαδικαστικού προγραμματισμού (Procedural programming features):

Η JavaScript σας επιτρέπει να δημιουργείτε υποκαταστήματα(branches), να ξεκινάτε τον έλεγχο υπό όρους(conditional checking), να ξεκινάτε βρόχους και πολλά άλλα, κάνοντας τη χρήση του ιστότοπού σας πολύ πιο διασκεδαστική.

Απόκριση στη δραστηριότητα των χρηστών (Response to user activity):

Η JavaScript είναι μια γλώσσα κωδικοποίησης που βασίζεται σε συμβάντα. Αυτό σημαίνει ότι διάφορα τμήματα κώδικα εκτελούνται αφού ένας χρήστης κάνει κλικ σε ένα κουμπί. Ως αποτέλεσμα, όταν φορτώνεται ο ιστότοπος, κανένας κώδικας δεν προετοιμάζεται. Κάνοντας αυτό, μπορείτε να είστε βέβαιοι ότι ο χρόνος φόρτωσης του ιστοτόπου σας δεν θα κινδυνεύει καθώς τον φορτώνετε με λειτουργίες προηγμένης τεχνολογίας.

Ενημερώσεις (Updates):

Από την κυκλοφορία του ECMAScript 5, της προδιαγραφής δέσμης ενεργειών στην οποία βασίζεται η JavaScript, η ECMA International έχει δώσει προτεραιότητα στην ενημέρωση της JavaScript ετησίως. 13 εκδόσεις λειτουργιών έχουν κυκλοφορήσει από το 1997. Η οικογένεια JavaScript περιλαμβάνει πλέον τις δυνατότητες ES2022 που έχουν περάσει την επαλήθευση του σταδίου 4.

Γραφικό περιβάλλον της εφαρμογής

4.1 Εισαγωγή

Το γραφικό περιβάλλον χρήστη (GUI) είναι μια διεπαφή μέσω της οποίας ένας χρήστης αλληλεπιδρά με ηλεκτρονικές συσκευές όπως υπολογιστές και smartphone μέσω της χρήσης εικονιδίων, μενού και άλλων οπτικών ενδείξεων ή αναπαραστάσεων (γραφικά). Τα GUI εμφανίζουν γραφικά πληροφορίες και σχετικά στοιχεία ελέγχου χρήστη, σε αντίθεση με τις διεπαφές που βασίζονται σε κείμενο, όπου τα δεδομένα και οι εντολές είναι αυστηρά σε κείμενο. Οι αναπαραστάσεις του γραφικού περιβάλλοντος διαχειρίζονται από μια συσκευή κατάδειξης όπως ποντίκι, trackball, γραφίδα ή με ένα δάχτυλο σε μια οθόνη αφής.

4.2 Ανάλυση του γραφικού περιβάλλοντος

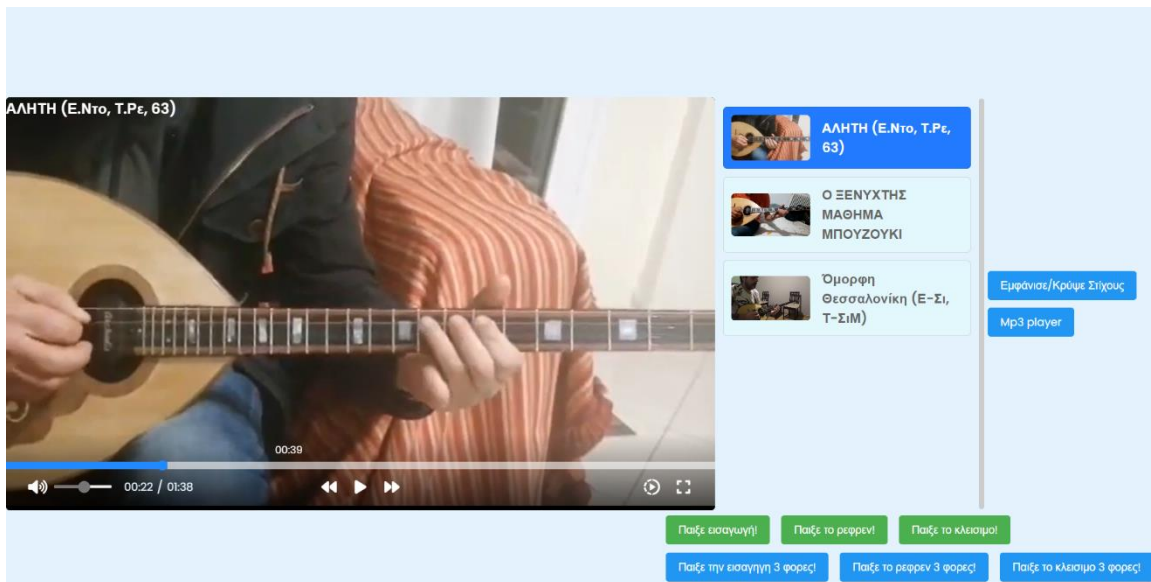
Το γραφικό περιβάλλον χρήστη της εφαρμογής μπορούμε να το χωρίσουμε σε δύο διαφορετικά τμήματα όπου το καθένα από αυτά έχει έναν δικό του διαφορετικό ρόλο στην εφαρμογή. Το πρώτο είναι αυτό του Video player και δεύτερο αυτό του Audio player.

4.3 Γραφικό περιβάλλον του Video player

Το πρώτο τμήμα του γραφικού περιβάλλοντος της εφαρμογής είναι αυτό του Video Player. Το συγκεκριμένο είναι περισσότερο και πλούσια εμπλουτισμένο από το δεύτερο τμήμα του GUI. Αυτή η διαφορά είναι λογική, διότι με τις οπτικές λειτουργίες που διαθέτει βοηθάει έναν αρχάριο χρήστη που επιθυμεί να μάθει και να μελετήσει ένα μουσικό έργο απλούστερο. Παρακάτω παρουσιάζονται αναλυτικότερα τα στοιχεία που περιέχει το GUI της σελίδας του Video Player και στην συνέχεια του Audio Player.

4.3.1 Τα στοιχεία του Video player

Αν και δεν αποτελεί μέρος του ίδιου του Video player, πρέπει να γίνει μια αρχική και σύντομη αναφορά στο background-βάση της σελίδας. Παρακάτω γίνεται αναφορά στο σχήμα 4.1. όπου φαίνεται το χρώμα που χρησιμοποιεί η σελίδα ένα απαλό γαλάζιο. Ο σκοπός της επιλογής αυτής είναι: πρώτον να μην αποσπά την προσοχή του χρήστη και δεύτερον να μην καθιστά κουραστικό στο μάτι του χρήστη. Για το δεύτερο, αν η επιλογή του χρώματος ήταν για παράδειγμα το κίτρινο, μπορούμε να αντιληφθούμε ότι σε μία επίσκεψη του χρήστη μεγάλης διάρκειας αυτή η επιλογή θα αποτελούσε εξουθενωτική για το μάτι.



Σχήμα 4.1: Συνολικό GUI Video Player

Παρακάτω γίνεται αναφορά στο σχήμα 4.2. όπου φαίνονται τα στοιχεία μέσα στο video player. Αρχικά στην πάνω αριστερά γωνία του player με γραμματοσειρά λευκού χρώματος φαίνεται ο τίτλος του αρχείου που έχει επιλεγθεί για αναπαραγωγή. Μέσα στο εσωτερικό πλαίσιο αναπαράγεται το video. Στο κάτω μέρος του πλαισίου υπάρχουν: μπάρα σημείου χρονικής ένδειξης, κουμπί mute/unmute, συρόμενη μπάρα για ρύθμιση έντασης ήχου, ακριβής ένδειξη λεπτού και δευτερολέπτου που βρίσκεται το βίντεο, ένδειξη συνολικής διάρκειας βίντεο, κουμπί

rewind, κουμπι pause/play, κουμπι forward, κουμπι ρύθμισης ταχύτητας αναπαραγωγής και κουμπι με λειτουργία πλήρους οθόνης.



Σχήμα 4.2: Στοιχεία Video Player

4.3.2 Κουμπιά ρυθμίσεων και μπάρες του Video player

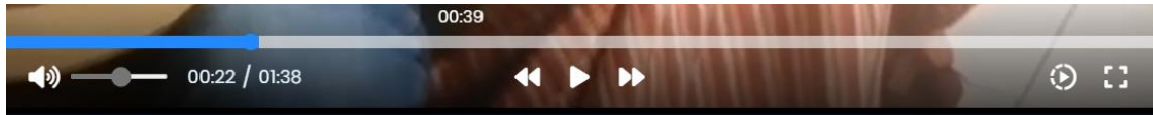
Στην συνέχεια θα αναφερθούμε αναλυτικά στα **κουμπιά ρυθμίσεων** του βίντεο player (βλ. Σχήμα 4.3). Ξεκινώντας με την **μπάρα σημείου χρονικής ένδειξης**.

Η μπάρα διαθέτει πάνω της μία τελεία χρώματος ανοικτού μπλε. Όσο το αρχείο αναπαράγεται ανάλογα με την συνολική διάρκεια του βίντεο και του ακριβούς χρονικού σημείου του βίντεο, η μπάρα μετακινείται από αριστερά προς τα δεξιά. Κατά την μετακίνηση της, αλλάζει το χρώμα της μπάρας από γκρι σε μπλε.

Επίσης παρέχει διάφορες δυνατότητες στον χρήστη. Όταν ο χρήστης αφήσει το ποντίκι σε οποιοδήποτε σημείο της μπάρας (hover). Τότε εμφανίζονται τα λεπτά και δευτερόλεπτα πάνω στον κέρσορα. Ανάλογα με το σημείο που θα γίνει το hover θα προσαρμοστούν και τα λεπτά-δευτερόλεπτα που θα εμφανιστούν. Επίσης αν ο χρήστης επιλέξει να κάνει click πάνω στην μπάρα τότε το αρχείο θα ξεκινήσει να παίζει από την ακριβής ένδειξη χρόνου που φαίνεται όταν κάνει hover με το ποντίκι. Επιπλέον ο χρήστης μπορεί να σύρει την μπάρα και την αφήσει σε οποιοδήποτε σημείο της μπάρας, με αποτέλεσμα να συνεχίσει η αναπαραγωγή του βίντεο από το σημείο της επιλογής του.

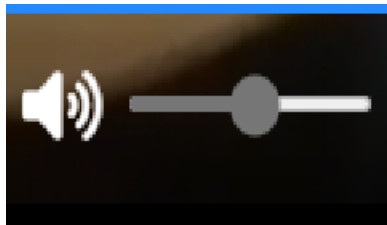
Στο κάτω αριστερά μέρος των κουμπιών ρυθμίσεων δίπλα από την συρόμενη μπάρα ήχου υπάρχουν τα: ακριβής ένδειξη λεπτού και δευτερολέπτου που βρίσκεται το βίντεο, ένδειξη συνολικής διάρκειας βίντεο (βλ Σχήμα 4.3). Το πρώτο αλλάζει δυναμικά ανάλογα με τον ακριβές

χρόνο που βρίσκεται η αναπαραγωγή του αρχείου και επηρεάζεται από οποιαδήποτε ενέργεια κάνει ο χρήστης πάνω στην μπάρα σημείου χρονικής ένδειξης.



Σχήμα 4.3: Ρυθμίσεις Video Player

Στο κάτω αριστερά μέρος του σχήματος 4.4 υπάρχουν οι **ρυθμίσεις ήχου**. Πρώτα το **κουμπί mute/unmute**. Παρέχει την δυνατότητα στον χρήστη να ενεργοποιήσει ή να απενεργοποιήσει τον ήχο από το αρχείο που αναπαράγεται στο Video Player. Από προεπιλογή το video player ξεκινάει με την επιλογή unmute (βλ. Σχήμα 4.4).



Σχήμα 4.4: Ρυθμίσεις Video Player

Επιπλέον αν ο χρήστης κάνει κλικ πάνω στο κουμπί mute/unmute πέρα οτι θα απενεργοποιηθεί ο ήχος το κουμπί θα αλλάξει και οπτικά (βλ. Σχήμα 4.5)



Σχήμα 4.5: Mute button

Τέλος για τις ρυθμίσεις ήχου ο χρήστης έχει την δυνατότητα να επηρεάσει κατα την προτίμηση του την ένταση του ήχου από την **μπάρα ήχου** (βλ. Σχήμα 4.4). Κάνοντας αριστερό κλικ και κρατώντας παρατεταμένα το κλικ πάνω στην μπάρα, μπορεί να μετακινήσει δεξιά ή αριστερά την μπάρα πάνω στην μπάρα και ανάλογα είτε ο ήχος θα αυξηθεί είτε θα μειωθεί.

Στο κάτω κεντρικό σημείο των ρυθμίσεων βρίσκονται τα 3 κουμπιά **rewind**, **play/pause**, **forward** (βλ. Σχήμα 4.6).

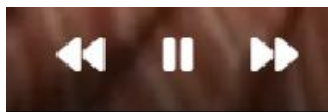
Το **κουμπί rewind** παρέχει την δυνατότητα στον χρήστη όταν το χρησιμοποιεί να πηγαίνει την αναπαραγωγή του βίντεο πέντε δευτερόλεπτα πίσω.

Το **κουμπί forward** παρέχει την δυνατότητα στον χρήστη όταν το χρησιμοποιεί να πηγαίνει την αναπαραγωγή του βίντεο πέντε δευτερόλεπτα μπροστά.



Σχήμα 4.6: Κεντρικά κουμπιά ρυθμίσεων

Το **κουμπί play/pause** από προεπιλογή, όταν ξεκινάει η αναπαραγωγή του βίντεο παίρνει την κοινή γνωστή οπτική μορφή του **pause button** που συναντάμε σε οποιοδήποτε σύγχρονο πρόγραμμα αναπαραγωγής (βλ. Σχήμα 4.7). Με την ίδια λογική όταν το αρχείο δεν αναπαράγεται και ο χρήστης έχει χρησιμοποιήσει το κουμπί **pause**, το κουμπί οπτικά παίρνει την εικόνα του **play button** (βλ. Σχήμα 4.6)



Σχήμα 4.7: Pause button

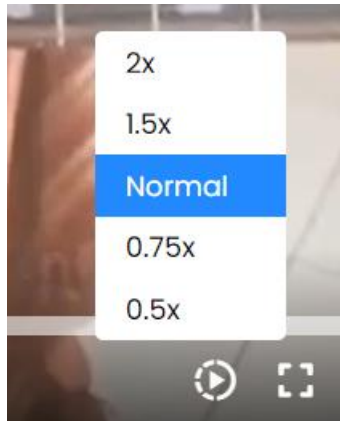
Στο σχήμα 4.8 φαίνονται τα δύο τελευταία κουμπιά ρυθμίσεων του video player. Τα οποία είναι το **κουμπί ρύθμισης ταχύτητας αναπαραγωγής** και το **κουμπί λειτουργίας πλήρους οθόνης**.



Σχήμα 4.8: Κουμπιά ρύθμισης ταχύτητας αναπαραγωγής και λειτουργίας πλήρους οθόνης

Το κουμπί ρύθμισης ταχύτητας αναπαραγωγής: Όταν ο χρήστης επιλέξει το συγκεκριμένο κουμπί τότε εμφανίζεται μία λίστα με επιλογές ταχύτητας αναπαραγωγής (βλ. Σχήμα 4.9). Ανάλογα με την επιλογή του χρήστη επηρεάζεται η ταχύτητα αναπαραγωγής. Για παράδειγμα αν επιλέξει την επιλογή "2x" τότε το βίντεο θα αναπαράγεται στην διπλάσια ταχύτητα σε σύγκριση

με την πραγματική του. Έτσι αν επιλέξει κάποια από τις επιλογές "0.75x" ή "0.5x" τότε η ταχύτητα του βίντεο θα μειωθεί αναλόγως.

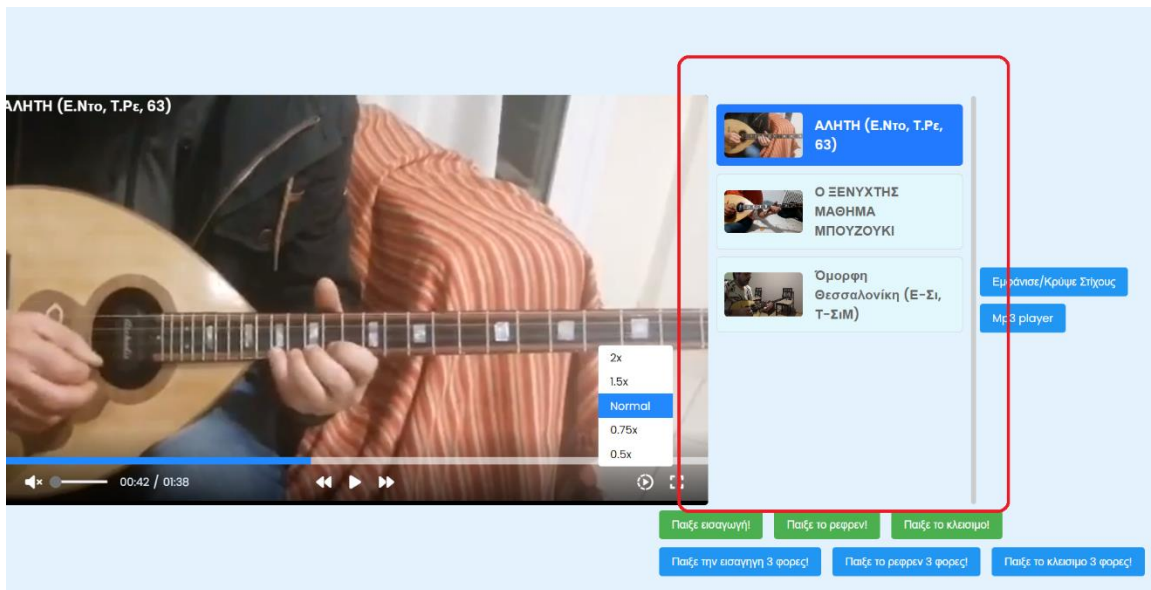


Σχήμα 4.9: Play speed settings

Το κουμπί λειτουργίας πλήρους οθόνης: Δίνει την δυνατότητα στον χρήστη να παρακολουθεί το βίντεο σε ολόκληρη την οθόνη του. Σε περίπτωση που θέλει να επιστρέψει στην κανονική προβολή μπορεί είτε να ξαναπατήσει το ίδιο κουμπί είτε να πατήσει το πλήκτρο «esc» στο πληκτρολόγιο του.

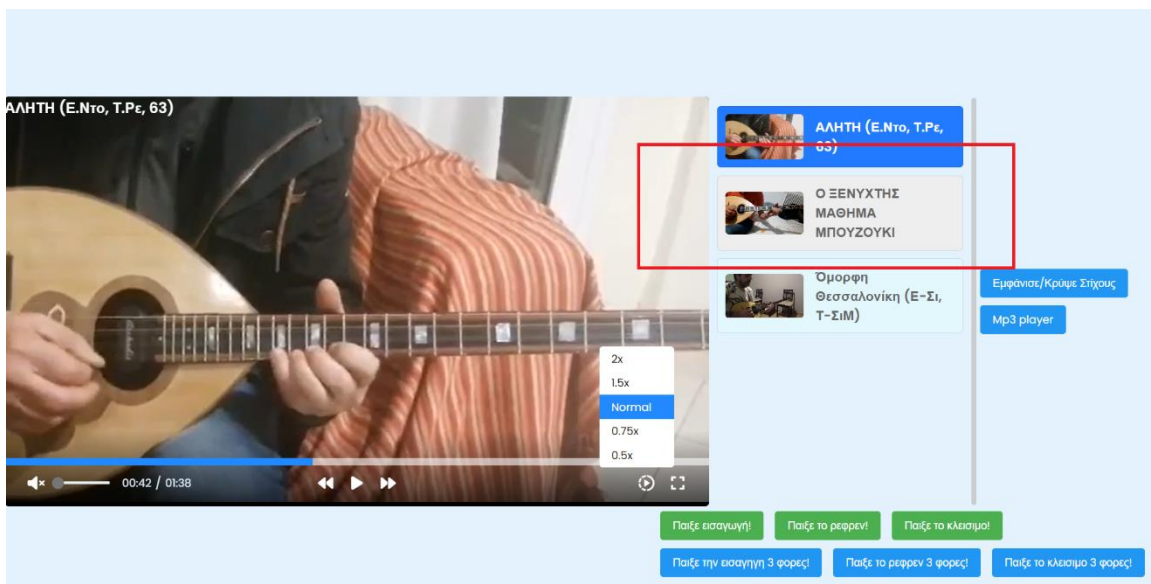
4.3.3 Λίστα αρχείων προς αναπαραγωγή

Ακριβώς δίπλα από το βασικό στοιχείο της πρώτης σελίδας δηλαδή του video player, βρίσκουμε την λίστα αρχείων προς αναπαραγωγή(βλ. Σχήμα 4.10). Η λίστα αυτή γεμίζει με τα αρχεία που έχει επιλέξει ο χρήστης να αναπαράξει. Η πρώτη δυνατότητα στην λίστα είναι ότι το επιλεγμένο αρχείο προς αναπαραγωγή μέσα στην λίστα τονίζεται με ένα έντονο μπλε χρώμα. Ίσως η πιο σημαντική δυνατότητα της λίστας είναι ότι όταν ο χρήστης επιλέξει με κλικ κάποιο αρχείο μέσα από την λίστα τότε αυτό περνάει μέσα στο video player και είναι έτοιμο το player να το αναπαράξει. Ταυτόχρονα με αυτή την λειτουργία-δυνατότητα, αλλάζουν και στοιχεία μέσα στο player όπως: ο τίτλος του αρχείου που έχει επιλεγεί για αναπαραγωγή, η ακριβής ένδειξη λεπτού και δευτερολέπτου που βρίσκεται το βίντεο και η ένδειξη συνολικής διάρκειας βίντεο.



Σχήμα 4.10: Λίστα αρχείων προς αναπαραγωγή

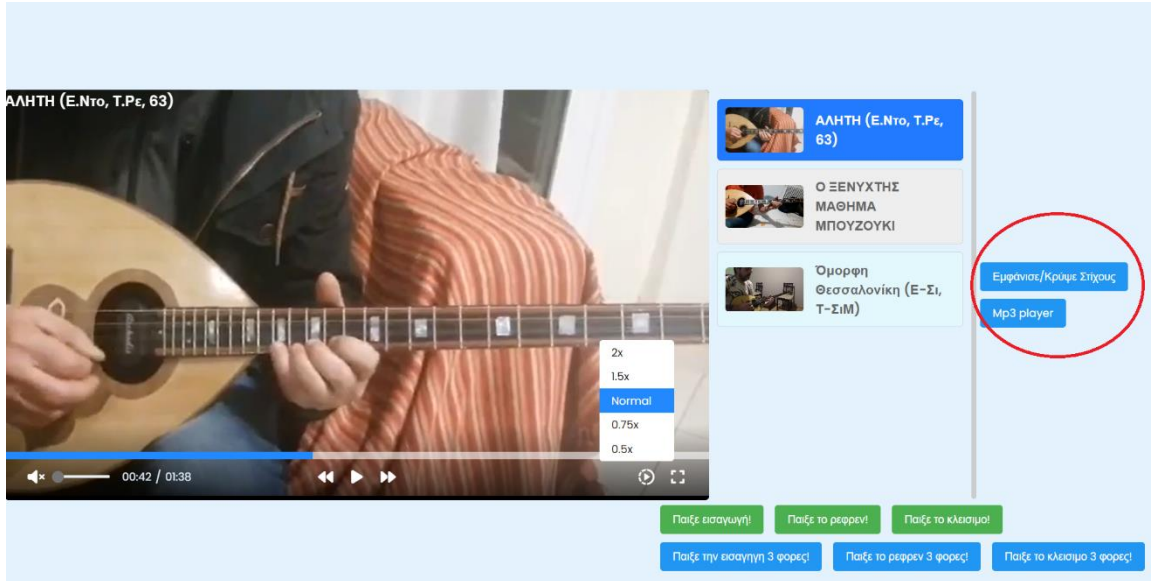
Επιπλέον η λίστα παρέχει στον χρήστη την δυνατότητα όταν κάνει hover με το ποντίκι του σε κάποιο στοιχείο της λίστας (πέρα από το αρχείο που ήδη είναι επιλεγμένο). Να αλλάζει τον κέρσορα του ποντικιού από τον κλασικό στο γνωστό "χεράκι" και να επιχρωματίζει το πλαίσιο του αρχείου που γίνεται hover με μπλε χρώμα ώστε να ξεχωρίζει (βλ. Σχήμα 4.11)



Σχήμα 4.11: Hover on list feature

4.3.4 Κουμπιά Εμφάνισε/Κρύψε Στίχους και MP3 player

Παρακάτω γίνεται αναφορά στο σχήμα 4.12. όπου φαίνονται τα δεξιά στην σελίδα δύο κουμπιά δεξιά από την λίστα. Το πρώτο με όνομα "Εμφάνισε/Κρύψε Στίχους" και το δεύτερο "MP3 player".



Σχήμα 4.12: Κουμπιά Εμφάνισε/Κρύψε Στίχους και MP3 player

Το κουμπί MP3 player παρέχει την δυνατότητα στον χρήστη όταν το επιλέξει να τον οδηγήσει στο δεύτερο τμήμα του GUI που είναι αυτό του Audio player όπου θα αναλυθεί σε επόμενο υποκεφάλαιο.

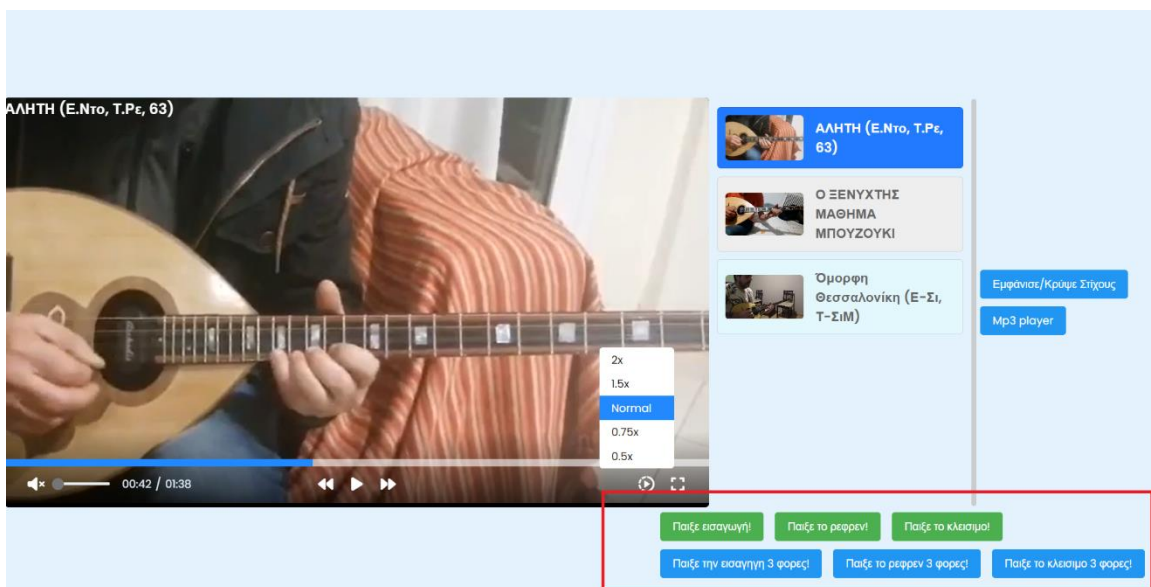
Το γραφικό περιβάλλον από προεπιλογή μόλις το ανοίξει ο χρήστης και επιλέξει κάποιο μουσικό κομμάτι, δεν εμφανίζει τους στίχους. Η δυνατότητα αυτή μπορεί να επιτευχθεί με την χρήση του κουμπιού Εμφάνισε/Κρύψε Στίχους. Με την προϋπόθεση ότι ο χρήστης έχει ήδη επιλέξει ένα αρχείο προς αναπαραγωγή, όταν χρησιμοποιήσει το κουμπί Εμφάνισε/Κρύψε Στίχους τότε στο κάτω μέρος της σελίδας (από το Video Player) θα εμφανιστούν οι στίχοι (βλ. Σχήμα 4.13). Επιπλέον δυνατότητα του κουμπιού είναι ότι για κάθε ένα ξεχωριστό αρχείο η σελίδα φορτώνει και το σωστό αρχείο που περιλαμβάνει τους στίχους για το κομμάτι που έχει επιλέξει προς αναπαραγωγή ο χρήστης. Κάνοντας scroll down με το ποντίκι στην σελίδα ο χρήστης μπορεί να δει και να διαβάσει τους στίχους του αρχείου.



Σχήμα 4.13: Πεδίο εμφάνισης στίχων

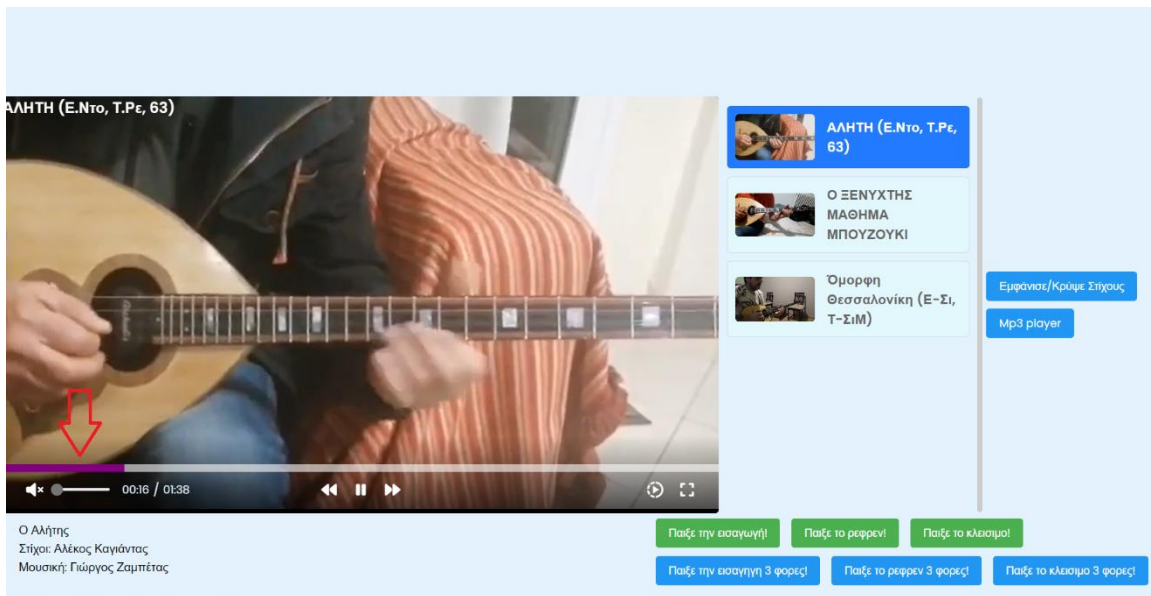
4.3.5 Κουμπιά Αναπαραγωγής από συγκεκριμένο σημείο του αρχείου

Στο σχήμα 4.14 μπορούμε να παρατηρήσουμε ότι υπάρχουν στο κάτω δεξιά μέρος της σελίδας επιπλέον έξι κουμπιά. Ονομαστικά είναι: "Παίξε την εισαγωγή!", "Παίξε το ρεφρέν!", "Παίξε το κλείσιμο!", "Παίξε την εισαγωγή 3 φορές!", "Παίξε το ρεφρέν 3 φορές", "Παίξε το κλείσιμο 3 φορές!".



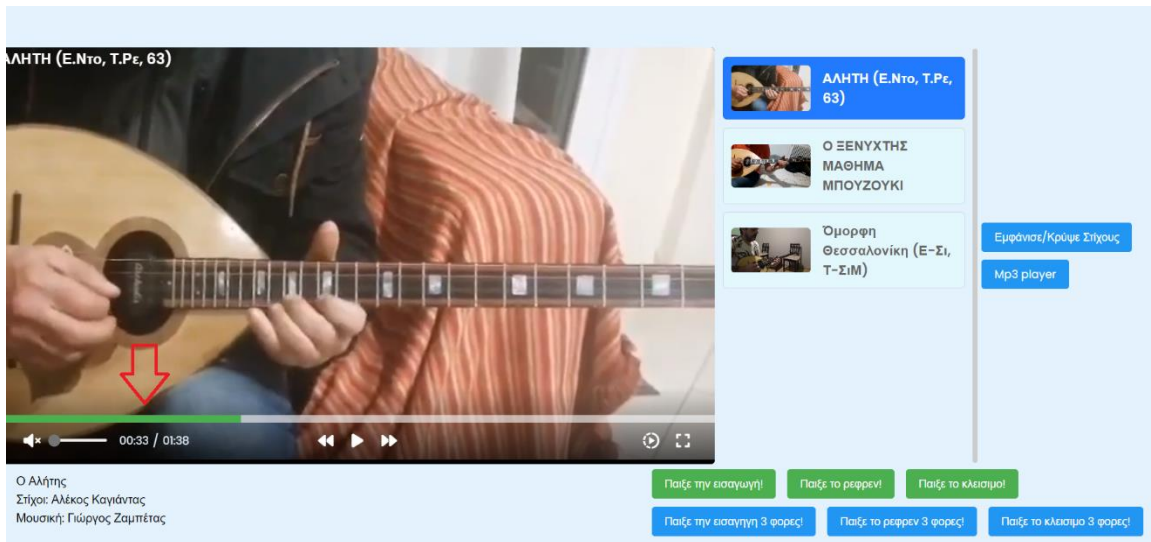
Σχήμα 4.14: Κουμπια αναπαραγωγής συγκεκριμένου σημείου και επαναλήψεων

Το κουμπί **“Παίξε την εισαγωγή!”** παρέχει τις εξής λειτουργίες. Αρχικά στην περίπτωση που ο χρήστης το επιλέξει ξεκινάει να παίζει από την αρχή μέχρι ένα προκαθορισμένο χρονικό σημείο του μουσικού έργου. Όταν η αναπαραγωγή φτάσει στο τέλος του προκαθορισμένου χρόνου τότε το βίντεο για την κατανόηση και την διευκόλυνση του χρήστη κάνει αυτόματα παύση. Ταυτόχρονα με αυτή την λειτουργία η μπάρα σημείου χρονικής ένδειξης αλλάζει από μπλέ χρώμα σε μωβ (βλ. Σχήμα 4.15).



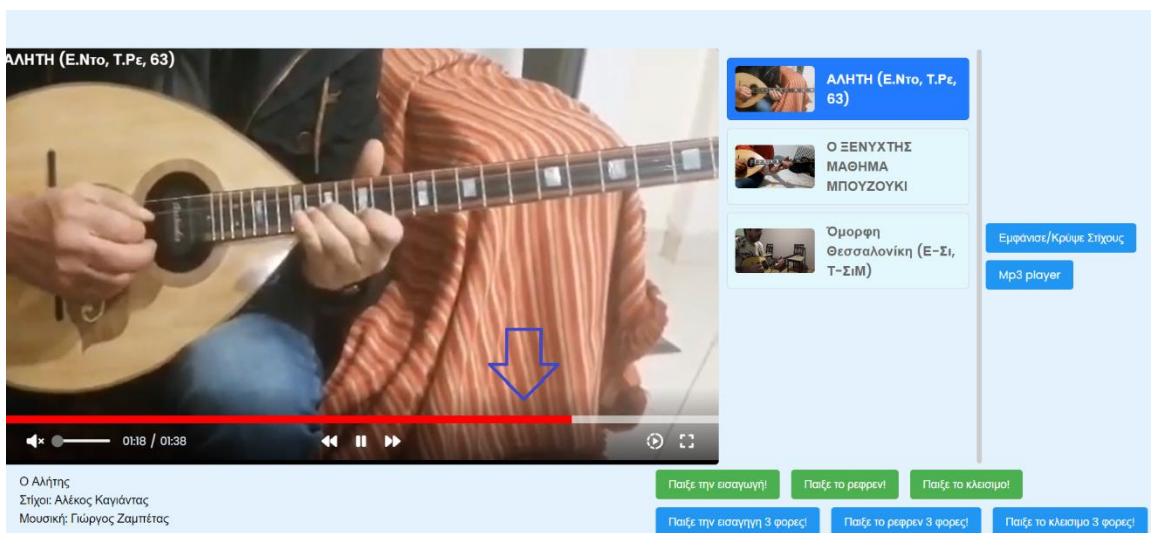
Σχήμα 4.15: Αποτέλεσμα χρήσης κουμπιού “Παίξε την εισαγωγή!”

Το κουμπί **“Παίξε το ρεφρέν!”** παρέχει τις εξής λειτουργίες. Όπως και στην περίπτωση του κουμπιού “Παίξε την εισαγωγή!” έτσι και αυτό το κουμπί όταν επιλεγθεί ξεκινάει να παίζει από ένα προκαθορισμένο σημείο μέχρι ένα άλλο προκαθορισμένο χρονικό σημείο του μουσικού έργου (συνήθως στην μέση της διάρκειας του έργου). Όταν η αναπαραγωγή φτάσει στο τέλος του προκαθορισμένου χρόνου τότε το βίντεο για την κατανόηση και την διευκόλυνση του χρήστη κάνει αυτόματα παύση. Ταυτόχρονα με αυτή την λειτουργία η μπάρα σημείου χρονικής ένδειξης αλλάζει από μπλέ χρώμα σε σκούρο πράσινο (βλ. Σχήμα 4.16).



Σχήμα 4.16: Αποτέλεσμα χρήσης κουμπιού "Παίξε το ρεφρέν!"

Το **κουμπί "Παίξε το κλείσιμο!"** παρέχει τις εξής λειτουργίες. Όπως και στις περιπτώσεις των προηγούμενων κουμπιών έτσι και αυτό το κουμπί όταν επιλεγθεί ξεκινάει να παίζει από ένα προκαθορισμένο σημείο με την διαφορά ότι η αναπαραγωγή συνεχίζεται μέχρι το τέλος του έργου. Όταν η αναπαραγωγή φτάσει στο τέλος του προκαθορισμένου χρόνου τότε το βίντεο για την κατανόηση και την διευκόλυνση του χρήστη κάνει αυτόματα παύση. Ταυτόχρονα με αυτή την λειτουργία η μπάρα σημείου χρονικής ένδειξης αλλάζει από μπλέ χρώμα σε κόκκινο (βλ. Σχήμα 4.17).

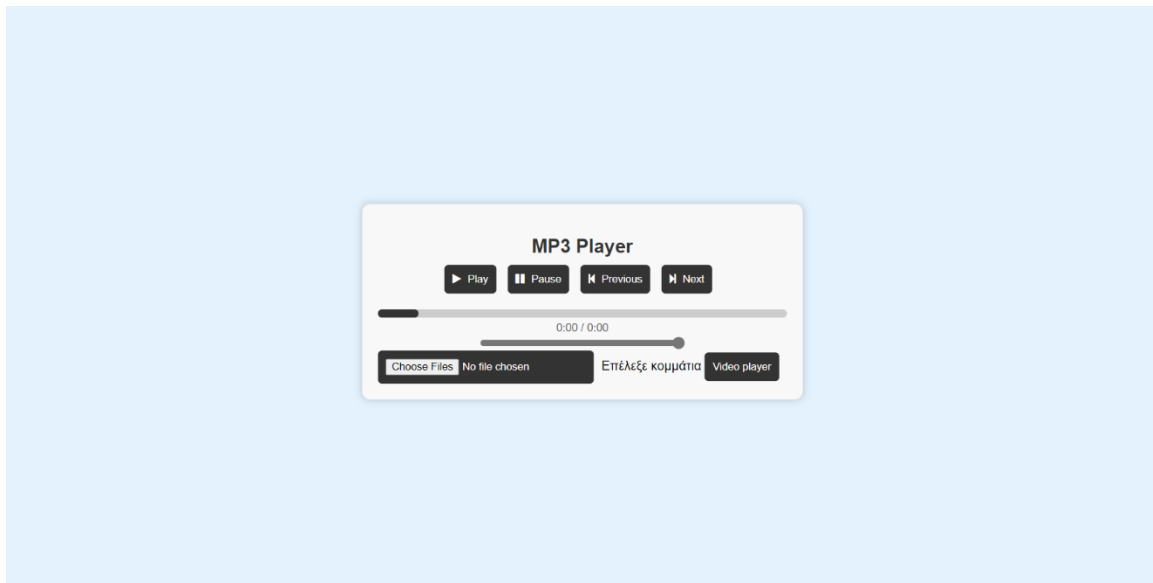


Σχήμα 4.17: Αποτέλεσμα χρήσης κουμπιού "Παίξε το κλείσιμο!"

Τέλος για τα 3 τελευταία κουμπιά του πρώτου τμήμας του GUI (“Παίξε την εισαγωγή 3 φορές”, “Παίξε το ρεφρέν 3 φορές”, “Παίξε το κλείσιμο 3 φορές”). Τα κουμπιά αυτά παρέχουν τις ίδιες λειτουργίες με τα προηγούμενα 3 κουμπιά που αναφέρθηκαν παραπάνω, με την βασική διαφορά ότι δεν σταματάνε την αναπαραγωγή του βίντεο την πρώτη φορά που ολοκληρωθεί το προκαθορισμένο χρονικό όριο αλλά επαναλαμβάνουν αυτή την διαδικασία 3 φορές. Η προσθήκη αυτών των κουμπιών μπορεί με μια πρώτη ματιά να θεωρηθεί πλεονασμός καθώς παρέχουν σχεδόν ίδιες λειτουργίες με τα κουμπιά που αναπαράγουν μια μονο φορα τα συγκεκριμένα χρονικά σημεία του έργου. Όμως συνηθίζεται σαν πρακτική άσκηση ανάμεσα στους μαθητεύμενους μουσικούς να παίζουν επαναλαμβανόμενα πολλές φορές επιλεγμένα σημεία των κομματιών που μαθαίνουν. Στηριζόμενος σε αυτό το γεγονός η χρήση και η προσθήκη των συγκεκριμένων κουμπιών αν μη τι άλλο τίθεται χρήσιμη και πρακτική.

4.4 Γραφικό περιβάλλον του Audio player

Όπως αναφέρθηκε στο προηγούμενο υποκεφάλαιο βασική προϋπόθεση ώστε ο χρήστης να αλληλεπιδράσει με το δεύτερο τμήμα του GUI αποτελεί η χρήση του κουμπιού “MP3 Player”. Το δεύτερο τμήμα του GUI θα μπορούσαμε να το πούμε λιγότερο εμπλουτισμένο από ότι το πρώτο. Αυτό στηρίζεται στην υπόθεση πως όταν ένας χρήστης έχει εξοικειωθεί σε έναν ικανοποιητικό βαθμό με ένα μουσικό έργο, δεν θεωρείται αναγκαία η βοήθεια από οπτικές λειτουργίες παρά μόνο αυτού του ήχου. Καθώς σαν τελικό στάδιο στην εκμάθηση του θα προσπαθεί να ακολουθεί και να παίζει με το μουσικό του όργανο το μουσικό έργο που θα έχει επιλέξει. Έτσι καθώς η προσοχή του βρίσκεται περισσότερο πάνω στο μουσικό του όργανο σκοπίμως και το δεύτερο τμήμα του GUI είναι απλούστερο(βλ. Σχήμα 4.18).



Σχήμα 4.18: Συνολικό GUI Audio Player

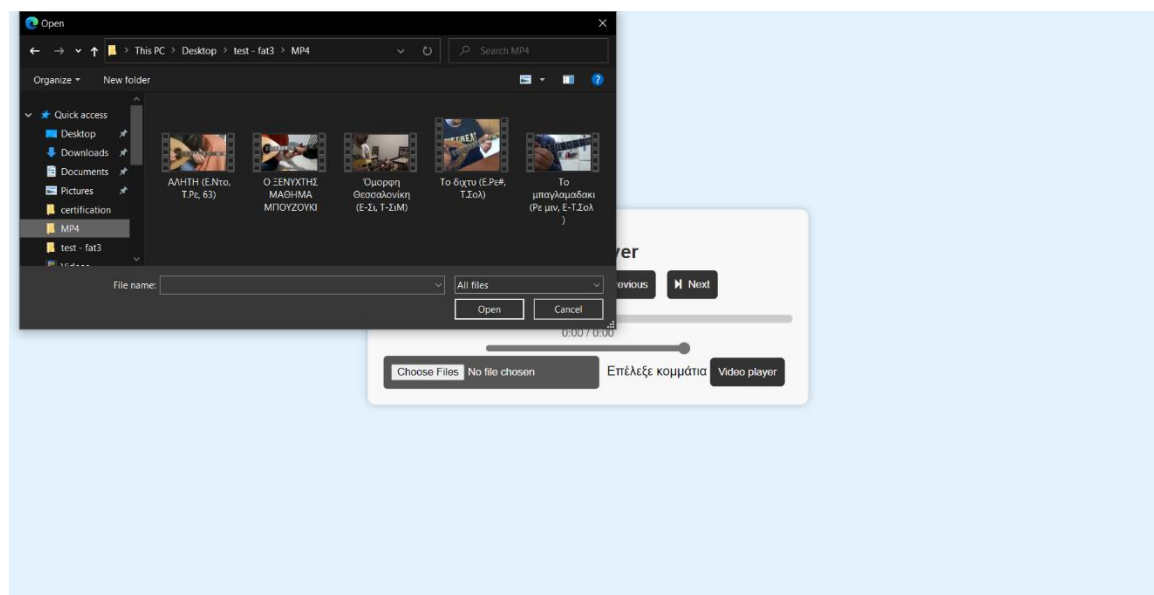
4.4.1 Ανάλυση γραφικών στοιχείων Audio Player

Το δεύτερο τμήμα του GUI αποτελείται από ένα πλαίσιο όπου στο εσωτερικό του υπάρχουν: μπάρα σημείου χρονικής ένδειξης, κουμπί play, κουμπί pause, συρόμενη μπάρα για ρύθμιση έντασης ήχου, ακριβής ένδειξη λεπτού και δευτερολέπτου που βρίσκεται το κομμάτι, ένδειξη συνολικής διάρκειας μουσικού έργου, κουμπί previous, κουμπί next, κουμπί επιλογή προσθήκης κομματιών και κουμπί επιστροφής στο Video Player.

4.4.2 Λειτουργία κουμπιών "Choose files" και "Video Player"

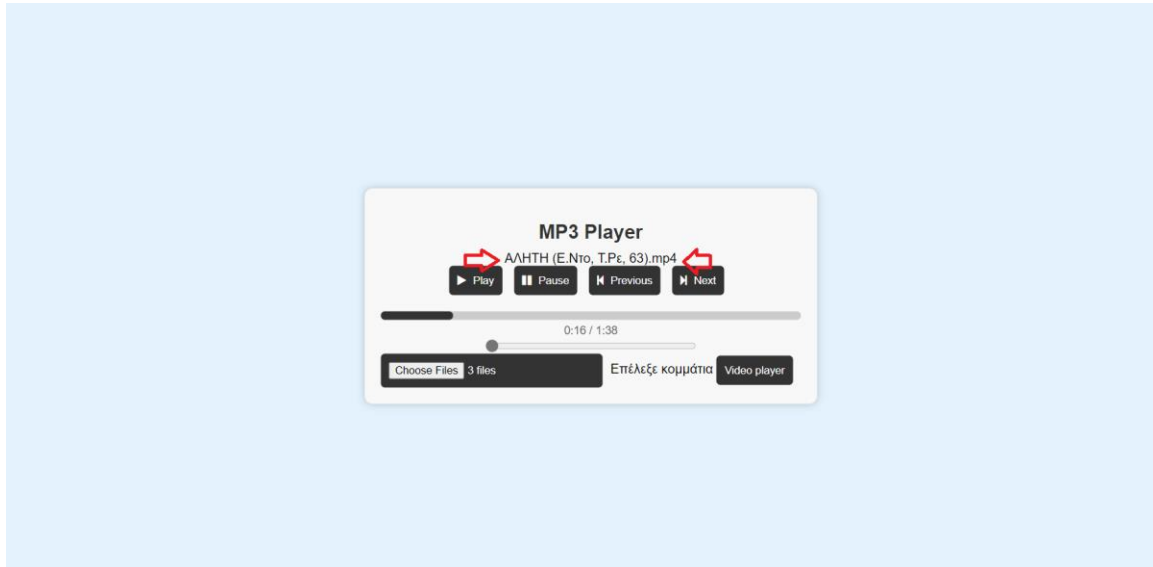
Η πιο απλή σε λειτουργία του MP3 Player αποτελεί το **κουμπί Video player**. Με την επιλογή αυτού του κουμπιού ο χρήστης μπορεί να επιστρέψει στο πρώτο τμήμα του GUI το Video Player.

Σαφώς για να είναι λειτουργικό το MP3 player πρέπει να παρέχεται η δυνατότητα στον χρήστη να προσθέτει κομμάτια προς αναπαραγωγή. Αυτή η λειτουργία και σκοπός επιτυγχάνεται μέσω του **κουμπιού Choose Files**. Όταν ο χρήστης κάνει κλικ πάνω σε αυτό το κουμπί τότε απευθείας θα του δοθεί η δυνατότητα να φορτώσει την λίστα με τα κομμάτια της επιλογής του απο τον προσωπικό του υπολογιστή (βλ. Σχήμα 4.19). Επιπλέον λειτουργία αυτού του κουμπιού είναι να θυμάται την σειρά με την οποία ο χρήστης επέλεξε τα κομμάτια ώστε να τα προσθέτει σε μία λίστα (μη ορατή προς τον χρήστη) με ακριβώς την ίδια σειρά που τα διάλεξε προηγουμένως.



Σχήμα 4.19: Στάδιο επιλογής μουσικού έργου

Εφόσον ο χρήστης έχει επιλέξει τα κομμάτια που επιθυμεί ο MP3 Player να αναπαράξει. Τότε ακριβώς κάτω από τα κουμπιά ρυθμίσεων θα εμφανιστεί ο τίτλος του μουσικού έργου που αναπαράγεται εκείνη την στιγμή (βλ. Σχήμα 4.20)



Σχήμα 4.20: Αποτέλεσμα χρήσης κουμπιού Choose Files

4.4.3 Λειτουργία κουμπιών ρυθμίσεων του Audio Player

Απο την στιγμή που ο χρήστης έχει επιλέξει τα κομμάτια χρειάζεται ένα τρόπο ώστε να ξεκινήσει η αναπαραγωγή του ήχου. Ακριβώς αυτό τον σκοπό εξυπηρετεί με την λειτουργία του το **κουμπί "Play"**. Αντίστοιχα αν επιθυμεί να θέσει σε παύση το MP3 Player το μόνο που έχει να κάνει είναι να χρησιμοποιήσει το **κουμπί "Pause"** καθώς αυτή είναι η μοναδική του λειτουργία.

Αναφερόμενοι στο σχήμα 4.20 μπορούμε να παρατηρήσουμε κάτω από τον τίτλο του μουσικού έργου την ύπαρξη των **κουμπιών "Previous"** και **"Next"**. Η πρώτη και βασική λειτουργία του κουμπιού "Next" είναι: από τα κομμάτια που έχει επιλέξει ο χρήστης και απο την λίστα που είχε δημιουργηθεί με την χρήση του κουμπιού "Choose Files", στην περίπτωση που ο χρήστης κάνει κλικ πάνω στο κουμπί Next να προχωράει στο επόμενο κομμάτι της λίστας και να ξεκινάει αυτόματα η αναπαραγωγή του. Στην περίπτωση που αναπαράγεται το τελευταίο κομμάτι της λίστα και ο χρήστης πατήσει το κουμπί τότε θα ξεκινήσει η αναπαραγωγή του πρώτου κομματιού που εντάχθηκε στην λίστα. Με την ίδια λογική το κουμπί "Previous" κάνει την ίδια λειτουργία με την μοναδική διαφορά ότι πηγαίνει στο προηγούμενο κομμάτι της λίστας. Όπως και στην ανάλογη περίπτωση του Next έτσι και στην περίπτωση του Previous, αν την στιγμή που πατηθεί το κουμπί η λίστα παίζει το πρώτο κομμάτι τότε θα οδηγηθεί και ξεκινήσει την αναπαραγωγή του τελευταίου κομματιού στην λίστα.

Επιπλέον δυνατότητες που παρέχουν τα κουμπιά Previous και Next είναι: Με την χρήση τους τότε αυτομάτως ο τίτλος αλλάζει αναλόγως με το αρχείο που αναπαράγεται εκείνη την στιγμή.

Επίσης με την εναλλαγή των κομματιών ενημερώνονται αυτόματα και τα πεδία ακριβής ένδειξη λεπτού και δευτερολέπτου που βρίσκεται το κομμάτι και ένδειξη συνολικής διάρκειας μουσικού έργου.

4.4.4 Λειτουργίες μπάρας σημείου χρονικής ένδειξης και συρόμενης μπάρας για ρύθμιση έντασης ήχου

Δεδομένου ότι ο χρήστης έχει επιλέξει μερικά μουσικά αρχεία προς αναπαραγωγή και επιπλέον ότι έχει επιλέξει ήδη το κουμπί Play μπορεί πλέον να αξιοποιήσει τις δυνατότητες που διαθέτουν οι μπάρες.

Πρώτα η **μπάρα σημείου χρονικής ένδειξης**. Με ένα απλό κλικ σε οποιοδήποτε σημείο πάνω της η αναπαραγωγή του ήχου θα συνεχίσει από το επιλεγμένο αυτό χρονικό σημείο. Δεύτερον η συρόμενη μπάρα ρύθμισης έντασης ήχου παρέχει τον έλεγχο στον χρήστη για το επίπεδο έντασης του ήχου. Την μείωση ή και την αύξηση αυτή μπορεί να επιτευχθεί κάνοντας αριστερό κλικ και κρατώντας παρατεταμένα το κλικ πάνω στην μπάρα. Η μετακίνηση προς τα δεξιά αυξάνει την ένταση του ήχου και η αριστερή μετακίνηση μειώνει την ένταση του ήχου.

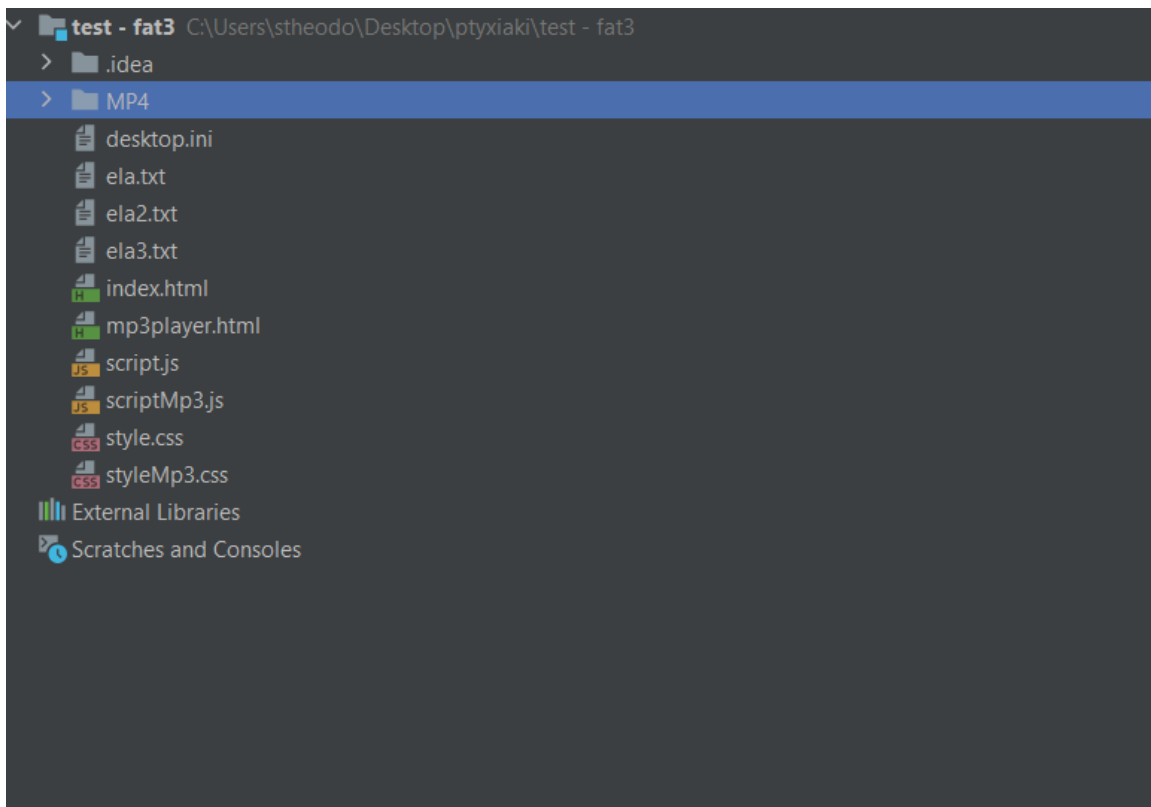
Ανάλυση κώδικα

5.1 Εισαγωγή

Όλες οι παραπάνω λειτουργίες πραγματοποιήθηκαν με την χρήση των τεχνολογιών και γλωσσών HTML, CSS και JavaScript οι οποίες αναφέρθηκαν προηγουμένως. Παρακάτω θα αναλύσουμε τον τρόπο με τον οποίο επιτεύχθηκε αυτό σε επίπεδο κώδικα.

5.2 Η δομή του project

Στο σχήμα 5.1 μπορεί κανείς να παρατηρήσει την δομή του συγκεκριμένου project και να εντοπίσει όλα τα αρχεία που το συνθέτουν.



Σχήμα 5.1: Δομή project

Περίληπτικά δίνεται παρακάτω ο λόγος ύπαρξης του κάθε αρχείου:

1. Αρχικά ο φάκελος MP4 περιέχει μερικά αρχεία μορφής .MP4 τα οποία χρησιμοποιούνται για τον εμπλουτισμό της λίστας του video player
21. Το αρχείο **desktop.ini** δεν έχει κάποια πρακτική λειτουργία στο project. Όμως δημιουργείται αυτόματα απο τον Code editor (IDE), με σκοπό να βοηθάει στην διόρθωση του συντακτικού τον προγραμματιστή καθώς αυτός γράφει τον κώδικα
22. Τα επόμενα **3 text files** αρχεία, είναι αρχεία όπου περιέχουν μέσα τους στίχους από τα μουσικά έργα. Το κάθε ένα αντιστοιχεί σε ένα από τα MP4 αρχεία
23. Το **index.html** είναι ο σκελετός της πρώτης σελίδας. Μέσα σε αυτό υπάρχουν όλα τα στοιχεία που δομούν την διεπαφή του Video Player. Αποτελεί ένα από τα πιο βασικά μέρη του project καθώς είναι υπεύθυνο για την αλληλεπίδραση του γραφικού περιβάλλοντος και των υπόλοιπων στοιχείων της εφαρμογής
24. Τον ίδιο σκοπό με το index.html εξυπηρετεί το αρχείο **mp3player.html**. Σε αυτό ζούν και βρίσκονται όλα τα στοιχεία της δεύτερης σελίδας του project, του Audio Player
25. Η λειτουργία του **style.CSS** είναι να εμπλουτίζει τα γραφικά στοιχεία και να ομορφαίνει το περιεχόμενο της σελίδας του Video Player. Αυτό επιτυγχάνεται με τον κώδικα CSS που διαθέτει μέσα το αρχείο και επηρεάζει άμεσα το index.html.
26. Επόμενο αρχείο που περιέχει κώδικα CSS είναι το **styleMp3.css**. Όπως περιγράψαμε για το προηγούμενο css αρχείο, το συγκεκριμένο εμπλουτίζει και τροποποιεί τα γραφικά στοιχεία του περιεχομένου της σελίδας του Audio Player

27. Το αρχείο **script.js** χρησιμοποιεί την γλώσσα προγραμματισμού JavaScript και επηρεάζει σε μεγάλο βαθμό το αρχείο index.html. Η λειτουργία του και η ύπαρξη του είναι απαραίτητη για το project καθώς μέσα σε αυτό βρίσκονται όλοι οι μέθοδοι που κάνουν λειτουργικό το πρώτο μέρος του project δηλαδή του Video Player.
28. Το αρχείο **scriptMp3.js** και αυτό είναι χρησιμοποιεί όπως και το προηγούμενο .js αρχείο την JavaScript και το αρχείο που επηρεάζει είναι το mp3player.html. Παρέχει με την σειρά του όλα τα απαραίτητα κομμάτια και μεθόδους που κάνουν λειτουργικό το Audio Player δηλαδή την δεύτερη σελίδα του project.

Στην συνέχεια θα αναλυθούν ένα προς ένα σε βάθος οι μέθοδοι και ο κώδικας που χρησιμοποιούν τα κύρια αρχεία που δομούν το project.

5.3 Το index.html

Το index.html αρχείο όπως είναι φανερό και από την ονομασία του αρχείου χρησιμοποιεί html κώδικα. Για να μπορέσουμε να κατανοήσουμε τον κώδικα αυτό θα πρέπει αρχικά να αναλυθούν τα html tags και όποια στοιχεία περιέχουν αυτά. Στην συνέχεια θα γίνει μία ειδικότερη ανάλυση στα κύρια html tags του όπως το <html>, <head>, <body>

5.3.1 Ανάλυση του index.html

Αρχικά το πρώτο tag που θα δει όποιος ανοίξει το συγκεκριμένο αρχείο είναι το <html>. Ονομάζεται "γονιός" (parent tag) και μέσα σε αυτό υπάρχουν όλα τα υπόλοιπα tags. Τα tags που βρίσκονται εσωτερικά σε αυτό ορίζονται ως "παιδιά" του (child tags). Οι ανάλυση των tags θα ξεκινήσει από πάνω προς τα κάτω (top-down analysis) δηλαδή από την πρώτη γραμμή και θα συνεχιστεί μέχρι και την τελευταία γραμμή κώδικα.

5.3.2 Το <head> tag του index.html

Συνολικά, αυτό το τμήμα κώδικα ρυθμίζει τους απαραίτητους πόρους για την ιστοσελίδα, συμπεριλαμβανομένων κωδικοποίησης χαρακτήρων(character encoding), τίτλου σελίδας(title), φύλλων στυλ(stylesheets) και εξωτερικών γραμματοσειρών(fonts). Διασφαλίζει ότι η ιστοσελίδα φαίνεται οπτικά ελκυστική και υποστηρίζει εικονίδια από το Font Awesome και το Google Fonts.

Περισσότερο έμφαση πρέπει να δοθεί στην συγκεκριμένη γραμμή κώδικα :

```
<link rel="stylesheet" href="style.css">
```

Στην πραγματικότητα αυτή η γραμμή ευθύνεται για την σύνδεση του αρχείου style.css με το index.html.

5.3.3 Το <body> tag του index.html

Μέσα στο συγκεκριμένο tag βρίσκονται όλα τα στοιχεία (elements) που απαρτίζουν το γραφικό περιβάλλον του Video Player. Συνολικά, ο κώδικας του ρυθμίζει τη δομή και τα κοντέινερ για μια ιστοσελίδα, συμπεριλαμβανομένου ενός προγράμματος αναπαραγωγής βίντεο, μιας λίστας βίντεο, στοιχείων ελέγχου κουμπιών, κοντέινερ στίχων και ενός script tag για πρόσθετη λειτουργικότητα.

Η βασικός σκελετός του κώδικά του συνοπτικά:

```
<body>
  <div class="container show-controls">
    <!-- ... -->
  </div>
  <div class="vList">
    <!-- ... -->
  </div>
  <div class="button-container">
    <!-- ... -->
  </div>
  <div class="KeysOpen">
    <!-- ... -->
  </div>
  <div id="text-file-contents" class="lyrics" style="display: none;">
    <!-- ... -->
  </div>
  <script src="script.js"></script>
</body>
```

Στο πρώτο **div tag "container show-controls"** βρίσκονται οτιδήποτε περιέχει μέσα το Video Player. Αναλυτικότερα συμπεριλαμβάνει: το τίτλο του αρχείου που έχει επιλεγεί για αναπαραγωγή, τη μπάρα σημείου χρονικής ένδειξης, το κουμπί mute/unmute, την συρόμενη μπάρα για ρύθμιση έντασης ήχου, την ακριβής ένδειξη λεπτού και δευτερολέπτου που βρίσκεται το βίντεο, την ένδειξη συνολικής διάρκειας βίντεο, το κουμπί rewind, το κουμπί pause/play, το κουμπί forward, το κουμπί ρύθμισης ταχύτητας αναπαραγωγής και το κουμπί της λειτουργίας πλήρους οθόνης.

Το δεύτερο **div tag "vList"** περιέχει την λίστα με τα αρχεία που είναι έτοιμα για επιλογή και αναπαραγωγή.

Στο τρίτο **div tag "button-container"** υπάρχουν τα κουμπιά "Παιξε την εισαγωγή!", "Παίξε το ρεφρέν!" και "Παίξε το κλείσιμο!". Επιπλέον μέσα του έχει ένα ακόμα child div tag που σε αυτο βρίσκονται τα κουμπιά επαναλήψεων.

Το τέταρτο **div tag "KeysOpen"** συμπεριλαμβάνει το κουμπί με την λειτουργία εμφάνισε/κρύψε στίχους καθώς και το κουμπί "Mp3 Player" που οδηγεί τον χρήστη στην σελίδα του Audio player

Το τελευταίο **div tag "lyrics"** αποτελεί το πλαίσιο στο οποίο η ιστοσελίδα φορτώνει και εμφανίζει τους στίχους.

Η τελευταία γραμμή κώδικα `<script src="script.js"></script>` έχει την ευθύνη να συνδέει το JavaScript αρχείο με το html αρχείο. Χωρίς την ύπαρξή της όλα τα στοιχεία πάνω στην σελίδα θα υστερούσαν σε λειτουργικότητα καθώς ο JavaScript κώδικας που περιέχει το script.js ζωντανεύει κάθε ένα από τα στοιχεία που αναφέρθηκαν προηγουμένως.

5.4 Ανάλυση του style.css

Προηγουμένως αναφερθήκαμε στην γραμμή κώδικα `<link rel="stylesheet" href="style.css">` που συνδέει το html με το css file. Η λειτουργία του style.css είναι να τροποποιεί οπτικά όλα τα προαναφερθέντα στοιχεία που βρίσκονται μέσα στα div tags του index.html αρχείου. Ο τρόπος με τον οποίο το πετυχαίνει αυτό είναι αρκετά απλός. Παρατηρούμε ότι σχεδόν για κάθε ένα div tag που έχει χρησιμοποιηθεί συνοδεύεται με την μεταβλητή class. Η μεταβλητή class στην ουσία δίνει ένα ξεχωριστό όνομα αναφοράς. Αυτός ο διαχωρισμός αποτελεί κλειδί για την σύνταξη κώδικα css καθώς μπορούμε απλά χρησιμοποιώντας το όνομα αναφοράς του div να του επηρεάσουμε στοιχεία οπτικά. Όπως το ύψος, πλάτος, πάχος διαχωριστικών γραμμών, χρώμα και άλλα πολλά.

Ένα μικρό παράδειγμα κώδικα είναι το εξής:

```
.options button{
    width: 40px;
    height: 40px;
    border: none;
    color: #fff;
    cursor: pointer;
    font-size: 19px;
    background: none;
}
```

Παραπάνω μπορούμε να δούμε ότι η αναφορά γίνεται για τα κουμπια που βρίσκονται στα `div` με `class name = options`. Συνοπτικά ορίζει τις διαστάσεις του, αφαιρεί το περίγραμμα, κάνει το κείμενο λευκό, αλλάζει τον κέρσορα σε δείκτη, καθορίζει το μέγεθος γραμματοσειράς και μετατρέπει το φόντο σε διαφανές.

Αναλυτικά, η κάθε γραμμή κάνει το εξής:

1. `width: 40px; and height: 40px;`: Ορίζει το πλάτος και το ύψος του κουμπιού σε 40 pixel το καθένα.
2. `border: none;`: Αφαιρεί το περίγραμμα γύρω από το κουμπί.
3. `color: #fff;`: Ορίζει το χρώμα κειμένου του κουμπιού σε λευκό (#fff)
4. `cursor: pointer;`: Αλλάζει τον κέρσορα του ποντικιού σε δείκτη όταν τοποθετείτε το δείκτη του ποντικιού πάνω από το κουμπί, υποδεικνύοντας ότι υπάρχει δυνατότητα κλικ
5. `font-size: 19px;`: Ορίζει το μέγεθος γραμματοσειράς του κειμένου του κουμπιού σε 19 pixel.
6. `background: none;`: Αφαιρεί το χρώμα φόντου ή την εικόνα από το κουμπί, αφήνοντάς το διαφανές.

5.5 Ανάλυση του `script.js`

Το συγκεκριμένο αρχείο έχει σκοπό να προσθέτει λειτουργικότητα στην ιστοσελίδα. Βασικά στοιχεία του `script.js` είναι οι σταθερές, οι μεταβλητές και οι μέθοδοι που χρησιμοποιεί για να πετύχει τον σκοπό του. Παρακάτω θα αναλυθούν όλα τα στοιχεία που περιέχει στον κώδικα του.

5.5.1 Σταθερές και μεταβλητές του `script.js`

Στις πρώτες γραμμές κώδικα του συγκεκριμένου αρχείου συναντάμε όλες τις σταθερές και μεταβλητές που χρησιμοποιεί. Για παράδειγμα:

```
const container = document.querySelector(".container"),
```

Αυτή η γραμμή κώδικα χρησιμοποιεί τη μέθοδο `document.querySelector()` για να επιλέξει το πρώτο στοιχείο στο HTML έγγραφο που ταιριάζει με τον επιλογέα CSS `".container"`. Εκχωρεί το επιλεγμένο στοιχείο στο `const container`.

Το προηγούμενο παράδειγμα αναφέρεται σε μια σταθερά. Ακολουθεί παράδειγμα με μία γραμμή κώδικα για τον χειρισμό μιας μεταβλητής.

```
vList = document.getElementById("vList");
```

Αυτή η γραμμή χρησιμοποιεί τη μέθοδο `getElementById()` για να επιλέξει το στοιχείο στο έγγραφο HTML με `id = "vList"`. Στη συνέχεια εκχωρεί το επιλεγμένο στοιχείο στη μεταβλητή `vList`.

5.5.2 Η μέθοδος `formatTime`

Αυτή η μέθοδος παίρνει μια χρονική τιμή (σε δευτερόλεπτα) ως όρισμα και τη μορφοποιεί ως συμβολοσειρά στη μορφή "ωω:μμ:δδ" ή "μμ:δδ" (αν οι ώρες είναι μηδέν). Συμπληρώνει τις τιμές με μηδενικά στην αρχή όταν είναι απαραίτητο. Την `formatTime` την χρησιμοποιούν οι περισσότερες μέθοδοι που θα αναφερθούν παρακάτω.

5.5.3 Τα event Listeners και μέθοδοι για την ανανέωση του χρόνου

1) `mainVideo.addEventListener("timeupdate", e => {...});`

Το συγκεκριμένο event listener έχει δύο λειτουργίες. Η πρώτη είναι να ενημερώνει το πλάτος της μπάρας γραμμής προόδου με βάση την τρέχουσα χρονική στιγμή στο βίντεο. Δεύτερη λειτουργία του είναι να ανανεώνει τον τρέχον χρονικό δείκτη που υπάρχει μέσα στο player.

2) `mainVideo.addEventListener("loadeddata", e => {...});`

Το παραπάνω event listener παρέχει την δυνατότητα να ενημερώνει τον χρονικό δείκτη της συνολικής διάρκειας του βίντεο.

Το επόμενο event listener είναι υπεύθυνο για την ενημέρωση της τρέχουσας ώρας του βίντεο με βάση τη θέση στην οποία έγινε κλικ στη μπάρα γραμμής προόδου:

3) `videoTimeline.addEventListener("click", e => {...});`

Η μέθοδος **`draggableProgressBar`** :

4) `const draggableProgressBar = e => {...};`

Προσφέρει τη λειτουργία ενημέρωσης του πλάτους της μπάρας γραμμής προόδου και την τρέχουσα ώρα του βίντεο με βάση την κίνηση του ποντικιού, επίσης ανανεώνει τον τρέχον χρονικό δείκτη που υπάρχει μέσα στο player

Οι επόμενοι δυο event Listeners λειτουργούν συνδυαστικά με την προηγούμενη μεθοδο:

5) `videoTimeline.addEventListener("mousedown", () => {...});`

6) `document.addEventListener("mouseup", () => {...});`

Κοινός τους σκοπός είναι να προσθέτουν στην μπάρα γραμμής προόδου την δυνατότητα που έχει η μέθοδος `draggableProgressBar` όταν πατηθεί αριστερό κλικ πάνω της και συρθεί το ποντίκι.

5.5.4 Τα event Listeners και μέθοδοι για τις ρυθμίσεις έντασης ήχου

Το event Listener `volumeBtn.addEventListener('click', () => {...});` επηρεάζει το κουμπί ρύθμισης ήχου του βίντεο player. Είναι υπεύθυνο για όταν ο χρήστης πατάει το κουμπί να αλλάζει το εικονίδιο του από `mute` σε `unmute` αναλόγως την κατάσταση που βρίσκεται εκείνη την στιγμή. Επίσης προσφέρει την δυνατότητα αν πατηθεί το κουμπί και τεθεί σε `mute` να αφαιρεί τον ήχο ή να βάζει τον ήχο στο υψηλότερο του επίπεδο αν τεθεί σε τιμή `unmute`

Το event Listener `volumeSlider.addEventListener("input", e => {...});` επηρεάζει τις λειτουργίες της συρόμενης μπάρας ήχου. Ανάλογα με το σημείο που μετακινεί και αφήνει ο χρήστης το ποντίκι του αυξομειώνει την ένταση του ήχου.

5.5.5 Τα event Listeners και μέθοδοι για τα κουμπιά ταχύτητας αναπαραγωγής και πλήρους οθόνης

Το event Listener `speedBtn.addEventListener("click", () => {...});` προσφέρει την δυνατότητα όταν πατηθεί το κουμπί να εμφανίσει την λίστα. Συνδυαστικά με το επόμενο `speedOptions.querySelectorAll("li").forEach(option => {...});` Αφού έχει εμφανιστεί η λίστα δίνει την δυνατότητα στον χρήστη να επιλέξει και να αλλάζει την ταχύτητα αναπαραγωγής.

Την δυνατότητα για την είσοδο και έξοδο στην προβολή πλήρους οθόνης την προσθέτει το event listener `fullscreenBtn.addEventListener('click', () => {...});`

5.5.6 Τα event Listeners και μέθοδοι για τα κουμπιά play/pause, backward, forward

Τα παρακάτω 3 event Listeners είναι υπεύθυνα για τις εξής λειτουργίες. Πρώτη αν το βίντεο βρίσκεται σε κατάσταση αναπαραγωγής να αλλάζει το εικονίδιο του κουμπι από play σε pause και αντίστροφα αν το βίντεο βρίσκεται σε κατάσταση παύσης. Δεύτερη και περισσότερο σημαντική λειτουργία όταν ο χρήστης πατάει το κουμπί play να ξεκινάει η αναπαραγωγή όπως και αν πατάει το κουμπί pause το βίντεο να τίθεται σε παύση

```
playPauseBtn.addEventListener("click", () => {...});
```

```
mainVideo.addEventListener("play", () => {...});
```

```
mainVideo.addEventListener("pause", () => {...});
```

5.5.7 Τα event Listeners και μέθοδοι για τους στίχους

Οι λειτουργία δυναμικής αλλαγής και επιλογής αρχείου των στίχων καθορίζεται από την ακόλουθη μέθοδο `listVideo.forEach(video => {`

```
    video.onclick = () =>{ ...  
});
```

Η μέθοδος `function toggleVisibility() {...}` προσφέρει την λειτουργικότητα στο κουμπί εμφάνισε/κρύψε στίχους

5.5.8 Μέθοδοι για λειτουργίες κουμπιών επανάληψης και αναπαραγωγής από συγκεκριμένο σημείο

Αρχικά οι 3 μέθοδοι που είναι υπεύθυνη για την αναπαραγωγή απο εισαγωγή, ρεφρεν και κλείσιμο είναι setTimeEisagwgi(), setTimeRefren(), setTimeKleisimo(). Οι κάθε μία από αυτές πρώτα θέτει τον προκαθορισμένο χρόνο και σημείο που θα αναπαραχθεί το βίντεο. Στην συνέχεια τροποποιεί την μπάρα προόδου αλλάζοντας το χρώμα.

Οι μέθοδοι που είναι υπεύθυνες για την λειτουργία των κουμπιών επανάληψης είναι setTimeEisagwgi3(), setTimeRefren3(), setTimeKleisimo3(). Κάνουν ακριβώς τις ίδιες ενέργειες με τα κουμπιά αναπαραγωγής από το ίδιο σημείο με την διαφορά ότι έχουν έναν πρόσθετο μετρητή επαναλήψεων

5.6 Ανάλυση του mp3player.html

Παρόμοια με το αρχείο index.html το αρχείο mp3player.html είναι υπεύθυνο για τα στοιχεία που φαίνονται στην δεύτερη σελίδα του project του Audio Player. Οι κυριότερες γραμμές κώδικα που πρέπει να δοθεί έμφαση είναι:

```
<link rel="stylesheet" href="styleMp3.css">
```

Όπως είχε αναφερθεί και στην ανάλυση του index.html αυτή η γραμμή είναι σημαντική διότι συνδέει το κομμάτι κώδικα css με αυτό του html

Δεύτερη σημαντική γραμμή κώδικα για την σύνδεση του scriptMp3 αρχείου με το mp3player.html αποτελεί η:

```
<script src="scriptMp3.js"></script>
```

5.6.1 Το <body> tag του mp3player.html

Μέσα στο <body> tag του mp3player.html αρχείου βρίσκεται σχεδόν όλος ο κώδικας που μπορούμε να βρούμε σε αυτό. Ο σκελετος του είναι ο ακόλουθος:

```
<div id="player">
  <h2>MP3 Player</h2>
  <div id="songTitle"></div>
  <button id="playBtn"><i class="fa fa-play"></i> Play</button>
  <button id="pauseBtn"><i class="fa fa-pause"></i> Pause</button>
  <button id="prevBtn"><i class="fa fa-step-backward"></i> Previous</button>
  <button id="nextBtn"><i class="fa fa-step-forward"></i> Next</button>
```

```

<div class="progress">
  <div class="progress-bar"></div>
</div>
<div class="time">0:00 / 0:00</div>
<input type="range" id="volumeSlider" min="0" max="1" step="0.1" value="1">
<input type="file" id="fileInput" class="chooseBtn" multiple>Επέλεξε κομμάτια</input>
<button onclick="location.href = 'index.html'" class="rightButtons">Video player</button>
</div>

```

Παρατηρούμε πως μέσα σε ένα μόλις `<div>` tag με `id=player` που είναι το parent tag είναι δυνατόν να χωρέσουν και όλα τα υπόλοιπα στοιχεία της σελίδας. Όπως ο τίτλος `<songTitle>`, όλα τα κουμπιά ρυθμίσεων του Audio player, η μπάρα προόδου, μπάρα ήχου. Ακόμη και το κουμπί για επιλογή κομματιών οπου θα γεμίσει η λίστα αναπαραγωγής.

Στο επόμενο υποκεφάλαιο θα αναφερθούμε κατευθείαν στο JavaScript κομμάτι δηλαδή το `scriptMp3.js` αρχείο και θα προσπεράσουμε το `css` αρχείο. Διότι όπως έχει επισημανθεί ήδη η συγκεκριμένη σελίδα προσφέρει σημαντικά λιγότερες οπτικές λειτουργίες από ότι αυτή που αναλύθηκε για το Video Player.

5.7 Ανάλυση του `scriptMp3.js`

Το συγκεκριμένο αρχείο χρησιμοποιεί και αυτό την γλώσσα προγραμματισμού JavaScript. Κύριος και βασικός σκοπός του είναι να προσθέτει λειτουργικότητα σε όλα τα πεδία και αντικείμενα που βρίσκονται στην δεύτερη σελίδα της εφαρμογής (Audio Player). Στην συνέχεια θα αναλυθούν τμηματικά τα event Listeners και μεθόδους που χρησιμοποιεί.

5.7.1 Μέθοδος `formatTime()`

Η μέθοδος `formatTime()` παίρνει μια χρονική τιμή (σε δευτερόλεπτα) ως όρισμα και τη μορφοποιεί ως συμβολοσειρά στη μορφή "ωω:μμ:δ" ή "μμ:δδ" (αν οι ώρες είναι μηδέν). Συμπληρώνει τις τιμές με μηδενικά στην αρχή όταν είναι απαραίτητο. Ο κώδικας που χρησιμοποιεί για να πετύχει τον σκοπό της είναι ίδιος με την `formatTime()` που αναφέρθηκε προηγουμένως για το αρχείο `script.js`

5.7.2 To event Listeners `fileInput.addEventListener('change', (event) => {...});`

Το συγκεκριμένο event Listener διαθέτει στο εσωτερικό του, κομμάτια κώδικα που αφορούν το μεγαλύτερο μέρος της λειτουργικότητας του Audio Player. Συνοπτικά, αυτός ο κώδικας που διαθέτει εσωτερικά χειρίζεται την επιλογή αρχείων ήχου. Δημιουργεί στοιχεία ήχου για κάθε επιλεγμένο αρχείο, ρυθμίζει τους event listeners για ενημέρωση της μπάρας-γραμμής προόδου, επίσης προσθέτει και την λειτουργία αναπαραγωγής του επόμενου τραγουδιού στη λίστα αναπαραγωγής όταν τελειώσει το τρέχον τραγούδι.

Ο βασικός σκελετός του είναι ο εξής:

```
let playlist = [];  
  
fileInput.addEventListener('change', (event) => {  
  for (let i = 0; i < files.length; i++) {  
  
    audio.addEventListener('timeupdate', () => {});  
  
    progressBar.parentNode.addEventListener('click', (event) => { });  
  
    audio.addEventListener('ended', () => {  
  }  
});
```

Αρχικά για την επιλογή αρχείων το event listener αξιοποιεί το κουμπί "Choose Files". Σημαντικό στοιχείο για την λειτουργία του αποτελεί η μεταβλητή πίνακας `playlist[]` διότι μέσα της αποθηκεύει το event listener τα κομμάτια που έχει επιλέξει ο χρήστης για αναπαραγωγή. Επίσης όταν το συγκεκριμένο κουμπί χρησιμοποιηθεί θα αποθηκεύσει τα αρχεία προς αναπαραγωγή με την σειρά που χρήστης τα επέλεξε.

Η λειτουργία του event listener `audio.addEventListener('timeupdate', () => {...});` είναι η εξής. Να ενημερώνει τη μπάρα-γραμμή προόδου και να εμφανίζει την τρέχουσα ώρα και διάρκεια του αρχείου ήχου.

Η λειτουργία του event listener `progressBar.parentNode.addEventListener('click', (event) => {...});` είναι η εξής. Όταν γίνει κλικ πάνω σε ένα τυχαίο σημείο στην μπάρα-γραμμή προόδου να συνεχίζει την αναπαραγωγή του ήχου από εκείνο το χρονικό σημείο. Επίσης ενημερώνει την τρέχουσα ώρα και διάρκεια του αρχείου ήχου.

Το event listener `audio.addEventListener('ended', () => {...})`. Διαθέτει τις ακόλουθες λειτουργίες: Παίζει το επόμενο τραγούδι στη λίστα αναπαραγωγής όταν τελειώσει το τρέχον τραγούδι. ενημερώνει το `currentAudioIndex` δηλαδή τον δείκτη πάνω στον πίνακα της λίστας αρχείων προς αναπαραγωγή, επαναφέρει την τρέχουσα ώρα του επόμενου τραγουδιού στο 0, ενημερώνει την εμφάνιση του τίτλου του τραγουδιού.

5.7.3 Τα event Listeners και μέθοδοι για τα κουμπιά ρυθμίσεων και μπάρας ήχου

Την λειτουργία του play button διαχειρίζεται το event listener `playBtn.addEventListener("click", () => {...})`; Εκτός από το να ξεκινάει την αναπαραγωγή προσθέτει και την δυνατότητα να φορτώνει τον τίτλο του κομματιού που αναπαράγεται εκείνη την στιγμή.

Την λειτουργία του pause button διαχειρίζεται το event listener `pauseBtn.addEventListener("click", () => {...})`; Το συγκεκριμένο listener διαθέτει μονάχα την λειτουργία παύσης ήχου.

Το `volumeSlider.addEventListener("input", () => {...})`; Με την λειτουργικότητα που διαθέτει δίνει την δυνατότητα στον χρήστη να επηρεάζει τα επίπεδα έντασης του ήχου απο την συρόμενη μπάρα ήχου.

Τα επόμενα δύο listener τα οποία ονομάζονται `nextBtn.addEventListener("click", () => {...})`; και `prevBtn.addEventListener("click", () => {...})`; Διαχειρίζονται τις λειτουργίες των κουμπιών next και previous. Το πρώτο δίνει τις εξής λειτουργίες στο κουμπί next. Αρχικά όταν χρησιμοποιηθεί το κουμπί ξεκινάει την αναπαραγωγή του επόμενου αρχείου στην λίστα. Επίσης ενημερώνει την εμφάνιση του τίτλου του τραγουδιού καθώς. Τις ίδιες λειτουργίες για το previous button διαχειρίζεται το δεύτερο event listener.

Είναι σημαντικό σε αυτό το σημείο να αναφέρουμε την μεταβλητή `currentAudioIndex`. Όλες οι προηγούμενες μέθοδοι την χρησιμοποιούν με διάφορους τρόπους για να πετύχουν τους σκοπούς τους.

Βιβλιογραφία