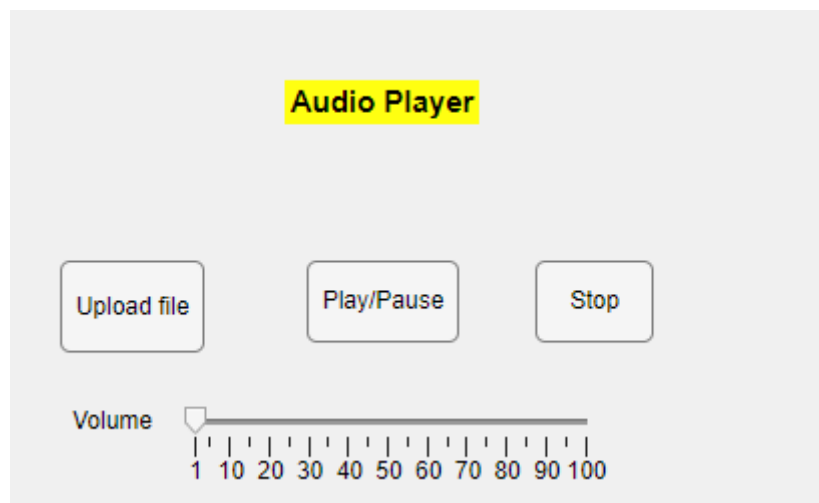


Αλέξανδρος Βυθούλκας 2013021
Απόστολος Καμπλιώνης 2015121
Επεξεργασία Ομιλίας και Ήχου - Απαλλακτική Εργασία:
“Μετατροπή του γραπτού λόγου σε ομιλία (Text To Speech – Speech To Text) και
αναγνώριση γλώσσας”

Checkpoint - Παραδοτέο 3 / Παρουσίαση προόδου εργασίας(15/4/22)

Όσον αφορά το 3ο Παραδοτέο, ερχόμενοι από την ολοκληρωμένη υλοποίηση της λειτουργίας **Speech-To-Text(STT)** εστιάσαμε στη δημιουργία ενός media player για τα ηχητικά αρχεία που υποστηρίζει η εφαρμογή μας. Το player αυτό, εφόσον ο χρήστης κάνει επιλογή του ηχητικού αρχείου μέσω του **Upload** button, παρέχει λειτουργίες αναπαραγωγής(**Play** button), αυξομείωσης της έντασης αναπαραγωγής(**Volume** button) αλλά και διακοπή αυτής(**Stop** button).

Σχεδιασμός και μορφή του κατάλληλου GUI για το audio player.



Για την υλοποίηση του **play/push button** σε συνδυασμό με την τιμή του **Volume Slider**, έγινε χρήση του **dsp.AudioFileReader** για να διαβάσουμε δεδομένα από το αρχείο ήχου και **audioDeviceWriter** για την αναπαραγωγή του ηχητικού, γράφοντας τα ηχητικά δείγματα στην συσκευή εξόδου. Ως αποτέλεσμα αποφεύχθηκε η επανεκκίνηση του αρχείου που αναπαράγεται κάθε φορά με την χρήση του audioplayer αφού έτσι κάθε φορά χρειάζεται να ξαναπαίξει το audio από το σημείο που έμεινε με την νέα τιμή του value. Στη συνέχεια, κάνουμε μια επανάληψη **while** για όσο δεν έχει τελειώσει το ηχητικό που έχουμε εκχωρήσει στο αντικείμενο **app.AFR** και το flag μας έχει την τιμή true. Ενώ τρέχει το while για κάθε επανάληψη καλούμε το αντικείμενο **app.AFR** σαν να είναι συνάρτηση και εκχωρούμε την τιμή στο **audio**, δηλαδή, **audio = app.AFR()**; Επίσης, για να παιχτεί το ηχητικό γράφουμε άλλη μια εντολή μέσα στο while, **app.ADW(app.VolumeSlider.Value/100 * audio)**; όπου παίζει το ηχητικό καλώντας το αντικείμενο **app.ADW** στο οποίο έχουμε εκχωρήσει την τιμή του **audioDeviceWriter** και δίνουμε σαν όρισμα το audio επί την τιμή του volume δια 100. Βάζουμε δια εκατό επειδή οι τιμές που έχουμε δώσει στον slider είναι “επί τοις εκατό” οπότε πρέπει να γίνει αυτό για να δουλέψει σωστά. Τέλος για να σιγουρευτούμε ότι το graphic callback θα προσπελαστεί γράφουμε την εντολή **drawnow limitrate** στο τέλος της επανάληψης. Αυτό χρειάζεται όταν έχουμε μεγάλες επαναλήψεις - loops και θέλουμε να διακόψουμε την επανάληψη για να προσπελάσουμε το callback κάποιου άλλου graphic αντικειμένου.

Ο κώδικας του play/pause button.

```
function PlayPauseButtonValueChanged(app, event)
    value = app.PlayPauseButton.Value;
    if value %elegxei ean h timh toy button einai true
        app.AFR = dsp.AudioFileReader(app.file); %diabazei to hxhtiko arxeio
        app.ADW = audioDeviceWriter(app.AFR.SampleRate); % grafei ta hxhtika deigmata sthn sysk
        app.VolumeSlider.Value = 1; %arxikopoioyme thn timh tou volume sto 1
        app.keepPlaying = value; % arxikopoioymai thn timh tou 'flag' sto true
        while ~isDone(app.AFR) && app.keepPlaying == true % while epanalhps h gia oso den exei t
            audio = app.AFR(); %kalei to antikeimeno app.AFR san na einai synarthsh kai to ekxv
            app.ADW(app.VolumeSlider.Value/100 * audio); % paizei to hxhtiko kalwntas to antike
            drawnow limitrate % drawnow einai gia na sigoyreytoyme oti to graphic callback tha
        end
        release(app.AFR); % afou teleiwsei to hxhtiko kleinei to arxeio kai stamataei thn x
        release(app.ADW);
        app.PlayPauseButton.Value = false;% vazei thn timh tou stete button sto false
    else
        app.keepPlaying = value; %alliws bbazei thn timh tou flag sto false
    end
end
```

Ο κώδικας του upload file button.

```
end

% Button pushed function: UploadfileButton
function UploadfileButtonPushed(app, event)
    app.file = uigetfile();
```

Ο κώδικας του stop button.

```
% Button pushed function: StopButton
function StopButtonPushed(app, event)
    app.keepPlaying = false;
```

Η μορφή της εφαρμογής μέχρι τώρα.

