

Σύστημα αναγνώρισης γραμμάτων στη νοηματική γλώσσα – 6^ο Δημοτικό Σχολείο Αμαλιάδας

Εκπαιδευτικοί: Δασιός Αθανάσιος ΠΕ86, Χριστοπούλου Θεώνη ΠΕ70 ΕΑΕ

Σκοπός του έργου ήταν να δώσει λύση στο πρόβλημα της επικοινωνίας ατόμων με αναπηρία που χρησιμοποιούν τη νοηματική γλώσσα, με άλλα άτομα τα οποία δεν γνωρίζουν τη νοηματική γλώσσα, μέσω ενός συστήματος το οποίο θα αναγνωρίζει χειρονομίες που αντιστοιχούν σε γράμματα ή και λέξεις στη νοηματική γλώσσα και θα τις μεταφράζει σε κείμενο, το οποίο και θα εμφανίζει σε μια οθόνη.

Οι επιμέρους εκπαιδευτικοί στόχοι του έργου ήταν οι παρακάτω:

- Η ευαισθητοποίηση των μαθητών σε θέματα συμπερίληψης ατόμων με αναπηρία.
- Η ανάπτυξη της ικανότητας των μαθητών να προσφέρουν στο κοινωνικό σύνολο, φτιάχνοντας ένα σύστημα με αξία για μια κοινωνία ισότητας, συμπερίληψης και αποδοχής.
- Η εξοικείωση των μαθητών με έννοιες και πρακτικές που αφορούν τη μηχανική μάθηση και την τεχνητή νοημοσύνη.
- Η εξοικείωση των μαθητών με έννοιες και πρακτικές που αφορούν τον προγραμματισμό.
- Η εξοικείωση των μαθητών με έννοιες και πρακτικές που αφορούν τη χρήση ανοιχτού υλικού (open hardware).
- Η εξοικείωση των μαθητών με έννοιες και πρακτικές που αφορούν τη χρήση ανοιχτού λογισμικού (open software).

Μαθητές που συμμετείχαν

Οι μαθητές που συμμετείχαν στο έργο ήταν όσοι μαθητές επιθυμούσαν από τις τάξεις Δ', Ε' και ΣΤ' του σχολείου, με την κάθε τάξη να αποτελείται από 2 τμήματα.

Υλικό που χρησιμοποιήθηκε

- PC
- Web camera
- BBC Micro:bit V2
- Display Module 1.8" 160x128 for BBC micro:bit

Λογισμικό – διεπαφές - βιβλιοθήκες που χρησιμοποιήθηκαν

- <https://teachablemachine.withgoogle.com/>
- <https://makeairobots.com/>
- <https://ai-robots.glitch.me/>
- <https://makecode.microbit.org/>
- <https://github.com/tkleisas/pxt-greekled-beta>
- <https://github.com/waveshare/PXT-WSLCD1in8>
- Εφαρμογή Pictoblox (διαθέσιμη και on line στο <https://pictoblox.ai/>)

Στάδια και περιγραφή υλοποίησης

1. Εκμάθηση του αλφάβητου της νοηματικής γλώσσας

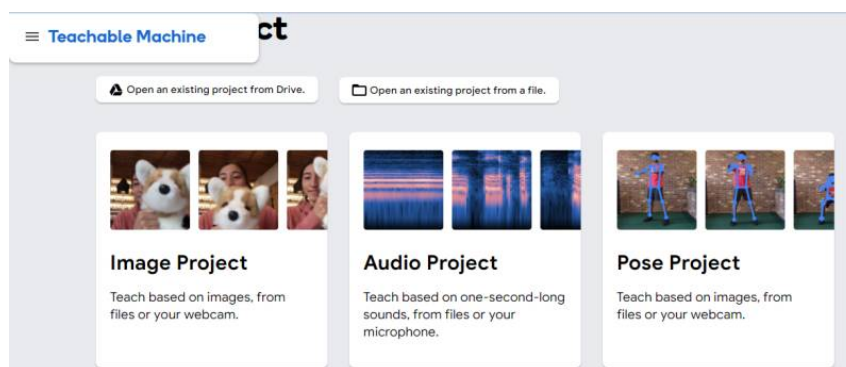
Σε αυτό το στάδιο δόθηκε στους μαθητές το ελληνικό αλφάβητο της νοηματικής γλώσσας, παρουσιάστηκαν ανάλογα video καθώς και έγινε ζωντανή επίδειξη από τους υπεύθυνους εκπαιδευτικούς του έργου. Παρακάτω φαίνεται το εν λόγω αλφάβητο:



Στη συνέχεια ανατέθηκε σε κάθε μαθητή η εκμάθηση 4 τουλάχιστον γραμμάτων, ωστόσο ορισμένοι μαθητές έμαθαν παραπάνω από 4 γράμματα, ενώ κάποιοι έμαθαν όλο το αλφάβητο.

2. Εκπαίδευση του συστήματος – δημιουργία του μοντέλου τεχνητής νοημοσύνης

Σε αυτό το στάδιο έγινε η εκπαίδευση του συστήματος τεχνητής νοημοσύνης (T.N.) με τη βοήθεια της πλατφόρμας Teachable Machine της Google (<https://teachablemachine.withgoogle.com/>):



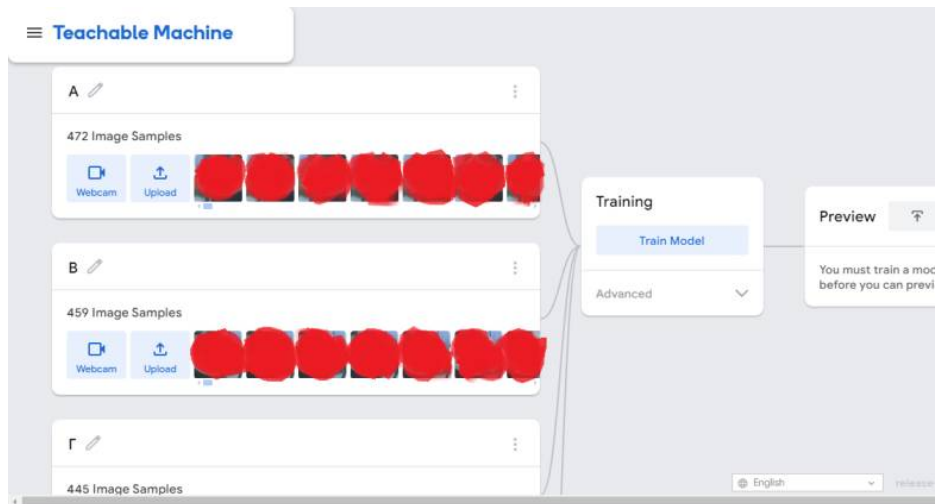
Μέσω αυτής της πλατφόρμας είναι δυνατή η εκπαίδευση και η εξαγωγή ενός μοντέλου τεχνητής νοημοσύνης το οποίο μπορεί να αναγνωρίζει είτε εικόνα, είτε ήχο, είτε κίνηση. Για την περίπτωση του συγκεκριμένου έργου, το επιθυμητό μοντέλο πρέπει να αναγνωρίζει την εικόνα από την κάμερα και να την ταξινομεί σε μία κλάση (class), η οποία αντιστοιχεί σε ένα γράμμα του ελληνικού αλφάβητου.

Έτσι δημιουργήθηκαν 25 κλάσεις, μία για κάθε ένα από τα 24 γράμματα του ελληνικού αλφάβητου και μία επιπλέον (κλάση none) για την περίπτωση όπου η εικόνα δεν αντιστοιχούσε σε κάποιο γράμμα, π.χ. λόγω μη ύπαρξης ατόμου, λανθασμένης χειρονομίας, παρεμβολής άλλου ατόμου μπροστά από την κάμερα κ.α.

Σε αυτό το στάδιο το κάθε παιδί ερχόταν μπροστά από το PC με την κάμερα και έκανε την αντίστοιχη χειρονομία για το κάθε γράμμα που είχε μάθει – εξασκήσει. Ο ένας υπεύθυνος εκπαιδευτικός το βοηθούσε και το καθοδηγούσε ως προς τη στάση και τη χειρονομία ενώ ο άλλος κατέγραφε μέσω της κάμερας και της πλατφόρμας την εικόνα. Με αυτό τον τρόπο προέκυψαν εκατοντάδες δείγματα (samples) για το κάθε γράμμα. Στις παρακάτω εικόνες φαίνονται από ένα δείγμα για το κάθε γράμμα του ελληνικού αλφάβητου της νοηματικής γλώσσας:

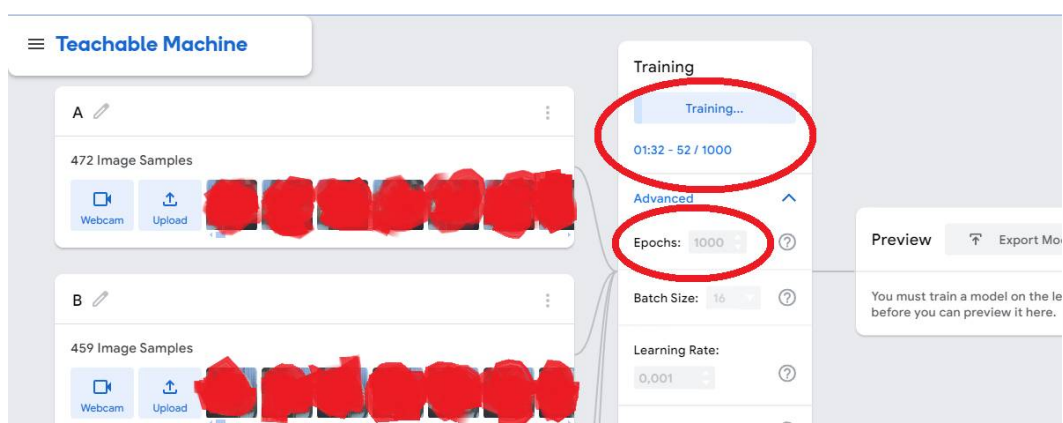


Στην επόμενη εικόνα φαίνεται το περιβάλλον της πλατφόρμας Teachable Machine κατά τη φάση της εισαγωγής των δειγμάτων με τη βοήθεια της κάμερας. Η πλατφόρμα δεν αποθηκεύει τις εικόνες – δείγματα, όμως δίνει τη δυνατότητα της αποθήκευσης των δειγμάτων τοπικά στον Η/Υ (download) όπως και του ανεβάσματος (upload) δειγμάτων από τον Η/Υ.

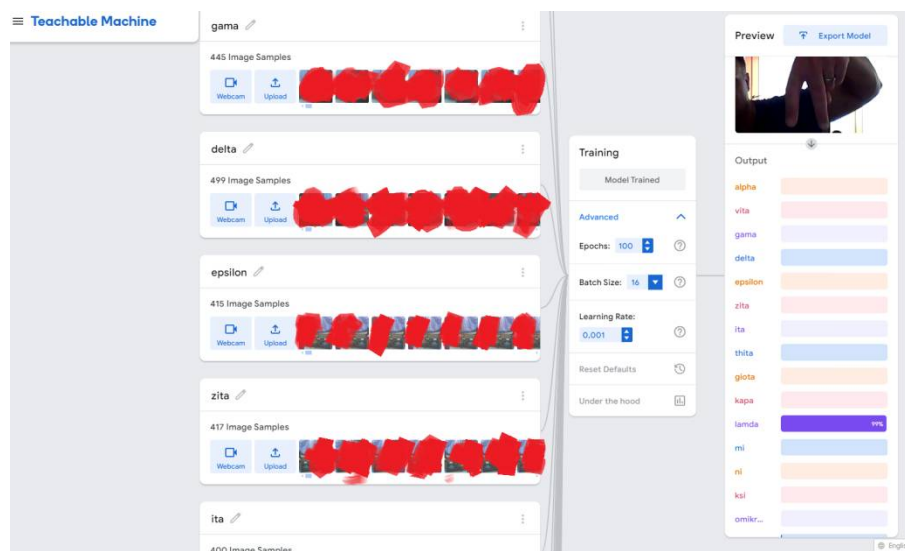


Η διαδικασία της καταγραφής των δειγμάτων επαναλήφθηκε πολλές φορές, αρχικά για δοκιμή με 4-8 γράμματα και λίγα παιδιά (δεκάδες δείγματα) και τελικά με όλα τα γράμματα και με όλα τα παιδιά (εκατοντάδες δείγματα). Τα δείγματα αποθηκεύονταν κάθε φορά προσωρινά στον Η/Υ και ανεβάζονταν εκ νέου στην πλατφόρμα για την επόμενη δοκιμή.

Σε αυτό το στάδιο αποδείχτηκε και η δυσκολία του εγχειρήματος – έργου, λόγω του ότι από την αρχή κιόλας προέκυψε ότι το ποσοστό επιτυχούς αναγνώρισης ήταν πολύ μικρό της τάξης 10% με 15%. Η πρώτη προσέγγιση για να ξεπεραστεί αυτή η δυσκολία ήταν να αυξηθεί ο αριθμός των παιδιών για κάθε γράμμα και επομένως και των δειγμάτων, η οποία δεν οδήγησε στο επιθυμητό αποτέλεσμα, δηλαδή στην αύξηση του ποσοστού επιτυχούς αναγνώρισης. Η δεύτερη προσέγγιση ήταν να αυξηθεί ο αριθμός των «περασμάτων» (epochs) κατά τη διαδικασία της εκπαίδευσης του μοντέλου. Έτσι πέρα από την προεπιλεγμένη τιμή για τα epochs που είναι 50, δοκιμάστηκαν οι τιμές 100, 500 και τελικά 1.000, με την τελευταία να καθιστά την εκπαίδευση του μοντέλου χρονοβόρο διαδικασία για πολλά δείγματα.



Ακόμα και με τη δεύτερη προσέγγιση για την επίλυση του προβλήματος του χαμηλού ποσοστού επιτυχούς αναγνώρισης, η βελτίωση ήταν μηδαμινή. Μια τρίτη προσέγγιση για την επίλυση του παραπάνω προβλήματος ήταν να προστεθούν δείγματα από άλλο χώρο, εκτός του εργαστηρίου Η/Υ του σχολείου, όπου το φυσικό φως πίσω από την κάμερα ήταν κατανεμημένο ομοιόμορφα και πίσω από περσίδες. Το μοντέλο εκπαιδεύτηκε εκ νέου συμπεριλαμβάνοντας τα νέα δείγματα με το νέο background, με χαμηλό αριθμό epochs (100) σε σχέση με την προηγούμενη προσπάθεια. Τα αποτελέσματα αυτής της τελευταίας προσέγγισης ήταν παραπάνω ενθαρρυντικά με το ποσοστό της επιτυχούς αναγνώρισης να φτάνει στο 67% (αναγνώριση 16 γραμμάτων από τα 24).



Παρακάτω φαίνεται, αριστερά το background του εργαστηρίου Η/Υ του σχολείου και δεξιά το νέο background:



Στη συνέχεια παρατίθεται οι σύνδεσμοι στους οποίους είναι διαθέσιμα τα μοντέλα των τριών τελευταίων εκπαιδεύσεων, με το τελευταίο να είναι αυτό που είχε τα καλύτερα αποτελέσματα:

<https://teachablemachine.withgoogle.com/models/95x3aR8O3/>

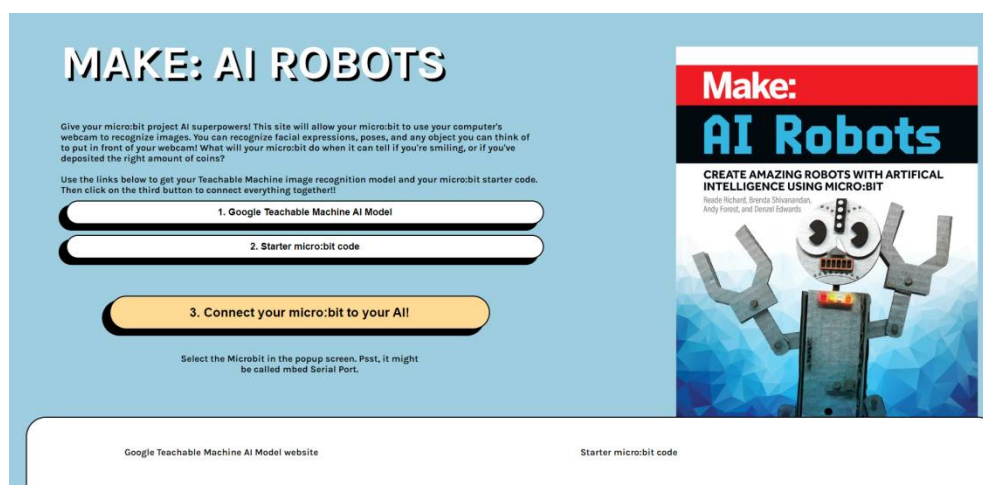
<https://teachablemachine.withgoogle.com/models/oxbuiDIRI/>

<https://teachablemachine.withgoogle.com/models/YCYaXM0pl/>

Η αντιμετώπιση του προβλήματος του χαμηλού ποσοστού επιτυχούς αναγνώρισης χρειάστηκε σημαντικό χρόνο, έτσι ώστε ο υπολειπόμενος διαθέσιμος χρόνος για την υλοποίηση του έργου να γίνει πολύ μικρός, ξεφεύγοντας σημαντικά από το αρχικό χρονοδιάγραμμα. Ωστόσο η προσπάθεια για να συνεχιστεί η υλοποίηση του έργου συνεχίστηκε.

3. Υλοποίηση με Micro::bit

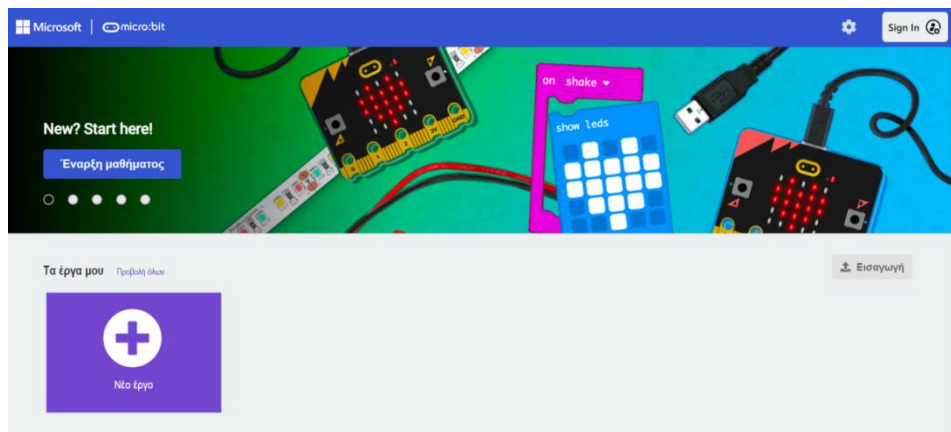
Σε αυτό το στάδιο το μοντέλο που έγινε εξαγωγή από το προηγούμενο στάδιο ανέβηκε στην πλατφόρμα <https://makeairobots.com/> (εναλλακτικά χρησιμοποιήθηκε και η πλατφόρμα <https://ai-robots.glitch.me/>) με σκοπό να μπορεί να τρέξει το μοντέλο εκεί, μιας και το υλικό που επιλέχθηκε για την υλοποίηση (του πρώτου σταδίου τουλάχιστον), το οποίο είναι το micro::bit, δεν μπορεί να «τρέξει» ένα τέτοιο μοντέλο. Η παραπάνω πλατφόρμα αποτελεί μια διεπαφή (interface) μεταξύ του micro::bit και του μοντέλου της τεχνητής νοημοσύνης.



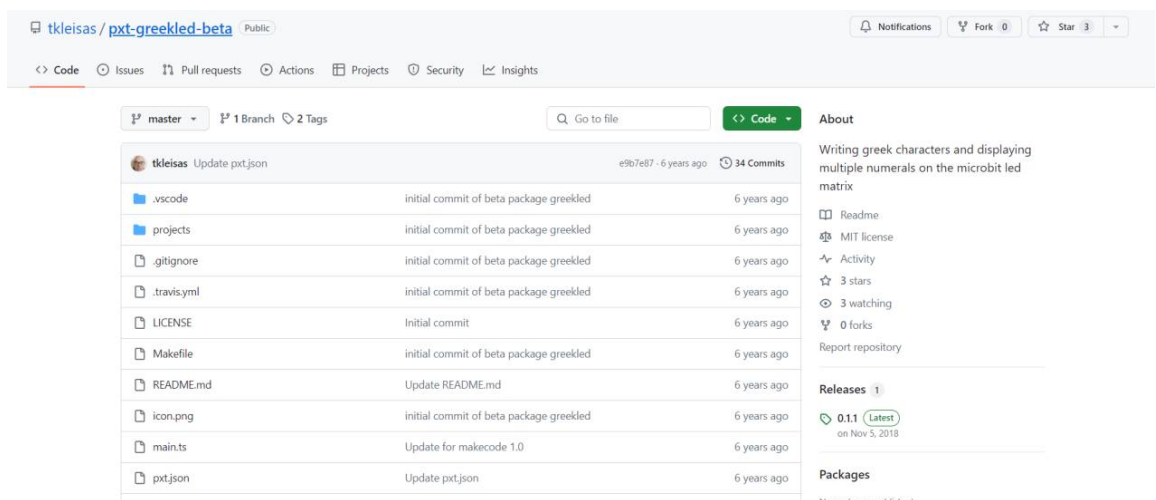
Στη συνέχεια συνδέθηκε η πλακέτα micro::bit, η οποία φαίνεται στη συνέχεια, με τον Η/Υ:



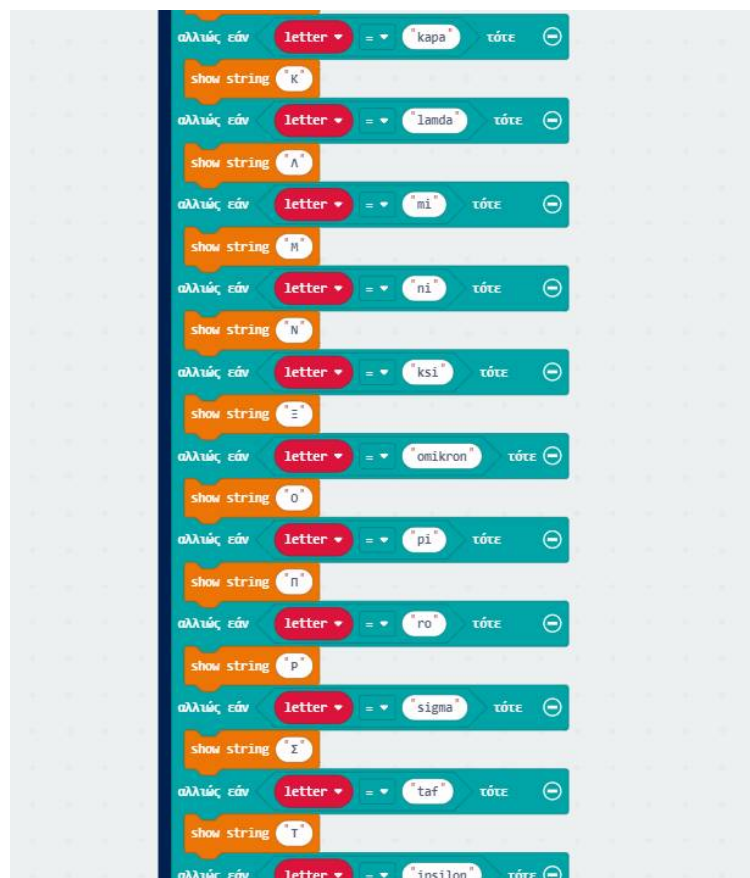
Για τη συγγραφή του κώδικα χρησιμοποιήθηκε η πλατφόρμα <https://makecode.microbit.org/>:



Αρχικά σκοπός του έργου ήταν η εμφάνιση των γραμμάτων του αλφάβητου, ύστερα από την οπτική αναγνώρισή τους από το σύστημα τεχνητής νοημοσύνης, στην «οθόνη» led matrix του micro::bit. Λόγω του ότι όμως το micro::bit δεν υποστηρίζει την εμφάνιση ελληνικών γραμμάτων στο led matrix, χρησιμοποιήθηκε η επέκταση – βιβλιοθήκη **greekled-beta** η οποία υποστηρίζει κάτι τέτοιο και βρίσκεται στο αποθετήριο <https://github.com/tkleisas/pxt-greekled-beta>:



Το πρόγραμμα (κώδικας) που γράφτηκε για το Micro::bit φαίνεται στη συνέχεια:





Όπως φαίνεται και παραπάνω το όνομα της κάθε κλάσης δεν είναι ο αντίστοιχος ελληνικός χαρακτήρας (Α, Β, Γ κλπ.) αλλά μια λέξη στα greeklish, δηλαδή alpha για το Α, vita για το Β, gama για το Γ κλπ. Αυτό έγινε διότι τα ελληνικά γράμματα δεν εμφανίζονταν στο led matrix με τον πρώτο τρόπο, γεγονός που σημαίνει ότι είτε η πλατφόρμα – διεπαφή που περιγράφεται παραπάνω, είτε η επικοινωνία αυτής με τη σειριακή θύρα του micro::bit δεν μπορεί να υποστηρίξει τους ελληνικούς χαρακτήρες. Έτσι αφού διαπιστώθηκε αυτό το πρόβλημα, έγινε εκ νέου εκπαίδευση του μοντέλου αλλάζοντας τα ονόματα των κλάσεων και το πρόγραμμα τροποποιήθηκε.

Επιπλέον έγινε προσπάθεια να χρησιμοποιηθεί, αντί του led matrix, μια LCD οθόνη συμβατή με το Micro::bit και συγκεκριμένα η Display Module 1.8" 160x128 for BBC micro:bit.



Ο κατασκευαστής της παραπάνω οθόνης παρέχει οδηγίες για τη χρήση της στην σελίδα:

https://www.waveshare.com/wiki/1.8inch_LCD_for_micro:bit

όπου και προτείνει τη χρήση της παρακάτω βιβλιοθήκης:

<https://github.com/waveshare/PXT-WSLCD1in8>

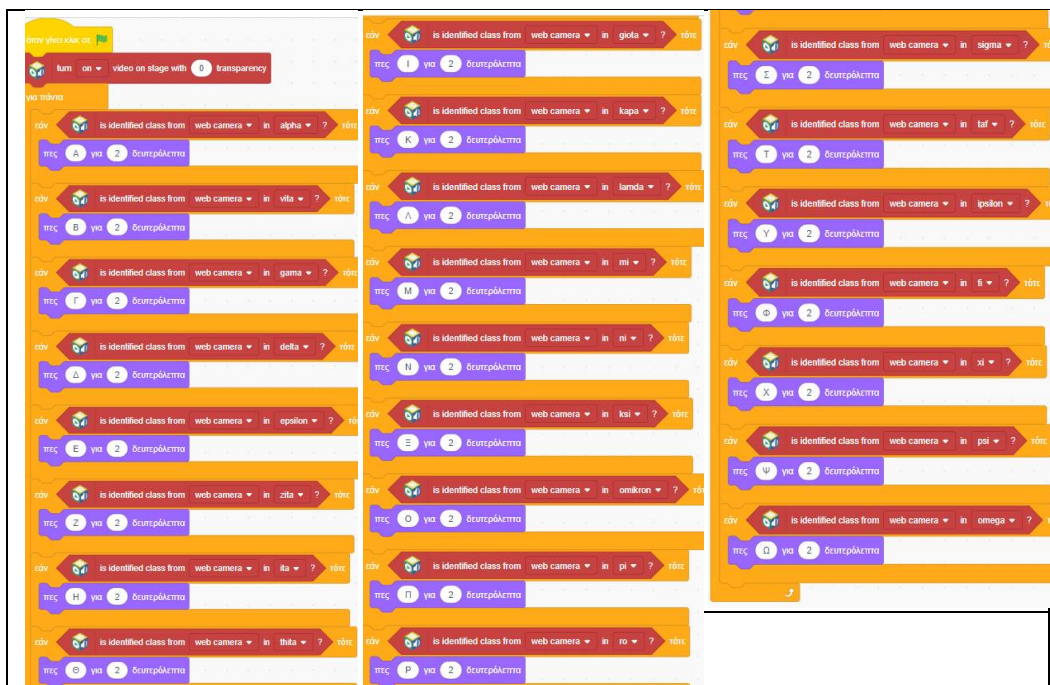
Ωστόσο δεν κατέστη δυνατή η λειτουργία της εν λόγω οθόνης παρόλο που ακολουθήθηκαν τα παραπάνω.

4. Υλοποίηση με το PictoBlox

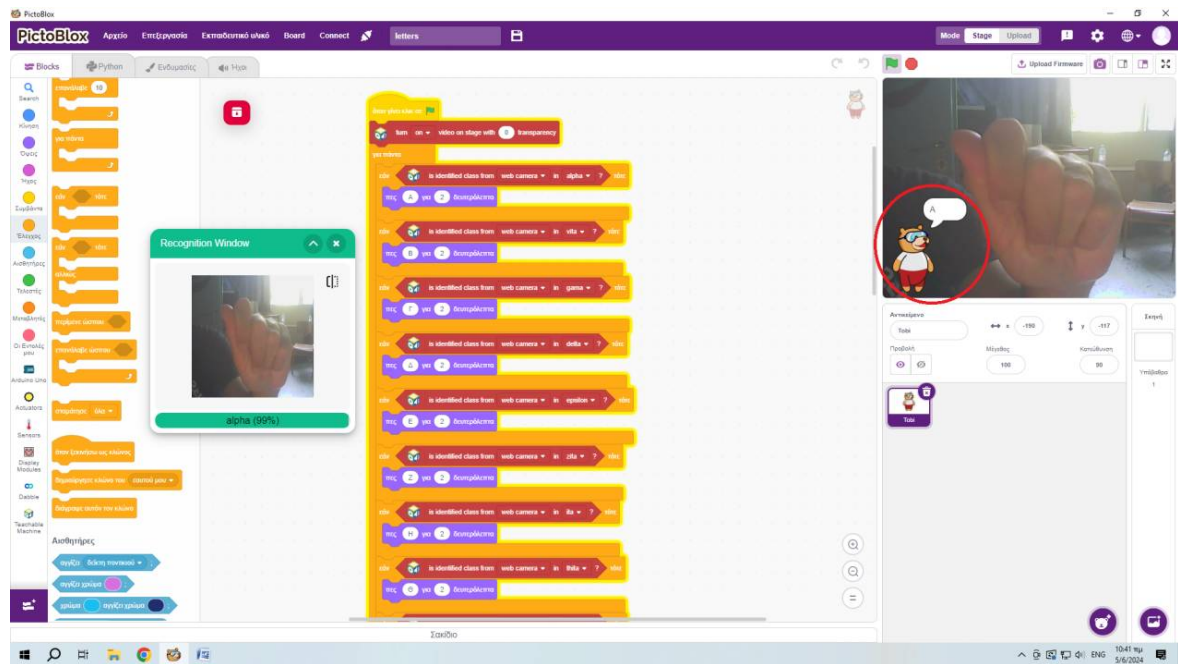
Για αυτή την υλοποίηση χρησιμοποιήθηκε η εφαρμογή **Pictoblox** (διαθέσιμο και on line στο <https://pictoblox.ai/>) η οποία είναι παρόμοια με το λογισμικό Scratch αλλά διαθέτει ένα μεγάλο αριθμό από επεκτάσεις για τη διασύνδεση με υλικό όπως το Micro::bit, το Arduino, τα Lego WeDo αλλά και για την υποστήριξη λειτουργιών που σχετίζονται με τεχνητή νοημοσύνη και μηχανική μάθηση.

Έτσι αφού έγινε εγγραφή στην πλατφόρμα (η οποία είναι απαραίτητη για τη χρήση των επεκτάσεων) προστέθηκε η επέκταση **ML with Teachable Machine** και στη συνέχεια εισήχθη το τελευταίο μοντέλο, το οποίο είχε και τα καλύτερα αποτελέσματα: <https://teachablemachine.withgoogle.com/models/YCYaXM0pl/>.

Το πρόγραμμα (κώδικας) που γράφτηκε φαίνεται στη συνέχεια:



Η λειτουργικότητα του παραπάνω προγράμματος είναι παρόμοια με αυτή του Micro::bit, με τη διαφορά ότι αντί να εμφανίζεται το γράμμα σε μια οθόνη το λέει ο χαρακτήρας της εφαρμογής (Tobi), όπως φαίνεται και στη συνέχεια:



Αποθετήριο στο Github

Η διεύθυνση του αποθετηρίου στο Github είναι η παρακάτω:

<https://github.com/DimSchool/6thellakopencontest>

Στο παραπάνω αποθετήριο βρίσκονται τα αρχεία που προέκυψαν από την εκπαίδευση του μοντέλου τεχνητής νοημοσύνης (tensorflow.js, tensorflow, tensorflow lite) καθώς και ο κώδικας για το Micro::bit και το Pictoblox.

Αποτελέσματα - Αποτίμηση - Συμπεράσματα

Τα αποτελέσματα του συνολικού εγχειρήματος ήταν ενθαρρυντικά, με το ποσοστό επιτυχούς αναγνώρισης να φτάνει στο 67% (αναγνώριση 16 γραμμάτων από τα 24), όμως απέχουν πολύ από το ιδανικό.

Η υλοποίηση του προτεινόμενου έργου αποδείχθηκε ιδιαίτερα απαιτητική και γεμάτη προκλήσεις. Ο ρόλος του background στο ποσοστό επιτυχούς αναγνώρισης φαίνεται ότι είναι κρίσιμος και η εύρεση του βέλτιστου background αποτελεί ένα δύσκολο πρόβλημα. Σε μελλοντικές υλοποιήσεις προτείνεται ο πειραματισμός πάνω σε αυτό το κομμάτι καθώς και ο πειραματισμός με άλλες παραμέτρους όπως ο φωτισμός του χώρου, η χρήση διαφορετικών καμερών, η χρήση σταθερού background κ.α.

Όσον αφορά το υλικό για την υλοποίηση του έργου σε μελλοντικές προσπάθειες προτείνεται το Arduino μαζί με κάποια οθόνη και σε συνδυασμό με το Pictoblox ή τη γλώσσα Python, κάτι το οποίο ήταν από τους αρχικούς σκοπούς του παρόντος έργου. Μια τέτοια υλοποίηση ωστόσο δεν έγινε λόγω έλλειψης χρόνου - εκτροχιασμού από το αρχικό χρονοδιάγραμμα.

Ως συμπέρασμα μπορούμε να πούμε ότι η υλοποίηση του έργου αποδείχθηκε συναρπαστική, παρόλο που ήταν γεμάτη με δυσκολίες και προκλήσεις, ενώ αποτέλεσε αφορμή για την απόκτηση νέων γνώσεων και εμπειριών γύρω από το αντικείμενο της τεχνητής νοημοσύνης και της μηχανικής μάθησης, καθώς και γύρω από το υλικό και το λογισμικό που σχετίζεται με αυτές.