

deAfault

by Vlad Constantinescu

Cuprins

- >Informatii Generale***
- >Ghid de instalare***
- >Arhitectura***
- >Testare***
- >Securitate***
- >Cerinte de sistem***
- >Dataset***
- >ML5 Api Reference***
- >p5.js Api Reference***
- >Node.js***
- >Machine Learning***
- >JavaScript vs TypeScript***

deAfault

by Vlad Constantinescu

Informatii generale:

deAfault este o aplicatie pentru telefon special dezvoltata pentru a usura invatarea limbajului semnelor (ASL) de catre persoanele **FARA** deficiente de auz in scopul integrarii mai usoare a persoanelor cu aceste deficiente.

Aplicatia se bazeaza pe premisa “punct ochit punct lovit”, astfel, utilizatorul isi poate alege campul lexical din care vrea sa asimileze, in locul unor lectiilor prestabilite.

deAfault prezinta mai multe metode atat de invatare cat si de evaluare/exersare precum:

- > categorii de invatare : utilizatorul alege campul lexical din care doreste sa invete urmand sa fie afisate diverse cuvinte (ilustrate in imagini cat si explicate)

- > videoclip-uri educative: sunt puse la dispozitia utilizatorului anumite videoclip-uri educative

- > quiz : cunostintele utilizatorului sunt testate (avand optiunea de testare generala sau dintr-o anumita arie de cuvinte)

- > jocuri interactive: prin intermediul camerei telefonului, sunt preluate imagini cu utilizatorul executand semne ce vor fi interpretate de joc. In momentul de fata sunt active 2 jocuri :

 - *"Color Game" : Joc in stilul aplicatiilor SnapChat/Boo/... ,
Aplicand anumite filtre in functie de semnele utilizatorului.

 - *"Guess the letter" : Joc in care trebuie aratata litera lipsa dintr-un cuvânt.

Ghid de instalare:

Aplicatia poate fi instalata ca orice alta aplicatie de telefon(apk pentru Android) si sper ca o sa se poata gasi curand si pe GooglePlay.

Arhitectura:

Proiectul deAfault este realizat/construit cu Apache Cordova.Astfel:

- >Pentru interfata am folosit HTML/CSS + Bootstrat/SweetAlert.
- >Pentru interactiunea cu utilizatorul am folosit JavaScript
- >Pe partea de ML am folosit libraria ml5.js
- >Recunoasterea faciala se bazeaza pe modelul PoseNet
- >Recunoasterea gesturilor se bazeaza pe modelul MobileNet.

De asemenea pentru design am folosit si libraria p5.js.

In versiunile anterioare, in locul libreriei ml5.js era folosit un model de Computer Vision ("hostat" pe Microsoft Azure).

->Pentru server am folosit Node.JS mentionand :

- Express pentru a administra requesturile
- Mysql2 pentru conectiunea cu baza de date
- JWT(JSON WEB TOKENS) pentru securitatea actiunilor ce tin de alterat continut din panel.

->Ca baza de date am folosit MariaDB

->Pentru panel-ul de administrare tehnologiile sunt asemanatoare cu cele folosite in aplicatia propriu-zisa.

Testare:

Aplicatia a fost testata de mai multi oameni dintre care :
prieteni/colegi de liceu/familie/...

Securitate:

Pentru partea de panou de administrare au fost luate in considerare urmatoarele :

- Pentru securitate in sesiune am folosit JWT(json web tokens)
- Query-urile pentru baza de date sunt parametrizate ca masura pentru SQL Injection
- Parola de administrator este criptata (BCrypt)
- Parola de administrator este criptata cu un caracter random inainte de a fi introdusa in baza de date ca masura impotriva atacurilor de tip Rainbow Table
- Serverul este rulat cu managerul de procese PM2 (asigurand load balancer/auto-run/...)

Cerinte de sistem:

- Conectiune la internet
- Telefon Android versiune minim 4.4 (sau versiune IOS suportata de cordova 5.0.0)
- Acces la camera telefonului.

DataSet :

Datorita faptului ca pe internet nu exista o baza de date cu imagini ale oamenilor ilustrand diverse semne in ASL, a trebuit sa o creez singur.

ML5 Api Reference :

General :

ml5.js este învățare automată pentru web din browserul dvs. web. Prin unele avansări inteligente și interesante, TensorFlow.js construiește oameni care au dat seama că este posibil să folosești browserul web integrat în unitatea de procesare grafică (GPU) pentru a face calcule care altfel ar rula foarte lent folosind unitatea centrală de procesare (CPU). O explicație cu adevărat frumoasă a ceea ce se întâmplă cu GPU-urile poate fi găsită aici - ml5 se străduiește să facă toate aceste noi dezvoltări în învățarea mașinii pe web mai accesibile tuturor.

ML5 ca open-source:

ML5 este un proiect open source care valorizează toate contribuțiile. Contribuțiile ml5 iau adesea forma de ateliere, contribuții de proiectare, ajutând la răspunsurile la întrebările

oamenilor despre Github, semnalizarea erorilor în cod, remedierea erorilor, adăugarea de noi funcții și multe altele.

Cum am invatat?

Ghidul începătorului de învățare automată cu ml5.js” al lui Dan Shiffman. Aici Dan explică ce este ml5.js și de unde provine totul.

P5JS Api Reference :

General:

p5.js este o bibliotecă JavaScript pentru codificare creativă, cu accent pe crearea accesibilă și incluzivă a codificării pentru artiști, designeri, educatori, începători și oricine altcineva! p5.js este gratuit și open-source, deoarece credem că software-ul și instrumentele de învățare a acestuia ar trebui să fie accesibile tuturor. Folosind metafora unei schițe, p5.js are un set complet de funcționalități de desen. Cu toate acestea, nu vă limitați la pânza de desen. Vă puteți gândi la întreaga pagină a browserului ca la schița dvs., inclusiv la obiectele HTML5 pentru text, intrare, video, cameră web și sunet.

Comunitate:

Comunitatea e formată din solidaritate cu persoane din fiecare identitate și expresie de gen, orientare sexuală, rasă, etnie, limbă, tip neuro, dimensiune, capacitate, clasă, religie, cultură, subcultură, opinie politică, vârstă, pricepere nivelul, ocupația și fundalul. Recunoaștem că nu toată lumea are timp, mijloace financiare sau capacitate de a participa activ, dar recunoaștem

și încurajăm implicarea de tot felul. Facilităm și încurajăm accesul și abilitarea. Cu toții suntem studenți. p5.js este o interpretare a procesării pentru web-ul de astăzi. Organizează evenimente și funcționăm cu sprijinul Fundației de procesare.

Node.js:

General:

Funcționează ca un timp de rulare JavaScript asincron, bazat pe evenimente, Node.js este proiectat pentru a construi aplicații de rețea scalabile. În următorul exemplu „lume de salut”, multe conexiuni pot fi gestionate concomitent. La fiecare conexiune, apelul este apelat, dar dacă nu există nicio lucrare de făcut, Node.js va dormi.

Design:

Node.js este similar în ceea ce privește designul și este influențat de sisteme precum Ruby Event Machine și Python's Twisted. Node.js duce modelul de eveniment ceva mai departe. Prezintă o buclă de eveniment ca o construcție de execuție în loc de bibliotecă. În alte sisteme, există întotdeauna un apel de blocare pentru a porni bucla evenimentului. De obicei, comportamentul este definit prin apeluri de apel la începutul unui script, iar la final un server este pornit printr-un apel de blocare precum EventMachine :: run (). În Node.js, nu există un astfel de apel start-the-event-loop. Node.js introduce pur și simplu bucla evenimentului după executarea scriptului de

intrare. Node.js iese din bucla evenimentului atunci când nu mai există apeluri pentru a efectua. Acest comportament este ca JavaScript browser-ului - bucla evenimentului este ascunsă de utilizator.

Pachete pentru node:

HTTP este un cetățean de primă clasă în Node.js, proiectat cu streaming și latență scăzută. Acest lucru face ca Node.js să se potrivească bine pentru crearea unei biblioteci sau cadru web. Node.js proiectat fără fire nu înseamnă că nu puteți profita de mai multe nuclee din mediul dvs. Procesele pentru copii pot fi generate prin utilizarea API-ului nostru `child_process.fork()` și sunt concepute pentru a fi ușor de comunicat cu. Construită pe aceeași interfață se află modulul `cluster`, care vă permite să partajați prize între procese pentru a permite echilibrarea încărcării pe nucleele dvs.

Comunitate:

Comitetul comunitar (CommComm) este un comitet de nivel superior al Fundației Node.js. CommComm are autoritate asupra eforturilor de divulgare a comunității care se confruntă spre exterior, inclusiv: Evangelizare comunitară Inițiative de educație Direcția culturală a fundației Node.js Obiectiv de organizare comunitară Traducere și internaționalizare Moderarea / Mediarea proiectului Publicitate și publicare.

Există patru tipuri de implicări cu Comitetul comunitar: Un contribuabil este orice persoană care creează sau comentează o problemă sau o solicitare de tragere. Un colaborator este un contribuabil căruia i s-a acordat acces la scriere la depozit Un observator este orice persoană care a solicitat sau a fost solicitată să participe la o ședință CommComm. De asemenea,

este primul pas pentru a deveni Membru. Un membru este un colaborator cu drepturi de vot, care a îndeplinit cerințele de participare și a votat prin procesul de vot al ComComm.

Versiuni:

Versiunile majore Node.js introduc starea de lansare curentă timp de șase luni, ceea ce le oferă autorilor bibliotecii timp pentru a adăuga suport pentru acestea. După șase luni, lansările cu numere impare (9, 11 etc.) devin neacceptate, iar versiunile cu număr par (10, 12 etc.) se mută la starea Active LTS și sunt gata pentru utilizare generală. Starea de eliberare LTS este „suport pe termen lung”, care garantează, de regulă, că bug-urile critice vor fi remediate în total 30 de luni. Aplicațiile de producție ar trebui să utilizeze versiuni Active LTS sau LTS de întreținere.

Machine Learning:

Modelul folosit:

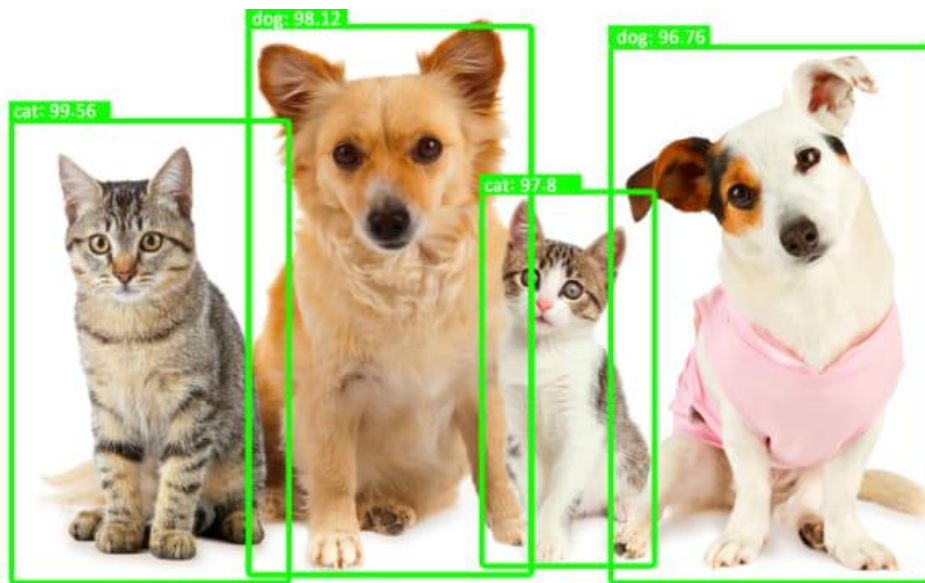
PoseNet este un model de învățare automată care permite estimarea în timp real a umane a poziției.



Primele Incercari:

Inainte de a avansa la modelul PoseNet, programul se baza pe o retea neuronală convolutională antrenată pe un calculator cu : placa video : NVIDIA GTX 1060 6GB, procesor : Intel core i7 8th gen.

Cu rezultate ok dar nu cele asteptate.



Host pe cloud:

Microsoft Azure este un serviciu de cloud computing creat de Microsoft pentru construirea, testarea, implementarea și gestionarea aplicațiilor și serviciilor prin centre de date gestionate de Microsoft. La baza sa, Azure este o platformă publică de cloud computing - cu soluții incluzând Infrastructure as a Service (IaaS), Platform as a Service (PaaS) și Software as a Service (SaaS) care poate fi utilizat pentru servicii precum analitice, virtuale calcul, stocare, rețea și multe altele.

De ce am renuntat la cloud computing?

In primul rand, avand modelul incarcat in cloud, user-ul era nevoit sa-l descarce pentru deschide aplicatia ceea ce ar crea o dependenta intre viteza de internet si performanta aplicatie, lucru nedorit.

In al doilea rand, Microsoft Azure nu e un serviciu gratuit, ceea ce inseamna ca proiectul nu poate fi sustinut pe aceasta platforma fara sa creeze o sursa de venit, lucru ce ar crea o contradictie, acesta fiind un proiect open-source deci gratuit.

De ce am renuntat la cloud computing?

In primul rand, avand modelul incarcat in cloud, user-ul era nevoit sa-l descarce pentru deschide aplicatia ceea ce ar crea o dependenta intre viteza de internet si performanta aplicatie, lucru nedorit.

In al doilea rand, Microsoft Azure nu e un serviciu gratuit, ceea ce inseamna ca proiectul nu poate fi sustinut pe aceasta platforma fara sa creeze o sursa de venit, lucru ce ar crea o contradictie, acesta fiind un proiect open-source deci gratuit.

JavaScript vs TypeScript:

Ce e TypeScript?

TypeScript este un limbaj de programare open source dezvoltat și menținut de Microsoft. Este un superset sintactic al limbajului JavaScript și asigură un sistem de tipuri opțional. TypeScript este proiectat pentru dezvoltarea de aplicații de mari dimensiuni și se compilează în JavaScript.

Originalar aplicatie cu Angular?

Originalar, deAfault a fost realizata in Angular, cu TypeScript .

Caracteristici JavaScript

Este un limbaj multiplă platformă Este folosit pentru partea clientului și pentru server Este ușor de învățat și de început Este un limbaj dinamic: flexibil și puternic Aveți „marea libertate” de a face orice doriți cu orice obiect Flux de lucru de testare puternică Dependente adăugate Cadru neacceptat

Ce e Angular?

Angular este o platformă de dezvoltare web cu sursă deschisă bazată pe limbajul TypeScript. Proiectul este dezvoltat de Echipa Angular de la Google și de o comunitate de utilizatori

individuali și companii. Angular este o rescriere completă, de către aceeași echipă, a frameworkului AngularJS.



De ce am renuntat?

Datorita unei dorinte personale, am renuntat la folosirea framework-ului Angular in favoarea unui framwork creat personal de mine.

La ce am renuntat?

Caracteristici ale Typescript
mentenabilitate
Ofertă de mare productivitate pentru dezvoltatori
Navigarea codului și prevenirea erorilor
Cod „descoperire” și refactorizare
Opțional Adnotare tip statică / Tipografie statică
Funcții suplimentare pentru funcții
Acceptă ES6 Acceptă interfețe, sub-interfețe, clase și subclase
Dezvoltare la nivel de client
HTML5 scalabilă
IDE bogat disponibil cu funcții de navigație automată și cod complet.
Orientat pe obiecte bazate pe clasă, cu moștenirea membrilor și interfețele private

Diferențe cheie

JavaScript este un limbaj de script care vă ajută să creați pagini web interactive, în timp ce Typescript este o supersetare a JavaScript. Codul typescript trebuie să fie compilat în timp ce codul JavaScript nu trebuie să compileze. Typescript acceptă o caracteristică de prototipare în timp ce JavaScript nu acceptă această caracteristică. Typescript utilizează concepte precum tipuri și interfețe pentru a descrie datele utilizate, în timp ce JavaScript nu are un astfel de concept. Typescript este un sistem de tip puternic, incluzând funcții generice și JS pentru proiecte de dimensiuni mari, în timp ce JavaScript este o opțiune ideală pentru proiecte de dimensiuni mici.

Deși se poate observa un anumit “regres”, renunțând la multe avantaje, folosind biblioteca p5.js, acestea nu au fost observabile în timpul dezvoltării

