

Teachable Machine

Echipă:

Clinici Rareș-Mihail
Ion Daniel
Matei Rareș-Andrei

Link GitHub:

<https://github.com/DaniPTRK/PythonProject>

Tehnologii folosite:

Python, Flask, SQLAlchemy, TensorFlow, HTML, CSS, GitHub

Descrierea aplicației:

O platforma web care oferă utilizatorului posibilitatea de a crea un teachable machine menit să distingă 2 clase de obiecte între mai multe poze. Platforma web are infrastructura de creare de conturi, zonă de creare a mașinilor și pentru încărcare a acestora. Un utilizator poate să acceseze orice mașină creată, meniu din care are opțiunea de a-o downloada sau de a o încerca pe alte imagini. Utilizatorul poate de asemenea să acceseze un meniu cu mașinile create de el, unde, pe lângă opțiunile prezentate mai sus, poate să le șteargă din baza de date.

Împărțirea muncii:

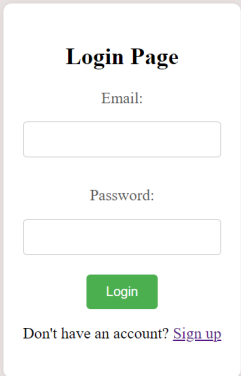
Clinici Rareș-Mihail: Flask, Bazele de date(account and machine management), puțin HTML

Ion Daniel: Partea de Machine Learning

Matei Rareș-Andrei: Flask, HTML, CSS

Flowul aplicației

Utilizatorul este rugat să se autentifice în cont și poate să-și creeze un cont în caz că nu are.



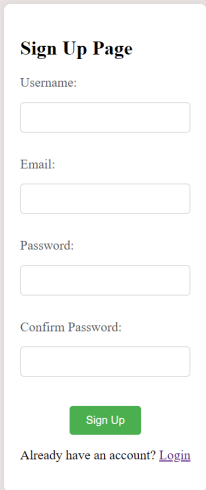
Login Page

Email:

Password:

Login

Don't have an account? [Sign up](#)



Sign Up Page

Username:

Email:

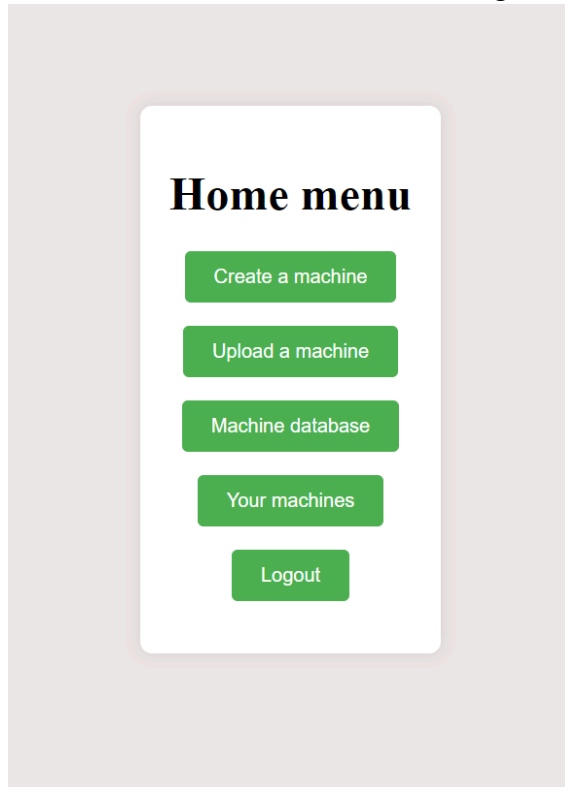
Password:

Confirm Password:

Sign Up

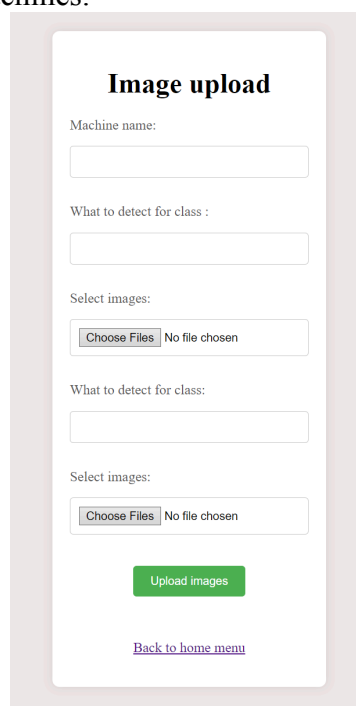
Already have an account? [Login](#)

După autentificarea în cont, utilizatorul este dus în Home Page.

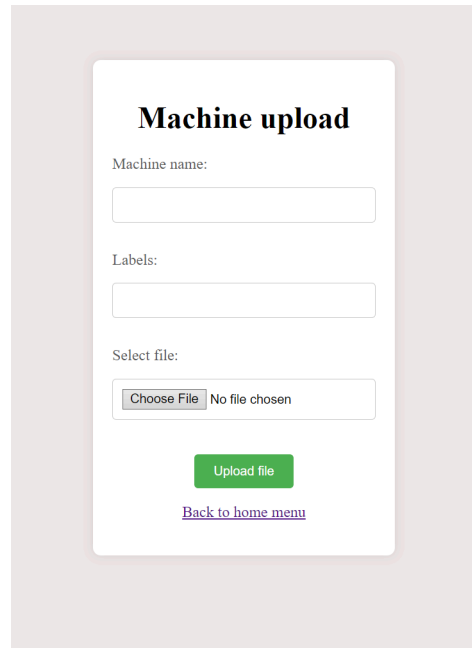


Aici utilizatorul poate alege să creeze o mașină nouă, să încarce în sistem o mașină din sistemul său, să vadă toate mașinile încărcate în sistem și mașinile încărcate de ei.

Meniul de încărcare mașină îți cere un nume pentru mașina și 2 clase dintre care se va detecta la testare. Pentru fiecare clasă se va da un nume și un set de poze de antrenament. Odată ce utilizatorul dă submit, se va genera mașina și va putea fi găsită în meniurile Machine Database și Your machines.

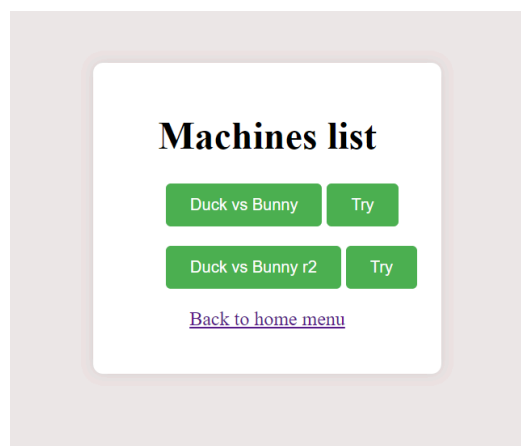
A screenshot of an 'Image upload' form. It has a white rounded rectangle on a light gray background. The title 'Image upload' is at the top in bold black font. Below it are two identical sections. Each section starts with 'Machine name:' followed by a text input field. Then 'What to detect for class :' followed by another text input field. Then 'Select images:' followed by a button labeled 'Choose Files' and the text 'No file chosen'. Below each section is 'What to detect for class:' followed by a text input field, and 'Select images:' followed by another 'Choose Files' button and 'No file chosen' text. At the bottom of the form is a green 'Upload images' button and a purple link 'Back to home menu'.

Meniul Upload machine îi permite utilizatorului să încarce o mașină(un fișier de tip .h5) cu un anumit nume și etichete pentru clase. Odată încărcată, va fi disponibilă în Machine Database pentru orice utilizator.



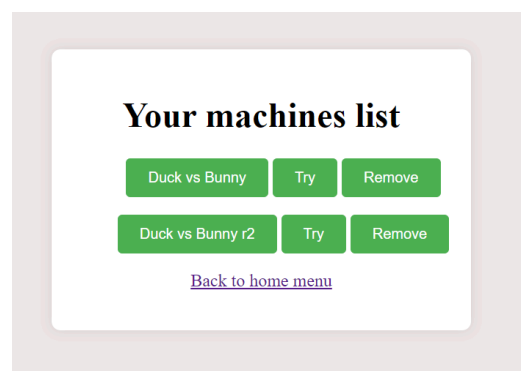
The image shows a web form titled "Machine upload". It contains three input fields: "Machine name:", "Labels:", and "Select file:". The "Select file:" field has a "Choose File" button and the text "No file chosen". Below the input fields is a green "Upload file" button and a link "[Back to home menu](#)".

În meniul Machine Database sunt toate mașinile încărcate de utilizatori. Oricine poate downloada o mașina apăsând pe butonul cu numele acesteia sau să o încerce apăsând pe Try.



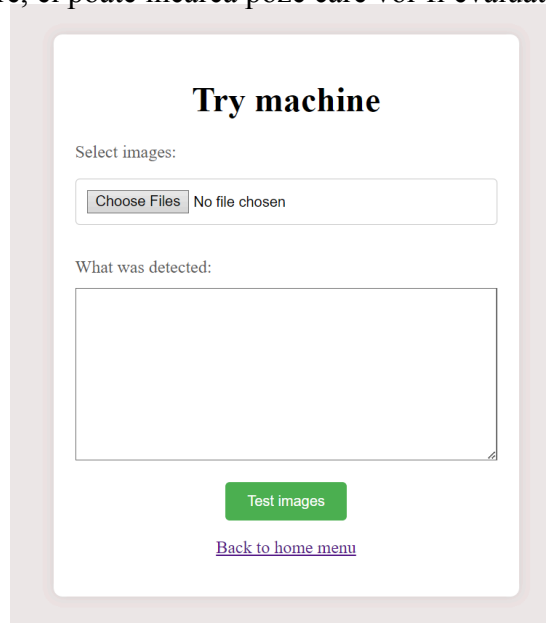
The image shows a web page titled "Machines list". It displays two rows of machine names with corresponding "Try" buttons: "Duck vs Bunny" and "Duck vs Bunny r2". Below the list is a link "[Back to home menu](#)".

Fereastra Your machines prezintă doar mașinile utilizatorului conectat. Pe lângă opțiunile din Machine Database, utilizatorul poate șterge mașini din sistem.



The image shows a web page titled "Your machines list". It displays two rows of machine names with corresponding "Try" and "Remove" buttons: "Duck vs Bunny" and "Duck vs Bunny r2". Below the list is a link "[Back to home menu](#)".

Din ferestrele Machine Database și Your machines, utilizatorul poate încerca orice mașină. În meniul pentru încercare, el poate încărca poze care vor fi evaluate.

The screenshot shows a web form titled "Try machine". It has a section "Select images:" with a file upload button labeled "Choose Files" and the text "No file chosen". Below this is a section "What was detected:" with a large empty rectangular box. At the bottom of the form is a green button labeled "Test images" and a link labeled "Back to home menu".

Detalii de implementare:

- în timpul sesiunii unei aplicații, utilizatorul este păstrat logat
- parola utilizatorului este ținută ca hash în baza de date pentru securitate
- pentru fiecare mașină uploadată, se generează un nume unic
- între utilizatori și mașini există o relație de one-to-many pentru a memora cine a creat o mașină
- mașina folosește modelul Sequential pentru a se antrena

Dificultăți în implementare:

- implementarea flaskului a început prima, cel care implementa lucra prima oară cu flask și nu avea încă html-ul sau mașina cu care să lucreze
- probleme cu merge-uri la github
- transmiterea pozelor ca argumente a unui request de la o pagina la alta pentru care s-a găsit un work around(unirea celor 2 pagini)
- a fost o situație după aplicarea claselor css în care butonul de Log In trimitea un GET request în loc de POST
- transmiterea datelor către zona de antrenare a mașinii și obținerea/prelucrarea datelor primite
- implementarea efectivă a mașinii

Cum rulăm proiectul?

flask --app app.py run