SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU

FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I

INFORMACIJSKIH TEHNOLOGIJA

Sveučilišni studij

KLASIFIKACIJA RIBA

Projektni zadatak

Ivan Smoljo

Stefan Uglješić

Leon Vučko

Osijek, 2023. godina.

##### SADRŽAJ

[1. UVOD 1](#_Toc138711576)

[1.1. Cilj projektnog zadatka 1](#_Toc138711577)

[2. OPIS KORIŠTENIH TEHNOLOGIJA I PODATAKA 2](#_Toc138711578)

[2.1. Microsoft Custom Vision AI 2](#_Toc138711579)

[2.2. Korišteni skup slika 2](#_Toc138711580)

[2.3. REST API 3](#_Toc138711581)

[3. TRENIRANJE MODELA 4](#_Toc138711582)

[3.1. Kreiranje projekta i modela 4](#_Toc138711583)

[3.2. Testiranje modela 6](#_Toc138711584)

[3.3. Kreiranje API-ja 6](#_Toc138711585)

[4. KLIJENTSKE APLIKACIJE 7](#_Toc138711586)

[4.1. Windows aplikacija 7](#_Toc138711587)

[4.2. Android aplikacija 8](#_Toc138711588)

[ZAKLJUČAK 10](#_Toc138711589)

# UVOD

Korištenjem strojnog učenja i dubokog učenja je moguće kreirati razne sustave i modele koji automatski prepoznaju objekte sa slike, koji se zatim mogu koristi za klasifikaciju raznih stvari.

Kroz ovaj rad se opisuje postupak na koji se način može kreirati model koji služi za raspoznavanje i klasifikaciju kao i pripadne aplikacije koje implementiraju RESTful API za komunikaciju sa serverom na kojemu se taj model nalazi.

Prvi korak u razvoju modela klasifikacije riba, ali i općenito pri razvoju modela, je prikupljanje i obrada podataka. Za tu svrhu se koriste već postojeće baze podataka ili datasetovi koji sadrže slike različitih riba koji su već klasificirani. .

Nakon što su podaci prikupljeni, korištenjem Microsoft Azure Custom Vision AI alata se trenira model koji klasificira ribe u nekoliko kategorija.

Kada je dobiven zadovoljavajući model, kreirana je Windows desktop i Android aplikacija koja koristi RESTful API kako bi klasificirala ribu na slici.

Rad je podijeljen na 5 osnovnih poglavlja: uvod, opis korištenih tehnologija i podataka, treniranje modela, klijentske aplikacije i zaključak. U prvom poglavlju je dan uvod u temu obrađenu u sklopu ovog projektnog zadatka. U drugom poglavlju objašnjene su tehnologije korištene u ovome radu, kao i skup slika za treniranje modela. U trećem poglavlju je opisan postupak obrade podataka kao i kreiranje i treniranje modela. U četvrtom poglavlju je dan uvid u kreiranje Windows Form aplikacije i Android aplikacije koja koristi kameru i prethodno trenirani model za klasifikaciju. Posljednje poglavlje predstavlja zaključak u kojem se komentiraju ovaj projekt i daju ideje za unaprjeđenje.

## Cilj projektnog zadatka

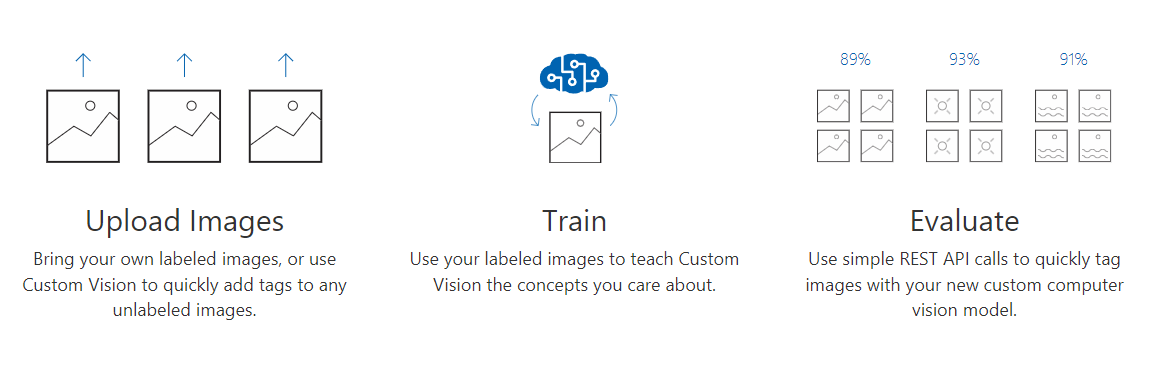
Napraviti i istrenirati model za klasifikaciju riba korištenjem Microsoft Azure Custom Vision AI alata, kreirati i implementirati API kroz klijentsku Windows desktop aplikaciju, kao i kroz Android mobilnu aplikaciju.

# OPIS KORIŠTENIH TEHNOLOGIJA I PODATAKA

Strojno učenje je područje umjetne inteligencije koje se bavi razvojem algoritama i modela koji omogućuju računalima da "uče" iz podataka i donose predikcije ili donose odluke bez izričitog programiranja. Umjesto da se programiraju specifične upute za obavljanje zadataka, strojno učenje se temelji na analizi podataka i automatskom izvlačenju uzoraka i znanja iz tih podataka [1].

## Microsoft Custom Vision AI

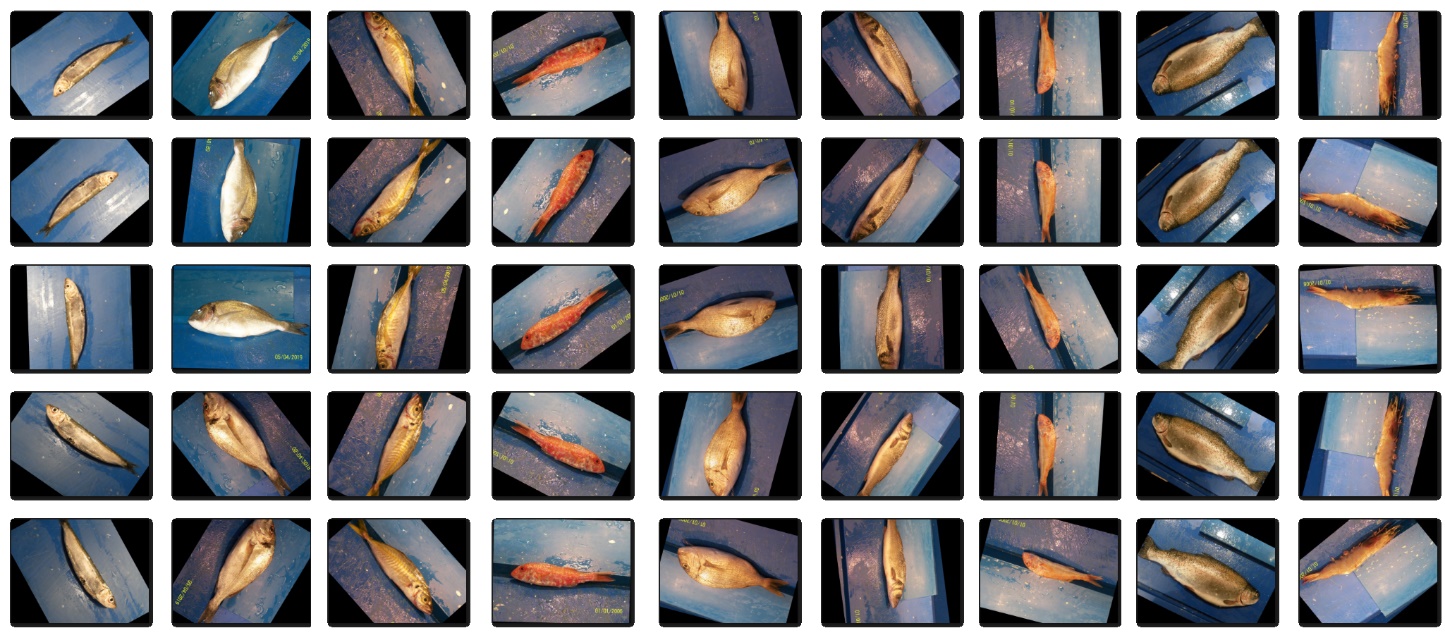
Microsoft Custom Vision AI je platforma temeljena na umjetnoj inteligenciji koja omogućava jednostavno stvaranje i treniranje vlastitih modela za prepoznavanje i klasifikaciju objekata na slikama [2]. Kombinirajući duboko učenje i napredne algoritme za obradu slika, Custom Vision AI pruža korisnicima mogućnost da stvore visoko precizne modele prepoznavanja slika specifičnih za njihove potrebe. Jednom istrenirani modeli se zatim mogu koristiti korištenjem REST API ili preuzeti kao model za lokalno korištenje. Princip rada Custom Vision AI je dan slikom 2.1.



**Sl. 2.1.** Princip rada Microsoft Custom Vision AI [2].

## Korišteni skup slika

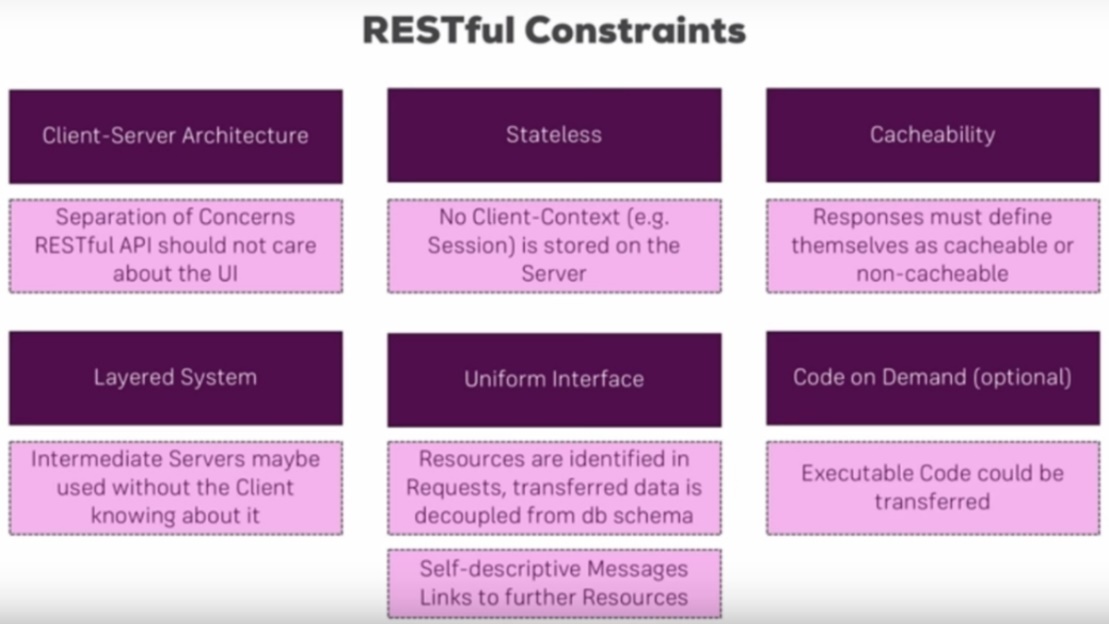
Za treniranje se koristi dataset *A Large Scale Fish Dataset* koji sadrži 9 različitih klasa riba od koje svaka sadrži 1000 slika, a slike su nastale slikanjem riba u supermarketu u Turskoj [3]. Dataset je dostupan za preuzimanje na poveznici: <https://www.kaggle.com/datasets/crowww/a-large-scale-fish-dataset>. Na slici 2.2. su vidljive primjeri slika iz tog dataseta.



**Sl. 2.2.** Primjer slika iz Clothing Dataseta.

## REST API

REST API (Representational State Transfer Application Programming Interface) je arhitektura i skup pravila za komunikaciju između računalnih sustava putem mreže, najčešće putem HTTP protokola [4]. REST API omogućuje razmjenu podataka između klijenta (npr. desktop aplikacije, web aplikacije, mobilne aplikacije) i poslužitelja (*server*) na način koji je jednostavan, skalabilan i neovisan o platformi. RESTful API posebno koristi HTTP metodologije specificirane u protokolu RFC 2616, a koristi se GET za dohvat resursa; PUT za izmjenu stanja ili ažuriranje resursa koji može biti objekt, datoteka ili blok; POST za stvaranje tog resursa i DELETE za brisanje tog resursa [5]. Kako bi API mogao biti REST, mora poštovati pravila na slici 2.3.



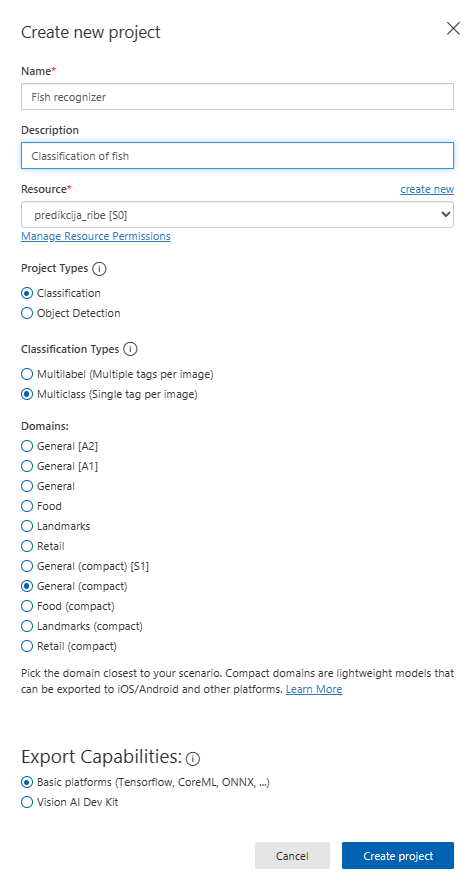
**Sl. 2.3.** RESTful načela [5].

# TRENIRANJE MODELA

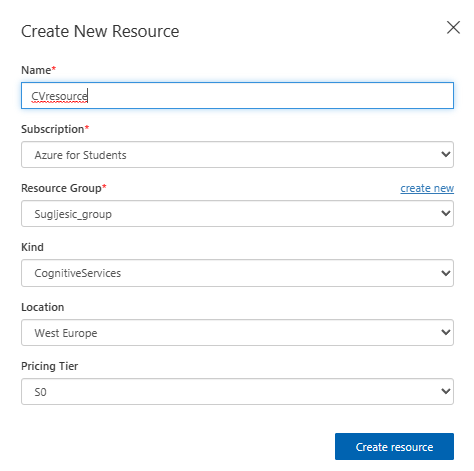
Kako bi se model mogao trenirati, potrebno je posjedovati Microsoft račun. Studenti imaju pravo na 100 besplatnih modela.

## Kreiranje projekta i modela

Nakon što smo se prijavili na Microsoft Azure Custom Vision AI, klikom na New Project otvara se izbornik kao na slici 3.1. gdje se unosi naziv našeg projekta, kratki opis, odabire se Resurs, koji ukoliko ne postoji, potrebno je kreirati novi kao na slici 3.2. Zatim se odabire tip projekta odabire klasifikacija, te se za tip klasifikacije odabire *multiclass* kako svaka riba pripada samo jednoj klasi. Zatim odabiremo domenu koja najviše odgovara našim potrebama, ukoliko želimo preuzeti modele, za obradu u drugim alatima, npr. *Python* i *Tensorflow*, potrebno je odabrati model koji je kompaktan (*compact*). Zatim ukoliko je odabran kompaktan model možemo odabrati na koje platforme želimo izvesti model.

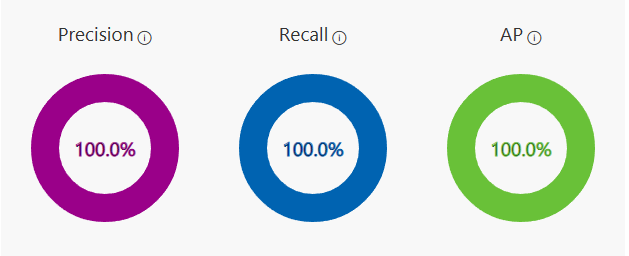


**Sl. 3.1.** Kreiranje novog modela.



**Sl. 3.2.** Kreiranje novog resursa.

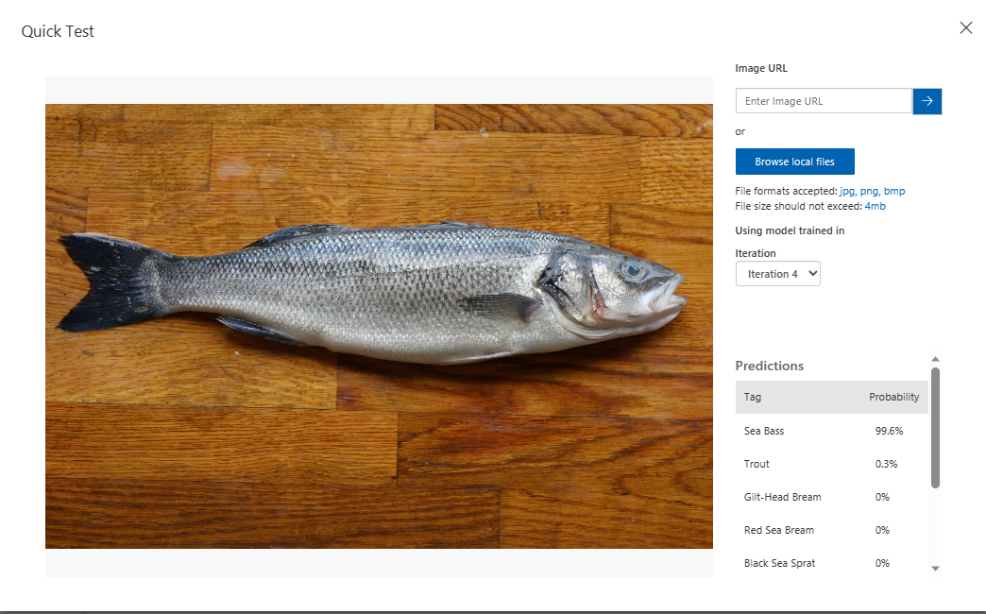
Nakon što je projekt kreiran, potrebno je dodati slike klikom na *Add Images*, otvara se izbornik te odabiremo slike. Nakon što smo odabrali slike potrebno im je dati oznaku (*tag*), u ovome slučaju to bi bila vrsta ribe koju dodajemo. Nakon što smo dodali sve potrebne slike i odabrali njihove tagove, možemo trenirati model klikom na gumb *Train*, otvara se izbornik gdje odabiremo želimo li brzo treniranje ili napredno treniranje. Klikom na *Advanced Training* otvara se izbornik gdje možemo odabrati koliko želimo dugo trenirati model od jednog sata do 96 sati. Pravilo vrijedi da je za složenije podatke bolje dati više vremena za treniranje, no valja napomenuti kako se naplaćuje vrijeme za treniranje. Ipak, ukoliko trening završi ranije, biti će naplaćeno samo onoliko koliko se model trenirao. Također, postoji mogućnost obavještavanja korisnika preko e-maila kada je trening gotov. Za potrebe ovoga projekta, odabire se *Quick Training*. Nakon što je trening gotov, možemo vidjeti njegove performanse kao na slici 3.3. *Precision* predstavlja točnost, tj. ako model predvidi oznaku, kolika je vjerojatnost da je ona ispravna. *Recall* govori od oznaka koje bi trebale biti ispravno predviđene, koliki postotak je vaš model ispravno pronašao? AP je mjera performansi ovoga modela, uzima u obzir *Precision* i *Recall* u različitim granicama.



**Sl. 3.3.** Performanse modela

## Testiranje modela

Model se može testirati klikom na tipku *Quick Test*. Tada se otvara izbornik, te se može odabrati datoteka s URL-a ili lokalna datoteka koja mora biti formata jpg, png ili bmp. Također, odabire se i iteracija modela ukoliko smo vršili više iteracija. Zatim se nakon nekoliko sekundi pojavljuje rezultat u kojemu vidimo vjerojatnosti. Izgled izbornika i predviđene klase je vidljiv na slici 3.4.

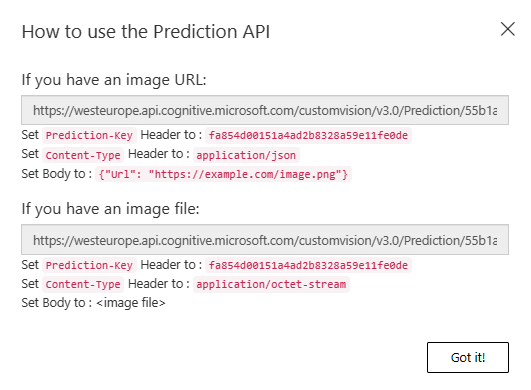


**Sl. 3.4.** Brzo testiranje modela (*Quick Test*).

U izborniku *Predictions* možemo vidjeti sve slike koje su testirane na modelu, kao i one koje su klasificirane preko API poziva.

## Kreiranje API-ja

Kako bi se kreirao API, potrebno je unutar izbornika *Performance* kliknuti na tipku *Publish* za iteraciju modela koju želimo koristiti. Nakon toga klikom na tipku *Prediction API* se otvaraju upute za korištenje API-ja kao na slici 3.4.



**Sl. 3.5.** Upute za Prediction API

# KLIJENTSKE APLIKACIJE

Za klijentske aplikacije su odabrane platforme Windows i Android zbog dostupnosti besplatnih i dostupnih alata za kreiranje aplikacija.

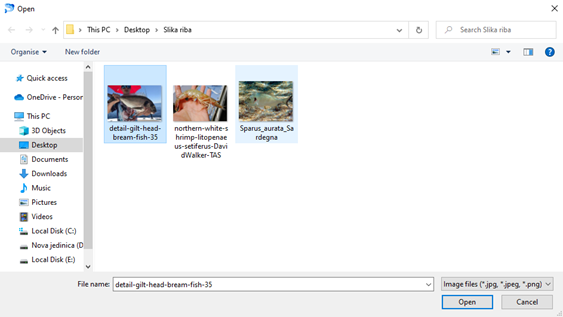
## Windows aplikacija

Izrađena aplikacija na slici 4.1. je C# Windows Forms aplikacija koja koristi Azure Custom Vision uslugu za prepoznavanje slika. Glavna svrha aplikacije je da korisnik može odabrati sliku putem korisničkog sučelja, poslati je na Custom Vision API za analizu i dobiti rezultate prepoznavanja naziva oznaka i vjerojatnosti za svaku oznaku.



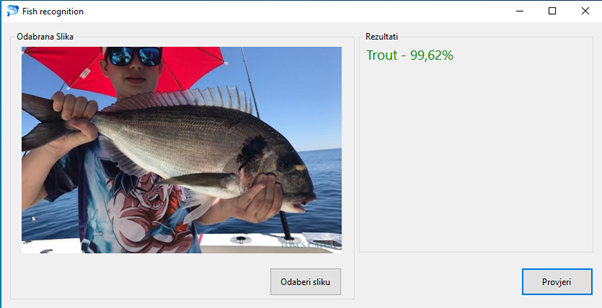
**Sl. 4.1.** Izgled *Fish recognition* aplikacije.

Klikom na tipku Odaberi sliku korisniku se nudi mogućnost odabira slike kao na slici 4.2., pri čemu su definirani prihvatljivi formati slike. Nakon što je slika odabrana, ona se prikazuje na mjestu crteža ribe i gumb Provjeri postaje aktivan.



**Sl. 4.2.** Odabir željene slike.

Pritiskom na tipku Provjeri aplikacija šalje HTTP zahtjev na Custom Vision API s odgovarajućim URL-om i binarnim podacima slike. U zahtjevu se također šalju potrebni zaglavlja, kao što su *Prediction-Key* za autentifikaciju i *Content-Type* za označavanje vrste podataka. Nakon slanja zahtjeva, aplikacija prima odgovor u JSON formatu koji sadrži rezultate prepoznavanja. Koristeći *Newtonsoft.Json* biblioteku, JSON rezultat se parsira i izvlače se informacije o oznakama i njihovim vjerojatnostima. Za svaku oznaku i vjerojatnost, aplikacija dinamički stvara novu *Label* kontrolu koja se dodaje u *GroupBox* kontejner na korisničkom sučelju. Labeli prikazuju nazive oznaka i vjerojatnosti u formatu "Naziv oznake - Vjerojatnost" kao što je to vidljivo na slici 4.3.



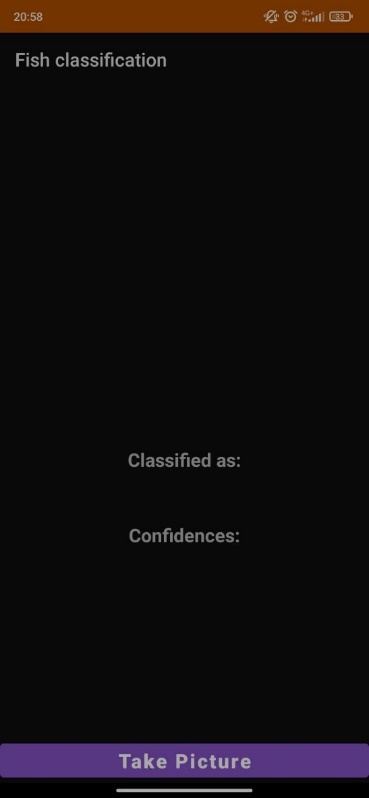
**Sl. 4.3.** Prikaz dobivenih rezultata za odabranu sliku.

Nakon prikaza rezultata, aplikacija omogućuje korisniku da odabere novu sliku i ponovi proces ili da zatvori aplikaciju.

Ova aplikacija omogućuje korisnicima da jednostavno koriste Azure Custom Vision uslugu za prepoznavanje slika i dobiju informacije o oznakama i njihovim vjerojatnostima na temelju analize slike.

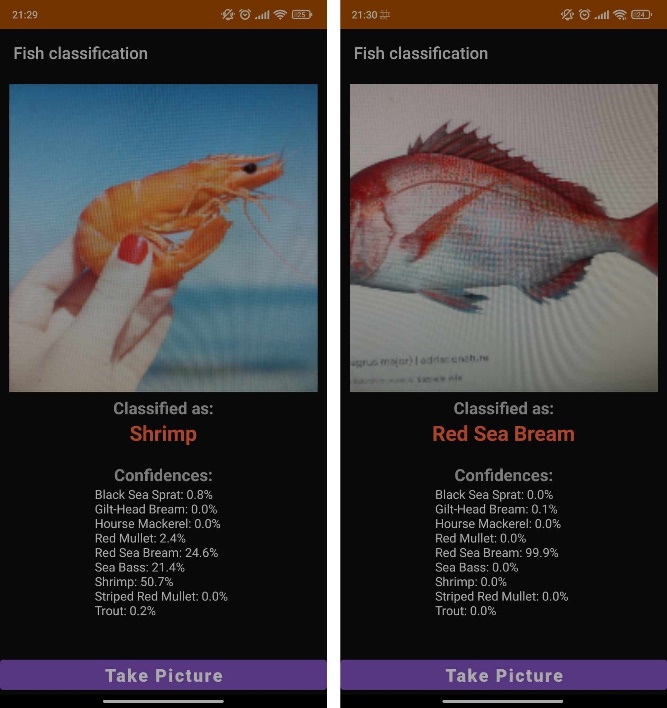
## Android aplikacija

Android aplikacija čiji dizajn je prikazan na slici 4.4. je kreirana u Android Studio-u. Ona također služi za prepoznavanje riba, ali na način da koristi kameru umjesto izbora slike koja se želi poslati.



**Sl. 4.4.** Dizajn Android aplikacije

Aplikacija radi tako da se kamerom uslika slika, zatim se ta slika konvertira u .jpg oblik koji je moguće poslati korištenjem API-ja. Nakon slanja zahtjeva, aplikacija prima odgovor u JSON formatu koji sadrži rezultate prepoznavanja. Rezultat sa najvećom vjerojatnošću se prikazuje u gornjem *textbox*-u ispod teksta „*Classified as:*“ dok se vjerojatnosti prikazuju u *textbox*-u ispod „*Confidences:*“. Primjer kako radi Android aplikacija je na slici 4.5.



**Sl. 4.5.** Izgled Android aplikacije

# ZAKLJUČAK

Kroz ovaj projekt smo se upoznali s korištenjem Microsoft Azure Custom Vision AI alata za klasifikaciju slika. Prvo je bilo potrebno prikupiti slike riba i postaviti ih na Custom Vision AI uz pripadajuće oznake klasa. Zatim se trenira model i nakon što smo zadovoljni sa rezultatima publisha. Nakon toga smo dobili potreban API za komunikaciju sa serverom kako bi smo mu poslali slike koje bi on zatim trebao klasificirati. Osim toga, kreirali smo Windows i Android aplikacije koje se oslanjaju na RESTful API za komunikaciju preko interneta. U Windows aplikaciji korisnik odabire željenu sliku koju program zatim predaje Custom Vision-u, te kao informaciju dobiva postotak kojoj klasi ta riba pripada. Dok je za mobilna aplikaciju realizirana na način da koristi kameru kako bi se uslikala željena slika, te se onda ta slika koristi za klasifikaciju ribe. Kako se sve slike poslane na Custom Vision AI spremaju, one koje su krivo kategorizirane se mogu ručno ispraviti, te se mogu koristiti za daljnje unaprjeđivanje ovog modela za klasifikaciju riba.

#### LITERATURA

1. Machine learning https://www.ibm.com/topics/machine-learning
2. Custom Vision Service, https://learn.microsoft.com/en-us/azure/cognitive-services/custom-vision-service/
3. Ulucan, Oguzhan and Karakaya, Diclehan and Turkan, Mehmet - A Large-Scale Dataset for Fish Segmentation and Classification – IEEE 2020.
4. REST API - https://www.ibm.com/topics/rest-apis
5. What is REST API - https://www.tutorialswebsite.com/what-is-a-restful-api/