



# Projekat Catapult

Kviz & Katalog znanja o mačkama

## Funkcionalni zahtevi

- **Kreiranje lokalnog naloga**

Prilikom prvog starta, da bi se aplikacija koristila, mora da se napravi lokalni nalog (profil) korisnika. Ukoliko nalog već postoji, prilikom svakog sledećeg starta preskače se kreiranje naloga i otvara se direktno glavni ekran. Obavezna polja detalja naloga su:

- Ime i prezime
- Nickname (bez razmaka, samo slova, brojevi i underscore \_)
- Email (validna email adresa)

- **Katalog informacija o rasama mačaka**

- **Lista rasa** sa opcijom pretraživanja rase.
  - a. ime rase,
  - b. opis rase (maksimalno 250 karaktera opisa; skratiti ukoliko je opis duži),
  - c. najviše tri osobine temperamenta (za rase koje imaju više osobina, nasumično izabrati tri). (predlog: možete koristiti chips)  
Alternativna imena rase ukoliko postoje su opcioni.
- **Detalji** rase mačke koji se otvaraju klikom na rasu mačke sa liste rasa.
  - a. minimalno jednu sliku rase,
  - b. ime rase,
  - c. kompletan opis rase,
  - d. spisak zemalja porekla,
  - e. sve osobine temperamenta,
  - f. životni vek,
  - g. prosečna težina i/ili visina rase,
  - h. dugme koje otvara stranicu o rasi na Vikipediji u browseru

- **Galerija** fotografija odabrane rase mačke u vidu grida fotografija. Otvara se iz detalja rase i prikazuje u zasebnom ekranu.
- **Photo viewer fotografija.** Prikazuje fotografiju preko celog ekrana, treba da se swipe-uje levo-desno sa pager komponentom. Otvara se iz galerije fotografija (grida) kada se klikne na fotografiju.
- **Kviz znanja o mačkama**
  - Kviz ima ukupno 20 pitanja. Pitanja se prikazuju pojedinačno, jedno po jedno, preko celog ekrana radi boljeg korisničkog iskustva.
  - Pitanja se generišu nasumično na osnovu tipa pitanja i podataka dostupnih sa CatsApi. Kviz ima tri glavne kategorije sa po nekoliko tipova pitanja. Više detalja se nalazi u sekciji [Dokumentacija Kviza](#).
  - Kviz se pokreće po kategorijama i bodovanje zavisi po kategorijama. U individualnoj izradi autor po svom izboru bira minimum jednu kategoriju za implementaciju.
  - Ukoliko aplikacija podržava više kategorija prilikom starta kviza, prvo treba izabrati kategoriju i tek onda početi, dok u slučaju samo jedne kategorije kviz se startuje bez biranja kategorije.
  - Aplikacija pamti rezultat svakog pitanja na osnovu formule bodovanja. Nakon odigranog kviza prikazuje se konačni rezultat. [Formula bodovanja](#) je data u tehničkim zahtevima.
  - Igranje kviza je vremenski ograničeno na 5 minuta. Po isteku vremena, korisnik se automatski prebacuje na stranu rezultata i pitanja na koja nije stigao da odgovori boduju se sa 0 poena. Više informacija u [sekciji za bodovanja](#).
  - Na strani rezultata kviza, korisnik treba da ima opciju da objavi (podeli) rezultat na globalnoj leaderboard listi za igranu kategoriju kviza koristeći Leaderboard api.
  - Korisnik treba da ima mogućnost da prekine kviz na bilo kom pitanju i u tom slučaju se rezultat ne računa. Prekidanje kviza treba da prikaže potvrdni dialog, jer je destruktivna operacija.
  - Nije dozvoljeno vraćati se na prethodna pitanja i menjati odgovore. Jednom dati odgovor na pitanje je konačan.
  - Klik na Back dugme u toku kviza treba da se onemogući ili da se ponaša na isti način kao i funkcionlanost otkazivanja kviza (prikazivanje potvrdnog dialog za prekid kviza). Up dugme ne treba da postoji na ekranu kviza.

- **Leaderboard ekran**

Ekran koji prikazuje sve prethodne rezultate objavljene na globalnoj leaderboard listi sortirani od najboljeg do najlošijeg rezultata. Rezultati se dobijaju putem Leaderboard apija. Svaka stavka liste treba da sadrži:

- globalni redni broj,
- nickname igrača,
- rezultat igre,
- ukupan broj odigranih kvizova tog igrača (javni tj. objavljeni).

Bitno je da može da se vidi rezultat po kategoriji bez rezultata drugih kategorija.

Treba prikazati sve rezultate u leaderboard listi (ne samo najbolje rezultate).

- **Detalji naloga**

Profil/Account ekran koji prikazuje informacije trenutnog naloga.

- Sadrži sve detalje (iz zahteva #1) lokalnog naloga.
- Ima opciju da otvori ekran za editovanje profila.
- Sadrži hronološku istoriju **svih** rezultata **svih** kvizova datog korisnika (uključujući i objavljene i neobjavljene rezultate).
- Sadrži informacije o najboljem rezultatu korisnika (takođe uključujući sve odigrane kvizove, objavljene i neobjavljene rezultate).
- Sadrži najbolju poziciju lokalnog naloga/profila na globalnoj leaderboard liste.

- **Izmena detalja naloga**

Ekran na kome korisnik može da promeni svoje podatke (iz zahteva #1) i da ih sačuva lokalno. Pristupa se iz detalja naloga. Neophodno je da postoji validacija polja kao i prilikom kreiranja naloga. Nije dozvoljeno čuvati nevalidne informacije.

## Tehnički zahtevi

### Arhitektura aplikacije

Obavezno je koristiti sledeće tehnologije (identični uslovi kao sa kolokvijumskog projekta):

Kotlin, Coroutines, Flow, Compose, MVI arhitektura, Jetpack Navigation, Retrofit, OkHttp, KotlinX Serialization.

Dodatni obavezni uslovi za ispitni projekat:

- Jetpack Room za rad sa bazom podataka;
- Jetpack DataStore za skladištenje jednostavnih struktura (možda informacije o lokalnom nalogu i slično);
- Hilt DI za upravljanje dependency-ijama.

### Lokalno čuvanje podataka

Kako bismo podigli korisnički doživljaj na većem nivou i izbegli nepotrebna čekanja odgovora sa api poziva u toku kviza, treba implementirati lokalno keširanje podataka u lokalnu bazu podataka.

Lokalna baza podataka je **single source of truth**, što znači da je to mesto odakle se čitaju podaci za prikazivanje na svim ekranima aplikacije, uključujući i kviz.

Direktno čitanje podataka sa api servisa i prikazivanje na ekran nije dozvoljeno za podatke o mačkama.

Leaderboard podaci se mogu direktno čitati i prikazivati sa api poziva tj. nije neophodno keširanje ovih podataka u lokalnu bazu podataka. Međutim, potrebno je lokalno pamtit i sopostvene rezultate kako bismo mogli da ih prikažemo u profil ekranu (jer tamo prikazuje sve objavljene i neobjavljene rezultate).

Autorima se ostavlja:

- sloboda u kreiranju database scheme lokalne baze podataka; i
- sloboda u implementaciji sinhronizacije podataka sa api servisa u bazu podataka;

## Tehnički zahtevi kviza

### Randomizacija & Dinamičnost

Randomizacija pitanja i odgovora kviza je obavezna kako bismo kviz učinili zanimljivijim i izazovnijim. Svako pitanje sa svojim potencijalnim odgovorima se dinamički kreira na osnovu tipa pitanja i podataka sa Cats apija koje smo sačuvali u lokalnu bazu podataka.

Na primer, da bismo kreirali pitanje “Koja je rasa mačke?” imamo nekoliko koraka:

- nasumično biramo neku rasu mačku; onda
- nasumično biramo fotografiju mačke, nasumično odabir fotografije je takođe vrlo bitan kako bismo povećali broj potencijalnih pitanja; zatim
- nasumično biramo 3 netačna odgovora (tačan odgovor već imamo).

Ovo pravilo randomizacija i dinamičnost važi za sve kategorije i tipova pitanja. Predlog implementacije je da definiše entitete baze podataka tako da nam bude lakše da generišemo pitanja i odgovora.

Dodatno pojašnjenje:

- Nije dozvoljeno da se slike mačaka ponavljaju u toku kviza.
- Generisanje pitanja može da se uradi na početku kviza (generisanje kviza), ili da se pitanju generišu po potrebi u toku igranja kviza.

### Bodovanje

Legenda:

- **BTO** - Broj tačnih odgovora;
- **MVT** - Maksimalno Vreme Trajanja kviza (**300** sekundi);
- **PVT** - Preostalo Vreme Trajanja kviza;
- **UBP** - Ukupan Broj Poena (šalje se na Leaderbord API).

**Formula:  $UBP = BTO * 2.5 * (1 + (PVT + 120) / MVT)$**

Kako bi maksimalnu vrednost UBP ograničili na 100, pre slanja na Leaderbord API potrebno je ograničiti UBP na 100. To možete da ostvarite u Kotlinu na sledeći način:

```
ubp.coerceAtMost(maximumValue = 100.00f)
```

## Vizuelni zahtevi

Za razliku od kolokvijumskog projekta, na ispitnom projektu će biti bitan važan i vizuelni aspekt aplikacije. Neophodno je koristiti Material Design 3 komponente i prilikom implementacije komponente upoznajte se sa specifikacijom komponente gde je Measurements najvažniji ([primer za list item](#)).

## Navigacija

Za navigaciju do dodatnih ekrana možete koristiti Drawer navigaciju (koju ćemo raditi 11. terminu), ili koristiti BottomNavigationBar, ili implementirati samostalno rešenje u skladu sa material dizajnom.

## Teme

- Aplikacija treba da podržava edge to edge funkcionalnost, odnosno da se sistemski barovi (status bar i navigation bar) uklope sa dizajnom i bojama ekrana.
- Neophodno je implementirati podršku za dark temu koristeći teme kao što smo radili na predavanju. Promena dark theme iz [quick actionsa u status notifications sekciji](#) (videti Dark theme dugme na linku) treba odmah da promeni temu u aplikaciji.

## Animacije

- Obavezno je koristiti animacije i tranzicije prilikom promene pitanja u kvizu.
- Ostale animacije su opcione u zavisnosti i ostavlja se sloboda autoru.

## Dokumentacija Web Servisa

### Cats Api

Za dobijanje podataka i fotografija o rasama mačaka korisaćmo TheCatApi.com servis.

Za korišćenje API-ja potreban je API ključ koji se dobija na email odmah nakon registracije.

Api dokumentaciju možete pronaći na sajtu api servisa ili na Postmanu preko linka u tabeli.

Website

<https://thecatapi.com>

Postman Api Dokumentacija

<https://documenter.getpostman.com/view/5578104/VUjSHPqo>

Registracija

<https://thecatapi.com/signup>

Prilikom registracije:

upišete svoj email na koji želite da dobijete Api ključ,  
kratak opis projekat,  
izaberite School/University Project opciju.

Bitne api rute za projekat

Spisak ključnih endpoint-ova neophodnih za realizaciju projekta.

Ruta za dobijanje liste rasa:

GET <https://api.thecatapi.com/v1/breeds>

Ruta za dobijanje detalja o konkretnoj rasi:

GET [https://api.thecatapi.com/v1/breeds/{breed\\_id}](https://api.thecatapi.com/v1/breeds/{breed_id})

Ruta za dobijanje informacija o specifičnoj fotografiji:

GET [https://api.thecatapi.com/v1/images/{image\\_id}](https://api.thecatapi.com/v1/images/{image_id})

Ruta za dobijanje fotografija o rasi:

GET [https://api.thecatapi.com/v1/images/search?breed\\_ids=id](https://api.thecatapi.com/v1/images/search?breed_ids=id)

Ruta za pretraživanje rasa:

GET <https://api.thecatapi.com/v1/breeds/search?q=query>

Ruta za dobijanje više fotografija o rasi:

{{cats\_endpoint}}/v1/images/search?breed\_ids=BREED\_ID&format=json

Zvanična endpoint dokumentacija je: <https://developers.thecatapi.com/view-account/yIX4bIBYT9FaoVd6OhvR?report=bOoHBz-8t>

Endpoint maksimalno vraća podrazumevano samo 1 fotografiju, a može maksimalno 10 fotografija sa parametrom limit=10, što znači da će biti potrebno bar nekoliko puta da pozovete endpoint da biste dobili malo više fotografija o rasi.

Podržava paginaciju i podrazumevani order je RAND koji vraća random fotografije. Verovatno je najbolja opcija da koristite ASC/DESC order i paginaciju da biste pokupili nekoliko strana rezultata.

# Leaderbord Api

Postoje dva endpointa koje su bitna za korišćenje Leaderboard APIja.

## Api Model Primeri

- **QuizResult:**  
{ "category": 2, "nickname": "111test", "result": 92.0, "createdAt": 1717621330048 }
  - **category:** id kateogrije (1, 2 ili 3);
  - **nickname:** samo slova, brojevi i underscore su dozvoljeni;
  - **result:** vrednost između 0.00 i 100.00;
  - **createdAt:** timestamp u milisekundama kada je rezultat snimljen na apiju;
- **Category** vrednosti Integera:
  - 1 - Guess the Fact;
  - 2 - Guess the Cat;
  - 3 - Left or Right;

## Endpoints

- **GET** leaderboard?category=Integer
  - **Category** query param je Integer vrednost kategorije.
  - Response je JSONArray svih rezultata (QuizResult) **sortiranih od najboljeg do najlošijeg**. Indeks u listi označava ranking rezultata, odnosno index 0 predstavlja 1. poziciju, index 1 predstavlja 2. poziciju, itd.

- **POST** leaderboard

{ "nickname": "rma", "result": 88.88, "category": 1 }

- **nickname:** samo slova, brojevi i underscore;
- **result:** vrednosti između 0.00 i 100.00 su dozvoljene;
- **category:** 1, 2, ili 3;

Response:

```
{  
  "result": { "category": 1, "nickname": "rma", "result": 88.88, "createdAt":  
1717624105670 },  
  "ranking": 1  
}
```

**ranking:** Poziciju na leaderboard listi za datu kategoriju.

API-iju se može pristupiti putem: <https://rma.finlab.rs>

Source code APIja se nalazi na: <https://github.com/Appollo41-Community/rma-quiz-leaderboard-api>

## Dokumentacija Kviza: Kategorije i tipovi pitanja

### Kategorija #1: Guess the Fact

(Cat picture question and 4 text answer options)

- Koja je rasa mačke?

Na ekranu se prikazuje slika mačke i 4 ponuđena odgovora od kojih je samo 1 odgovor tačan. Potrebno je da korisnik izabere tačnu rasu mačke.

- Izbaci uljeza!

Na ekranu se prikazuje slika mačke i 4 temperamenta u vidu odgovora. Samo 1 temperament ne pripada mački sa slike. Potrebno je da korisnik izabere temperament koji ne pripada rasi mačke sa slike.

- Koji temperament pripada zadatoj mački?

Na ekranu se prikazuje slika mačke i 4 temperamenta u vidu odgovora. Samo 1 temperament pripada mački sa slike. Potrebno je da korisnik izabere temperament koji pripada rasi mačke sa slike.

### Kategorija #2: Guess the Cat

(Text question and 4 cat images as answers)

- Koja je mačka <naziv\_temperamenta>?

Na ekranu se prikazuju 4 slike mačaka. Samo jedna od navedenih mačaka ima temperament iz pitanja. Potrebno je da korisnik izabere tačnu sliku mačke.

- Koja je mačka rase <naziv\_rase>?

Na ekranu se prikazuje 4 slike mačaka različitih rasa. Korisniku je zadata jedna rasa i potrebno je da pronađe mačku te rase u ponuđenim opcijama.

### Kategorija #3: Left or Right Cat

(Text question and 2 cat images as answers)



- Koja mačka je u proseku teža?

Na ekranu se prikazuju pitanje “Koja mačka je u proseku teža?” i slike 2 različite rase mačke sa nazivom rase. Potrebno je da korisnik izabere sliku mačke koja ima veću prosečnu težinu. Nije dozvoljeno prikazivati rase mačke sa istim prosečnim težinama.

- Koja mačka u proseku živi duže?

Na ekranu se prikazuje pitanje “Koja mačka u proseku živi duže?” i slike 2 različite rase mačke sa nazivom rase. Potrebno je da korisnik izabere sliku mačke koja ima duži prosečni životni vek. Nije dozvoljeno prikazivati rase mačke sa istim prosečnim životnim vekom.