



Implementation of a System for Image Describing on Mobile Phones

Presenter:

Bahar Amirian Varnousefaderani

Supervisor:

Dr. Mohsen Ebrahimi Moghadam

**Shahid Beheshti University
Computer Science and Engineering Department
Summer 2021**

Purpose

Implementing a system to use Artificial Intelligence models and particularly Computer Vision models on mobile phones with Android operating system



Introduction of implementation methods

Introduction of the implemented system

Client Side

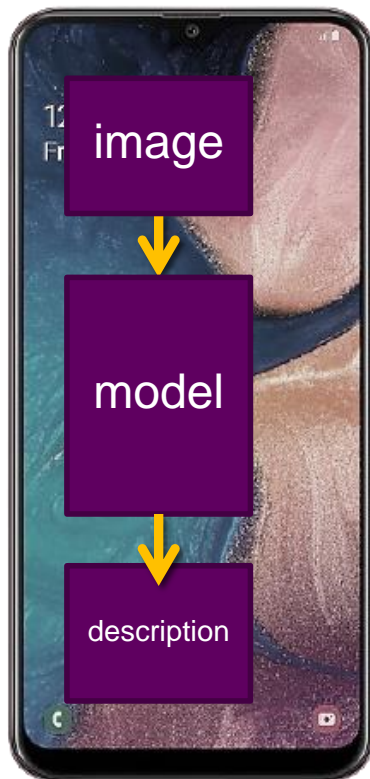
Server Side

Applications

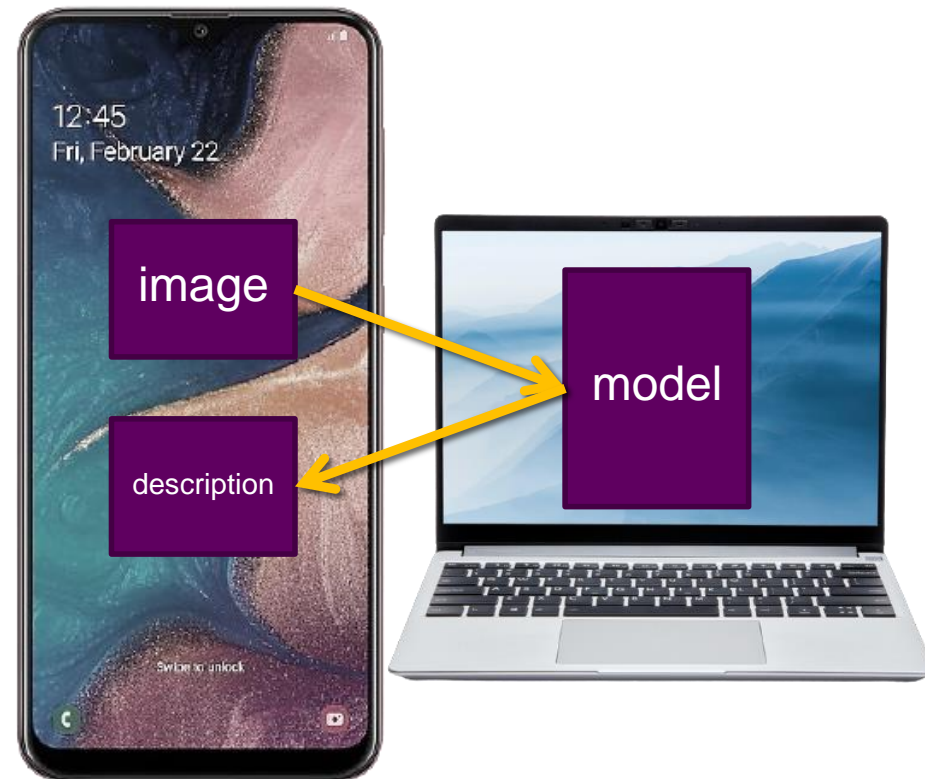
Implementation Methods

Model on Device

(e.g. using TensorFlow Lite)



Model on Server



Comparison between Implementation Methods

Issue	Model on Device	Model on Server
Latency	Lower latency enhances the real-time experience	Asynchronous communication and available bandwidth can affect latency
Resources	The particular device's resources, like processing power and storage, can limit performance	Cloud-based resources are more powerful and storage is more plentiful
Offline/Online	The ability to operate offline is a plus for running with poor or non-existing network infrastructure	A network connection is required
Cost	Battery usage, model download time for end users	Bandwidth for data transfer for end users, computing charges for developers
Privacy	User data never leaves the device	Data may leave the device, additional precautions may be necessary

**Introduction of
implementation methods**

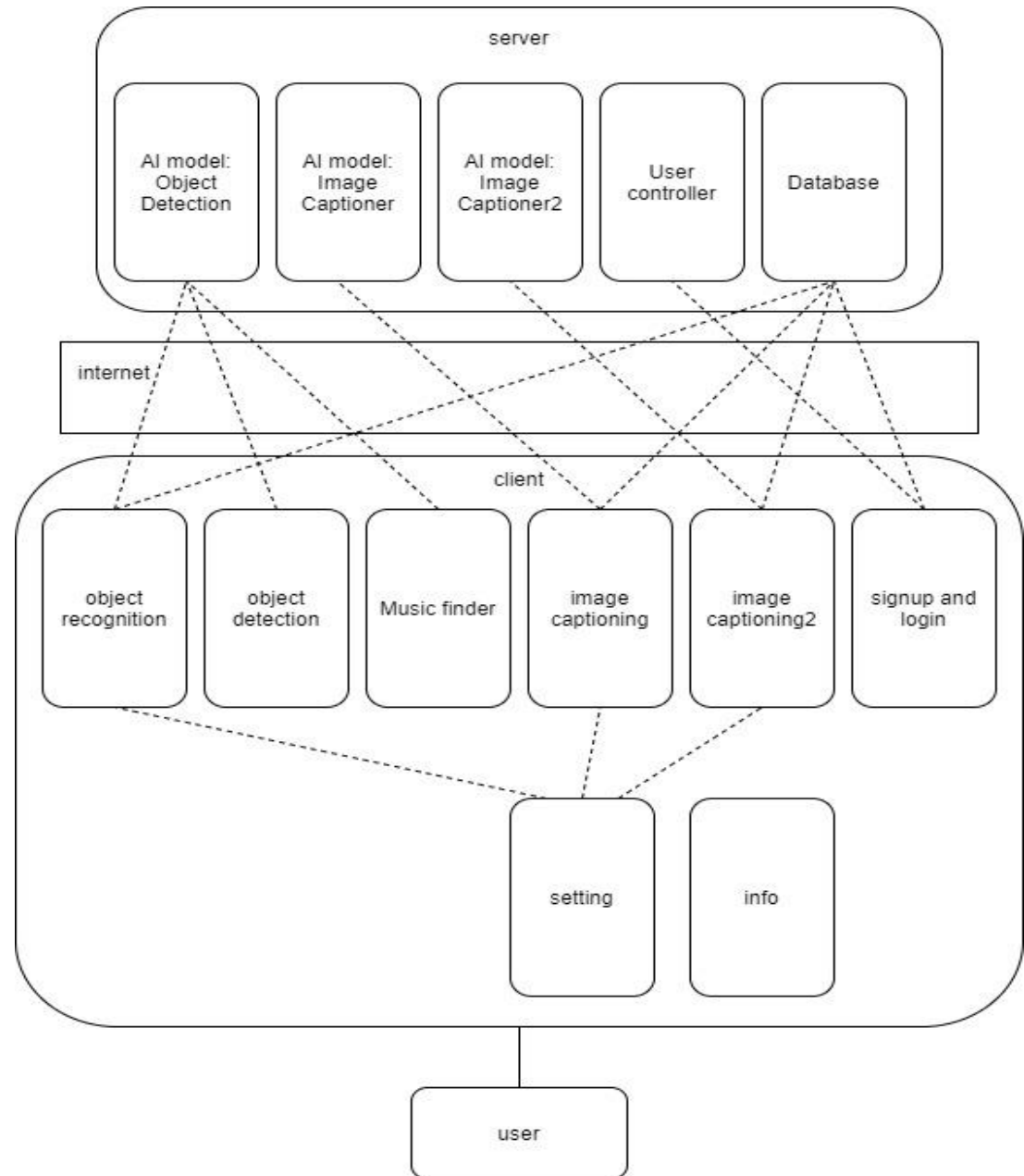
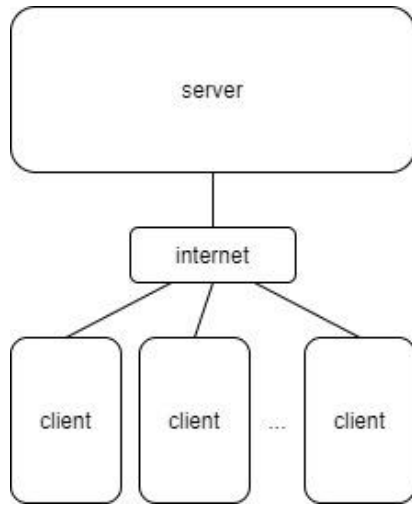
**Introduction of the
implemented system**

Client Side

Server Side

Applications

Overall Look



**Introduction of
implementation methods**

**Introduction of the
implemented system**

Client Side

Server Side

Applications

Server Side

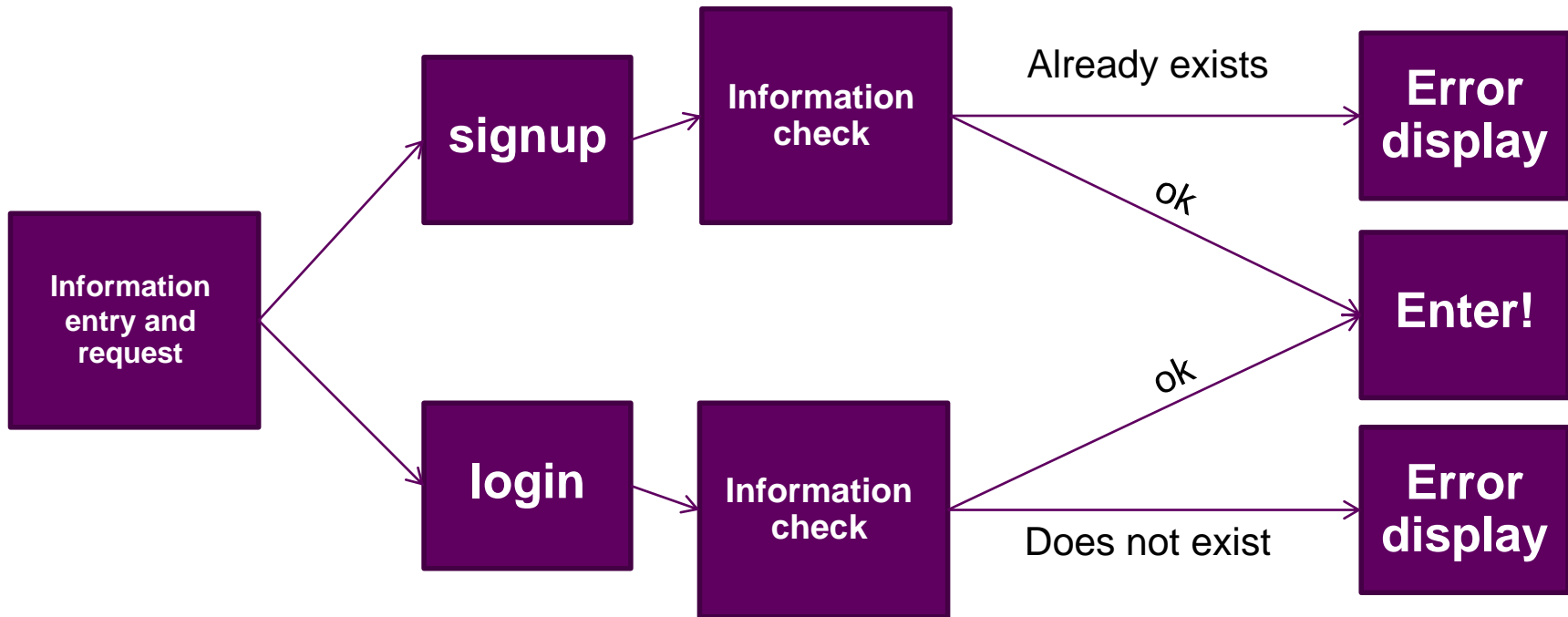
Implemented using python and flask framework under the Microservice Architecture

Includes services for:

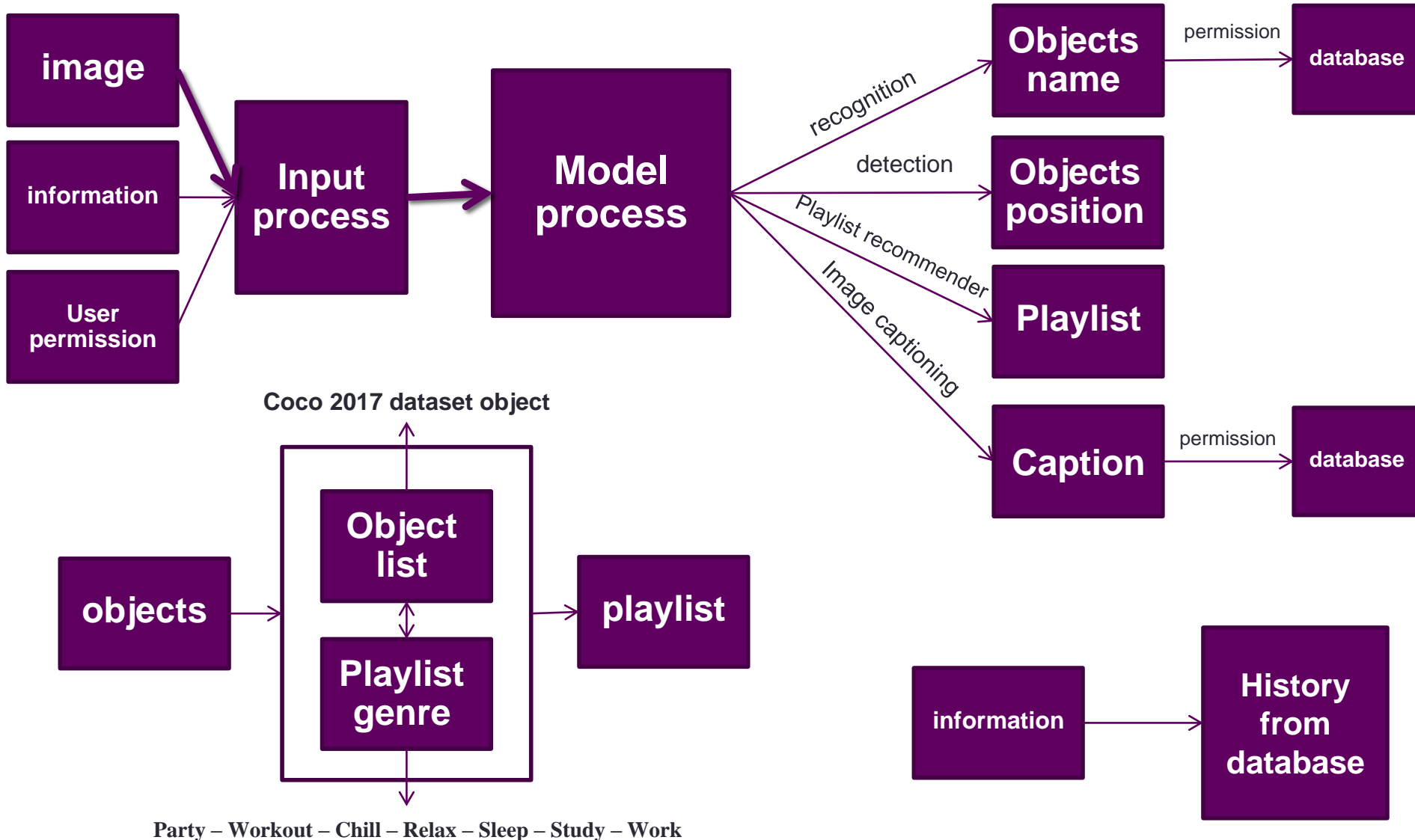
- Signup and login of a user
- Using the describer models (Object Recognition and Detection, Image Captioning, Music Play List Recommender)
- Configuring the database

Any of these services can be run separately as normal or using Docker and Docker Compose

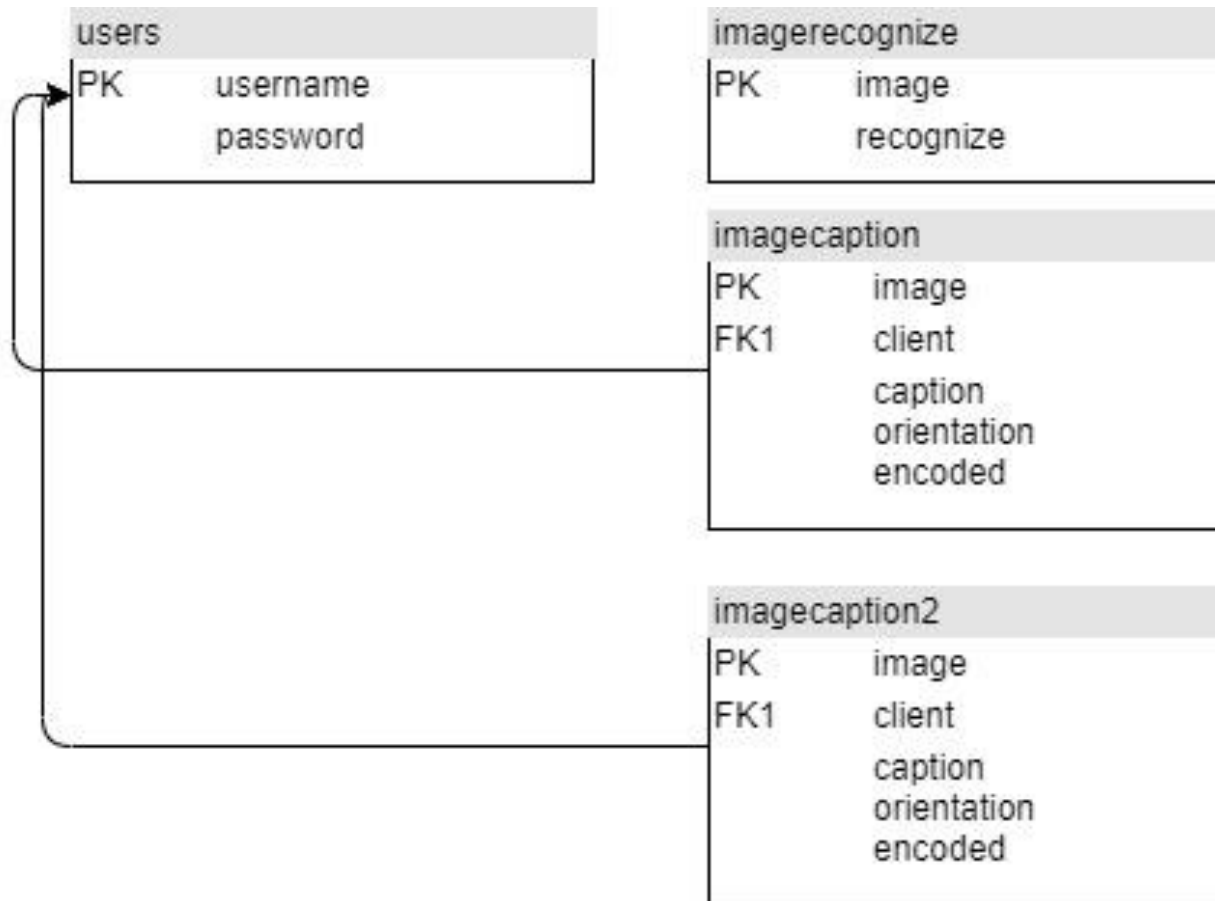
Server Side – Signup and Login



Server Side – Descriptor Models



Server Side – Database



**Introduction of
implementation methods**

**Introduction of the
implemented system**

Client Side

Server Side

Applications

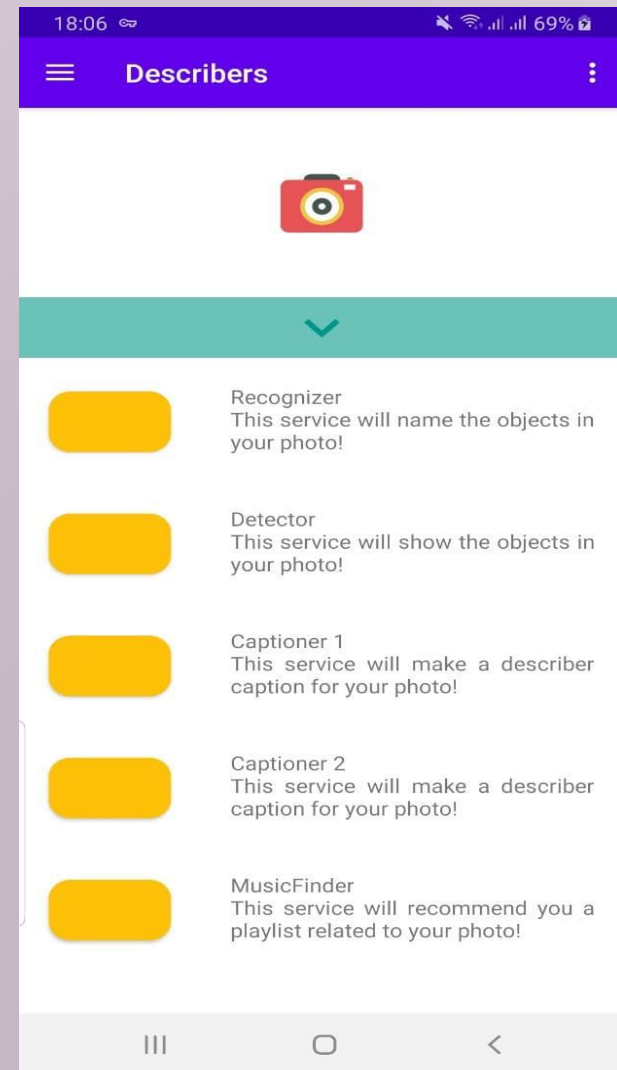
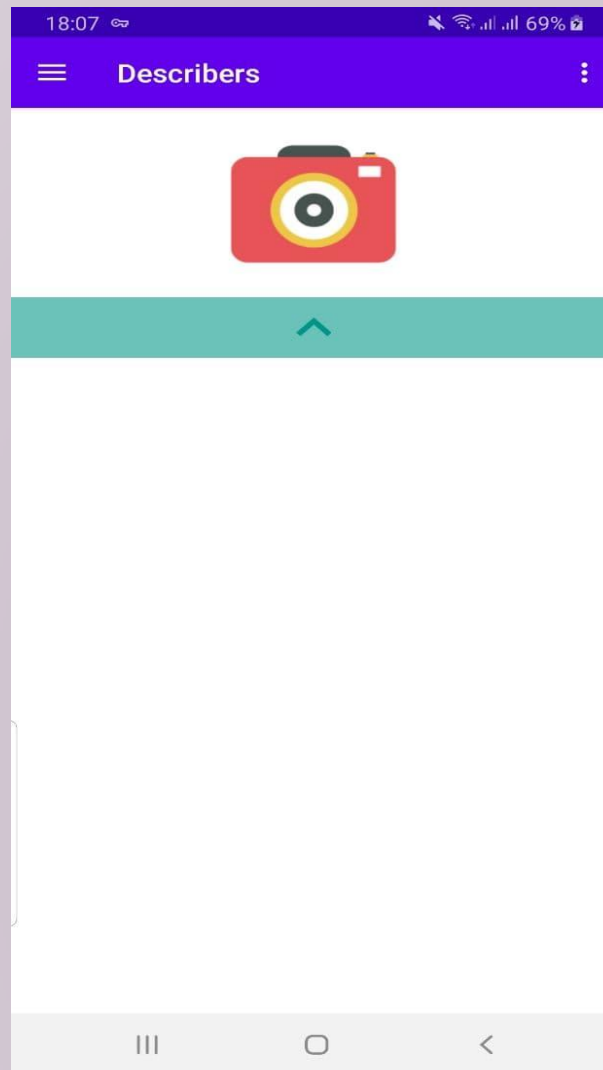
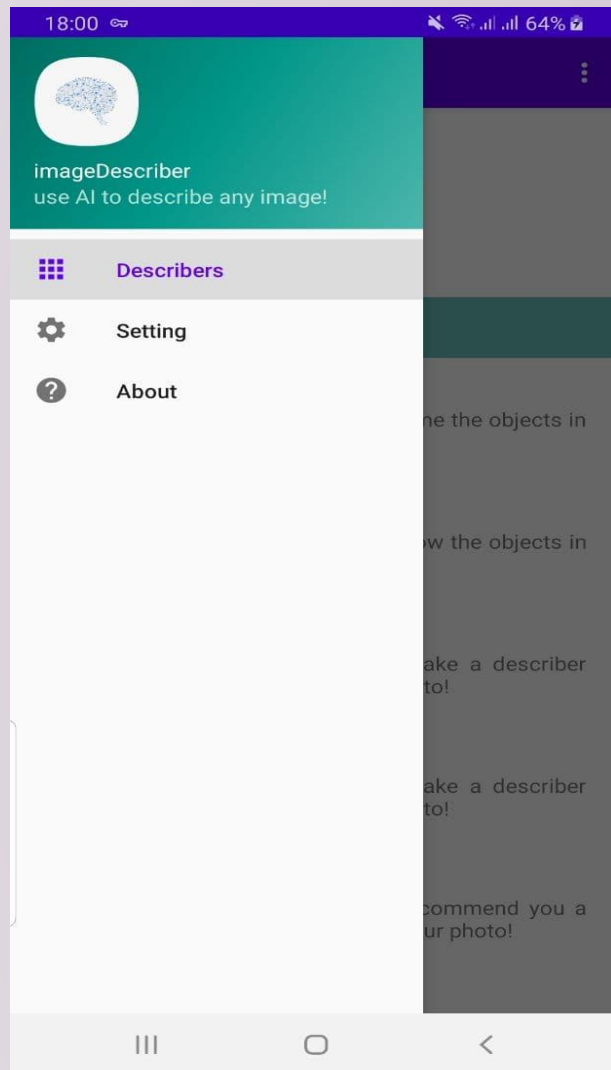
Client Side

Implemented using Java on Android Studio for mobile phones with Android operating system

Includes facilities for:

- Accessing services (Object Recognition and Detection, Image Captioning, and Music Play List Recommendation based on the models on server)
- Showing an album of previous results produced in the image captioning services
- Information
- Setting
- Login and Signup

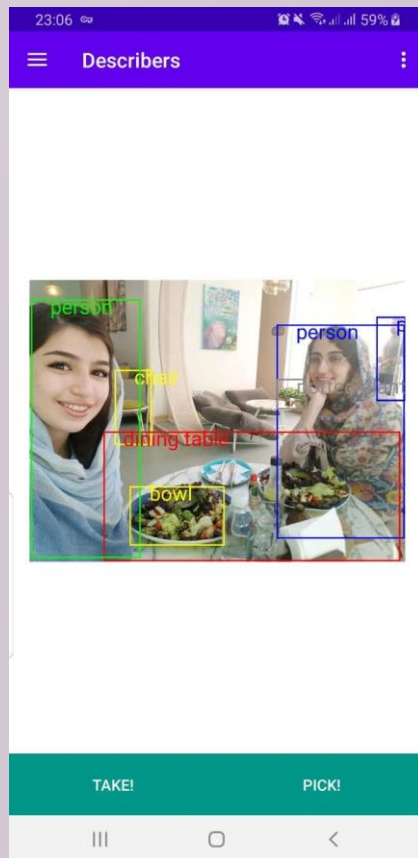
Client Side – Menus



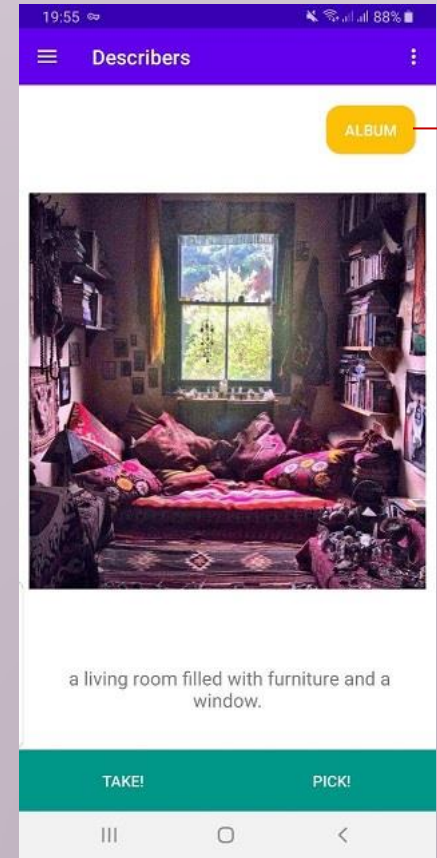
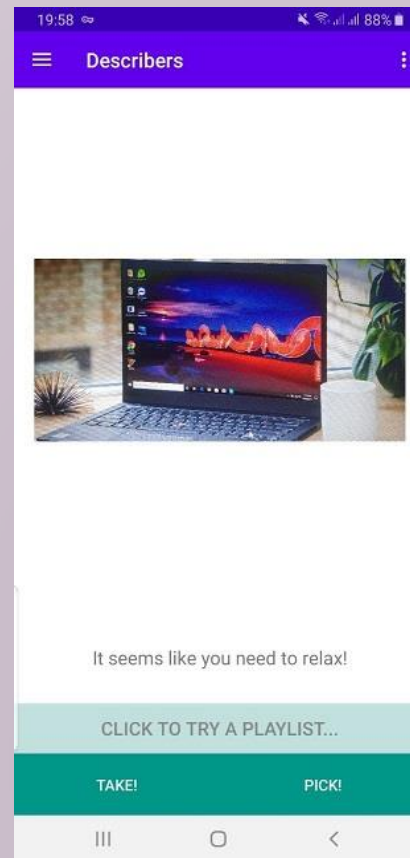
Client Side - Services



Taking a picture

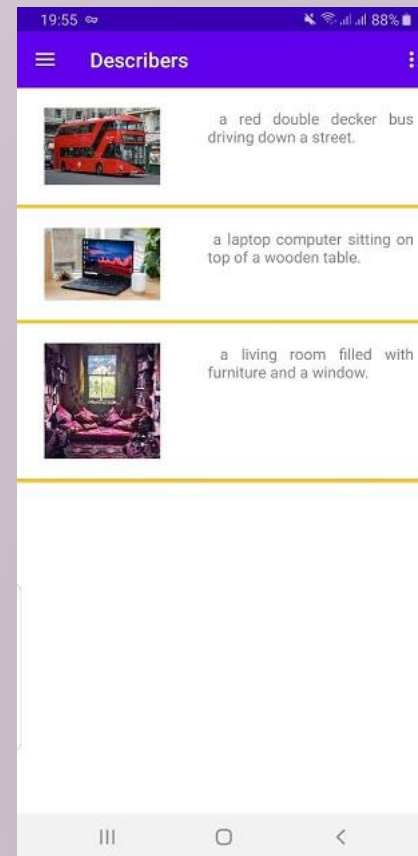
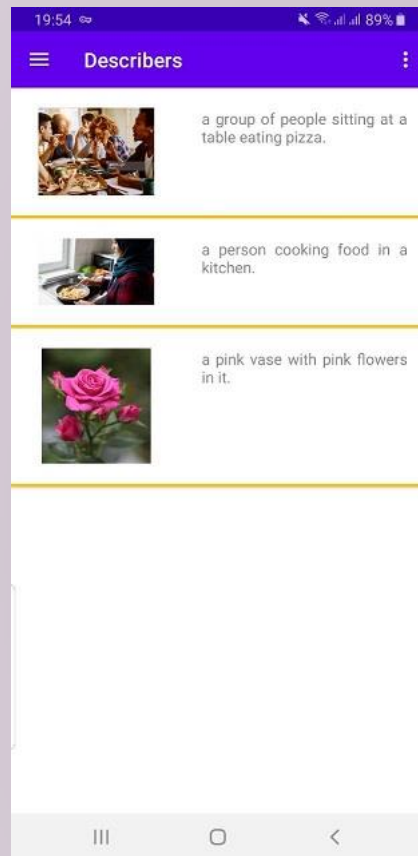


Picking picture form gallery

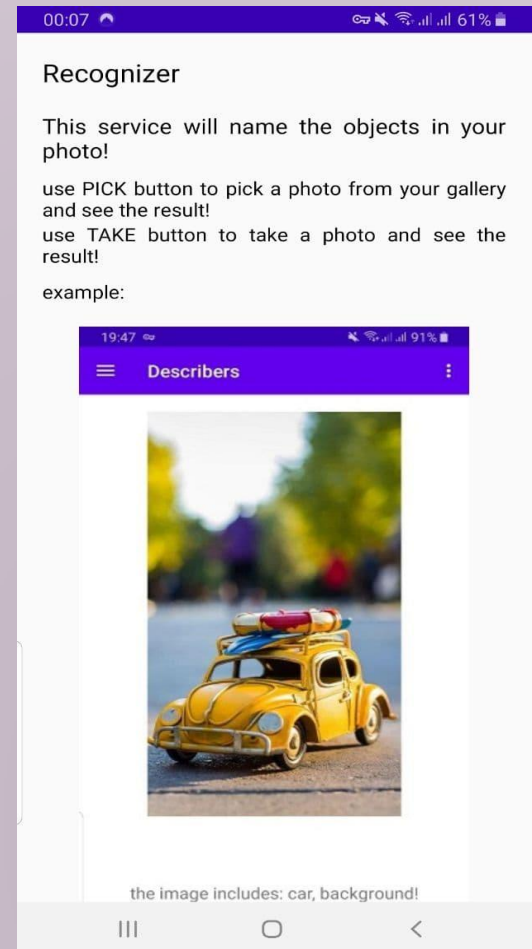
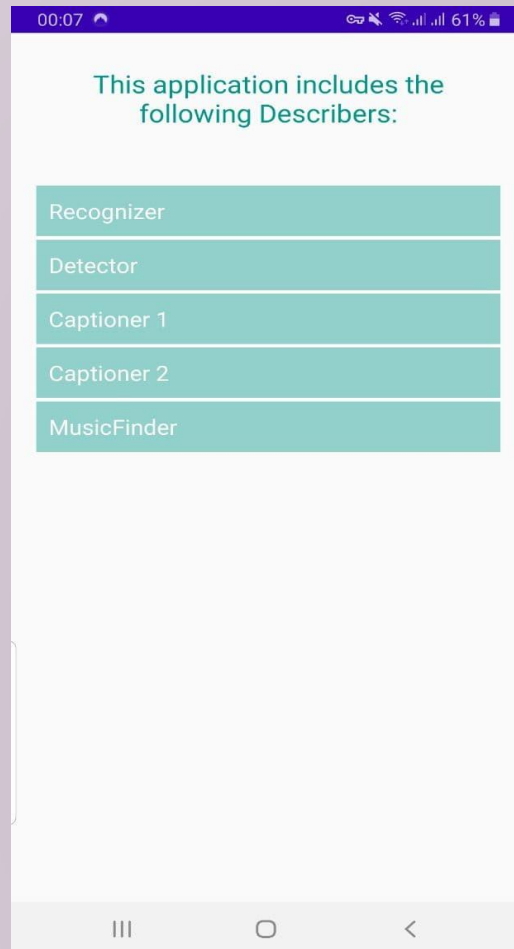
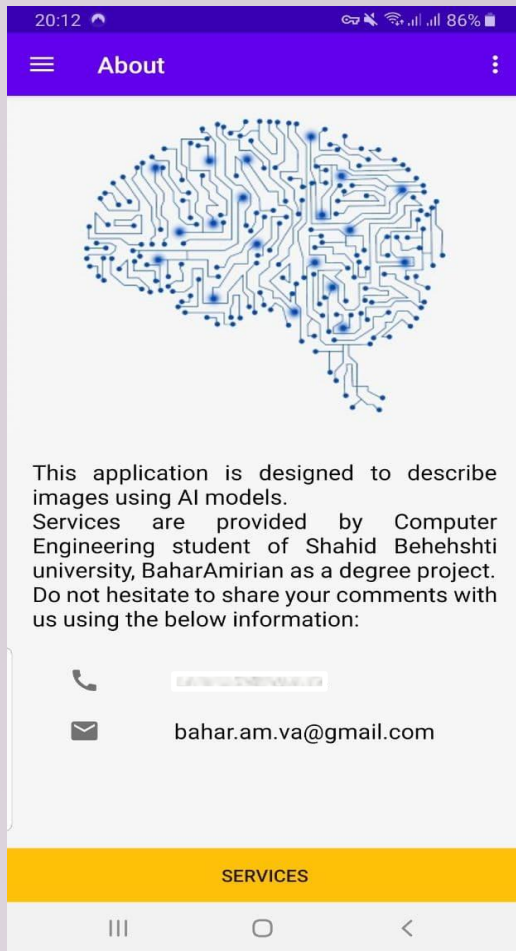


Showing the album of results

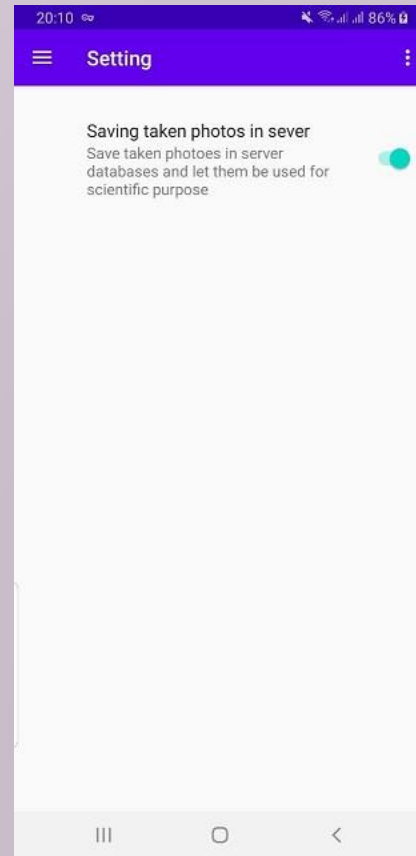
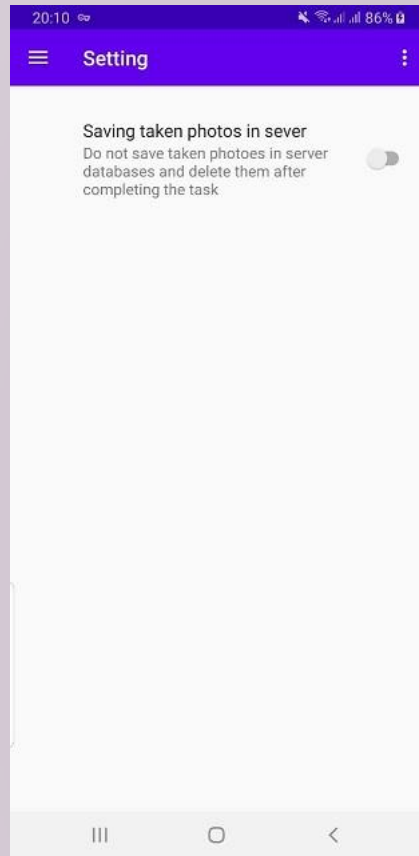
Client Side – Album of Results



Client Side – Information



Client Side – Setting: User Permission



Client Side – Login and Signup

17:35

username

password

LOGIN

SIGN UP

17:42

bhr

password

username should be 4 characters at least!

LOGIN

SIGN UP

**Introduction of
implementation methods**

**Introduction of the
implemented system**

Client Side

Server Side

Applications

Some Applications

- Assistive tool for visually impaired people to gain an understanding of the surrounding environment or images
- Platform for using other Artificial Models, specially Computer Vision Models
- Producing datasets