

# Dina Othman Emam

✉ dinaosman581@gmail.com ☎ 01100725368 📍 Cairo, Egypt 🔗 LinkedIn 🐙 Github

## PROFILE

I'm a Faculty of Computers and Artificial Intelligence Cairo University graduate, with a strong focus on Android development. I have gained expertise through immersive courses and by creating diverse applications. Committed to continuous learning, I consistently expand my proficiency in Android development.

## EDUCATION

**Bachelor's degree, Information System,**

09/2020 – 07/2024

*Faculty of Computers and Artificial Intelligence, Cairo University.*

**GPA: 3.13/4.00**

## PROJECTS

**Graduation Project (Grade A+), Disease Prediction and Management Mobile Application** 📄

A mobile application aimed at assisting patients in predicting the likelihood of various diseases and managing their health more effectively.

The app assists patients in predicting diseases and managing their health with features such as disease prediction based on symptoms, AI-based skin disease diagnosis, chronic disease management tips, medication information, medical calculations, AI chat, and support for both Arabic and English languages. The project also includes dark and light modes for user convenience.

**News App** 📄

Mobile App enables users to browse, bookmark favorites, and share, search for news articles, offering a seamless reading experience with onboarding screens and a splash screen. Built with Android Kotlin and Jetpack Compose UI, the app utilizes MVVM architecture, Retrofit for API integration, Room for offline storage, and Hilt for dependency injection, ensuring smooth functionality and responsive navigation

**Movies App** 📄

The Movie App enables users to explore movies through a visually engaging interface with features like infinite scrolling and detailed movie views. Built with Android Kotlin and Jetpack Compose, it employs MVVM architecture, Retrofit for API integration, and Room for local data storage. The app also includes dynamic color adjustments and seamless navigation using Hilt for dependency injection and Accompanist Pager for tab-based navigation.

**Gemini Chat App** 📄

an Android chat application developed with Jetpack Compose and Kotlin, featuring integration with a generative AI model for text and image-based conversations. It employs MVVM architecture along with coroutine-based asynchronous data handling and state management to provide real-time chat interactions, including image uploads and AI-generated responses.

**To Do App** 📄

To-Do List App that allows users to manage tasks, including marking them as important, editing, and deleting tasks. Built using Jetpack Compose, Kotlin, and MVVM architecture, with Room for local storage and Hilt for dependency injection. Integrated smooth navigation transitions and snackbar notifications for task actions.

**GymsAround App** 📄

Gyms Around app where users can explore gym locations, view details, and mark gyms as favorites. The app uses Kotlin with Jetpack Compose for UI, employs the MVVM architecture, integrates Retrofit for network calls, and utilizes Hilt for dependency injection. Additionally, the app includes unit testing for UI components and a splash screen for improved user experience.

**Calculator App** 📄

Calculator App allows users to perform mathematical calculations with features for basic operations, percentage calculations, and a responsive UI. Developed using Android Kotlin with ConstraintLayout for the interface and ScriptEngine for expression evaluation. The app includes essential functions like clear, backspace, and handling multi-digit inputs.

**MNIST Digit Classification Project** 📄

Implemented a machine learning model to classify handwritten digits using the MNIST dataset. Conducted data exploration, preprocessing, and normalization of image data. Applied K-Nearest Neighbors (K-NN) with hyperparameter tuning and built multiple Artificial Neural Network (ANN) architectures. Evaluated model performance using accuracy metrics and confusion matrices, and deployed the best-performing model for predictions on unseen test data.

**Fawry\_System\_API** 📄

The Fawry Services project is a comprehensive Spring Boot application designed to facilitate online payment and refund processes. It includes features such as user authentication, payment handling (via credit card, wallet, and cash), service and provider selection, and transaction tracking. Built with Java, Spring Boot, and RESTful APIs, it showcases proficiency in backend development and financial transaction management.

### **Parking-Garage-System**

This project is a parking management system implemented in Java. It includes features for parking in and out, slot selection strategies like first-come-first-serve and best-fit, along with functionalities to track total income and the number of vehicles parked.

### **Game-Rental-Database-System**

Developed a Windows Forms application using C# for managing game rentals, including functionalities for adding, updating, and searching rental records in a SQL Server database. Implemented features for data display and user interaction through DataGridViews and various input controls.

### **Higher-Studies-Website**

This project is a Django-based web management system allowing CRUD operations on student and course records. It includes features like client-side form validation, search/filter functionality, and AJAX-powered row deletion for improved user interaction. The system enhances efficiency in managing student and course data within educational institutions.

### **Registration Form**

This project is a Laravel-based registration form featuring form validation and file uploads. It includes functionalities such as input validation, password strength requirements, and image uploads. The system provides error messages for incorrect inputs and follows MVC architecture for clean and organized code

## **SKILLS**

---

- |                                     |                                  |
|-------------------------------------|----------------------------------|
| • Android development with Kotlin   | • MySQL, SQL server              |
| • Dagger-Hilt, Coroutines           | • XML, Compose UI                |
| • Unit Testing                      | • Algorithms and Data Structures |
| • Design Patterns, SOLID Principles | • REST APIs, Retrofit            |
| • Architecture Patterns (MVVM, MVC) | • Clean Architecture             |
| • Git, GitHub                       | • Java, C++, Python, Kotlin      |
| • Object-Oriented Programming       | • Flows                          |