

# FLUTTER MOBILE PROGRAMMING

## 1. Mục tiêu

Sinh viên áp dụng toàn bộ kiến thức đã học trong môn PRM393 để **xây dựng một ứng dụng di động hoàn chỉnh bằng Flutter**, có kết nối **backend và cơ sở dữ liệu thực tế**, triển khai theo đúng tiến độ các slot học.

Project nhằm đánh giá:

- Hiểu và sử dụng Flutter & Dart
- Xây dựng UI đa nền tảng
- Quản lý state, navigation, form
- Lập trình bất đồng bộ
- Kết nối RESTful API
- Lưu trữ dữ liệu và triển khai ứng dụng thực tế

====

## 2. Đề tài

Sinh viên **tự chọn 01 đề tài cá nhân**, theo mẫu:

**Ứng dụng + chức năng chính + đối tượng sử dụng**

**Ví dụ đề tài hợp lệ**

- Ứng dụng **đặt lịch cắt tóc** cho **30Shine**
- Ứng dụng **bán giày** cho **Adidas**
- Ứng dụng **order đồ ăn**
- Ứng dụng **đặt lịch sửa xe**

- Ứng dụng **quản lý phòng trọ**
- Ứng dụng **đặt lịch khám bệnh**

### Lưu ý

- Mỗi sinh viên làm **01 project riêng** (không làm nhóm)
- Đề tài phải **thực tế**, có **luồng chức năng rõ ràng**
- Không chấp nhận app demo đơn giản, app chỉ có UI hoặc hardcode dữ liệu

===

## 3. Công nghệ bắt buộc

### Frontend (Mobile App)

- **Flutter + Dart** (bắt buộc)
- Chạy trên Android (iOS là điểm cộng)

### Backend (chọn 1)

- PHP / Node.js / Java / C# / Python (Flask, FastAPI, Django REST)

### Database (chọn 1)

- MySQL
- SQL Server
- MongoDB
- Firebase (Firestore / Realtime Database)

### Không sử dụng SQLite làm database chính

(Có thể dùng SharedPreferences cho token, setting)

===

## 4. Yêu cầu ứng dụng Flutter

Ứng dụng phải đáp ứng các yêu cầu sau:

- Giao diện rõ ràng, thân thiện, tối ưu cho mobile
- Có chức năng **đăng nhập / đăng ký**
- Có **chức năng chính theo đề tài** (đặt lịch, mua hàng, tìm kiếm, quản lý,...)
- Kết nối backend qua **RESTful API**
- Dữ liệu lấy từ **server thực tế**, không hardcode
- Có:
  - Navigation giữa các màn hình
  - State management (setState / Provider / Riverpod / Bloc,...)
  - Lập trình bất đồng bộ với async/await
- (Khuyến khích) chức năng nâng cao:
  - Thống kê, báo cáo
  - Thanh toán VNPAY / MoMo
  - Local / Push Notification
  - Kết nối nhiều user

---

### Cuối Tuần 3 (Kết thúc Slot 6) – Giai đoạn Thiết Kế

Sinh viên phải **commit lên GitHub** các nội dung sau:

1. **Flutter source code**

- Project Flutter được khởi tạo đầy đủ
- Cấu trúc thư mục rõ ràng
- Các màn hình chính:
  - Login / Register
  - Home
  - Các màn hình chức năng theo đề tài
- Navigation giữa các màn hình

## 2. Thiết kế giao diện

- Mockup (Figma / ảnh / PDF) **hoặc**
- UI Flutter thể hiện layout chính

## 3. Thiết kế cơ sở dữ liệu

- Sơ đồ CSDL (ERD)
- Danh sách bảng / collection
- Không dùng SQLite

## 4. README.md

- Tên đề tài
- Mô tả ứng dụng
- Chức năng dự kiến
- Công nghệ sử dụng

===

## Cuối Tuần 6 (Kết thúc Slot 12) – Giai đoạn Backend & API

Sinh viên phải **commit lên GitHub** các nội dung sau:

### 1. Source code Backend

- Backend chạy được
- Kết nối database thành công

### 2. Database

- File .sql hoặc cấu trúc MongoDB / Firebase
- Có dữ liệu mẫu

### 3. RESTful API

- Login / Register
- CRUD chức năng chính
- API test được bằng Postman / Swagger

### 4. (Khuyến khích) Web Admin

- CRUD dữ liệu chính

### 5. Cập nhật README

- Hướng dẫn chạy backend
- Danh sách API

===

## Cuối Tuần 9 (Kết thúc Slot 16) – Giai đoạn Tích Hợp Hoàn Chỉnh

Sinh viên phải **commit lên GitHub** các nội dung sau:

### 1. Flutter App hoàn chỉnh

- Tích hợp API thật
- Các chức năng chính hoạt động đầy đủ

- Xử lý loading, lỗi, validation

## 2. State Management & Async

- Áp dụng state management hợp lý
- Sử dụng async/await, Future

## 3. Lưu trữ cục bộ

- SharedPreferences (token, login state)

## 4. Hoàn thiện Backend

- Fix lỗi
- Đồng bộ với app Flutter

## 5. Cập nhật README

- Hướng dẫn chạy app
- Tài khoản test
- Link demo (nếu có)

===

## 6. Nộp bài cuối cùng (Slot 18)

Sinh viên nộp lên Classroom:

- Link **YouTube demo** ứng dụng
- Link **GitHub** (Flutter + Backend)
- Link trên các chợ ứng dụng như **Google Play / apkpure.com**

====

## 7. Chấm điểm (Slot 19)

- Sinh viên trình bày project trực tiếp
- Giảng viên hỏi về code, API, luồng xử lý

## Thang điểm

- **5 điểm:** App chạy được luồng cơ bản, không lỗi
- **6 điểm:** Có database server
- **7 điểm:** Có API + web admin CRUD + + thanh toán online qua vnpay, momo, paypal (không phải qua QRcode)
- **8 điểm:** Có đầy đủ báo cáo thống kê quan trọng của người quản lý, các user có thể trao đổi với nhau qua hệ thống, commit đúng deadline, đầy đủ từng phần như mô tả lên github
- **9 điểm:** App live trên Store trong trước slot 18
- **10 điểm:** Ứng dụng bảo mật tốt, hoàn hảo, kiến trúc có thể mở rộng, không còn lỗi