

LAPORAN PRAKTIKUM MOBILE PROGRAMMING
Desain GUI, Navigasi, dan Integrasi API dengan JSON Serialization di
Flutter



Nama : Sausan Shalihah Alfirdausi
NIM : 230605110064
Kelas : E
Tanggal : 9 Oktober 2025

JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI

UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM

MALANG

GANJIL 2025/2026

I. Tujuan

- Mempelajari cara membuat halaman detail pada aplikasi Flutter yang menampilkan data dari API eksternal.
- Memahami penggunaan model class dengan `json_serializable` untuk mengubah data JSON menjadi objek Dart.
- Mengimplementasikan `FutureBuilder` untuk menangani proses asynchronous pengambilan data dari API.
- Menampilkan informasi game berupa gambar, deskripsi, sistem requirement, serta screenshot dengan tampilan yang rapi dan interaktif.

II. Langkah Kerja

Membuat file **detail.dart** di folder `view/` untuk menampilkan halaman detail game.

- Menambahkan konstruktor `gameTerpilih` sebagai ID game yang diterima dari halaman Home.
- Menggunakan `FutureBuilder` untuk menampilkan data dari API.
- Menambahkan widget `Header`, `Expanded`, `ListView`, dan `ReadMoreText` untuk menampilkan detail game.

Membuat file **detailgame.dart** di folder `model/`.

- Mendefinisikan class `MinimumSystemRequirements`, `Screenshot`, dan `DetailGame`.
- Memberikan anotasi `@JsonSerializable()` agar dapat digenerate otomatis file `detailgame.g.dart`.

Membuat file **fetchgame.dart** di folder `viewmodel/`.

- Membuat fungsi `fetchDataFromAPI(int idGame)` untuk mengambil data detail game dari API `freetogame.com`.
- Mengembalikan data JSON dalam bentuk `Map<String, dynamic>`.

Menjalankan perintah:

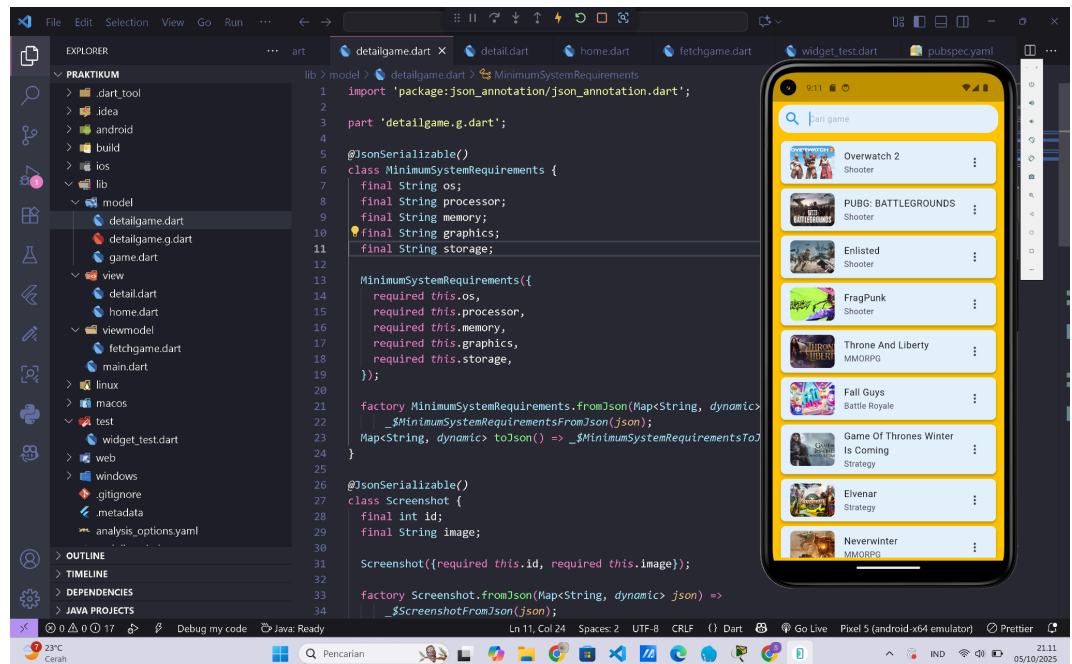
`flutter pub get`

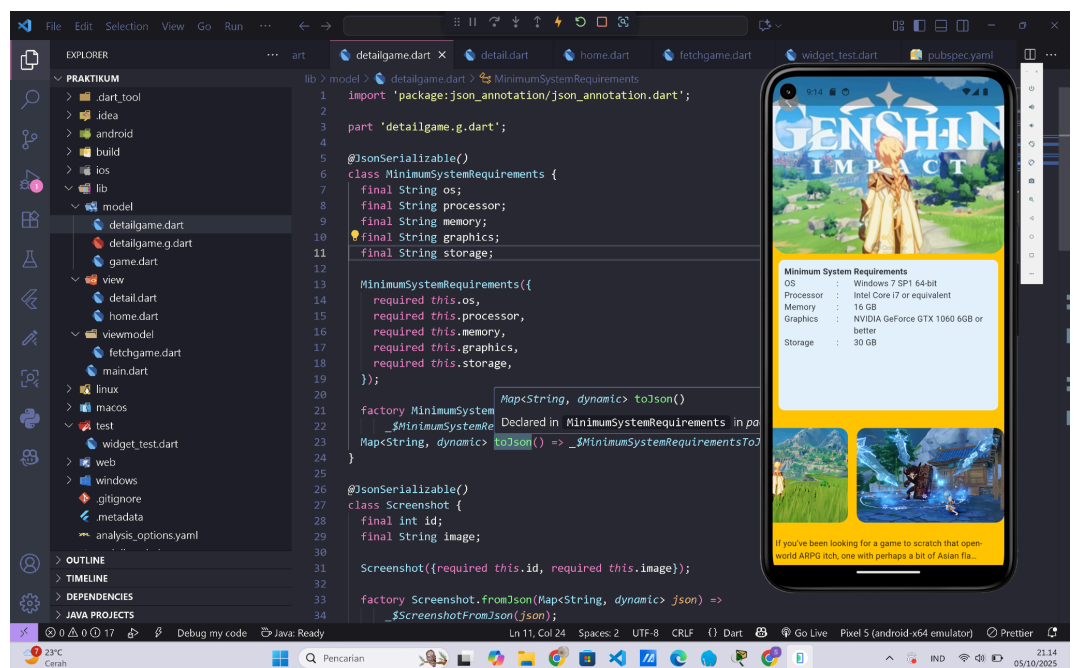
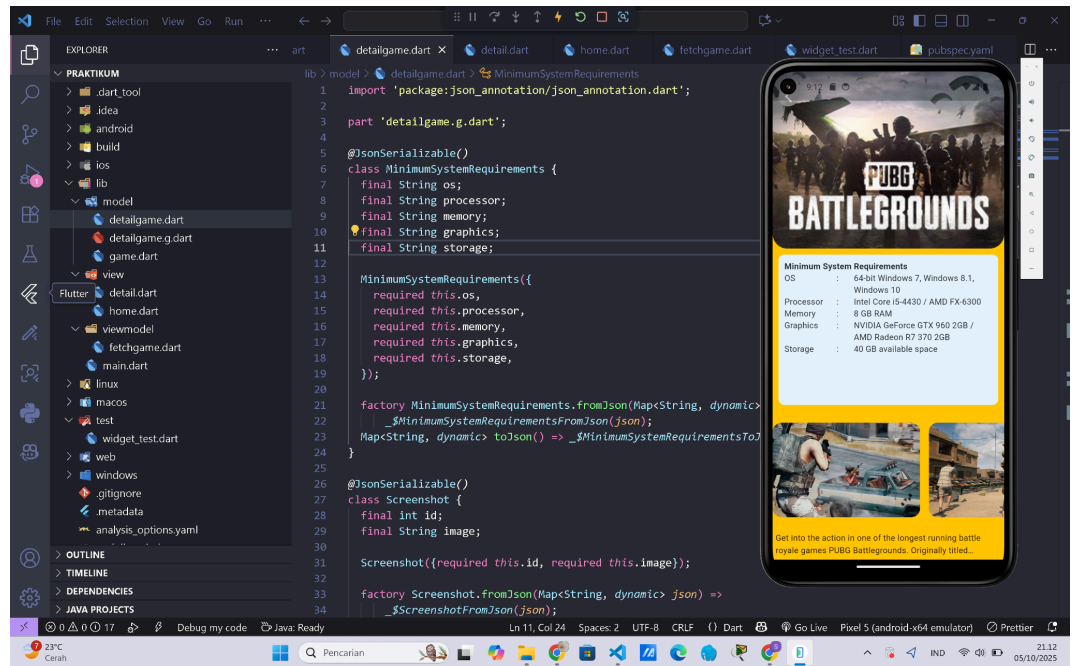
`flutter pub run build_runner build --delete-conflicting-outputs`

untuk mengunduh dependency dan menghasilkan file otomatis `detailgame.g.dart`.

Menghubungkan route di **main.dart** agar halaman detail dapat diakses dari halaman Home dengan parameter ID game.

III. Screenshot Hasil





IV. Kesimpulan

- Dengan menggunakan model data dan package json_serializable, proses parsing JSON ke objek Dart menjadi lebih mudah dan terstruktur.

- FutureBuilder mempermudah penanganan data asynchronous sehingga UI dapat menampilkan indikator loading, error, maupun data yang sudah berhasil dimuat.
- Flutter menyediakan widget yang fleksibel (seperti Stack, ClipRRect, ListView, dan ReadMoreText) untuk membangun tampilan detail game yang interaktif dan menarik.
- Praktikum ini berhasil memperlihatkan bagaimana data dari API eksternal dapat diolah dan ditampilkan dalam aplikasi Flutter secara dinamis.