

Rancangan Fitur Lengkap

1. Homepage

- Menampilkan generator utama: pilihan hero (filter lane termasuk **exp**, **mid**, **roam**, **jungle**, **gold lane**), battle spell, item (kategori physical, mage, tank, all).
- Tombol **Generate** → membuat build baru dalam format visual.
- Di bawah generator ditampilkan beberapa **thumbnail hasil-hasil build terakhir** (misalnya 4–6 paling baru).
- Tombol **Selanjutnya** menuju ke halaman gallery.

2. Gallery Page

- Menampilkan **semua build yang sudah dibuat** oleh user (atau dapat juga oleh semua user jika diatur).
- Fitur filter:
 - Pilih berdasarkan hero tertentu atau kategori hero/role.
 - Tombol “Random dari Gallery” → akan menampilkan satu build secara acak dari hasil yang sudah ada sesuai filter.
- Setiap thumbnail clickable untuk melihat detail full-size image.

3. Login & Roles

- **Login/Register** wajib untuk bisa menggunakan fitur generator.
 - Jika belum login dan klik generate → diarahkan ke halaman login dulu.
- Sistem **3 role pengguna**:
 - **Super Admin**: Bisa menambah/edit hero/spell/item, mengelola user dan role, melihat semua gallery.
 - **Admin**: Bisa memoderasi konten (hapus/approve build), mengedit data kategori hero atau item jika diberikan izin.
 - **User biasa**: Bisa generate build, lihat gallery milik sendiri (atau publik), filter & random di gallery.
- Struktur database role standar terpisah: tabel roles + tabel relationship users-roles [Django Forum+1Designmodo+1Laracasts+2Frontegg+2Dittofi Docs+2Stack Overflow+3Kong Docs+3Medium+3](#).
- Inisialisasi super admin pertama dilakukan via *database seeder* atau script migrasi saat deployment [Reddit](#).

4. Filter Hero & Item

- **Hero:** filter kategori termasuk exp, mid, roam, jungle, **gold lane**.
- **Battle Spell:** bebas pilih satu atau banyak.
- **Item:** pilih berdasarkan kategori (physical, mage, tank, all); bisa satu atau banyak kategori sekaligus.
- Kombinasi valid dilakukan dengan memilih dari subset sesuai filter.

5. Visual Build Generator

- Output berupa **gambar build**: icon hero, spell, items dalam layout estetika seperti contoh.
- Nama challenge bisa dicustom ditampilkan di header.
- Bisa mengunduh gambar PNG/JPEG.
- Tersedia opsi share ke media sosial.

6. Backend & Frontend

- **Backend** (Node.js / Python Flask / Django / Laravel): autentikasi + role-based access control (RBAC) [FreeCodeCampFrontegg](#).
- **Frontend:** React / Vue / Svelte atau kombinasi HTML + JS.
- **Database:** tabel users, roles, hero, spell, item, build_results.
- **APIs:**
 - listing hero/spell/item,
 - submit build (POST → generate dan simpan),
 - fetch builds, filter builds.

7. Admin Panel

- Super admin / admin bisa menambah/edit data hero/spell/item/kategori lane.
- Mengelola user (membuat user/admin dan assign role).
- Melihat statistik penggunaan build.

Struktur Data (Contoh Tabel)

Tabel Kolom Utama

users id, email, password, role_id, dsb.

roles id, role_name (super-admin, admin, user)

heroes id, name, kategori_lane (multi-flag)

spells id, name, icon_url

items id, name, kategori_item (physical, mage, tank)

builds id, user_id, name_challenge, hero_id, spell_ids, item_ids, created_at