Proyecto Aplicación web para una peluquería en PHP, te sugiero incluir las siguientes funcionalidades básicas:

**Funcionalidades:**

1. **Gestión de citas**:
   * Reservar citas para clientes con fecha, hora y servicios específicos.
   * Consultar y administrar las citas.
2. **Gestión de servicios**:
   * Listado de servicios ofrecidos (corte, peinado, tintes, etc.) con precios.
   * Agregar, editar y eliminar servicios.
3. **Gestión de clientes**:
   * Registro y consulta de clientes frecuentes.
4. **Panel de administración**:
   * Controlar el sistema desde un panel seguro.
5. **Notificaciones**:
   * Confirmación de cita por correo electrónico o mensaje en la web.
6. **Estadísticas** (opcional):
   * Visualizar citas realizadas, servicios más solicitados, ingresos aproximados.

**Pasos para construirla:**

**1. Configurar el entorno:**

* Instala un servidor local como XAMPP o WAMP.
* Crea una base de datos llamada peluqueria con tablas como:
  + clientes
  + servicios
  + citas
  + usuarios (para administración).

**2. Estructura del proyecto:**

* /config/ → Archivos de configuración (base de datos).
* /models/ → Lógica de la base de datos (CRUD).
* /views/ → Interfaces HTML.
* /controllers/ → Procesos de negocio.
* /public/ → Archivos estáticos (CSS, JS, imágenes).

**3. Crear la base de datos:**

Diseño básico:

sql

Copiar código

CREATE TABLE clientes (

id INT AUTO\_INCREMENT PRIMARY KEY,

nombre VARCHAR(100) NOT NULL,

telefono VARCHAR(20),

email VARCHAR(100),

fecha\_registro TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);

CREATE TABLE servicios (

id INT AUTO\_INCREMENT PRIMARY KEY,

nombre VARCHAR(100) NOT NULL,

precio DECIMAL(10, 2) NOT NULL

);

CREATE TABLE citas (

id INT AUTO\_INCREMENT PRIMARY KEY,

cliente\_id INT,

servicio\_id INT,

fecha DATE NOT NULL,

hora TIME NOT NULL,

FOREIGN KEY (cliente\_id) REFERENCES clientes(id),

FOREIGN KEY (servicio\_id) REFERENCES servicios(id)

);

**4. Desarrollar funcionalidades:**

* **Conexión a la base de datos**:

php

Copiar código

<?php

try {

$pdo = new PDO("mysql:host=localhost;dbname=peluqueria", "root", "");

$pdo->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

} catch (PDOException $e) {

die("Error al conectar: " . $e->getMessage());

}

?>

* **CRUD para servicios**:

php

Copiar código

// Insertar servicio

$stmt = $pdo->prepare("INSERT INTO servicios (nombre, precio) VALUES (:nombre, :precio)");

$stmt->execute(['nombre' => 'Corte de cabello', 'precio' => 15.00]);

**5. Diseñar la interfaz:**

* Usa HTML5, CSS3 y Bootstrap para crear una UI atractiva.
* Ejemplo:

html

Copiar código

<form action="reservar\_cita.php" method="POST">

<label for="cliente">Cliente:</label>

<select name="cliente" id="cliente">

<!-- Opciones generadas dinámicamente -->

</select>

<label for="servicio">Servicio:</label>

<select name="servicio" id="servicio">

<!-- Opciones generadas dinámicamente -->

</select>

<label for="fecha">Fecha:</label>

<input type="date" name="fecha" required>

<label for="hora">Hora:</label>

<input type="time" name="hora" required>

<button type="submit">Reservar</button>

</form>

**6. Validación y seguridad:**

* **Validación del lado del cliente** con JavaScript.
* **Sanitización de datos** para evitar inyecciones SQL.

**7. Pruebas y despliegue:**

* Prueba cada funcionalidad.
* Publica la aplicación en un hosting compatible con PHP.