

## Business Requirements Document (BRD)

### A. Tujuan Proyek

Deskripsi Proyek: Proyek ini bertujuan untuk mengelola pembayaran siswa di sebuah institusi pendidikan. Aplikasi ini mencakup pendaftaran siswa, catatan pembayaran, dan manajemen status pembayaran.

Tujuan: Memudahkan manajemen pembayaran, memungkinkan pelaporan keuangan, dan menyediakan transparansi terkait pembayaran siswa.

### B. Skop Proyek

#### 1. Fitur Utama:

- Pendaftaran siswa.
- Rekaman jenis pembayaran (SPP, buku, seragam, dll.).
- Pengelolaan status pembayaran (lunas, menunggu, belum dibayar).

#### 2. Output:

- Tabel students untuk menyimpan data siswa.
- Tabel payments untuk menyimpan data pembayaran siswa.

#### 3. Tabel:

- Students (id, name, class, timestamps).
- payments (id, student\_id, payment\_type, amount, due\_date, status, timestamps)

### C. Fitur dan Fungsionalitas:

#### 1. Fitur Pengelolaan Siswa:

- Pendaftaran Siswa: Fitur untuk menambahkan, memodifikasi, dan menghapus informasi siswa (name dan class).
- Lihat Informasi Siswa: Pengguna dapat melihat daftar siswa yang terdaftar.
- Pencarian Siswa: Fitur pencarian berdasarkan nama atau kelas.

#### 2. Fitur Pengelolaan Pembayaran:

- Menambahkan Pembayaran: Pengguna dapat menambahkan pembayaran seperti SPP, buku, seragam, dll.
- Melihat Riwayat Pembayaran: Pengguna dapat memeriksa detail pembayaran siswa, termasuk jenis pembayaran, jumlah, dan status.
- Mengedit Status Pembayaran: Pengguna dapat memperbarui status pembayaran menjadi "lunas", "menunggu", atau "belum dibayar".
- Melihat Riwayat Pembayaran Siswa berdasarkan ID siswa.
- Laporan Pembayaran: Fitur untuk membuat laporan terkait pembayaran siswa, mencakup jumlah pembayaran, dan status pembayaran.

#### 3. Proses:

- Proses Pendaftaran Siswa:
- Masukan: Informasi siswa (nama, kelas) dari form pendaftaran.
- Proses: Menyimpan informasi tersebut ke tabel students di database.

- Keluaran: Data siswa tersimpan di tabel students.
4. Proses Pembayaran:
- Masukan: Informasi pembayaran (student\_id, payment\_type, amount, due\_date, status) dari form pembayaran.
  - Proses: Menyimpan informasi tersebut ke tabel payments di database.
  - Keluaran: Data pembayaran tersimpan di tabel payments

D. Kebutuhan Fungsional:

1. Validasi Data:

- Pastikan input valid berdasarkan format yang telah ditentukan (misalnya, jenis pembayaran, format angka).
- Validasi status pembayaran validasi untuk lunas, menunggu, dan belum dibayar.

2. Hak Akses:

- Pengguna dengan peran super\_admin dapat menambahkan, mengedit, dan menghapus data.
- Pengguna dengan peran admin hanya dapat menampilkan informasi terkait siswa dan pembayaran.

E. Kebutuhan Non-Fungsional:

1. Performa:

- Sistem harus dapat menangani pengelolaan data siswa dan pembayaran secara efisien.
- Query database harus diproses dalam waktu singkat (misalnya, pencarian dan laporan).

2. Keamanan:

- Pastikan semua operasi CRUD diotorisasi berdasarkan peran pengguna.
- Data sensitif (misalnya, status pembayaran) dilindungi dengan izin yang sesuai.

F. Skema Database:

Tabel students: "id", "name", "class", "created\_at", "updated\_at".

Tabel payments: "id", "student\_id", "payment\_type", "amount", "due\_date", "status", "created\_at", "updated\_at".