**Bab III**

**ANALISIS DAN PERANCANGAN**

Pengembangan perangkat lunak ini menggunakan metode *waterfall*, yang terdiri atas tahap studi literatur, analisis data, perancangan, implementasi, evaluasi, dan pemeliharaan. Pada bab ini akan dijelaskan secara rinci dua tahapan pengembangan, yaitu analisis dan perancangan.

**3.1. Analisis**

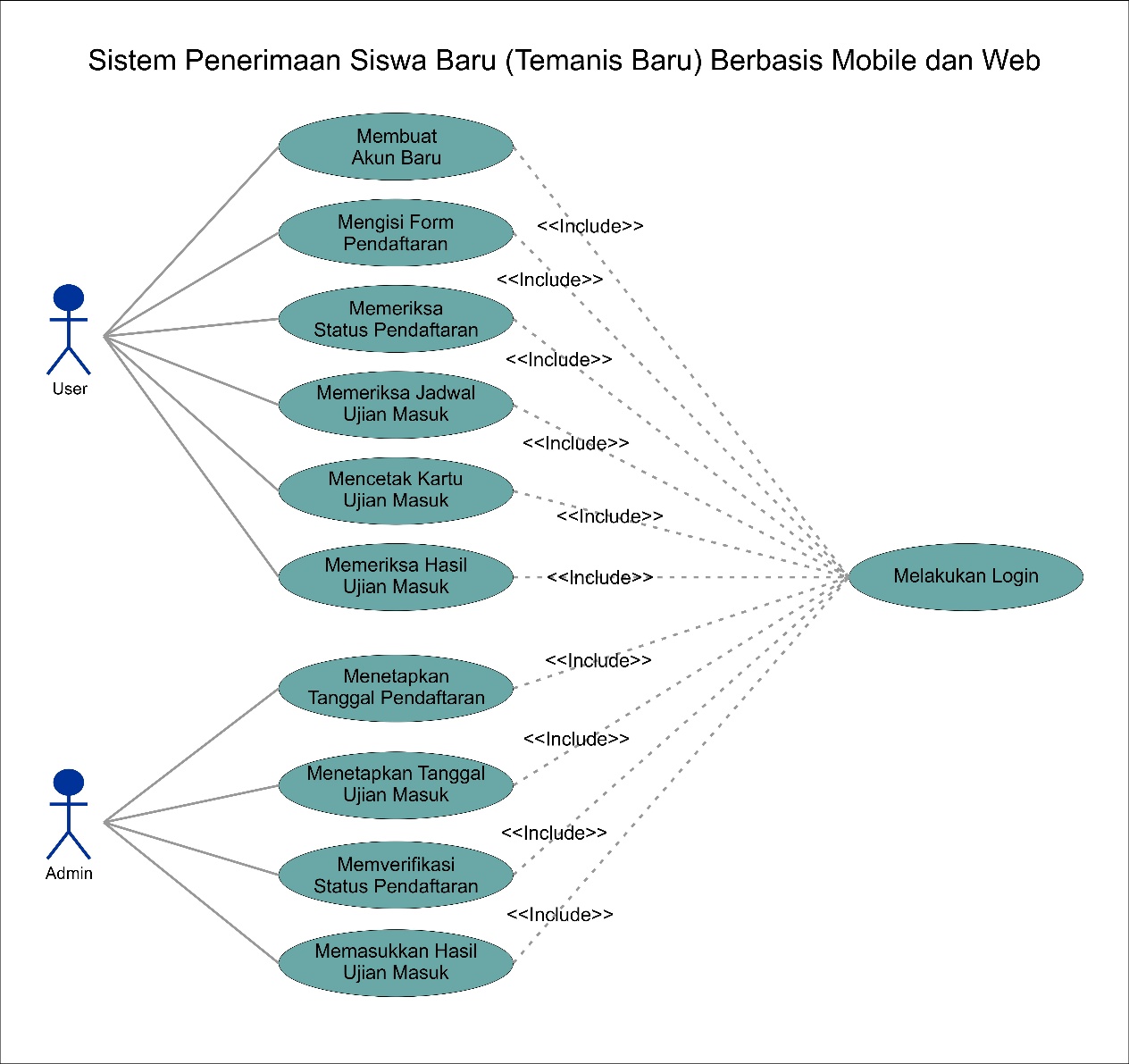
Pada subbab ini akan dijelaskan tahapan analisis, yaitu analisis kebutuhan. Berikut adalah penjelasan masing-masing analisis tersebut.

**3.1.1. Analisis Kebutuhan**

Analisis kebutuhan merupakan tahapan penting dalam pengembangan sistem untuk mengetahui secara detail sistem yang ingin dikembangkan. Kebutuhan sistem akan ditentukan oleh pemilik dan pengguna sistem. Analisis kebutuhan dibagi menjadi dua, yaitu analisis kebutuhan fungsional yang merupakan spesifikasi inti mengenai hal-hal yang bisa dilakukan oleh sistem dan kebutuhan non fungsional yang merupakan komponen pendukung pada sistem.

**3.1.1.1. Analisis Kebutuhan Fungsional**

Analisis kebutuhan fungsional dapat digambarkan dengan menggunakan *Use Case Diagram.*



**Gambar 3.1 Use Case Diagram**

Ada 2 aktor yang terlibat dalam sistem, yaitu *user* dan *admin*. Tabel 3.1 adalah definisi 2 aktor yang terlibat dalam sistem.

**Tabel 3.1 Definisi Aktor**

|  |  |  |
| --- | --- | --- |
| **No.** | **Aktor** | **Deskripsi** |
| 1. | *User* | Aktor yang melakukan pendaftaran ke sekolah |
| 2. | *Admin* | Aktor yang memverifikasi pendaftaran dari *user* |

*Use case* menyatakan fungsi lengkap yang dilakukan oleh aktor dan menghasilkan nilai ke aktor serta menggambarkan di mana level *user* dan respon.

Tabel 3.2 menyatakan definisi *use case* pada Gambar 3.1.

**Tabel 3.2 Definisi *Use Case***

|  |  |  |
| --- | --- | --- |
| **No.** | **Use Case** | **Definisi** |
| 1. | Melakukan *login* | Melakukan *login* untuk masuk ke dalam sistem |
| 2. | Membuat akun baru | Membuat akun untuk dapat melakukan *login* |
| 3. | Mengisi *form* pendaftaran | Mengisi *form* pendaftaran terkait informasi data diri *user* |
| 4. | Memeriksa status pendaftaran | Melakukan pemeriksaan status form pendaftaran yang belum diverifikasi |
| 5. | Memeriksa jadwal ujian masuk | Melakukan pemeriksaan jadwal kapan *user* akan mengikuti ujian masuk |
| 6. | Mencetak kartu ujian masuk | Melakukan pencetakan kartu ujian masuk sebagai syarat mengikuti ujian masuk |
| 7. | Memeriksa hasil ujian masuk | Melakukan pemeriksaan hasil ujian masuk terkait lulus atau tidak |
| 8. | Menetapkan tanggal pendaftaran | Melakukan penetapan tanggal buka dan tutup pendaftaran |
| 9. | Menetapkan tanggal ujian masuk | Melakukan penetapan tanggal ujian masuk untuk *user* |
| 10. | Memverifikasi status pendaftaran | Melakukan verifikasi terhadap status form pendaftaran *user* |
| 11. | Memasukkan hasil ujian masuk | Melakukan pemasukkan hasil ujian masuk terkait kelulusan *user* |

Selanjutnya adalah membuat skenario (*flow of event)* yang menggambarkan urutan interaksi aktor dengan sistem dari *use case* pada Gambar 3.4.

**Tabel 3.3 Skenario *use case* melakukan *login***

|  |  |
| --- | --- |
| **Identifikasi** | |
| **Nomor** | 1 |
| **Nama** | Melakukan *login* |
| **Tujuan** | Untuk melakukan *login* ke dalam sistem |
| **Deskripsi** | |
| **Aktor** | *User* |
| **Skenario Utama** | |
| **Kondisi Awal** | Tampilan halaman *login* |
| **Aksi Aktor** | **Reaksi Sistem** |
| Mengisi *username & password user* | Sistem akan menampilkan kotak *username & password* yang harus diisi oleh *user* untuk masuk ke dalam sistem |
| Mencentang tombol “I AM NOT A ROBOT” | Sistem akan menampilkan 9 pilihan gambar yang harus dipilih *user* |
| Menekan tombol *login* | Sistem akan memproses dan melakukan pengecekan pada *database* |
| **Kondisi Akhir** | Tampilan halaman *user* |

**Tabel 3.4 Skenario *use case* membuat akun baru**

|  |  |
| --- | --- |
| **Identifikasi** | |
| **Nomor** | 2 |
| **Nama** | Membuat akun baru |
| **Tujuan** | Untuk melakukan pembuatan akun baru |
| **Deskripsi** | |
| **Aktor** | *User* |
| **Skenario Utama** | |
| **Kondisi Awal** | Tampilan halaman *register* |
| **Aksi Aktor** | **Reaksi Sistem** |
| Mengisi *email, password & reenter password* | Sistem akan menampilkan kotak informasi yang harus dilengkapi oleh *user* |
| Menekan tombol *register* | Sistem akan memproses dan menyimpan data *user* ke dalam *database* |
| **Kondisi Akhir** | Tampilan halaman setelah *login user* |

**Tabel 3.5 Skenario *use case* mengisi form pendaftaran**

|  |  |
| --- | --- |
| **Identifikasi** | |
| **Nomor** | 3 |
| **Nama** | Mengisi *form* pendaftaran |
| **Tujuan** | Untuk melakukan pendaftaran ke sekolah |
| **Deskripsi** | |
| **Aktor** | *User* |
| **Skenario Utama** | |
| **Kondisi Awal** | Tampilan halaman *form* pendaftaran |
| **Aksi Aktor** | **Reaksi Sistem** |
| Mengisi informasi data diri | Sistem akan menampilkan kotak berisi identitas diri yang harus dilengkapi oleh *user* |
| Mengisi informasi *history* sekolah | Sistem akan menampilkan kotak berisi data *history* sekolah sebelumnya yang harus dilengkapi oleh *user* |
| Mengupload dokumen yang bersangkutan | Sistem akan menampilkan kotak dimana *user* dapat mengupload dokumen yang dibutuhkan |
| Menekan tombol *save* | Sistem akan memproses dan menyimpan data-data yang telah diisi/diupload oleh *user* |
| Memeriksa status pendaftaran | Sistem akan menampilkan status siswa apakah *verified, pending, passed, failed* |
| **Kondisi Akhir** | Tampilan halaman *form* pendaftaran |

**Tabel 3.6 Skenario *use case* mencetak kartu ujian**

|  |  |
| --- | --- |
| **Identifikasi** | |
| **Nomor** | 4 |
| **Nama** | Mencetak kartu ujian |
| **Tujuan** | Untuk melakukan pencetakan kartu ujian masuk |
| **Deskripsi** | |
| **Aktor** | *User* |
| **Skenario Utama** | |
| **Kondisi Awal** | Tampilan halaman cetak kartu ujian |
| **Aksi Aktor** | **Reaksi Sistem** |
| Mengklik tombol cetak | Sistem akan mencetak kartu ujian siswa |
| **Kondisi Akhir** | Tampilan halaman cetak kartu ujian |