**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**

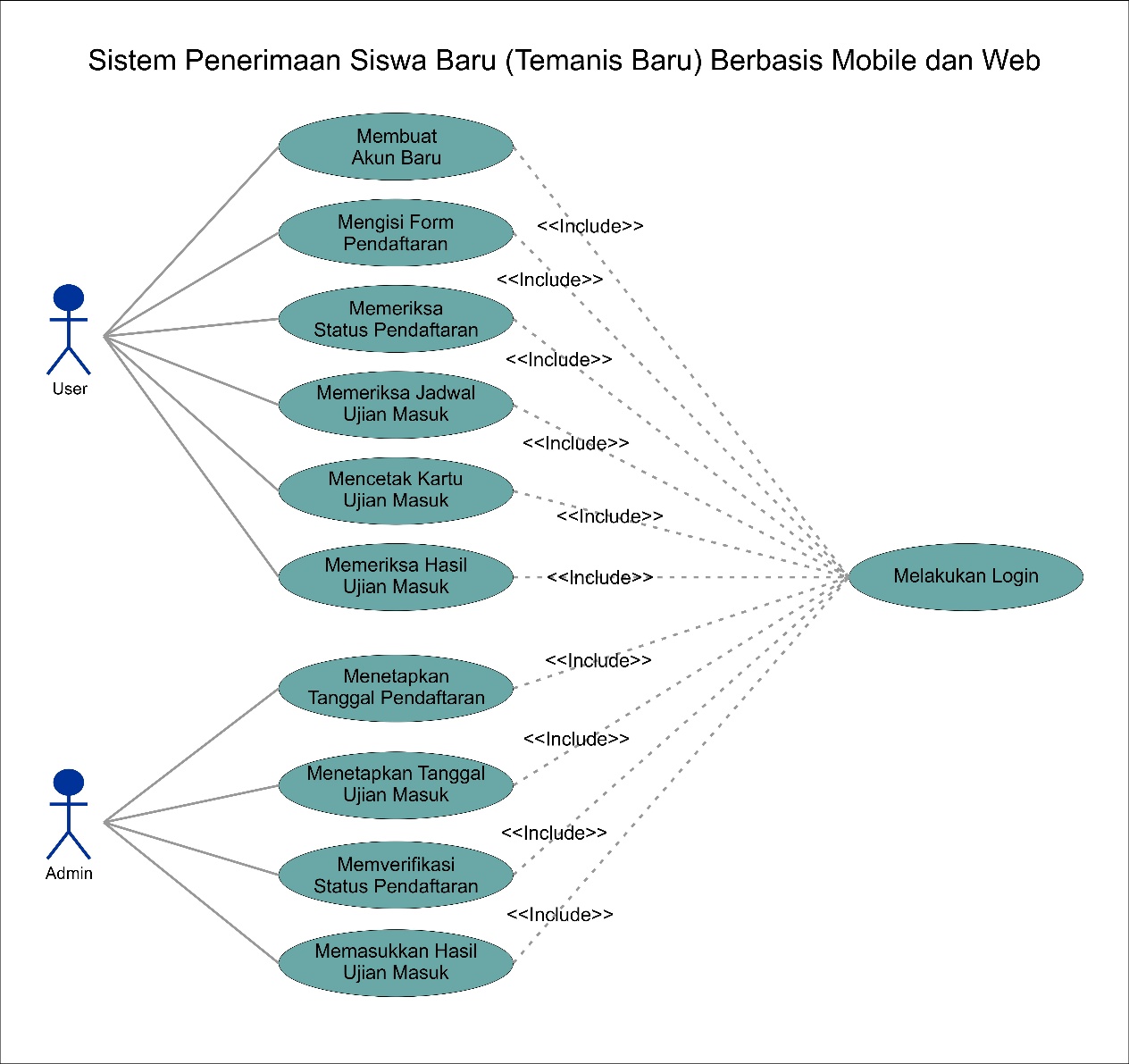
Pada subbab ini akan dijelaskan tahapan analisis, yaitu analisis kebutuhan. Berikut adalah penjelasan masing-masing analisis tersebut.

**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 calon siswa dan admin. Tabel 3.1 adalah definisi 2 aktor yang terlibat dalam sistem.

**Tabel 3.1 Definisi Aktor**

|  |  |  |
| --- | --- | --- |
| **No.** | **Aktor** | **Deskripsi** |
| 1. | Calon siswa | Aktor yang melakukan pendaftaran ke sekolah |
| 2. | *Admin* | Aktor yang memverifikasi pendaftaran dari calon siswa |

*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 *register* | Melakukan pembuatan akun baru agar terdaftar dalam sistem |
| 2. | Melakukan *login* | Masuk ke dalam sistem agar dapat melakukan pendaftaran |
| 3. | Mengisi *form* pendaftaran | Melakukan pengisian *form* pendaftaran yang terdiri dari data diri calon siswa, *history* sekolah dan *upload* dokumen |
| 4. | Mengisi data diri | Melakukan pengisian terhadap data diri dari calon siswa yang bersangkutan |
| 5. | Mengisi data *history* sekolah | Melakukan pengisian terhadap data dari sekolah sebelumnya |
| 6. | Meng-*upload* dokumen | Melakukan *upload* dokumen yang dibutuhkan untuk proses pendaftaran |
| 7. | Memeriksa status pendaftaran | Melakukan pemeriksaan terhadap status pendaftaran |
| 8. | Memeriksa jadwal ujian saringan masuk | Melakukan pemeriksaan jadwal untuk mengikuti ujian saringan masuk |
| 9. | Mencetak kartu ujian masuk | Melakukan pencetakan kartu ujian masuk sebagai syarat mengikuti ujian |
| 10. | Menetapkan tanggal pendaftaran | Melakukan penetapan tanggal buka dan tutup pendaftaran |
| 11. | Memverifikasi status pendaftaran | Melakukan verifikasi terhadap status *form* pendaftaran calon siswa |
| 12. | Memasukkan hasil ujian masuk | Melakukan *input* hasil ujian masuk terkait kelulusan calon siswa |
| 13. | Menetapkan tanggal ujian saringan masuk | Melakukan penetapan tanggal ujian saringan masuk untuk calon siswa |

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* membuat akun baru**

|  |  |
| --- | --- |
| **Identifikasi** | |
| **Nomor** | 1 |
| **Nama** | Membuat akun baru |
| **Tujuan** | Untuk melakukan pembuatan akun baru agar terdaftar dalam sistem |
| **Deskripsi** | |
| **Aktor** | Calon siswa |
| **Skenario Utama** | |
| **Kondisi Awal** | Tampilan halaman *register* |
| **Aksi Aktor** | **Reaksi Sistem** |
| Mengisi *username, password & reenter password* | Sistem akan menampilkan kotak *username, password & reenter password* yang harus diisi oleh calon siswa agar dapat terdaftar dalam sistem |
| Mencentang tombol “I AM NOT A ROBOT” | Sistem akan menampilkan 9 pilihan gambar yang harus dipilih calon siswa sesuai dengan permintaan sistem |
| Menekan tombol *register* | Sistem akan memproses dan menyimpan data calon siswa ke dalam *database* |
| **Kondisi Akhir** | Tampilan halaman *login* |

**Tabel 3.4 Skenario *use case* melakukan *login***

|  |  |
| --- | --- |
| **Identifikasi** | |
| **Nomor** | 2 |
| **Nama** | Melakukan *login* |
| **Tujuan** | Untuk masuk ke dalam sistem |
| **Deskripsi** | |
| **Aktor** | Calon siswa |
| **Skenario Utama** | |
| **Kondisi Awal** | Tampilan halaman login |
| **Aksi Aktor** | **Reaksi Sistem** |
| Mengisi *email & password* | Sistem akan menampilkan kotak *email & password* yang harus diisi dengan benar agar dapat masuk ke dalam sistem |
| Mencentang tombol “I AM NOT A ROBOT” | Sistem akan menampilkan 9 pilihan gambar yang harus dipilih calon siswa sesuai dengan permintaan sistem |
| Menekan tombol *login* | Sistem akan memproses dan melakukan pengecekan pada *database* |
| **Kondisi Akhir** | Tampilan halaman beranda calon siswa |

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

|  |  |
| --- | --- |
| **Identifikasi** | |
| **Nomor** | 3 |
| **Nama** | Mengisi *form* pendaftaran |
| **Tujuan** | Untuk melakukan pengisian *form* pendaftaran |
| **Deskripsi** | |
| **Aktor** | Calon siswa |
| **Skenario Utama** | |
| **Kondisi Awal** | Tampilan halaman *form* pendaftaran |
| **Aksi Aktor** | **Reaksi Sistem** |
| Mengisi data diri | Sistem akan menampilkan kotak berisi identitas data diri yang harus dilengkapi oleh calon siswa |
| Mengisi data *history* sekolah | Sistem akan menampilkan kotak berisi data *history* sekolah sebelumnya yang harus dilengkapi oleh calon siswa |
| Meng-*upload* dokumen | Sistem akan menampilkan kotak dimana *user* dapat meng-*upload* dokumen yang dibutuhkan |
| Menekan tombol *save* | Sistem akan memproses dan menyimpan data-data yang telah diisi/di-*upload* oleh calon siswa |
| 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* memeriksa jadwal ujian saringan masuk**

|  |  |
| --- | --- |
| **Identifikasi** | |
| **Nomor** | 4 |
| **Nama** | Memeriksa jadwal ujian saringan masuk |
| **Tujuan** | Untuk melakukan pemeriksaan jadwal untuk mengikuti ujian saringan masuk |
| **Deskripsi** | |
| **Aktor** | Calon siswa |
| **Skenario Utama** | |
| **Kondisi Awal** | Tampilan halaman ujian saringan masuk |
| **Aksi Aktor** | **Reaksi Sistem** |
| Memeriksa jadwal ujian masuk | Sistem akan menampilkan tanggal untuk jadwal ujian masuk |
| Mengklik tombol cetak | Sistem akan mencetak kartu ujian siswa |
| **Kondisi Akhir** | Tampilan halaman ujian saringan masuk |

**Tabel 3.7 Skenario *use case* menetapkan tanggal pendaftaran**

|  |  |
| --- | --- |
| **Identifikasi** | |
| **Nomor** | 5 |
| **Nama** | Menetapkan tanggal pendaftaran |
| **Tujuan** | Melakukan penetapan tanggal buka dan tutup pendaftaran |
| **Deskripsi** | |
| **Aktor** | *Admin* |
| **Skenario Utama** | |
| **Kondisi Awal** | Tampilan halaman buka/tutup pendaftaran |
| **Aksi Aktor** | **Reaksi Sistem** |
| Menetapkan tanggal buka pendaftaran | Sistem akan menampilkan kalender yang akan dipilih admin sebagai tanggal pembukaan pendaftaran |
| Menetapkan tanggap tutup pendaftaran | Sistem akan menampilkan kalender yang akan dipilih admin sebagai batas tanggal pendaftaran |
| **Kondisi Akhir** | Tampilan halaman buka/tutup pendaftaran |

**Tabel 3.9 Skenario *use case* memverifikasi status pendaftaran**

|  |  |
| --- | --- |
| **Identifikasi** | |
| **Nomor** | 5 |
| **Nama** | Memverifikasi status pendaftaran |
| **Tujuan** | Melakukan verifikasi terhadap status form pendaftaran calon siswa |
| **Deskripsi** | |
| **Aktor** | *Admin* |
| **Skenario Utama** | |
| **Kondisi Awal** | Tampilan halaman verifikasi pendaftar |
| **Aksi Aktor** | **Reaksi Sistem** |
| Mengklik nama calon siswa untuk diverifikasi | Sistem akan menampilkan data calon siswa beserta dokumen yang telah diupload dan admin akan memverifikasinya |
| Memasukkan hasil ujian masuk dengan mengklik *pass / fail* | Sistem akan mengubah status murid sesuai dengan pilihan admin |
| **Kondisi Akhir** | Tampilan halaman verifikasi pendaftar |

**Tabel 3.10 Skenario *use case* menetapkan tanggal ujian saringan masuk**

|  |  |
| --- | --- |
| **Identifikasi** | |
| **Nomor** | 6 |
| **Nama** | Menetapkan tanggal ujian saringan masuk |
| **Tujuan** | Melakukan penetapan tanggal ujian saringan masuk untuk calon siswa |
| **Deskripsi** | |
| **Aktor** | *Admin* |
| **Skenario Utama** | |
| **Kondisi Awal** | Tampilan halaman ujian saringan masuk |
| **Aksi Aktor** | **Reaksi Sistem** |
| Menetapkan tanggal ujian saringan masuk | Sistem akan menampilkan kalender yang akan dipilih admin sebagai tanggal ujian saringan masuk |
| **Kondisi Akhir** | Tampilan halaman ujian saringan masuk |

**3.1.1.2. Analisis Kebutuhan Non Fungsional**

Analisis kebutuhan non fungsional dijabarkan dengan menggunakan analisis PIECES (*Performance, Information, Economics, Control, Efficiency, Service*).

Berikut adalah penjabaran dari analisis yang digunakan:

1. Performance

Sistem dirancang sedemikian rupa sehingga setiap data maupun tampilan akan diproses secara cepat.

2. Information

Sistem memberikan informasi yang lengkap bagi calon siswa mulai dari awal pendaftaran hingga hasil kelulusan.

3. Economy

Dalam segi ekonomi, calon siswa dan adminhanya membutuhkan *browser* dan *internet* untuk dapat mengakses sitem.

4. Control

Sistem menawarkan kemudahan bagi admin sehingga dapat dengan mudah mengkontrol dan memproses setiap calon siswa yang melakukan pendaftaran

5. Efficiency

Pendaftaran dapat dilakukan secara efisien baik melalui *web* maupun *mobile* sehingga dapat digunakan kapan saja dan dimana saja.

6. Service

Sistem menawarkan fitur-fitur sederhana dan mudah dipakai sehingga calon siswa maupun admin dapat dengan mudah dan cepat melakukan kegiatannya.

**3.2. Perancangan**

Tahapan perancangan akan merincikan gambaran sistem sebelum memasuki tahap implementasi. Perancangan terbagi menjadi 2 bagian, yaitu: