PENGEMBANGAN SISTEM APLIKASI PENCARIAN DAN PEMESANAN GURU LES PRIVAT DI KOTA MALANG

SKRIPSI

Disusun oleh:

Elsa Akhmal Rohmianti

NIM: 155150201111097



PROGRAM STUDI TEKNIK INFORMATIKA

JURUSAN TEKNIK INFORMATIKA

FAKULTAS ILMU KOMPUTER

UNIVERSITAS BRAWIJAYA

MALANG

2019

DAFTAR ISI

[PENGEMBANGAN SISTEM APLIKASI PENCARIAN DAN PEMESANAN GURU LES PRIVAT DI KOTA MALANG i](#_Toc5655831)

[DAFTAR ISI ii](#_Toc5655832)

[DAFTAR TABEL v](#_Toc5655833)

[DAFTAR GAMBAR vi](#_Toc5655834)

[DAFTAR LAMPIRAN vii](#_Toc5655835)

[BAB 1 pendahuluan 1](#_Toc5655836)

[**1.1** **Latar Belakang** 1](#_Toc5655837)

[**1.2** **Rumusan Masalah** 2](#_Toc5655838)

[**1.3** **Tujuan** 2](#_Toc5655839)

[**1.4** **Manfaat** 2](#_Toc5655840)

[**1.4.1** **Bagi Pencari Les/Murid** 3](#_Toc5655841)

[**1.4.2** **Bagi Guru/Pengajar** 3](#_Toc5655842)

[**1.5** **Batasan Masalah** 3](#_Toc5655843)

[**1.6** **Sistematika Pembahasan** 3](#_Toc5655844)

[**BAB 2 LANDASAN KEPUSTAKAAN** 5](#_Toc5655845)

[**2.1** **Kajian Pustaka** 5](#_Toc5655846)

[**2.1.1** **Pengertian Les Privat** 7](#_Toc5655847)

[**2.1.2** **Manfaat Les Privat** 8](#_Toc5655848)

[**2.3** **Rekayasa Perangkat Lunak** 8](#_Toc5655849)

[**2.4** **Software Development Life Cycle(SDLC)** 8](#_Toc5655850)

[**2.5** **Metode *Waterfall*** 8](#_Toc5655851)

[**2.6** **Pendekatan Object Oriented** 10](#_Toc5655852)

[**2.6.1** **Use Case Diagram** 10](#_Toc5655853)

[**2.6.2** **Use Case Skenario** 11](#_Toc5655881)

[**2.6.3** **Sequence Diagram** 12](#_Toc5655882)

[**2.6.4** **Class Diagram** 13](#_Toc5655904)

[**2.7** **Pengujian** 13](#_Toc5655905)

[**2.7.1** **White Box** 13](#_Toc5655906)

[**2.7.2** **Black Box** 14](#_Toc5655907)

[***2.8*** ***Analityc Hierarchy Process(AHP)*** 15](#_Toc5655908)

[**2.9** **Pustaka Pendukung** 15](#_Toc5655909)

[**2.6.1** **Hyper Text Markup Language** 15](#_Toc5655910)

[**2.6.2** **Cascanding Style Sheet (CSS)** 15](#_Toc5655911)

[**2.6.3** **Javascript** 16](#_Toc5655912)

[**2.6.4** **PHP** 16](#_Toc5655913)

[**2.6.5** **Model View Controller (MVC)** 16](#_Toc5655914)

[**2.6.6** **MySql** 17](#_Toc5655915)

[**2.6.7** **Codeigniter** 17](#_Toc5655916)

[**BAB 3 METODOLOGI** **Error! Bookmark not defined.**](#_Toc5655917)

[**3.1** **Studi Literatur** 18](#_Toc5655918)

[**3.2** **Analisis Kebutuhan** 19](#_Toc5655919)

[**3.3** **Perancangan** 19](#_Toc5655920)

[**3.4** **Implementasi** 20](#_Toc5655921)

[**3.5** **Pengujian** 20](#_Toc5655922)

[**3.6** **Kesimpulan dan Saran** 20](#_Toc5655923)

[**BAB 4 REKAYASA KEBUTUHAN** **Error! Bookmark not defined.**](#_Toc5655924)

[**4.1** **Gambaran Umum Sistem** 21](#_Toc5655925)

[**4.2** **Analisis Kebutuhan Perangkat Lunak** 21](#_Toc5655926)

[**4.2.1** **Identifikasi Aktor** 21](#_Toc5655927)

[**4.2.2** **Analisis dan Spesifikasi Kebutuhan** 23](#_Toc5655928)

[**4.2.3** **Kebutuhan Fungsional Murid** 27](#_Toc5655929)

[**4.2.4** **Kebutuhan Fungsional Guru Les Privat** 29](#_Toc5655930)

[**4.2.5** **Kebutuhan Fungsional Admin** 31](#_Toc5655931)

[**4.3** **Pemodelan Kebutuhan Object Oriented** 32](#_Toc5655932)

[**4.3.1** **Use Case Diagram** 33](#_Toc5655933)

[**4.3.2** **Use Case Skenario** 35](#_Toc5655934)

[**BAB 5 PERANCANGAN DAN IMPLEMENTASI** **Error! Bookmark not defined.**](#_Toc5655935)

[**5.1** **Perancangan Sistem** 56](#_Toc5655936)

[**5.1.1** **Perancangan Class Diagram** 56](#_Toc5655937)

[**5.1.2** **Perancangan Sequence Diagram** 57](#_Toc5655938)

[daftar pustaka 99](#_Toc5655939)

DAFTAR TABEL

[Tabel 2.1 Kajian Pustaka 5](#_Toc5476564)

[Tabel 2.2 Pengertian Tahap Metode Waterfall 9](#_Toc5476565)

[Tabel 2.3 Komponen Use Case Diagram 10](#_Toc5476566)

[Tabel 2.4 Komponen Sequence Diagram 12](#_Toc5476567)

DAFTAR GAMBAR

[Gambar 2.1.1 Metode Waterfall 9](#_Toc5656767)

[Gambar 2.1.2 Different control Flow Graph Notations 14](#_Toc5656768)

[Gambar 2.1.3 *Black Box* Diagram 14](#_Toc5656769)

[Gambar 0.1.1 Identifikasi Aktor 22](#_Toc5656770)

[Gambar 5.1 Perancangan Class Diagram **Error! Bookmark not defined.**](#_Toc5656771)

DAFTAR LAMPIRAN

# pendahuluan

* 1. **Latar Belakang**

Les Privat adalah bantuan yang diberikan oleh pembimbing kepada siswa yang mengalami kesulitan belajar (Sularso, Kokotiasa, & Paramedica, 2015). Les privat biasanya dilakukan oleh guru atau tentor atau bisa juga dilakukan oleh pemberi jasa les. Les privat biasanya dilakukan kepada satu orang murid atau tidak lebih dari 4 orang, dengan ini metode pembelajaran yang dilakukan akan maksimal. Dalam kegiatan belajarnya les privat tetap menggunakan kurikulum dari sekolah sebagai acuan. Les privat biasanya dilakukan di rumah siswa ataupun bisa dilakukan di rumah pengajar. Dalam kegiatan les ini cocok dan bermanfaat untuk menekuni atau mendalami suatu bidang mata pelajaran tertentu.

Dalam kegiatan les private ini tentunya harus memilih calon pengajar atau pembimbing yang sesuai kriteria dan keahlian pengajar. Dengan memilih pengajar dengan keahlian yang tepat maka akan mendapatkan manfaat dan menguasai mata pelajaran yang dianggap sulit sebelumnya. Murid ataupun orang tua harus memeriksa kualitas pengajar dengan melihat sisi akademik dan dari segi pengalaman yang dimiliki serta dari sisi sikap dan tata krama saat mengajar. Selain itu disisi lain guru juga harus dapat menanamkan nilai-nilai moral kepada murid dan bagaimana menanamkan sikap yang baik. Dari sini maka metode pembelajaran akan lebih optimal.

Untuk menguasai bidang mata pelajaran tertentu atau yang diinginkan les privat sangat mempunyai manfaat yang besar bagi murid. Sedikitnya jumlah murid membuat belajar menjadi lebih intensif. Selain itu waktu yang fleksibel dalam melakukan kegiatan mengajar karena bisa dilakukan kapan saja diluar jam sekolah. Guru les privat juga dapat membantu murid dalam menetapkan tujuan dan sasaran belajar. Selain itu cara penyampaian materi dalam les privat lebih rinci sehingga mata pelajaran yang diingin dikuasai lebih mudah untuk dicerna.

Di kota Malang sendiri biasanya murid atau orang tua murid dalam mencari guru les privat hanya mendapatkan info dari kerabat atau teman yang mengetahui info guru les ataupun mereka telah melakukan kegiatan les privat sebelumnya. Sehingga terkadang guru yang dicari tidak sesuai dengan apa yang dibutuhkan antara murid yang satu dengan yang lain. Karena berdasarkan hasil penelitian lebih dari 50% murid menginginkan guru yang memiliki kemampuan dalam mengajar dan menguasai materi, sekitar 23% menginginkan guru yang mempunyai pengalaman yang lebih seperti pengalaman mengajar sehingga dapat menguasai materi dan dapat mengetahui metode belajar yang tepat untuk murid serta sisanya murid juga menginginkan karakter guru yang baik dan dapat dicontoh.

Berdasarkan permasalahan yang penulis sampaikan, maka penulis ingin membuat sebuah website pencarian dan pemesanan guru les privat di Kota Malang. Sehingga penulis mengangkat sebuah judul “ PENGEMBANGAN SISTEM APLIKASI PENCARIAN DAN PEMESANAN GURU LES PRIVAT DI KOTA MALANG ” dengan harapan sistem tersebut dapat membantu dan mempermudah murid dan orang tua dalam mencari guru les privat yang sesuai dengan keinginan. Selain itu sistem tersebut dapat membantu guru les atau tentor dalam mempromosikan dirinya sebagai pengajar sehingga mendapatkan pekerjaan sebagai guru les yang sesuai dengan potensi yang dimilikinya.

* 1. **Rumusan Masalah**

Berdasarkan latar belakang tersebut, maka ditarik beberapa rumusan masalah sebagai berikut:

1. Bagaimana analisis kebutuhan pengguna dan perancangan sistem pencarian dan pemesanan guru les privat di Kota Malang.
2. Bagaimana rancangan sistem aplikasi yang digunakan dalam pencarian dan pemesanan guru les privat di Kota Malang ?
3. Bagaimana implementasi sistem tersebut dari hasil analisis dan perancangan ?
4. Bagaimana hasil pengujian fungsional dan non fungsional sistem tersebut?
   1. **Tujuan**

Berdasarkan rumusan masalah yang telah ditulis sebelumnya, maka muncul beberapa tujuan penulisan sebagai berikut:

1. Menyusun dan menganalisis sistem untuk sistem aplikasi dalam pencarian dan pemesanan guru les privat di Kota Malang.
2. Menganalisis kebutuhan dan merancang sistem yang sesuai kebutuhan pengguna dalam pencarian dan pemesanan guru les privat.
3. Mengimplementasikan sistem dari hasil analisis dan perancangan sistem tersebut.
4. Menguji sistem perangkat lunak tersebut secara fungsional dan non fungsional.
   1. **Manfaat**

Dengan adanya website tersebut maka diharapkan dapat memberikan manfaat untuk beberapa pihak. Berikut ini merupakan beberapa manfaat yang dapat diambil yaitu:

* + 1. **Bagi Pencari Les/Murid**

1. Dapat memberikan kemudahan dalam mencari guru les privat yang sesuai dengan keinginan.
2. Mempermudah dalam memesan guru atau pengajar yang sesuai dengan bidang mata pelajaran yang diinginkan.
   * 1. **Bagi Guru/Pengajar**

Bagi guru atau pengajar yaitu dapat membantu dalam mencari pekerjaan sebagai pengajar les khususnya les privat. Serta dapat membantu dalam mencari penghailan tambahan.

* 1. **Batasan Masalah**

Berikut ini merupakan beberapa batasan masalah yang dapat diambil yaitu sebagai berikut:

1. Sistem hanya digunakan untuk mencari dan memesan guru les privat di Kota Malang.
2. Sistem hanya menampilkan guru yang berdomisili di Kota Malang.
3. Sistem hanya dapat membantu siswa dalam mendapatkan guru les privat mulai dari jenjang Sekolah Dasar hingga Sekolah Menengah Atas.
4. Sistem menggunakan metode Analityc Hierarchy Process(AHP)
   1. **Sistematika Pembahasan**

Penelitian ini disusun dalam beberapa bab pembahasan sebagai acuan mendasar, adapun rancangan sistematika pembahasan seperti berikut :

**Bab I Pendahuluan**, pada bab pendahuluan ini dimana terdapat gambaran umum dari penelitian yang terdiri dari: latar belakang yang digunakan untuk mendeskripsikan masalah apa yang ada pada penelitian, rumusan masalah, tujuan, manfaat, batasan masalah dan sistematika pembahasan dari sistem aplikasi pencarian dan pemesanan guru les privat di kota Malang.

**Bab II Landasan Kepustakaan**, berisi uraian dan pembahasan tentang teori, konsep, model, metode, sistem atau analisis literatur ilmiah yang berkaitan sistem aplikasi pencarian dan pemesanan guru les privat di kota Malang..

**Bab III Metodologi Penelitian**, berisi mengenai metode atau teknik penelitian yang diterapkan dalam sistem aplikasi pencarian dan pemesanan guru les privat di kota Malang. Dalam bab ini menjelaskan bagaimana solusi yang tepat untuk menyelesaikan suatu masalah.

**Bab IV Rekayasa Kebutuhan**, pada bab ini memaparkan mengenai kebutuhan apa saja yang dibutuhkan dalam pembuatan sistem aplikasi pencarian dan pemesanan guru les privat di kota Malang berdasarkan hasil wawancara dan observasi.

**Bab V Perancangan dan Implementasi**, berisi mengenai rancangan sistem yang akan dibuat dan bagaimana mengimplementasikan ke dalam suatu kode untuk menciptakan suatu sistem yang tepat sesuai dengan kebutuhan.

**Bab VI Pengujian**, berisi mengenai pemaparan hasil pengujian perangkat lunak yang telah dibuat sebelumnya pada tahap implementasi. Pengujian sistem dengan menggunakan *white box testing* dan *black box testing.*

**Bab VII Penutup**, pada bab ini berisi mengenai kesimpulan dan saran, kesimpulan dibuat untuk menjawab dari rumusan masalah yang telah dibuat. Dan saran dibuat agar nantinya sistem aplikasi pencarian dan pemesanan guru les privat di kota Malang dapat digunakan dalam pengembangan yang lebih lanjut.

# LANDASAN kEPUSTAKAAN

1. **Kajian Pustaka**

Penelitian sebelumnya telah dilakukan oleh Putra, A. K., Nyoto, R. D., & Pratiwi, H. S. (2017) dengan judul Rancang Bangun Aplikasi Marketplace Penyedia Jasa Les Privat Di Kota Pontianak Berbasis Web. Dalam penelitian tersebut bersumber dari beberapa masalah yaitu pencarian yang cukup lama dalam menemukan guru les privat, kesesuaian lokasi dan waktu antara guru les dan murid dan tidak bisa mengetahui tentang potensi dan prestasi guru les tersebut. Sehingga tidak dapat menemukan guru les terbaik. Sistem yang dibangun menggunakan metode waterfall, dengan menggunakan bahasa pemrograman PHP, Framework Laravel dan database MySQL. Pada hasil penelitian ini dalam membangun sebuah sistem terdapat tahapan seperti analisis kebutuhan, perancangan, implementasi dan pengujian. Sebelumnya dengan menguraikan beberapa komponen yang menunjang marketplace tersebut. Sehingga nantinya akan dapat memudahkan masyarakat untuk menemukan guru les privat dengan kriteria yang diinginkan.

Penelitian sebelumnya juga telah dilakukan oleh Nike Essyana & Anita Qoiriah(2016) dengan judul “Sistem Pencarian Guru Les Privat Berbasis Web”. Penelitian tersebut didasarkan pada masalah yaitu bergesernya peran orang tua sebagai pengasuh anak karena kesibukan orang tua karena rutinitas sehari-hari. Kebutuhan cara belajar anak yang berbeda-beda sehingga juga memerlukan pengajaran khusus seperti les privat. Namun dari masalah tersebut juga tidak diimbangi dengan kemudahan pencarian guru les privat. Sehingga dari penelitian tersebut menciptakan sebuah sistem pencarian guru les privat bernama ‘Go-Teacher’ yang dapat membantu dalam mencari guru les privat. Dalam penggunaannya menggunakan fitur dari Google Maps Api untuk menampilkan guru di area murid. Serta dengan fitur geocoding dan geolocation dapat menghasilkan sistem yang dapat mengetahui lokasi guru di area rumah murid.

Tabel 2.1 Kajian Pustaka

|  |  |  |  |
| --- | --- | --- | --- |
| Judul | Metode yang Digunakan | Permasalahan | Hasil Penelitian |
| Rancang Bangun Aplikasi Marketplace Penyedia Jasa Les Privat Di Kota Pontianak Berbasis Web. Putra, A. K., Nyoto, R. D., & Pratiwi, H. S. (2017). | Sistem yang dibangun menggunakan metode waterfall, dengan menggunakan bahasa pemrograman PHP, Framework Laravel dan database MySQL. | Dalam penelitian tersebut bersumber dari beberapa masalah yaitu pencarian yang cukup lama dalam menemukan guru les privat, kesesuaian lokasi dan waktu antara guru les dan murid dan tidak bisa mengetahui tentang potensi dan prestasi guru les tersebut. Sehingga tidak dapat menemukan guru les terbaik. | Pada hasil penelitian ini dalam membangun sebuah sistem terdapat tahapan seperti analisis kebutuhan, perancangan, implementasi dan pengujian. Sebelumnya dengan menguraikan beberapa komponen yang menunjang marketplace tersebut. Sehingga  nantinya akan dapat memudahkan masyarakat untuk menemukan guru les privat dengan kriteria yang diinginkan. |
| ‘Go-Teacher’  Sistem Penarian Guru Les Privat Berbasis Web. Esyana, Nike & Qoiriah, Anita.(2016) | Sistem tersebut dibangun dengan menggunakan metode GeoCoding dan Geolocation. Dan memanfaatkan fitur dari Google Maps | Penelitian tersebut didasarkan pada masalah yaitu bergesernya peran orang tua sebagai pengasuh anak karena kesibukan orang tua karena rutinitas sehari-hari. Kebutuhan cara belajar anak yang berbeda-beda sehingga juga memerlukan pengajaran khusus seperti les privat. Namun dari masalah tersebut juga tidak diimbangi dengan kemudahan pencarian guru les privat. | Dalam penelitian tersebut dilakukan analisa sistem, analisa manajemen sumberdaya manusia, kemudian mendesain sistem dan implementasi serta hasil dari sistem yang dapat mencari guru les yang dapat diakses langsung dengan guru dan murid yang dapat diakses oleh admin. |

Bepacu dari penelitian sebelumnya, di Kota Malang belum terdapat suatu sistem yang digunakan dalam mencari dan memesan guru les privat. Maka, pengembangan suatu sistem untuk memesan dan mencari guru les privat sekiranya dibutuhkan dan dapat membantu masyarakat terutama para siswa di Kota Malang. Sistem tersebut akan membantu para masyarakat terutama siswa di Kota Malang dalam membantu dan mencari dan memesan guru les privat yang sesuai dengan potensi atau keahlian guru yang diinginkan. Dalam pengembangan aplikasi tersebut maka digunakan sutu metode yaitu dengan menggunakan Metode *Waterfall*. Pengembangan perangkat lunak dengan menggunakan metode Waterfall memiliki kelebihan yaitu membuat kualitas *software* tetap terjaga karena pengembangannya yang terstruktur.

1. **Definisi Les privat**

Di bawah ini terdapat pengertian mengenai les privat dan juga manfaat yang kita dapatkan apabila mengikuti kegiatan les privat, yaitu sebagai berikut:

* + 1. **Pengertian Les Privat**

Les Privat adalah bantuan yang diberikan oleh pembimbing kepada siswa yang mengalami kesulitan belajar (Sularso, Kokotiasa, & Paramedica, 2015). Les privat biasanya dilakukan oleh guru atau tentor atau bisa juga dilakukan oleh pemberi jasa les. Les privat biasanya dilakukan kepada satu orang murid atau tidak lebih dari 4 orang, dengan ini metode pembelajaran yang dilakukan akan maksimal. Dalam kegiatan belajarnya les privat tetap menggunakan kurikulum dari sekolah sebagai acuan. Les privat biasanya dilakukan di rumah siswa ataupun bisa dilakukan di rumah pengajar. Dalam kegiatan les ini cocok dan bermanfaat untuk menekuni atau mendalami suatu bidang mata pelajaran tertentu.

* + 1. **Manfaat Les Privat**

Dalam les privat sendiri tentunya memiliki banyak manfaat, yaitu antara lain

: (Sularso, Kokotiasa, & Paramedica, 2015)

* Murid dengan cepat menguasai bidang studi atau mata pelajaran yang dianggap sulit
* Lebih konsentrasi dalam belajar
* Memiliki kesempatan untuk mengulang mata pelajaran yang diajarkan di sekolah sehingga lebih paham lagi dengan pelajaran tersebut.

1. **Rekayasa Perangkat Lunak**

Rekayasa Perangkat Lunak atau yang disebut dengan Software Enggineering dipopulerkan pada tahun 1968 pada Software Enggineering Conference oleh NATO. Rekayasa Perangkat Lunak adalah suatu ilmu yang digunakan dalam pembuataan atau pengembangan perangkat lunak. Rekayasa Perangkat Lunak adalah suatu bidang profesi yang mendalami bagaimana cara pengembangan perangkat lunak mulai dari pembuatan, pengembangan hingga pemeliharaan (Simarmata, 2010). Dalam pengembangan yang dimaksud yaitu aspek-aspek pembuatan perangkat lunak mulai dari tahap awal hingga pemeliharaan perangkat lunak. Tahapan-tahapan tersebut yaitu mengenai analisis kebutuhan, desain, pemrograman atau pengkodean, pengjian dan pemeliharaan sistem. Tujuan Rekayasa Perangkat Lunak sendiri yaitu agar dapat menghasilkan kinerja yang baik, waktu penyelesaian yang cepat dan menghemat biaya.

1. **Software Development Life Cycle(SDLC)**

Software Development Life Cycle (SDLC) merupakan sebuah siklus dalam pengembangan perangkat lunak. SDLC merupakan sebuah proses atau langkah-langkah yang digunakan oleh seorang *system analyst* dan *programmer* dalam membangun sistem (Dwanoko, 2016). SDLC merupakan sebuah pola dalam pengembangan perangkat lunak dari sebuah tahapan. Tahap atau langkah- langkah yang dimaksud yaitu: tahap analisis kebutuhan, perancangan atau desain, implementasi, pengujian dan *maintenance*. Dalam SDLC terdapat beberapa model atau metode yang digunakan dalam pengembangannya, yaitu: Waterfall, Prototype, Rapid Application Development(RAD), Spirall dan Agile.

1. **Metode *Waterfall***

Metode Waterfall merupakan suatu proses pengembangan perangkat lunak tradisional yang pada umumnya digunakan dalam proyek pembangunan perangkat lunak. Metode yang digunakan dalam penelitian ini adalah dengan menggunakan metode Waterfall. Metode waterfall adalah salah satu metode dari SDLC yaitu metode yang digunakan dalam pengembangan perangkat lunak. Dalam pengembangan metode waterfall ini mempunyai beberapa tahapan yaitu *requirement*(analisis kebutuhan), *design*(desain atau perancanga), implementasi(pengkodean), *Testing*(pengujian) dan *Maintenance*. Metode waterfall memiliki kelebihan yaitu membuat kualitas *software* tetap terjaga karena pengembangannya yang terstruktur (Fahrurrozi & SN, 2012).

**Analisis**

**Perancangan**

**Implementasi**

**Pengujian**

**Maintenance**

Gambar 2.2.1 Metode Waterfall

(Sumber: Fahrurrozi & SN, 2012)

Tabel 2.2 Pengertian Tahap Metode Waterfall

|  |  |
| --- | --- |
| Tahap | Pengertian |
| Analisis | Requirement atau tahap analisis kebutuhan ini merupakan tahap awal dimana pengembang mencari tau kebutuhan apa saja yang dibutuhkan pengguna. |
| Perancangan | Design atau tahap perancangan merupakan tahap dimana pengembang akan merancang sistem secara keseluruhan sesua dengan kebutuhan penggguna. |
| Implementasi | Pada tahapan ini pengembang akan melakukan pengkodean sistem dimana dari perancangan diimplementasikan dengan kode-kode sehingga menjadi sebuah sistem. |
| Pengujian | Testing atau pengujian adalah langkah dimana sistem akan diuji apakah sesuai dengan kebutuhan atau tidak. |
| *Maintenance* | Tahap ini adalah tahap pemeliaraan sistem, pemeliharaan termasuk dalam memperbaiki kesalahan pada sistem. |

1. **Pendekatan Object Oriented**

Pendekatan Object Oriented merupakan salah satu pendekatan dalam pengembangan perangkat lunak. Pendekatan ini merupakan suatu pendekatan yang berorientasi pada objek sehingga dekomposisi permasalahan berdasarkan objek-objek. Pada pendekatan object oriented objek-objek saling bekerja sama dan berinteraksi satu sama lain. Dalam pendekatan ini terdapat istilah OOA atau *Object Oriented Analysis* yaitu analisis kebutuhan dengan menggunakan pendekatan object oriented. Dan pada perancangan sistemnya dengan *Object Oriented Design*(OOD). Pendekatan berorientasi objek berpusat pada teknik yang mengacu pada pemodelan objek (Whitten, Bentley, & Dittman, 2004).

1. **Use Case Diagram**

Use case merupakan suatu diagram yang digunakan untuk menjelaskan hubungan antara aktor dengan sistem. Use Case menggambarkan fungsi sistem dari sudut pandang pengguna, sebuah cara dan terminologi yang dipahami (Whitten, Bentley, & Dittman, 2004). Use case diagram menunjukkan bagaimana hubungan antara sistem dengan lingkungan luar. Use Case merupakan suatu diagram yang terdiri dari beberapa komponen yaitu: aktor, use case dan relasi. Aktor merupakan sesorang yang berinteraksi dengan sistem. sedangkan use case merupakan elemen atau gambaran fungsional sistem. Serta relasi-relasinya seperti: association, generalization dan aggregation. Relasi-relasi tersebut memiliki fungsi seperti yang ada pada tabel 2.3.

Tabel 2.3 Komponen Use Case Diagram

|  |  |  |
| --- | --- | --- |
| Simbol | Nama | Fungsi |
|  | Actor | Seseorang yang menggunakan atau yang berinteraksi dengan sistem. |
|  | Use Case | Gambaran fungsional sistem |
|  | Assosciation | Menghubungkan aktor dengan use case |
|  | Generalization | Spesialisasi dari elemen yang lain |
|  | Dependency | Relasi antar elemen yang bergantungan |
|  | Aggregation | Hubungan antar elemen dimana elemen berisi elemen lainnya |
| << include >> | Include | Kondisi dimana use case bergantung dengan use case yang lain |
| << extends >> | Extends | Tambahan fungsional dari use case yang lain |

1. **Use Case Skenario**

Use Case Skenario adalah sebuah urutan dari keadaan yang akan terjadi ketika aktor menggunakan sebuah sistem. Dalam tahap analisis kebutuhan use case skenario sangat penting digunakan agar mengetahui alur sistem yang jelas. Dalam penjabaran use case skenario digunakan use case diagram yang telah dibuat dan kemudian dibuat use case skenario dari setiap use case yang ada. Use case skenario sangat memudahkan dalam membuat sistem pada tahap selanjutnya.

1. **Sequence Diagram**

Sequence diagram merupakan sebuah diagram yang ada pada Unified Modelling Language(UML). Sequence Diagram yaitu menggambarkan bagaimana objek saling berinteraksi satu sama lain melalui eksekusi dari suatu use case atau operasi (Whitten, Benley, & Dittman, 2001). Sequence Diagram adalah gambaran dari interaksi atar objek yang satu dengan yang lain. Sequence diagram menunjukkan bagaimana objek-objek tersebut saling berinteraksi dan saling mengirim pesan. Komponen-komponen yang ada dalam sequence diagram yaitu: aktor, bundary, controller, entity, lifeline dan line message. Pesan-pesan yang dikirimkan akan di eksekusi oleh setiap objek.

Tabel 2.4 Komponen Sequence Diagram

|  |  |  |
| --- | --- | --- |
| Simbol | Nama | Fungsi |
|  | Aktor | Merupakan seseorang yang mengunakan sistem |
|  | Boundary | Berhubungan dengan aktor. Berisi tampilan suatu sistem |
|  | Controller | Penghubung antara boundary dan entity. Merupakan Logic dari sebuah sistem |
|  | Entity | Berhubungan dengan data |
|  | Lifeline | Menggambarkan pesan dimulai dan diakhiri |
|  | Line Message | Perintah untuk mengirim pesan |

1. **Class Diagram**

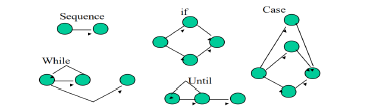
Class Diagram merupakan salah satu diagram dari Unified Modelling Language(UML). Class diagram menggambarkan kelas-kelas yang ada pada sebuah sistem. Class Diagram menggambarkan struktur objek sistem dimana mereka menunjukkan objek dari suatu kelas dan kelas-kelas tersebut berhubungan satu sama lain (Whitten, Benley, & Dittman, 2001) Class Diagram sendiri terdiri dari nama kelas, atribut, operasi dan relasi. Atribut memiliki sifat-sifat diantaranya: private, protected dan public. Dalam atribut yang bersifat private maka penulisan tribut dalam class diagram diawali dengan tanda”-“. Sedangkan protected dalam penulisannya diawali dengan tanda pagar “#”. Dan public dalam penulisannya diawali dengan tanda”+”. Serta relasi dalam class diagram yaitu: association, generalization dan dependency.

1. **Pengujian**

Dalam membangun sebuah perangkat lunak tentunya juga harus melakukan tahap pengujian guna untuk mengetahui apakah sistem layak digunakan atau tidak serta untuk mengetahui kesalahan yang ada pada sistem. Di bawah ini merupakan cara atau metode yang digunakan dalam pengujian perangkat lunak yaitu:

1. **White Box**

Pengujian *White Box* atau *white box testing* merupakan pengujian perangkat lunak dimana pengujian didasarkan pada alur logika kode program. Metode pengujian perangkat lunak ini digunakan dalam pengecekan detail atau mekanisme suatu sistem atau aliran data program dengan teknik secara struktural. Dengan menggunakan pengujian *white box* maka dapat digunakan dalam menemukan suatu kesalahan program atau error pada kode program. Dalam *white box testing* terdapat 2 teknik, yaitu *Static White Box Testing* dan *Struktural White Box Testing*. Dalam Static White Box Testing dalam pengujiannya dibagi menjadi *Desk Checking, Code Walktrhough*, dan *Formal Inspections*. Sedangkan pada *Struktural White Box Testing* dibagi menjadi *Control flow/ Coverage testing*, *Basic Path Testing, Loop Testing* dan *Data Flow Testing.* (Nidhra & Dondeti, 2012)



Gambar 2.2.2 Different control Flow Graph Notations

(Sumber: Nidhra & Dondeti, 2012)

1. **Black Box**

Pengujian *black box* atau *black box testing* merupakan sebuah pengujian perangkat lunak dimana pengujiannya berdasarkan fungsional yang ada pada sistem. Pengujian ini digunakan untuk mengetahui struktur luar sistem apakah sesuai dengan kebutuhan tanpa mengenal struktur internal sistem (Nidhra & Dondeti, 2012). *Black box* testing hanya digunakan untuk menguji tampilan dan alur program dari tampak luar. Tujuan dari *black box* testing untuk mengetahui letak kesalahan *interface*, kesalahan kinerja sistem dan kesalahan struktur data. Sehingga pengujian ini lebih mudah dan efisien untuk diterapkan dalam pengujian perangkat lunak.

**Black Box**

Gambar 2.2.3 *Black Box* Diagram

1. ***Analityc Hierarchy Process(AHP)***

AHP merupakan salah satu metode dalam sistem pendukung keputusan yang digunakan dalam menyelesaikan masalah dan mengambil suatu keputusan. Metode AHP dikembangkan oleh Thomas L. Saaty yang merupakan hierarki fungsional dimana input utama berupa persepsi manusia. Sehingga dengan adanya hierarki maka masalah yang tidak terstruktur akan dipecah dan dijadikan alternatif dan menghasilkan suatu solusi (Kristiyanti, Sugiharto, & W, 2013).

Tahap-tahap dalam pengambilan keputusan dalam metode AHP pada dasarnya sebagai berikut:

Mendefinisikan masalah dan menentukan solusi

Membuat struktur hierarki

Membentuk matriks perbandingan

Menormalisasi data

Menghitung nilai eigen vektor

Menguji konsistensi hierarki

1. **Pustaka Pendukung** 
   * 1. **Hyper Text Markup Language**

Hypertext Markup Language (HTML) adalah bahasa yang digunakan dalam membuat suatu web. HTML merupakan bahasa yang dikenali web browser guna menampilkan informasi yang menarik (Oktavian, 2010). HTML merupakan standart internet yang dikendalikan oleh Word Wide Web Consortium (W3C). HTML digunakan untuk membuat tampilan atau interface pada halaman suatu web. Dengan menggunakana bahasa markup, html memiliki banyak fungsi dan struktur dasar yang digunakan dalam membangun sebuah tampilan website. HTML memiliki banyak simbol yang digunakan yang dimasukkan kedalam sebuah file atau dokumen. Dengan adanya html maka dapat dengan mudah membuat tampilan website yang mudah dibaca dan dipahami oleh pengguna.

* + 1. **Cascanding Style Sheet (CSS)**

Cascanding Style Sheet(CSS) merupakan suatu bahasa program yang digunakan untuk mendesain tampilan html. CSS berhubungan dengan HTML karena dengan adanya CSS maka dapat mengubah tampilan yang lebih baik dan menarik. CSS memberikan peran yang sangat besar karena dapat digunakan untuk mengelola tampilan seperti text, warna, latar belakang, tata letak dan gambar (Meyer, 2004). Bahasa atau program css sendiri dapat menjadi satu dengan file html atau dapat menjadi file terpisah. Keuntungn menggunakan CSS yaitu dapat memngubah atau mengatur tampilan dengan cepat dan mudah.

* + 1. **Javascript**

Javascript merupakan bahasa pemrograman yang berbentuk dari sekumpulan script. Javascript merupakan bahasa yang ditafsirkan, yang artinya bahwa javascript tidak dikompilasi ke dalam program yang dieksekusi terpisah namun javascript tertanam di dalam file html (Gandy & Stobart, 2005). Javascript dapat menyempurnakan tampilan website. Dengan menggunakan javascript maka tampilan website akan terlihat dinamis dan lebih interaktif. Javascript merupakan bahasa pemrograman yang bersifat *Client Side Programming* *Language* yaitu bahasa pemrograman yang pemrosesannya dilakukan oleh client. Javascript memiliki beberapa kelebihan, yaitu mudah dipelajari, ringan dan mudah dikolaborasikan.

* + 1. **PHP**

PHP atau *Hypertext Processor* merupakan bahasa pemrograman yang digunakan dalam pengembangan sebuah situs web. PHP digunakan dalam web yang bersifat dinamis. Dimana kode programnya disisipkan ke dalam dokumen HTML, sehinga disebut dengan Scripting Language. PHP juga disebut dengan bahasa pemrograman server side karena PHP diproses pada komputer server (Haryana, 2008). PHP dapat berjalan di hampir semua web server seperti IIS, apache, Xitami, Lighttpd, dan lain-lain. PHP juga sering digunakan untuk membuat program atau sistem informasi, seperti sistem managemen, keuangan, akademik, dan lain-lain.

* + 1. **Model View Controller (MVC)**

Model View Controller merupakan sebuah metode yang digunakan dalam membuat suatu perangkat lunak dengan membagi menjadi Model, View dan Controller. MVC merupakan pemisah tampilan pengguna dari kendali asupan pengguna dan model informasi yang mendasarinya (Nugroho, 2009). Model merupakan bagian kode aplikasi yang behubungan dengan data atau basis data. View merupakan bagian yang mengatur tampilanweb dan berhubungan dengan pengguna. Controller yaitu bagian yang menghubungkan antara Model dan View, yang berisi perintah-perintah untuk memproses data. Dengan MVC maka peragkat lunak akan lebih mudah untuk dirawat dan dikembangkan.

Controller

View

Model

**Gambar 2.4** **Model View Controller**

* + 1. **MySql**

Menurut *Wahana Komputer* (2010) MySql pertamakali dikemukakan oleh Michael Widenius yang merupakan seorang programmer database. MySql merupakan database yang dapat mengirim dan menerima data yang cepat dan multiuser. MySql adalah database yang open source dan cukup populer dibanding dengan database yang lainnya. MySql database server yaitu RDBMS atau Relasional Database Management System dimana server dapat menampung data yang cukup besar namun tidak menuntut resource yang besar. Terdapat beberapa keuntungan dengan menggunakan MySql yaitu, Database MySql mengerti bahasa SQL (Structured Query Language), dapat menyimpan data yang sangat besar, mempunyai performa yang simpel dan server MySql dapat diakses dalam satu waktu.

* + 1. **Codeigniter**

Codeigniter adalah framework atau kerangka kerja yang digunakan dalam membangun suatu aplikasi berbasis website. Framework merupakan kumpulan intruksi atau prosedur serta fungsi dengan tujuan tertentu yang telah tersedia dan siap digunakan sehingga dapat membantu dalam mempercepat pekerjaan. Framework Codeigniter bersifat open source dan tidak berbayar. Codeigniter digunakan untuk para developer yang digunakan untuk mempermudah dalam pembuatan dan tidak perlu membuat kode program dari awal. Codeigniter memiliki dokumentasi yang sangat lengkap serta ringan dan cepat (Hustinawati, Himawan, & Latifah, 2014).

# METODOLOGI

Pada bab ini dijelaskan mengeai metodologi yang akan digunakan dalam “Pengembangan Sistem Aplikasi Pencarian dan Pemesanan Guru Les Privat di Kota Malang”. Tahapan yang dilakukan yaitu studi literatur, analisis kebutuhan, perancangan, implementasi, pengujian serta pengambilan kesimpulan dan saran. Berikut merupakan diagram alir yang menjelaskan mengenai metodologi yang digunakan seperti pada gambar 3.1.

Gambaran umum sistem

Analisis Kebutuhan

Pemodelan Kebutuhan

Class Diagram

Sequence Diagram

Perancangan User Interface

Pengujian White Box

Pengujian Black Box

**Gambar 3.1** **Diagram Alur**

* 1. **Studi Literatur**

Pada tahap ini digunakan untuk mempelajari literatur atau dasar-dasar teori dan kajian pustaka yang digunakan untuk menunjang penulisan yang dilakukan dalam pengembangan sistem aplikasi pencarian dan pemesanan guru les privat di Kota Malang. Pada tahap ini digunakan dalam mencari informasi yang berhubungan dengan penelitian. Sehingga dari informasi-informasi yang didapat akan membantu dan mempermudah dalam melakukan penelitian. Literatur yang digunakan didapat dari jurnal, skripsi dan dokumentasi dari internet.

* 1. **Analisis Kebutuhan**

Pada tahap analisis kebutuhan, dilakukan penggalian kebutuhan atau identifikasi kebutuhan yang dibutuhkan dalam membangun sebuah sistem. Analisis kebutuhan merupakan kebutuhan yang harus dipenuhi dalam membangun sebuah sistem. Metode analisis yang digunakan dalam penelitian ini adalah *Object Oriented Analysis*(OOA). Dalam analisis kebutuhan terdapat beberapa tahapan, yaitu gambaran umum sistem, identifikasi aktor, kebutuhan fungsional dan kebutuhan non fungsional. Dalam analisis kebutuhan terdapat kebutuhan fungsional dan kebutuhan non fungsional. Selain itu langkah selanjutnya dalam analisis kebutuhan yaitu dengan membuat use case diagram. Use case diagram digunakan untuk mengetahui hubungan pengguna dengan sistem.

* 1. **Perancangan**

Perancangan sistem berisi rancangan langkah kerja dari sistem secara menyeluruh. Perancangan sistem digunakan untuk mempermudah implementasi. Dalam merancang suatu sistem berisi gambaran jelas dan menghasilkan rancangan bangun yang lengkap. Dalam membuat model perancangan tersebut menggunakan Object Oriented Design ( OOD) yaitu pemodelan UML(*Unified Modelling Language*). Pemodelan UML merupakan pemodelan yang terdiri dari beberapa diagram, yaitu class diagram dan sequence diagram.

Dalam tahap perancangan perangkat lunak terdapat beberapa tahap perancangan, yaitu perancangan class diagram, sequence diagram dan perancangan antarmuka. Perancangan class diagram digunakan untuk menggambarkan struktur class yang ada pada sistem. Sequence diagram digunakan untuk menunjukan bagaimana pesan yang dikirim antar objek, atau bagaimana interaksi antara objek satu dengan yang lain. Perancangan antamuka merupakan gambaran awal sistem yang nantinya sebagai acuan dalam tahap implementasi.

* 1. **Implementasi**

Implementasi merupakan proses pembuatan aplikasi dengan menggunakan kode program. Tahap implementasi merupakan tahap selanjutnya setelah analisis kebutuhan dan perancangan sehingga dalam implementasinya berpacu dengan analisis kebutuhan dan perancangan. Dalam implementasi membuat program yang dapat menampilkan fitur-fitur yang sesuai dengan kebutuhan fungsional serta kebutuhan non fungsional yang mendukung kinerja sistem. Implementasi perangkat lunak dilakukan dengan menggunakan HTML & CSS, PHP dan javascript. Serta membuat basis data dengan menggunakan MySql.

* 1. **Pengujian**

Pengujian sistem digunakan untuk mengetahui bagaimana kerja sistem yang telah dibuat. Pengujian ini dibuat agar dapat menunjukkan bahwa perangkat lunak telah mampu bekerja sesuai dengan spesifikasi dari kebutuhan yang melandasinya. Selanjutnya melakukan evaluasi terhadap sistem sehingga mengetahui hasil dari sistem yang nantinya dijadikan sebagai kesimpulan. Dalam tahap pengujian menggunakan pengujian *white box* dan *black box*. Pengujian *White Box*, untuk mengetahui cara kerja suatu perangkat lunak secara internal. Pengujian *Black Box*, untuk mengetahui dan memeriksa fungsional dari perangkat lunak serta kinerja sistem.

* 1. **Kesimpulan dan Saran**

Pengambilan kesimpulan dan saran dilakukan pada tahap akhir dari proses penelitian setelah tahapan perancangan, implementasi dan pengujian. Kesimpulan dapat diambil dari kelebihan serta kekurangan sistem yang telah dibuat. Kesimpulan bertujuan untuk menjawab permasalahan yang ada pada rumusan masalah. Saran yang dimaksud untuk memberikan masukan terhadap kekurangan-kekurangan yang ada di dalam sistem. Dan saran yang telah dibuat nantinya dapat sebagai pertimbangan untuk penelitian selanjutnya.

# REKAYASA kEBUTUHAN

1. **Gambaran Umum Sistem**

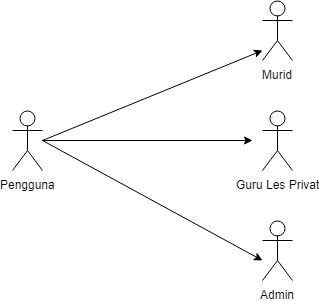
Sistem pencarian dan pemesanan guru les privat dapat disebut dengan BRAINY, yang merupakan sebuah sistem atau perangkat lunak berbasis website yang digunakan untuk mempertemukan antara dua pengguna yaitu murid dan guru les privat. Dimana murid sebagai pencari dan pemesan guru les dapat menemukan guru les privat yang ia inginkan. Dan sistem dapat digunakan oleh para guru les privat dimana sistem digunakan sebagai wadah bagi guru les privat supaya dapat lebih cepat untuk mendapatkan murid. Terdapat tiga pemangku kepentingan dalam sistem, yaitu murid, guru les privat dan admin. Murid berperan dalam mencari dan memesan guru les privat. Guru les privat dapat mendaftarakan dirinya ke sistem supaya terdaftar menjadi member atau salah satu guru les privat di sistem. Sedangkan admin dapat mengetahui guru siapa saja yang mendaftarkan diri di website dan admin dapat menyeleksi dan memvalidasi guru yang benar-benar ingin menjadi salah satu member dari Smart Student.

1. **Analisis Kebutuhan Perangkat Lunak**

Analisis kebutuhan perangkat lunak merupakan tahap yang digunakan untuk mengetahui atau mendapatkan kebutuhan-kebutuhan pada sistem. Dalam tahap analisi kebutuhan ini terdapat beberapa hal yang dilakukan yaitu identifikasi aktor, analisis dan spesifikasi kebutuhan, kebutuhan fungsional murid, kebutuhan fungsional guru les privat dan kebutuhan fungsional admin.

1. **Identifikasi Aktor**

Identifikasi aktor digunakan untuk mengetahui pengguna-pengguna yang terlibat dalam sistem. Terdapat empat aktor yang terlibat ke dalam sistem, dan di bawah ini merupakan gambaran dari identifikasi aktor yang tampak pada Gambar 4.1



Gambar 4.4.1 Identifikasi Aktor

Pada Tabel 4.1 merupakan tabel yang mendeskripsikan mengenai tugas atau peran pada setiap aktornya. Pengguna merupakan aktor yang belum melakukan login atau merupakan pengguna yang umum. Sedangkan murid, guru les privat dan admin adalah aktor yang telah login dan siap menggunakan sistem dengan peran masing-masing.

Tabel 4.1 Identifikasi Aktor

|  |  |  |
| --- | --- | --- |
| **Aktor** | **Spesifikasi** | **Deskripsi** |
| Pengguna | Murid | Murid merupakan aktor yang dapat mencari dan memesan guru. |
| Guru Les Privat | Guru dapat melakukan pendaftaran jika ingin menjadi member guru dengan mengisi form pendaftaran. |
| Admin | Admin dapat mengetahui data murid dan data guru dan admin dapat memvalidasi guru yang ingin bergabung. |

1. **Analisis dan Spesifikasi Kebutuhan**

Pada tabel 4.2 merupakan tabel analisis dan spesifikasi kebutuhan yang ada pada sistem. Dalam tabel tersebut terdapat kode yang merupakan sebuah penomoran kebutuhan, dan kemudian terdapat deskripsi yang dimana dalam deskripsi terdapat kebutuhan dan spesifikasi kebutuhan.

**Tabel 4.2 Analisis dan Spesifikasi Kebutuhan**

|  |  |
| --- | --- |
| **Kode** | **Deskripsi** |
| BRAINY-1-100 | Sistem harus menyediakan halaman untuk login. |
| Spesifikasi Kebutuhan:   * Sistem harus menyediakan menu login untuk pengguna baik untuk murid maupun guru les privat yang terdiri dari username dan password.( BRAINY-1-101) * Sistem harus menyediakan menu daftar untuk melakukan registrasi atau pendaftaran. (BRAINY-1-102) |
| BRAINY-1-200 | Sistem harus menyediakan menu registrasi atau pendaftaran. |
| Spesifikasi Kebutuhan:   * Sistem harus menyediakan halaman untuk registrasi bagi murid yang berisi nama, email, username dan password. (BRAINY-1-201) * Sistem harus menyediakan halaman registrasi bagi guru les privat yang terdiri dari nama, alamat, no.telp, email dan password.( BRAINY-1-202) |
| BRAINY-1-300 | Sistem harus menyediakan fitur untuk pencarian guru serta fitur untuk memesan guru |
| Spesifikasi Kebutuhan:   * Sistem harus menyediakan form yang digunakan untuk mencari guru les privat yang terdiri dari jenjang pendidikan, mata pelajaran, sesi, hari dan jam (BRAINY-1-401) * Sistem harus menyediakan tombol cari untuk mencari guru rekomendasi berdasarkan form yang telah diisi. (BRAINY-1-402) * Sistem harus mampu menampilkan daftar rekomendasi guru les privat yang sesuai dengan permintaan dari murid. (BRAINY-1-403) * Sistem harus menyediakan tombol lihat untuk melihat profil guru les privat rekomendasi. (BRAINY-1-404) * Sistem harus menyediakan tombol pilih untuk melanjutkan ke tahap selanjutnya. (BRAINY-1-405) * Sistem harus menyediakan form permintaan mengajar. (BRAINY-1-406) * Sistem menyediakan tombol kirim untuk mengirim form permintaan mengajar kepada guru les privat yan telat terpilih. (BRAINY-1-407) * Sistem harus menyediakan menu gaji untuk mengetahui perolehan gaji yang telah didapat. (BRAINY-1-408) |
| BRAINY-1-400 | Sistem harus menyediakan dashboard untuk murid |
| Spesifikasi kebutuhan:   * Sistem harus menampilkan profil murid yang menampilkan biodata lengkap dari murid yang disertai foto pribadi. (BRAINY-1-501) * Sistem harus memiliki tombol untuk edit profil. (BRAINY-1-502) * Sistem harus menyediakan menu Permintaan Terkirim yang digunakan sebagai informasi mengenai permintaan belajar yang telah terkirim. (BRAINY-1-503) * Sistem harus menyediakan Permintaan Disetujui yang digunakan untuk mengetahui bahwa permintaan belajar yang diterima oleh guru les privat. (BRAINY-1-504) * Sistem harus menyediakan Permintaan Disetujui yang digunakan untuk mengetahui bahwa permintaan belajar yang diterima oleh guru les privat. (BRAINY-1-505) * Sistem harus menyediakan menu Bimbingan Saat ini yang digunakan untuk mengetahui bimbingan saat ini yang telah berjalan. (BRAINY-1-506) * Sistem harus menyediakan menu Riwayat dimana dapat menampung mengenai riwayat belajar sebelumnya. (BRAINY-1-507) * Sistem harus memiliki fitur hapus riwayat belajar. (BRAINY-1-508) * Sistem harus menyediakan fitur akhiri, ketika ingin mengakhiri kegiatan belajar. (BRAINY-1-509) * Sistem harus dapat menyediakan form untuk memberikan ulasan dan rating. (BRAINY-1-510) * Sistem harus menyediakan fitur untuk mengirim pesan kepada guru les privat. (BRAINY-1-511) * Sistem harus menyediakan fitur upload bukti pembayaran. (BRAINY-1-512) * Sistem harus memiliki fitur logout. (BRAINY-1-513) |
| BRAINY-1-500 | Sistem harus menyediakan dashboard untuk guru les privat |
|  | Spesifikasi Kebutuhan:   * Sistem harus menampilkan menu lamaran pekerjaan. (BRAINY-1-601) * Sistem harus menampilkan status diproses, diterima atau ditolak mengenai permintaan lamaran menjadi guru les privat pada menu. (BRAINY-1-602) * Sistem harus memiliki fungsi untuk mengirim lamaran pekerjaan bagi guru untuk mendaftarkan diri sebagai guru les privat. (BRAINY-1-603) * Sistem harus dapat menampilkan profil guru seperti biodata lengkap dan foto pribadi. (BRAINY-1-604) * Sistem harus menampilkan tombol untuk edit profil. (BRAINY-1-605) * Sistem harus menampilkan menu Daftar Permintaan. (BRAINY-1-606) * Sistem harus menampilkan menu Permintaan Disetujui. (BRAINY-1-607) * Sistem harus menampilkan menu Bimbingan saat ini. (BRAINY-1-608) * Sistem harus menyediakan Riwayat Mengajar. (BRAINY-1-609) * Sistem harus menyediakan fitur untuk menghapus riwayat mengajar. (BRAINY-1-610) * Sistem harus menampilkan status aktif untuk guru yang sedang aktif mengajar dan menerima bimbingan belajar. (BRAINY-1-611) * Sistem harus menyediakan fitur tidak aktif apabila guru tidak sedang menerima bimbingan belajar. (BRAINY-1-612) * Sistem harus menyediakan fitur untuk menyetujui dan menolak permintaan belajar. (BRAINY-1-613) * Sistem harus menyediakan fitur untuk mengirim alasan penolakan. (BRAINY-1-614) * Sistem harus menyediakan halaman untuk melihat gaji yang telah didapatkan. * Sistem harus memiliki fitur logout. (BRAINY-1-615) |
| BRAINY-1-600 | Sistem harus menyediakan halaman utama admin untuk melakukan manajemen data. |
| Spesifikasi Kebutuhan:   * Sistem harus menyediakan halaman untuk melakukan login. ( BRAINY-1-701) * Sistem harus dapat menampilkan daftar guru yang telah menjadi member di BRAINY.( BRAINY-1-702) * Sistem harus memiliki tombol detail untuk melihat profil guru pada halaman daftar guru. ( BRAINY-1-702) * Sistem harus dapat menampilkan halaman daftar permintaan menjadi guru les privat di BRAINY. ( BRAINY-1-703) * Sistem harus memiliki tombol untuk menerima dan menolak guru les privat yang daftar di sistem. ( BRAINY-1-704) * Sistem harus dapat menampilkan bimbingan antar murid dan guru les privat. ( BRAINY-1-705) * Sistem dapat mengetahui status guru yang aktif dan tidak. ( BRAINY-1-706) * Sistem harus menyediakan menu transaksi yang digunakan untuk menyimpan hasil bukti pembayaran yang dilakukan oleh murid. ( BRAINY-1-707) * Sistem harus meyediakan fungsi yang digunakan untuk mengelola gaji guru. ( BRAINY-1-708) * Sistem harus menyediakan menu transaksi yang digunakan untuk melihat hasil transaksi pembayaran yang dilakukan oleh murid. * Sistem harus menyediakakan fitur yang digunakan untuk menerima transaksi pembayaran apabila transaksi tersebut benar dan sah. * Sistem harus menyediakan menu untuk menolak transaksi pembayaran apabila transaksi tersebut tidak benar atau terdapat pemalsuan transaksi. * Sistem harus menyediakan fitur untuk mengambil gaji guru apabila terdapat guru yang ingin menarik atau mengambil gaji yang dimiliki. * Sistem harus memiliki fitur untuk logout. (BRAINY-1-709) |

1. **Kebutuhan Fungsional Murid**

Pada Tabel 4.3 merupakan penjabaran dari kebutuhan-kebutuhan fungsionalitas murid. Terdapat 18 kebutuhan fungsional murid yang nantinya ada pada sistem.

**Tabel 4.3 Kebutuhan Fungsional Murid**

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Kode** | **Nama Fungsi** | **Deskripsi** |
| 1 | BRAINY-2-001 | Login | Fitur ini berfungsi untuk mendapatkan status atau masuk sebagai murid |
| 2 | BRAINY-2-002 | Daftar | Fitur ini berfungsi untuk registrasi ke sistem sebelum melakukan login. |
| 3 | BRAINY-2-003 | Cari Guru | Fitur ini berfungsi untuk melakukan pencarian guru les privat berdasarkan form. |
| 4 | BRAINY-2-004 | Melihat Halaman Rekomendasi Guru | Fitur ini berfungsi untuk menampilkan hasil rekomedasi guru les privat. |
| 5 | BRAINY-2-005 | Melihat Profil Guru | Fitur ini digunakan untuk melihat profil guru les privat sebelum melakukan pemesanan. |
| 6 | BRAINY-2-007 | Memilih Guru | Fitur ini digunakan untuk memilih guru les privat yang diinginkan. |
| 7 | BRAINY-2-008 | Mengirim Permintaan Mengajar | Fitur ini digunakan untuk mengirim permintaan mengajar ke guru les privat yang diinginkan. |
| 8 | BRAINY-2-008 | Melihat Dashboard | Fitur ini digunakan untuk melihat dashboard atau profil pribadi. |
| 9 | BRAINY-2-009 | Melihat Permintaan Terkirim | Fitur ini digunakan untuk melihat permintaan mengajar yang telah terkirim ke guru. |
| 10 | BRAINY-2-010 | Melihat Permintaan Diterima | Fitur ini digunakan untuk melihat permintaan belajar yang telah diterima. |
| 11 | BRAINY-2-011 | Melihat Permintaan Ditolak | Fitur ini digunakan untuk melihat permintaan belajar yang ditolak beserta alasan penolakan. |
| 12 | BRAINY-2-012 | Melihat Bimbingan Belajar Saat Ini | Fitur ini digunakan untuk menampilkan bimbingan belajar saat ini dengan guru yang telah terpilih sebelumnya. |
| 13 | BRAINY-2-013 | Melihat Riwayat Belajar | Fitur ini digunakan untuk menampilkan riwayat bimbingan belajar dengan guru les privat. |
| 14 | BRAINY-2-014 | Menghapus Riwayat | Fitur ini digunakan untuk menghapus riwayat bimbingan belajar. |
| 15 | BRAINY-2-015 | Akhiri | Fitur ini digunakan untuk mengakhiri kegiatan belajar saat ini. |
| 16 | BRAINY-2-016 | Memberikan Rating | Fitur ini digunakan untuk memberikan rating. |
| 17 | BRAINY-2-017 | Mengirim ulasan | Fitur ini digunakan untuk mengirim ulasan. |
| 18 | BRAINY-2-018 | Mengirim Pesan | Fitur ini digunakan untuk mengirim pesan kepada guru les privat. |
| 19 | BRAINY-2-019 | Edit Profil Murid | Fitur ini digunakan untuk mengedit profil pribadi. |
| 20 | BRAINY-2-020 | Upload Bukti Pembayaran | Fitur ini digunakan untuk mengunggah buti pembayaran. |
| 21 | BRAINY-2-021 | Pesan Lagi | Fitur ini digunakan untuk memesan kembali guru les privat yang telah masuk di riwayat pemesanan. |
| 22 | BRAINY-2-022 | Logout | Fitur ini digunakan untuk keluar dari autorisasi murid. |

1. **Kebutuhan Fungsional Guru Les Privat**

Pada Tabel 4.4 merupakan fungsionalitas sistem yang dimiliki oleh guru les privat. Kebutuhan-kebutuhan tersebut dapat diakses oleh guru les privat. Dan dari tabel tersebut terdapat 16 kebutuhan fungsional.

**Tabel 4.4 Kebutuhan Fungsional Guru Les Privat**

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Kode** | **Nama Fungsi** | **Deskripsi** |
| 1 | BRAINY-3-001 | Login | Fitur ini digunakan untuk mendapatkan autorisasi sebagai guru les privat |
| 2 | BRAINY-3-002 | Daftar | Fitur ini digunakan untuk melakukan registrasi sebelum melakukan login. |
| 3 | BRAINY-3-003 | Melihat Dashboard | Fitur ini digunakan untuk melihat dashboard atau profil pribadi. |
| 4 | BRAINY-3-004 | Mengirim Lamaran Pekerjaan | Fitur ini digunakan untuk mengirim lamaran pekerjaan dengan mengisi form yang tersedia. |
| 5 | BRAINY-3-005 | Melihat Lamaran Pekerjaan | Fitur ini digunakan untuk menampilkan menu lamaran pekerjaan yang berisi lamaran pekerjaan sebelumnya. |
| 6 | BRAINY-3-006 | Melihat status lamaran pekerjaan | Fitur ini digunakan untuk melihat status lamaran pekerjaan seperti diproses, diterima atau ditolak. |
| 7 | BRAINY-3-007 | Edit Profil Guru | Fitur ini digunakan untuk mengedit profil pribadi. |
| 8 | BRAINY-3-008 | Melihat Daftar Permintaan | Fitur ini digunakan untuk melihat daftar permintaan bimbingan belajar. |
| 9 | BRAINY-3-009 | Menyetujui Permintaan | Fitur ini digunakan untuk menyetujui permintaan belajar. |
| 10 | BRAINY-3-010 | Menolak Permintaan | Fitur ini digunakan untuk menolak permintaan belajar. |
| 11 | BRAINY-3-011 | Melihat Bimbingan Saat Ini | Fitur ini digunakan untuk melihat bimbingan yang dilakukan saat ini. |
| 12 | BRAINY-3-012 | Melihat Riwayat Mengajar | Fitur ini digunakan untuk melihat riwayat bimbingan mengajar |
| 13 | BRAINY-3-013 | Menghapus Riwayat | Fitur ini digunakan untuk menghapus riwayat mengajar. |
| 14 | BRAINY-3-014 | Mengganti Status Aktif atau Tidak Aktif | Fitur ini digunakan untuk mengganti status aktif apabila aktif mengajar dan tidak aktif ketika tidak bersedia mengajar. |
| 15 | BRAINY-3-015 | Mengirim Alasan Penolakan | Fitur ini digunakan untuk mengirim alasan menolak bimbingan belajar. |
| 16 | BRAINY-3-016 | Mengirim Pesan | Fitur ini dapat digunakan untuk mengirim pesan dengan murid. |
| 17 | BRAINY-3-017 | Melihat Gaji | Fitur ini digunaan untuk melihat gaji yang telah dimiliki oleh guru les privat selama mengajar. |
| 18 | BRAINY-3-018 | Logout | Fitur ini digunakan untuk keluar dari autorisasi guru les privat. |

1. **Kebutuhan Fungsional Admin**

Pada tabel 4.5 merupakan pemaparan mengenai fungsionalitas yang dimiliki oleh admin. Terdapat 12 kebutuhan fungsional yang telah dijabarkan mulai dari login sampai melakukan proses logout.

**Tabel 4.5 Kebutuhan Fungsional Admin**

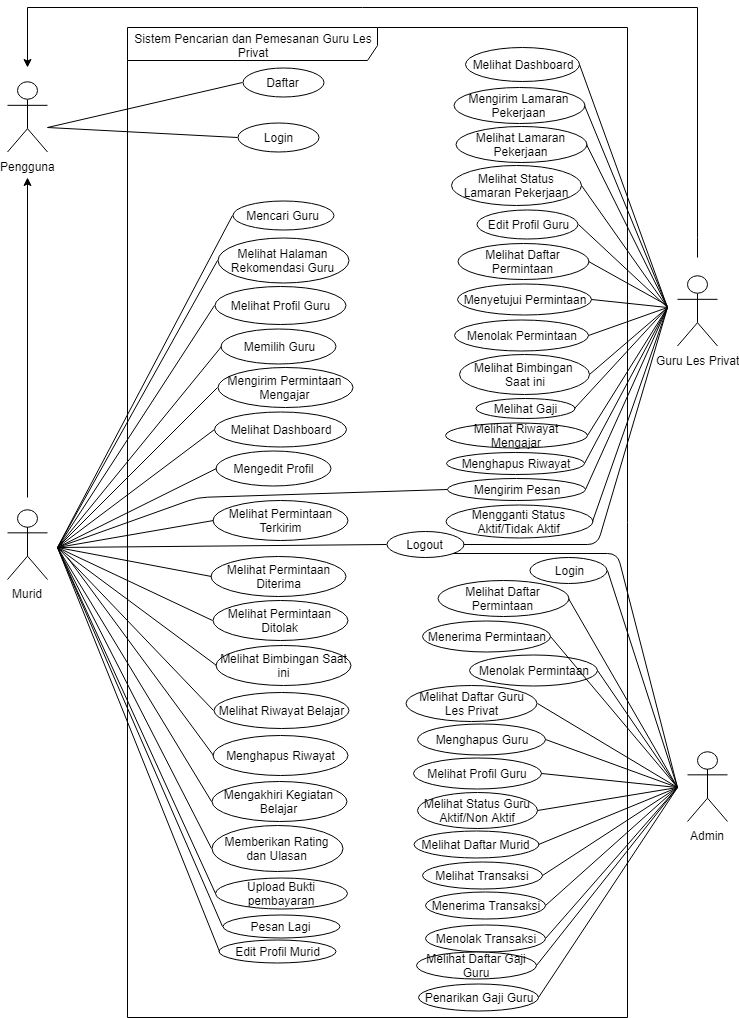
|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Kode** | **Nama Fungsi** | **Deskripsi** |
| 1 | BRAINY-4-001 | Login | Fitur ini digunakan untuk autorisasi sebagai admin. |
| 2 | BRAINY-4-002 | Melihat Daftar Permintaan | Fitur ini digunakan untuk melihat daftar permintaan menjadi guru les privat. |
| 3 | BRAINY-4-003 | Menerima Permintaan | Fitur ini digunakan untuk menyetujui permintaan untuk menjadi member guru les privat. |
| 4 | BRAINY-4-004 | Menolak Permintaan | Fitur ini digunakan untuk menolak permintaan untuk menjadi member guru les privat. |
| 5 | BRAINY-4-005 | Melihat Daftar Guru Les Privat | Fitur ini digunakan untuk melihat daftar guru les privat yang sudah terdaftar. |
| 6 | BRAINY-4-006 | Menghapus Guru | Fitur ini digunakan untuk menghapus guru les privat. |
| 7 | BRAINY-4-007 | Melihat Profil Guru | Fitur ini digunakan untuk melihat profil pribadi guru. |
| 8 | BRAINY-4-008 | Melihat Status Guru Aktif/Non Aktif | Fitur ini digunakan untuk melihat status guru yang aktif mengajar dan tidak aktif mengajar. |
| 9 | BRAINY-4-009 | Melihat Daftar Murid | Fitur ini digunakan untuk melihat daftar murid. |
| 10 | BRAINY-4-010 | Melihat Transaksi | Fitur ini digunakan untuk melihat struk bukti pembayaran biaya les dari murid kepada guru. |
| 11 | BRAINY-4-011 | Menerima Transaksi | Fitur ini digunakan untuk menerima transaksi bahwa transaksi tersebut benar dan telah sah. |
| 12 | BRAINY-4-012 | Menolak Transaksi | Fitur ini digunakan untuk menolak transaksi. |
| 13 | BRAINY-4-013 | Melihat Daftar Gaji Guru | Fitur ini digunakan untuk melihat daftar gaji para guru. |
| 14 | BRAINY-4-014 | Penarikan Gaji Guru | Fitur ini digunakan untuk menarik gaji guru apabila guru tersebut ingin mengambil gaji. |
| 15 | BRAINY-4-015 | Logout | Fitur ini digunakan untuk melakukan logout. |

1. **Pemodelan Kebutuhan Object Oriented**

Pemodelan kebutuhan merupakan suatu proses yang digunakan dimana akan membangun model yang menyajikan suatu domain masalah. Dengan adanya pemodelan akan lebih mudah dalam menganalisisnya. Pemodelan kebutuhan ini digunakan untuk lebih memperjelas mengenai fungsionalitas sistem yang sedang dianalisis. Dalam pemodelan kebutuhan dibagi menjadi dua pendekatan yaitu pendekatan terstruktur dan object oriented. Dalam penelitian menggunakan pendekatan *object oriented*. Pada pendekatan *object oriented* objek-objek saling bekerja sama dan berinteraksi satu sama lain. dalam pemodelan ini akan dibuat *use case* diagram, *use case* skenario dan class diagram analysis.

1. **Use Case Diagram**

Use case diagram dibawah ini terdapat 4 Pengguna yaitu: Pengguna, Murid, Guru Les Privat dan Admin. Pengguna memiliki 2 use case yaitu Login dan registrasi. Murid memiliki 20 use case. Guru Les Privat memiliki 15 use case. Sedangkan Admin memiliki 15 use case.



Gambar 4.2 Use Case Diagram

1. **Use Case Skenario**

Use Case Skenario merupakan deskripsi atau penjabaran mengenai use case yang ada pada use case diagram. Dimana dalam use case skenario mendeskripsikan mengenai alur atau aksi yang dilakukan oleh aktor yang telah didefinisikan pada suatu kebutuhan.

**Tabel 4.6 Use Case Skenario Login**

|  |  |  |
| --- | --- | --- |
| **Use Case: Login** | | |
| Objektif | Aktor dapat melakukan login | |
| Aktor | Pengguna | |
| Pre – Condition | Aktor membuka halaman beranda | |
| Main Flow | Aktor | Sistem |
| 1. Aktor menekan tombol login  3. Aktor mengisi form login berupa username dan password dan menekan tombol login. | 2.Sistem menampilkan halaman login  4.Sistem memproses data username dan password yang telah dimasukkan oleh aktor. |
| Alternatif Flow | 1. Jika aktor memasukkan username dan password salah maka akan muncul allert “ Cek Username dan Password” 2. Jika username tidak diisi maka muncul tulisan “username tidak boleh kosong”. 3. Jika password tidak diisi maka muncul tulisan “password tidak boleh kosong”. | |
| Post Condition | Sistem menampilkan halaman utama. | |

**Tabel 4.7 Use Case Skenario Daftar**

|  |  |  |
| --- | --- | --- |
| **Use Case: Daftar** | | |
| Objektif | Aktor dapat melakukan registrasi | |
| Aktor | Pengguna | |
| Pre – Condition | Aktor membuka halaman beranda | |
| Main Flow | Aktor | Sistem |
| 1.Aktor menekan tombol daftar  3. Aktor mengisi form untuk registrasi dan menekan tombol daftar. | 2.Sistem menampilkan halaman daftar  4.Sistem memproses data dari form dan di proses ke database. |
| Alternatif Flow | Jika salah satu field dalam form registrasi tidak diisi maka akan muncul allert “ Tidak boleh kosong”. | |
| Post Condition | Sistem menampilkan halaman utama. | |

**Tabel 4.8 Use Case Skenario Mencari Guru**

|  |  |  |
| --- | --- | --- |
| **Use Case: Mencari Guru** | | |
| Objektif | Aktor dapat mencari guru | |
| Aktor | Murid | |
| Pre – Condition | Aktor membuka halaman beranda dan telah login. | |
| Main Flow | Aktor | Sistem |
| 1. Aktor memilih menu cari guru  3. Aktor mengisi form untuk mencari guru les privat dan menekan tombol cari. | 2. Sistem menampilkan halaman cari guru  4. Sistem memproses dan melakukan pencarian guru dari hasil masukan dari form. |
| Alternatif Flow | Jika salah satu field dalam form pencarian tidak diisi maka akan muncul allert “ Tidak boleh kosong”. | |
| Post Condition | Sistem melakukan pencarian guru dan sistem menampilkan halaman rekomendasi guru. | |

**Tabel 4.9 Use Case Skenario Melihat Halaman Rekomendasi Guru**

|  |  |  |
| --- | --- | --- |
| **Use Case: Melihat Halaman Rekomendasi Guru** | | |
| Objektif | Aktor dapat melihat halaman rekomendasi guru | |
| Aktor | Murid | |
| Pre – Condition | Aktor telah menekan tombol cari | |
| Main Flow | Aktor | Sistem |
| 1. Aktor memilih menu cari guru  3. Aktor memasukkan field untuk mencari guru les privat dan menekan tombol cari. | 2. Sistem menampilkan halaman cari guru  4. Sistem menampilkan halaman rekomendasi guru dari hasil pencarian. |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan halaman rekomendasi guru. | |

**Tabel 4.10 Use Case Skenario Melihat Profil Guru**

|  |  |  |
| --- | --- | --- |
| **Use Case: Melihat Profil Guru** | | |
| Objektif | Aktor dapat melihat profil guru | |
| Aktor | Murid | |
| Pre – Condition | Berada pada halaman rekomendasi guru | |
| Main Flow | Aktor | Sistem |
| 1. Aktor menekan tombol lihat. | 2. Sistem menampilkan halaman profil guru |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan halaman detail guru | |

**Tabel 4.11 Use Case Skenario Memilih Guru**

|  |  |  |
| --- | --- | --- |
| **Use Case: Memilih Guru** | | |
| Objektif | Aktor dapat memilih | |
| Aktor | Murid | |
| Pre – Condition | Berada pada halaman rekomendasi guru | |
| Main Flow | Aktor | Sistem |
| 1. Aktor menekan tombol lihat.  3. Aktor menekan tombol pilih | 2. Sistem menampilkan halaman detail guru  4. Sistem menampilkan halaman untuk mengirim halaman permintaan mengajar. |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan halaman permintaan mengajar. | |

**Tabel 4.12 Use Case Skenario Mengirim Permintaan Mengajar**

|  |  |  |
| --- | --- | --- |
| **Use Case: Mengirim Permintaan Mengajar** | | |
| Objektif | Aktor dapat mengirim permintaan mengajar | |
| Aktor | Murid | |
| Pre – Condition | Berada pada halaman rekomendasi guru | |
| Main Flow | Aktor | Sistem |
| 1. Aktor menekan tombol lihat.  3. Aktor menekan tombol pilih  5. Aktor mengisi form untuk mengirim permintaan mengajar kepada guru les privat yang dipilih dan menekan tombol kirim. | 2. Sistem menampilkan halaman detail guru  4. Sistem menampilkan halaman untuk mengirim halaman permintaan mengajar.  6. Sistem menampilkan halaman dashboard murid. |
| Alternatif Flow | Jika form ada yang belum diisi maka sistem menampilkan allert “Tidak boleh kosong” | |
| Post Condition | Sistem menampilkan halaman dashboard murid. | |

**Tabel 4.13 Use Case Skenario Melihat Dashboard**

|  |  |  |
| --- | --- | --- |
| **Use Case: Melihat Dashboard** | | |
| Objektif | Aktor dapat melihat dashboard | |
| Aktor | Murid, Guru Les Privat | |
| Pre – Condition | Aktor membuka halaman beranda dan telah login. | |
| Main Flow | Aktor | Sistem |
| 1. Aktor menekan menu dengan simbol akun pribadi.  3. Aktor memilih lihat profil | 2. Sistem menampilkan menu dropdown berupa lihat profil dan logout.  4. Sistem menampilkan halaman profil atau dashboard. |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan halaman dashboard | |

**Tabel 4.14 Use Case Skenario Mengedit Profil**

|  |  |  |
| --- | --- | --- |
| **Use Case: Mengedit Profil** | | |
| Objektif | Aktor dapat mengedit profil | |
| Aktor | Murid , Guru Les Privat | |
| Pre – Condition | Aktor membuka halaman beranda dan telah login. | |
| Main Flow | Aktor | Sistem |
| 1. Aktor menekan menu dengan simbol akun pribadi.  3. Aktor memilih lihat profil  5. Aktor menekan tombol edit profil  7. Aktor mengedit bagian yang ingin diedit dan menekan tombol simpan. | 2. Sistem menampilkan menu dropdown berupa lihat profil dan logout.  4. Sistem menampilkan halaman profil atau dashboard.  6. Sistem menampilkan halaman profil yang siap di edit.  8. Sistem menampilkan halaman profil yang telah diedit. |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan halaman dashboard | |

**Tabel 4.15 Use Case Skenario Melihat Permintaan Terkirim**

|  |  |  |
| --- | --- | --- |
| **Use Case: Melihat Permintaan Terkirim** | | |
| Objektif | Aktor dapat melihat permintaan terkirim | |
| Aktor | Murid | |
| Pre – Condition | Berada pada halaman dashboard | |
| Main Flow | Aktor | Sistem |
| 1. Aktor memilih menu permintaan terkirim | 2. Sistem menampilkan halaman permintaan terkirim. |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan halaman permintaan terkirim | |

**Tabel 4.16 Use Case Skenario Melihat Permintaan Diterima**

|  |  |  |
| --- | --- | --- |
| **Use Case: Melihat Permintaan Diterima** | | |
| Objektif | Aktor dapat melihat permintaan diterima | |
| Aktor | Murid | |
| Pre – Condition | Berada pada halaman dashboard | |
| Main Flow | Aktor | Sistem |
| 1. Aktor memilih menu permintaan diterima | 2. Sistem menampilkan halaman permintaan diterima |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan halaman permintaan diterima | |

**Tabel 4.17 Use Case Skenario Melihat Permintaan Ditolak**

|  |  |  |
| --- | --- | --- |
| **Use Case: Melihat Permintaan Ditolak** | | |
| Objektif | Aktor dapat melihat permintaan ditolak | |
| Aktor | Murid | |
| Pre – Condition | Berada pada halaman dashboard | |
| Main Flow | Aktor | Sistem |
| 1. Aktor memilih menu permintaan ditolak | 2. Sistem menampilkan halaman permintaan ditolak |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan halaman permintaan ditolak | |

**Tabel 4.18 Use Case Skenario Upload Bukti Pembayaran**

|  |  |  |
| --- | --- | --- |
| **Use Case: Upload Bukti Pembayaran** | | |
| Objektif | Aktor dapat upload bukti pembayaran | |
| Aktor | Murid | |
| Pre – Condition | Berada pada halaman dashboard | |
| Main Flow | Aktor | Sistem |
| 1. Aktor memilih menu permintaan diterima  3. Aktor menekan tombol upload bukti pembayaran dan menekan tombol upload. | 2. Sistem menampilkan halaman permintaan diterima.  4. Sistem menyimpan hasil bukti pembayaran dan mengalihkan ke menu bimbingan saat ini. |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan halaman permintaan diterima. | |

**Tabel 4.19 Use Case Skenario Melihat Bimbingan Saat Ini**

|  |  |  |
| --- | --- | --- |
| **Use Case: Melihat Bimbingan Saat ini** | | |
| Objektif | Aktor dapat melihat bimbingan saat ini | |
| Aktor | Murid | |
| Pre – Condition | Berada pada halaman dashboard | |
| Main Flow | Aktor | Sistem |
| 1. Aktor memilih menu bimbingan saat ini | 2. Sistem menampilkan halaman bimbingan saat ini |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan halaman bimbingan saat ini. | |

**Tabel 4.21 Use Case Skenario Melihat Riwayat Belajar**

|  |  |  |
| --- | --- | --- |
| **Use Case: Melihat Riwayat Belajar** | | |
| Objektif | Aktor dapat melihat riwayat belajar | |
| Aktor | Murid | |
| Pre – Condition | Berada pada halaman dashboard | |
| Main Flow | Aktor | Sistem |
| 1. Aktor memilih menu riwayat belajar | 2. Sistem menampilkan halaman riwayat belajar |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan halaman riwayat belajar. | |

**Tabel 4.22 Use Case Skenario Menghapus Riwayat Belajar**

|  |  |  |
| --- | --- | --- |
| **Use Case: Menghapus Riwayat Belajar** | | |
| Objektif | Aktor dapat menghapus riwayat belajar | |
| Aktor | Murid | |
| Pre – Condition | Berada pada halaman dashboard | |
| Main Flow | Aktor | Sistem |
| 1. Aktor memilih menu riwayat belajar  3. Aktor menekan tombol hapus | 2. Sistem menampilkan halaman riwayat belajar  4. Sistem meghapus bagian yang dihapus oleh aktor. |
| Alternatif Flow | - | |
| Post Condition | Sistem menghapus riwayat belajar | |

**Tabel 4.23 Use Case Skenario Mengakhiri Kegiatan Belajar**

|  |  |  |
| --- | --- | --- |
| **Use Case: Mengakhiri Kegiatan Belajar** | | |
| Objektif | Aktor dapat mengakhiri kegiatan belajar. | |
| Aktor | Murid | |
| Pre – Condition | Berada pada halaman dashboard | |
| Main Flow | Aktor | Sistem |
| 1. Aktor memilih menu bimbingan belajar saat ini  3. Aktor menekan tombol akhiri | 2. Sistem menampilkan halaman bimbingan saat ini. |
| Alternatif Flow | - | |
| Post Condition | Sistem menghapus paket bimbingan belajar di menu bimbingan saat ini dan memasukkan ke daftar riwayat. | |

**Tabel 4.24 Use Case Skenario Memberikan Rating dan Ulasan**

|  |  |  |
| --- | --- | --- |
| **Use Case: Memberikan Rating dan Ulasan** | | |
| Objektif | Aktor dapat memberikan rating dan ulasan | |
| Aktor | Murid | |
| Pre – Condition | Berada pada halaman dashboard | |
| Main Flow | Aktor | Sistem |
| 1. Aktor memilih menu bimbingan belajar saat ini  3. Aktor menekan tombol akhiri.  5. Aktor memberikan rating dan ulasan. | 2. Sistem menampilkan halaman bimbingan saat ini  4. Sistem menampilkan halaman untuk memberikan rating dan ulasan.  6. Sistem memproses masukan dari aktor. |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan halaman beranda. | |

**Tabel 4.25 Use Case Skenario Pesan Lagi**

|  |  |  |
| --- | --- | --- |
| **Use Case: Pesan Lagi** | | |
| Objektif | Aktor dapat memesan kembali guru les privat yang terdapat dalam riwayat belajar | |
| Aktor | Murid | |
| Pre – Condition | Berada pada halaman beranda | |
| Main Flow | Aktor | Sistem |
| 1. Aktor memilih menu akun atau profil pribadi  3. Aktor memilih menu riwayat belajar.  5. Aktor menekan tombol pesan lagi untuk guru yag ingin dipesan kembali.  7. Aktor mengisi permintaan mengajar dan menekan tombol kirim. | 2. Sistem menampilkan halaman profil atau dashboard  4. Sistem menampilkan daftar riwayat belajar.  6. Sistem menampilkan form permintaan mengajar.  8. Sistem akan memproses lebih lanjut. |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan halaman riwayat belajar. | |

**Tabel 4.26 Use Case Skenario Mengirim Pesan**

|  |  |  |
| --- | --- | --- |
| **Use Case: Mengirim Pesan** | | |
| Objektif | Aktor dapat mengirim pesan | |
| Aktor | Murid dan Guru Les Privat | |
| Pre – Condition | Berada pada halaman dashboard | |
| Main Flow | Aktor | Sistem |
| 1. Aktor memilih menu bimbingan saat ini  3. Aktor menekan tombol kirim pesan | 2. Sistem menampilkan halaman bimbingan saat ini  4. Sistem menampilkan halaman untuk saling mengirim pesan. |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan halaman detail guru | |

**Tabel 4.27 Use Case Skenario Mengirim Lamaran Pekerjaan**

|  |  |  |
| --- | --- | --- |
| **Use Case: Mengirim Lamaran Pekerjaan** | | |
| Objektif | Aktor dapat mengirim lamaran pekerjaan | |
| Aktor | Guru Les Privat | |
| Pre – Condition | Aktor membuka halaman beranda dan telah login. | |
| Main Flow | Aktor | Sistem |
| 1. Aktor memilih menu dashboard  3. Aktor memilih menu lamaran pekerjaan.  5. Aktor menekan tombol buat.  7. Aktor mengisi form lamaran pekerjaan dan menekan tombol kirim. | 2. sistem menampilkan menu dashboard.  4. Sistem menampilkan halaman untuk membuat lamaran pekerjaan.  6. Sistem menampilkan form lamaran pekerjaan guru les privat yang harus diisi oleh guru.  4. Sistem memproses lamaran pekerjaan dan dikirimkan ke admin yang akan memvalidasi. |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan halaman lamaran pekerjaan pada menu dashboard dan menampilkan form yang telah terisi. | |

**Tabel 4.28 Use Case Skenario Melihat Lamaran Pekerjaan**

|  |  |  |
| --- | --- | --- |
| **Use Case: Melihat Lamaran Pekerjaan** | | |
| Objektif | Aktor dapat melihat lamaran pekerjaan | |
| Aktor | Guru Les Privat | |
| Pre – Condition | Berada pada halaman beranda dan telah login. | |
| Main Flow | Aktor | Sistem |
| 1. Aktor menekan menu dengan simbol akun pribadi.  3. Aktor memilih lihat profil  5. Aktor memilih menu lamaran pekerjaan | 2. Sistem menampilkan menu dropdown berupa lihat profil dan logout.  4. Sistem menampilkan halaman profil atau dashboard.  6. Sistem menampilkan halaman lamaran pekerjaan. |
| Alternatif Flow | Jika belum mengirim lamaran pekerjaan maka halaman lamaran pekerjaan kosong. | |
| Post Condition | Sistem menampilkan halaman lamaran pekerjan | |

**Tabel 4.29 Use Case Skenario Melihat Status Lamaran Pekerjaan**

|  |  |  |
| --- | --- | --- |
| **Use Case: Melihat Status Lamaran Pekerjaan** | | |
| Objektif | Aktor dapat melihat status lamaran pekerjaan | |
| Aktor | Guru Les Privat | |
| Pre – Condition | Berada pada halaman beranda dan telah login. | |
| Main Flow | Aktor | Sistem |
| 1. Aktor menekan menu dengan simbol akun pribadi.  3. Aktor memilih lihat profil  5. Aktor memilih menu lamaran pekerjaan | 2. Sistem menampilkan menu dropdown berupa lihat profil dan logout.  4. Sistem menampilkan halaman profil atau dashboard.  6. Sistem menampilkan halaman lamaran pekerjaan dan menampilkan status lamaran pekerjaan (terkirim, diterima atau ditolak) |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan halaman lamaran pekerjan | |

**Tabel 4.30 Use Case Skenario Melihat Daftar Permintaan**

|  |  |  |
| --- | --- | --- |
| **Use Case: Melihat Daftar Permintaan** | | |
| Objektif | Aktor dapat melihat daftar permintaan | |
| Aktor | Guru Les Privat | |
| Pre – Condition | Berada pada halaman beranda dan telah login. | |
| Main Flow | Aktor | Sistem |
| 1. Aktor menekan menu dengan simbol akun pribadi.  3. Aktor memilih lihat profil  5. Aktor memilih menu daftar permintaan | 2. Sistem menampilkan menu dropdown berupa lihat profil dan logout.  4. Sistem menampilkan halaman profil atau dashboard.  6. Sistem menampilkan daftar permintaan |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan halaman daftar permintaan | |

**Tabel 4.31 Use Case Skenario Menyetujui Permintaan**

|  |  |  |
| --- | --- | --- |
| **Use Case: Menyetujui Permintaan** | | |
| Objektif | Aktor dapat menyetujui permintaan | |
| Aktor | Guru Les Privat | |
| Pre – Condition | Berada pada halaman beranda dan telah login. | |
| Main Flow | Aktor | Sistem |
| 1. Aktor menekan menu dengan simbol akun pribadi.  3. Aktor memilih lihat profil  5. Aktor memilih menu daftar permintaan  7. Aktor menekan tombol setujui | 2. Sistem menampilkan menu dropdown berupa lihat profil dan logout.  4. Sistem menampilkan halaman profil atau dashboard.  6. Sistem menampilkan daftar permintaan  8. Sistem memproses dan menampilkan di halaman bimbingan saat ini |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan halaman daftar permintaan | |

**Tabel 4.32 Use Case Skenario Menolak Permintaan**

|  |  |  |
| --- | --- | --- |
| **Use Case: Menolak Permintaan** | | |
| Objektif | Aktor dapat menolak permintaan | |
| Aktor | Guru Les Privat | |
| Pre – Condition | Berada pada halaman beranda dan telah login. | |
| Main Flow | Aktor | Sistem |
| 1. Aktor menekan menu dengan simbol akun pribadi.  3. Aktor memilih lihat profil  5. Aktor memilih menu daftar permintaan  7. Aktor menekan tombol tolak | 2. Sistem menampilkan menu dropdown berupa lihat profil dan logout.  4. Sistem menampilkan halaman profil atau dashboard.  6. Sistem menampilkan daftar permintaan  8. Sistem memproses dan menampilkan di halaman bimbingan saat ini |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan halaman daftar permintaan | |

**Tabel 4.33 Use Case Skenario Melihat Bimbingan Saat Ini**

|  |  |  |
| --- | --- | --- |
| **Use Case: Melihat Bimbingan Saat Ini** | | |
| Objektif | Aktor dapat melihat bimbingan saat ini | |
| Aktor | Guru Les Privat | |
| Pre – Condition | Berada pada halaman beranda dan telah login. | |
| Main Flow | Aktor | Sistem |
| 1. Aktor menekan menu dengan simbol akun pribadi.  3. Aktor memilih lihat profil  5. Aktor memilih menu bimbingan saat ini | 2. Sistem menampilkan menu dropdown berupa lihat profil dan logout.  4. Sistem menampilkan halaman profil atau dashboard.  6. Sistem menampilkan halaman bimbingan saat ini. |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan halaman bimbingan saat ini | |

**Tabel 4.34 Use Case Skenario Melihat Riwayat Mengajar**

|  |  |  |
| --- | --- | --- |
| **Use Case: Melihat Riwayat Mengajar** | | |
| Objektif | Aktor dapat melihat riwayat mengajar | |
| Aktor | Guru Les Privat | |
| Pre – Condition | Berada pada halaman beranda dan telah login. | |
| Main Flow | Aktor | Sistem |
| 1. Aktor menekan menu dengan simbol akun pribadi.  3. Aktor memilih lihat profil  5. Aktor memilih menu riwayat mengajar | 2. Sistem menampilkan menu dropdown berupa lihat profil dan logout.  4. Sistem menampilkan halaman profil atau dashboard.  6. Sistem menampilkan halaman riwayat mengajar |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan halaman riwayat mengajar | |

**Tabel 4.35 Use Case Skenario Menghapus Riwayat Mengajar**

|  |  |  |
| --- | --- | --- |
| **Use Case: Menghapus Riwayat Mengajar** | | |
| Objektif | Aktor dapat menghapus riwayat mengajar | |
| Aktor | Guru Les Privat | |
| Pre – Condition | Berada pada halaman beranda dan telah login. | |
| Main Flow | Aktor | Sistem |
| 1. Aktor menekan menu dengan simbol akun pribadi.  3. Aktor memilih lihat profil  5. Aktor memilih menu riwayat mengajar  7. Aktor menekan | 2. Sistem menampilkan menu dropdown berupa lihat profil dan logout.  4.  Sistem menampilkan halaman profil atau dashboard.  6. Sistem menampilkan halaman riwayat mengajar |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan halaman riwayat mengajar | |

**Tabel 4.36 Use Case Skenario Melihat Gaji**

|  |  |  |
| --- | --- | --- |
| **Use Case: Melihat Daftar Permintaan** | | |
| Objektif | Aktor dapat melihat gaji yang diperoleh | |
| Aktor | Guru les Privat | |
| Pre – Condition | Berada pada halaman utama | |
| Main Flow | Aktor | Sistem |
| 1. Aktor menekan menu Gaji | 2. Sistem menampilkan gaji yang telah didapatkan selama mengajar. |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan halaman gaji | |

**Tabel 4.37 Use Case Skenario Melihat Daftar Permintaan**

|  |  |  |
| --- | --- | --- |
| **Use Case: Melihat Daftar Permintaan** | | |
| Objektif | Aktor dapat melihat daftar permintaan | |
| Aktor | Admin | |
| Pre – Condition | Berada pada halaman utama | |
| Main Flow | Aktor | Sistem |
| 1. Aktor menekan menu Daftar Permintaan | 2. Sistem menampilkan halaman daftar permintaan |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan halaman daftar permintaan. | |

**Tabel 4.38 Use Case Skenario Menerima Permintaan**

|  |  |  |
| --- | --- | --- |
| **Use Case: Menerima Permintaan** | | |
| Objektif | Aktor dapat menerima permintaan | |
| Aktor | Admin | |
| Pre – Condition | Berada pada halaman utama | |
| Main Flow | Aktor | Sistem |
| 1. Aktor menekan menu Daftar Permintaan  3. Aktor menekan tombol terima untuk menerima lamaran pekerjaan. | 2. Sistem menampilkan halaman daftar permintaan  4. Sistem menambahkan ke dalam daftar guru les privat. |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan halaman daftar permintaan. | |

**Tabel 4.39 Use Case Skenario Menolak Permintaan**

|  |  |  |
| --- | --- | --- |
| **Use Case: Menolak Permintaan** | | |
| Objektif | Aktor dapat menolak permintaan | |
| Aktor | Admin | |
| Pre – Condition | Berada pada halaman utama | |
| Main Flow | Aktor | Sistem |
| 1. Aktor menekan menu Daftar Permintaan  3. Aktor menekan tombol tolak untuk menolak lamaran pekerjaan. | 2. Sistem menampilkan halaman daftar permintaan  4. Sistem menampilkan pop-up untuk mengisi alasan penolakan dan menghapus data lamaran dari daftar lamaran. |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan halaman daftar permintaan. | |

**Tabel 4.41 Use Case Skenario Melihat Daftar Guru Les Privat**

|  |  |  |
| --- | --- | --- |
| **Use Case: Melihat Daftar Guru Les Privat** | | |
| Objektif | Aktor dapat melihat Daftar Guru Les Privat | |
| Aktor | Admin | |
| Pre – Condition | Berada pada halaman utama | |
| Main Flow | Aktor | Sistem |
| 1. Aktor menekan menu Daftar Guru | 2. Sistem menampilkan halaman daftar guru |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan daftar guru les privat | |

**Tabel 4.42 Use Case Skenario Menghapus Guru**

|  |  |  |
| --- | --- | --- |
| **Use Case: Menghapus Guru** | | |
| Objektif | Aktor dapat menghapus guru | |
| Aktor | Admin | |
| Pre – Condition | Berada pada halaman utama | |
| Main Flow | Aktor | Sistem |
| 1. Aktor menekan menu Daftar Guru  3. Aktor menekan tombol hapus kepada guru les privat yang ingin dihapus | 2. Sistem menampilkan halaman daftar guru  4. Sistem menghapus data dari daftar guru. |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan daftar guru les privat | |

**Tabel 4.43 Use Case Skenario Melihat Profil Guru**

|  |  |  |
| --- | --- | --- |
| **Use Case: Melihat Profil Guru** | | |
| Objektif | Aktor dapat melihat profil guru | |
| Aktor | Admin | |
| Pre – Condition | Berada pada halaman utama | |
| Main Flow | Aktor | Sistem |
| 1. Aktor menekan menu Daftar Guru  3. Aktor menekan tombol lihat profil | 2. Sistem menampilkan halaman daftar guru  4. Sistem menampilkan halaman profil guru les privat. |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan halaman profil guru les privat | |

**Tabel 4.44 Use Case Skenario Melihat Status Guru Aktif/Non Aktif**

|  |  |  |
| --- | --- | --- |
| **Use Case: Melihat Status Guru Aktif/Non Aktif** | | |
| Objektif | Aktor dapat melihat profil guru yang sedang aktif/ non aktif | |
| Aktor | Admin | |
| Pre – Condition | Berada pada halaman utama | |
| Main Flow | Aktor | Sistem |
| 1. Aktor menekan menu Daftar Guru  3. Aktor menekan tombol lihat profil | 2. Sistem menampilkan halaman daftar guru  4. Sistem menampilkan halaman profil guru les privat dan menampilkan status guru (aktif atau non aktif) |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan halaman profil guru les privat dan status aktif atau tidak aktif. | |

**Tabel 4.45 Use Case Skenario Melihat Daftar Murid**

|  |  |  |
| --- | --- | --- |
| **Use Case: Melihat Daftar Murid** | | |
| Objektif | Aktor dapat melihat daftar murid | |
| Aktor | Admin | |
| Pre – Condition | Berada pada halaman utama | |
| Main Flow | Aktor | Sistem |
| 1. Aktor menekan menu Daftar Murid | 2. Sistem menampilkan halaman daftar murid |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan daftar murid | |

**Tabel 4.46 Use Case Skenario Melihat Transaksi**

|  |  |  |
| --- | --- | --- |
| **Use Case: Melihat Transaksi** | | |
| Objektif | Aktor dapat melihat transaksi | |
| Aktor | Admin | |
| Pre – Condition | Berada pada halaman utama | |
| Main Flow | Aktor | Sistem |
| 1. Aktor menekan menu Transaksi  3. Aktor menekan tombol lihat untuk melihat bukti pembayaran. | 2. Sistem menampilkan halaman daftar transaksi pembayaran yang dilakukan murid ke guru.  4. Sistem menampilkan struk bukti transaksi pembayaran. |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan struk transaksi pembayaran. | |

**Tabel 4.47 Use Case Skenario Menerima Transaksi**

|  |  |  |
| --- | --- | --- |
| **Use Case: Menerima Transaksi** | | |
| Objektif | Aktor dapat menerima transaksi | |
| Aktor | Admin | |
| Pre – Condition | Berada pada halaman utama | |
| Main Flow | Aktor | Sistem |
| 1. Aktor menekan menu Transaksi  3. Aktor menekan tombol terima untuk menerima transaksi. | 2. Sistem menampilkan halaman daftar transaksi pembayaran yang dilakukan murid ke guru.  4. Sistem memproses dan mengubah status di dashboard murid. |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan halaman transaksi. | |

**Tabel 4.48 Use Case Skenario Menolak Transaksi**

|  |  |  |
| --- | --- | --- |
| **Use Case: Menolak Transaksi** | | |
| Objektif | Aktor dapat menolak transaksi | |
| Aktor | Admin | |
| Pre – Condition | Berada pada halaman utama | |
| Main Flow | Aktor | Sistem |
| 1. Aktor menekan menu Transaksi  3. Aktor menekan tombol tolak untuk menolak transaksi. | 2. Sistem menampilkan halaman daftar transaksi pembayaran yang dilakukan murid ke guru.  4. Sistem memproses dan mengubah status di dashboard murid. |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan halaman transaksi. | |

**Tabel 4.49 Use Case Skenario Melihat Gaji Guru**

|  |  |  |
| --- | --- | --- |
| **Use Case: Melihat Gaji Guru** | | |
| Objektif | Aktor dapat melihat gaji guru | |
| Aktor | Admin | |
| Pre – Condition | Berada pada halaman utama | |
| Main Flow | Aktor | Sistem |
| 1. Aktor menekan menu Daftar Gaji Guru | 2. Sistem menampilkan halaman daftar guru dengan gaji masing-masing. |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan halaman transaksi. | |

**Tabel 4.50 Use Case Skenario Penarikan Gaji Guru**

|  |  |  |
| --- | --- | --- |
| **Use Case: Penarikan Gaji Guru** | | |
| Objektif | Aktor dapat penarikan gaji guru | |
| Aktor | Admin | |
| Pre – Condition | Berada pada halaman utama | |
| Main Flow | Aktor | Sistem |
| 1. Aktor menekan menu Daftar Gaji Guru  3. Aktor menekan tombol penarikan | 2. Sistem menampilkan halaman daftar guru dengan gaji masing-masing.  4. Sistem menampilkan pop-up untuk melakukan penarikan dengan jumlah yang diinginkan. |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan halaman transaksi. | |

**Tabel 4.51 Use Case Skenario Logout**

|  |  |  |
| --- | --- | --- |
| **Use Case: Logout** | | |
| Objektif | Aktor dapat melakukan logout | |
| Aktor | Murid, Guru Les Privat dan Admin | |
| Pre – Condition | Berada pada halaman utama | |
| Main Flow | Aktor | Sistem |
| 1. Aktor menekan tombol simbol akun  3. Aktor memilih menu Logout | 2. Sistem menampilkan pilihan Logout  4. Sistem menampilkan halaman utama |
| Alternatif Flow | - | |
| Post Condition | Sistem menampilkan halaman utama | |

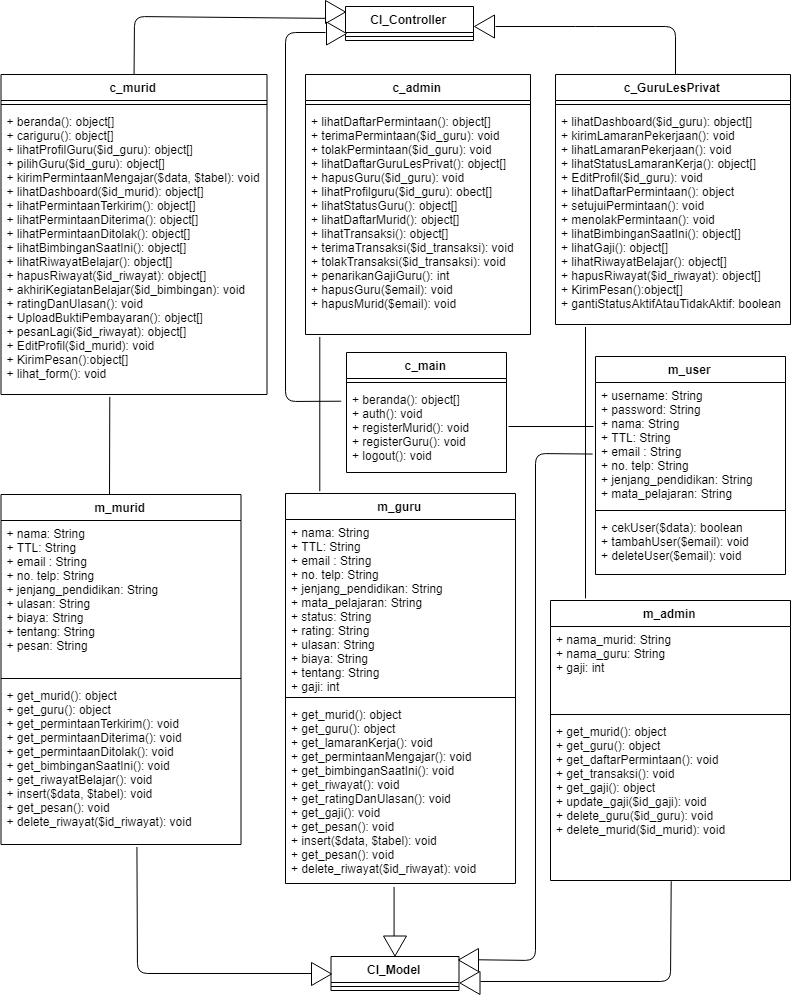
# PERANCANGAN DAN IMPLEMENTASi

1. **Perancangan Sistem**

Perancangan sistem berisi rancangan langkah kerja dari sistem secara menyeluruh. Perancangan sistem digunakan untuk mempermudah implementasi. Dalam merancang suatu sistem berisi gambaran jelas dan menghasilkan rancangan bangun yang lengkap. Dalam membuat model perancangan tersebut menggunakan Object Oriented Design ( OOD).

1. **Perancangan Class Diagram**

Class Diagram merupakan salah satu diagram dari Unified Modelling Language(UML). Class diagram menggambarkan kelas-kelas yang ada pada sebuah sistem. Class Diagram menggambarkan struktur objek sistem dimana mereka menunjukkan objek dari suatu kelas dan kelas-kelas tersebut berhubungan satu sama lain.



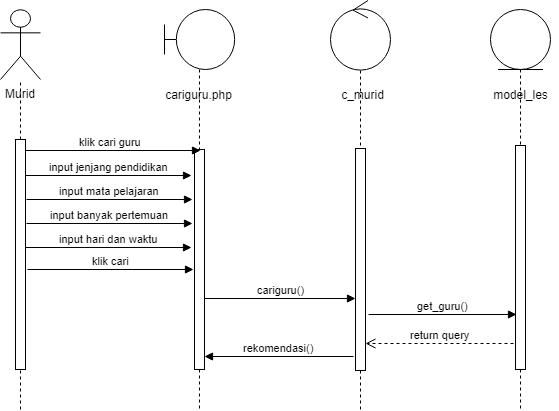
Gambar 5.1 Perancangan Class Diagram

1. **Perancangan Sequence Diagram**

Sequence diagram menunjukkan bagaimana objek-objek tersebut saling berinteraksi dan saling mengirim pesan. Di bawah ini terdapat 3 sequence diagram yaitu sequence diagram Upload Bukti pembayaran. Menyetujui Permintaan dan Menerima Transaksi.

1. **Sequence Diagram Cari Guru**

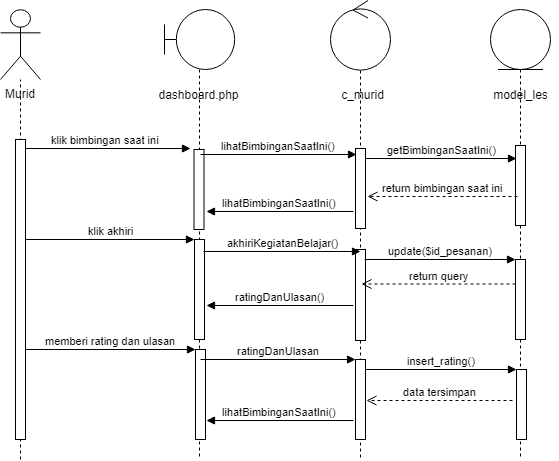
Pada Gambar 5.4 merupakan sequence diagram untuk mencari guru. Cari guru dilakukan oleh aktor murid. Dalam sequence diagram tersebut terdapat view dengan nama cariguru.php, controller dengan nama c\_murid dan model\_les. Semua object tersebut saling berhubungan satu sama lain.



Gambar 5.2 Sequence Diagram Cari Guru

1. **Sequence Diagram Memberikan Rating dan Ulasan**

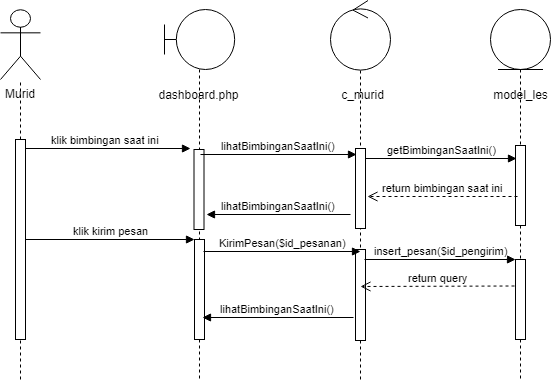
Sequence Diagram Memberikan Rating dan Ulasan yang dilakukan oleh aktor murid. Dalam sequence diagram tersebut terdapat aktor yaitu murid, view dengan nama dashboard.php, controller c\_murid dan model\_les. Setiap objek tersebut saling berinteraksi dan mengirim pesan.



Gambar 5.3 Sequence Diagram Memberikan Rating dan Ulasan

1. **Sequence Diagram Mengirim Pesan**

Pada Gambar 5.3 merupakan sequence diagram mengirim pesan. Dan dalam gambar tersebut menunjukkan bagaimana view, model dan controller yang saling berinteraksi dan mengirim pesan.



Gambar 5.4 Sequence Diagram Mengirim Pesan

1. **Perancangan Algoritma**

Perancangan Algoritma merupakan salah satu tahap dari perancangan sistem dimana di dalam perancangan algoritma terdapat susunan atau langkah-langkah dari pemecahan masalah yang dibuat secara sistematis. Dalam perancangan algoritma dibawah ini terdapat perancangan algoritma dari login, cari guru dan memberikan rating dan ulasan.

Perancangan Algoritma Cari Guru

1. **Perancangan Algoritma Cari Guru**

|  |
| --- |
| variabel data = mengambil data jenjang pendidikan, mata pelajaran, banyak pertemuan, hari dan waktu  jenjang\_pendidikan = post (jenjang\_pendidikan)  mata\_pelajaran = post (mata\_pelajaran)  banyak\_pertemuan = post(banyak\_pertemuan)  hari = post(hari)  waktu = post(waktu)  panggil method cariguru(data)  panggil halaman rekomendasi |

1. **Perancangan Algoritma Rating dan Ulasan**

|  |
| --- |
| variabel data = mengambil data rating dan ulasan  rating, ulasan = post(rating, ulasan)  panggil method review(rating, ulasan)  show review(data) |

1. **Perancangan Algoritma KirimPesan**

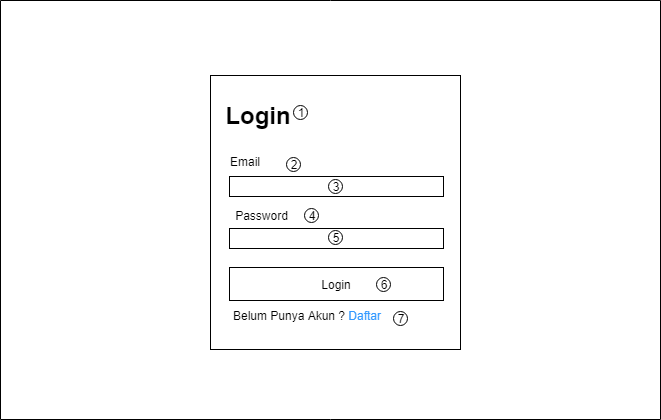
|  |
| --- |
| variabel pengirim = mengambil id pengirim  variabel pesan = menyimpan data isi pesan  isi\_pesan = post(isi\_pesan)  panggil method chat (pesan)  show pesan (pesan) |

1. **Perancangan Antarmuka**

Perancangan Antarmuka atau *User Interface Design* merupakan salah satu tahap yang digunakan pada tahap perancangan. Tahap perancangan ini digunakan untuk menghasilkan antarmuka sistem yang digunakan pengguna sebagai sarana komunikasi dengan sistem.

1. **Perancangan Antarmuka Login**

Pada Gambar 5.5 menunjukkan user interface login yang dilakukan oleh pengguna. Login ke sistem untuk masuk ke autorisasi sebagai murid atau guru les privat. Pengguna memasukkan Username dan Password untuk melakukan login.

****

Gambar 5.5 Perancangan Antarmuka Login

Table 5.1 Keterangan Halaman Login

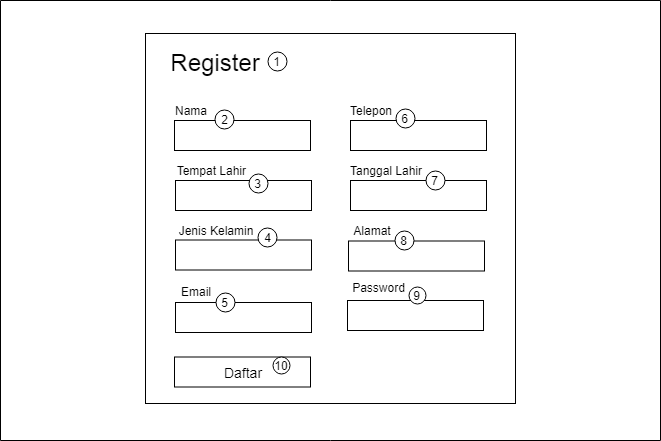
|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Objek** | **Tipe** | **Keterangan** |
| 1 | Login | Teks | Menunjukkan bahwa form tersebut adalah form untuk melakukan login |
| 2 | Email | Label | Untuk memberi keterangan kepada pengguna bahwa yang harus diisi adalah berupa email |
| 3 | Field Email | Text Field | Digunakan untuk mengisi email |
| 4 | Password | Label | Keterangan untuk isian password |
| 5 | Field Password | Teks Field | Digunakan untuk mengisi password |
| 6 | Login | Button | Tombol untuk melakukan login |
| 7 | Daftar | Link | Link yang digunakan untuk register apabila belum memiliki akun. |

1. **Perancangan Antarmuka Register**

User Interface Register digunakan oleh pengguna, dimana pengguna akan mendaftarkan sebagai murid atau sebagai guru les privat. Pada halaman register terdapat form untuk melakukan pendaftaran ke sistem. Dalam melakukan register terdapat perbedaan form antara murid dan guru les privat sebagai berikut:

* 1. **Perancangan Antarmuka Register Guru**

Register Guru memiliki form untuk melakukan pendaftaran seperti nama, tempat lahir, tanggal lahir, Jenis Kelamin, Alamat, Telepon, Email dan Password.



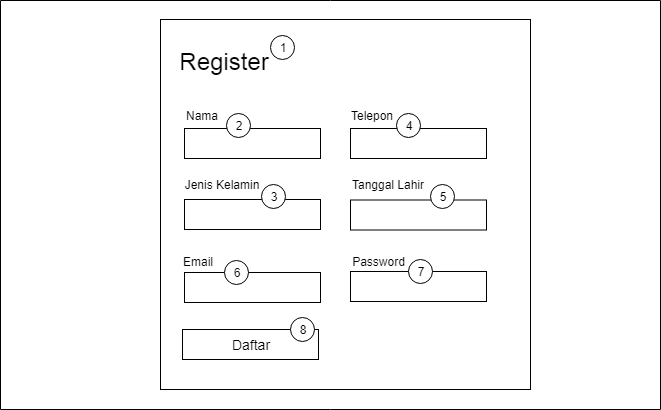
Gambar 5.6 Perancangan Antarmuka Register Guru

Table 5.2 Keterangan Halaman Register Guru

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Objek** | **Tipe** | **Keterangan** |
| 1 | Register | Teks | Untuk memberi informasi bahwa form tersebut merupakan form untuk melakukan registrasi |
| 2 | Field Nama | Text Field | Untuk memberikan informasi bahwa isian berupa nama |
| 3 | Field Tempat Lahir | Text Field | Untuk memberikan informasi bahwa isian berupa tempat lahir |
| 4 | Field Jenis Kelamin | Text Field | Untuk memberikan informasi bahwa isian berupa jenis kelamin |
| 5 | Field Email | Text Field | Untuk memberikan informasi bahwa isian berupa email |
| 6 | Field Telepon | Text Field | Untuk memberikan informasi bahwa isian berupa nomor telepon |
| 7 | Field Tanggal Lahir | Text Field | Untuk memberikan informasi bahwa isian berupa tanggal lahir |
| 8 | Field Alamat | Text Field | Untuk memberikan informasi bahwa isian berupa alamat |
| 9 | Field Password | Text Field | Untuk memberikan informasi bahwa isian berupa password |
| 10 | Daftar | Button | Tombol untuk melakukan registrasi atau pendaftaran |

* 1. **Perancangan Antarmuka Register Murid**

Register murid memiliki form untuk melakukan pendaftaran sebagai murid dengan mengisi form seperti nama lengkap, TTL, Jenis Kelamin, Alamat, No. Telp, Email dan Password.



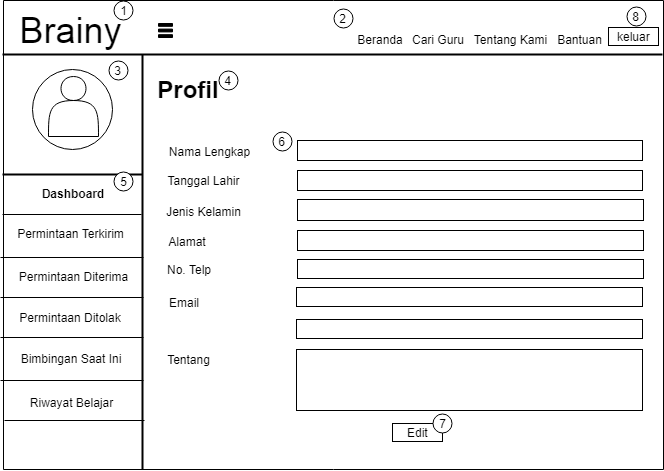
Gambar 5.7 Perancangan Antarmuka Register Murid

Table 5.3 Keterangan Halaman Register Murid

|  |  |  |  |
| --- | --- | --- | --- |
| No | Nama Objek | Tipe | Keterangan |
| 1 | Register | Label | Memberikan informasi bahwa form tersebut merupakan form untuk melakukan registrasi atau pendaftaran. |
| 2 | Nama | Text Field | Isian yang menunjukkan bahwa isian berupa nama |
| 3 | Jenis Kelamin | Text Field | Isian yang menunjukkan bahwa isian berupa jenis kelamin |
| 4 | Telepon | Text Field | Isian yang menunjukkan bahwa isian berupa nomor telepon |
| 5 | Tanggal Lahir | Text Field | Isian yang menunjukkan bahwa isian berupa tanggal lahir |
| 6 | Email | Text Field | Isian yang menunjukkan bahwa isian berupa email |
| 7 | Password | Text Field | Isian yang menunjukkan bahwa isian berupa password |
| 8 | Daftar | Button | Tombol yang digunakan untuk melakukan registrasi apabila semua isian sudah terisi |

1. **Perancangan Antarmuka Dashboard Murid**

Dalam dashboard murid terdapat beberapa menu seperti Permintaan Terkirim, Permintaan Diterima, Permintaan Ditolak, Bimbingan Saat Ini dan Riwayat Belajar. Serta dapat melihat profilpribadi dan mengedit profil.



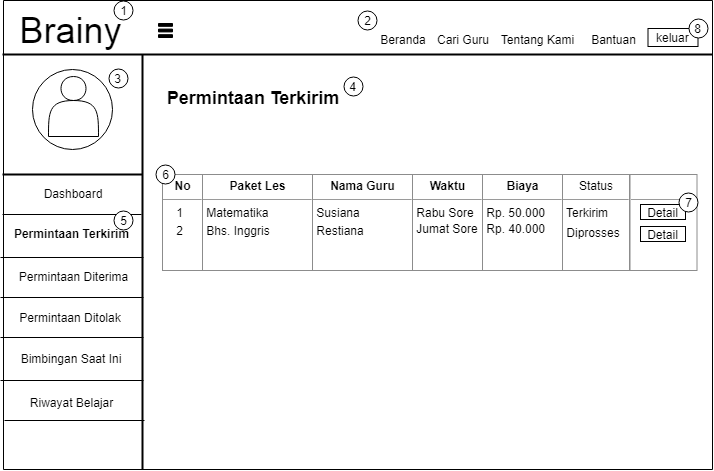
**Gambar 5.8 Perancangan Antarmuka Dashboard Murid**

Table 5.4 Tabel Keterangan Halaman Dashboard Murid

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Objek** | **Tipe** | **Keterangan** |
| 1 | Logo | Label | Nama dari sistem |
| 2 | Navbar Menu | Link | Di dalam navbar terdapat menu-menu yang dapat diakses seperti: beranda, cari guru, tentang kami dan bantuan. |
| 3 | Photo Profil | Image | Foto akun pribadi |
| 4 | Judul | Label | Digunakan untuk memberi informasi bahwa di bawah merupakan profil pribadi. |
| 5 | Sidebar Menu | Link | Dalam sidebar menu terdapat beberapa menu seperti: dashboard, permintaan terkirim, permintaan diterima, permintaan ditolak, bimbingan saat ini dan riwayat. |
| 6 | Form Profil | Text | Berisi informasi pribadi yang telah diisi dan harus dilengkapi. |
| 7 | Edit | Button | Dilakukan untuk melakukan *update* atau perubahan pada informasi pribadi |
| 8 | Keluar | Button | Digunakan untuk keluar dari autorisasi murid. |

* 1. **Perancangan Antarmuka Terkirim**

Dalam permintaan terkirim terdapat detail tentang permintaan mengajar atau pesanan yang dikirim kepada guru.



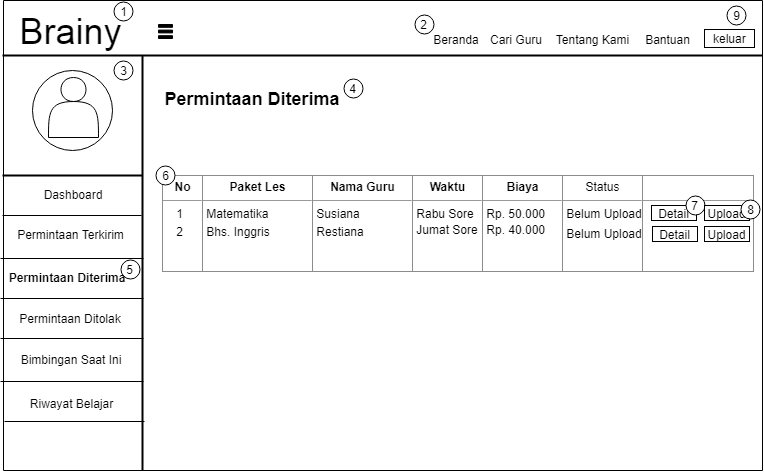
Gambar 5.9 Perancangan Antarmuka Permintaan Terkirim

Table 5.5 Keterangan Halaman Permintaan Terkirim

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Objek** | **Tipe** | **Keterangan** |
| 1 | Logo | Label | Nama dari sistem |
| 2 | Navbar Menu | Link | Di dalam navbar terdapat menu-menu yang dapat diakses seperti: beranda, cari guru, tentang kami dan bantuan. |
| 3 | Photo Profil | Image | Foto akun pribadi |
| 4 | Judul | Label | Digunakan untuk memberi informasi bahwa di bawah merupakan profil pribadi. |
| 5 | Sidebar Menu Permintaan Terkirim | Link | Menu permintaan terkirim yang ada di sidebar merupakan menu yang digunakan untuk mengakses halaman permintaan terkirim. |
| 6 | Tabel | Table | Tabel yang berisi informasi mengenai pemesanan guru les privat yang telah terkirim. |
| 7 | Detail | Button | Tombol yang digunakan apabila kita ingin melihat detail atau form yang telah dikirimkan kepada guru les privat yang dituju. |
| 8 | Keluar | Button | Digunakan untuk keluar dari autorisasi murid. |

* 1. **Perancangan Antarmuka Permintaan Diterima**

Permintaan Diterima merupakan halaman dimana permintaan mengajar telah diterima oleh guru les privat. Dalam permintaan diterima harus melakukan upload bukti pembayaran agar dapat melakukan bimbingan belajar. Murid akan menerima status belum upload bukti apabila belum melakukan upload bukti pembayaran dan diproses apabila telah upload bukti dan menunggu persetujuan admin.



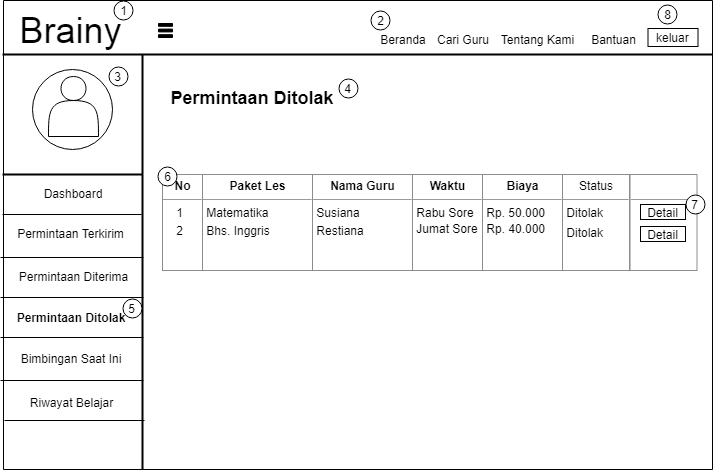
Gambar 5.10 Perancangan Antarmuka Permintaan Diterima

Table 5.6 Keterangan Halaman Permintaan Diterima

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Objek** | **Tipe** | **Keterangan** |
| 1 | Logo | Label | Nama dari sistem |
| 2 | Navbar Menu | Link | Di dalam navbar terdapat menu-menu yang dapat diakses seperti: beranda, cari guru, tentang kami dan bantuan. |
| 3 | Photo Profil | Image | Foto akun pribadi |
| 4 | Judul | Label | Digunakan untuk memberi informasi bahwa di bawah merupakan halaman permintaan diterima |
| 5 | Sidebar Menu Permintaan Diterima | Link | Menu permintaan diterima yang ada di sidebar merupakan menu yang digunakan untuk mengakses halaman permintaan diterima. |
| 6 | Tabel | Table | Tabel yang berisi informasi mengenai pemesanan guru les privat yang telah diterima |
| 7 | Detail | Button | Tombol yang digunakan apabila kita ingin melihat detail atau form yang telah dikirimkan kepada guru les privat yang dituju. |
| 8 | Upload | Button | Tombol yang digunakan untuk melakukan upload bukti pembayaran. |
| 9 | Keluar | Button | Digunakan untuk keluar dari autorisasi murid. |

* 1. **Perancangan Antarmuka** **Permintaan Ditolak**

Permintaan Ditolak menampilkan halaman apabila permintaan untuk melakukan les atau form yang diajukan ditolak.



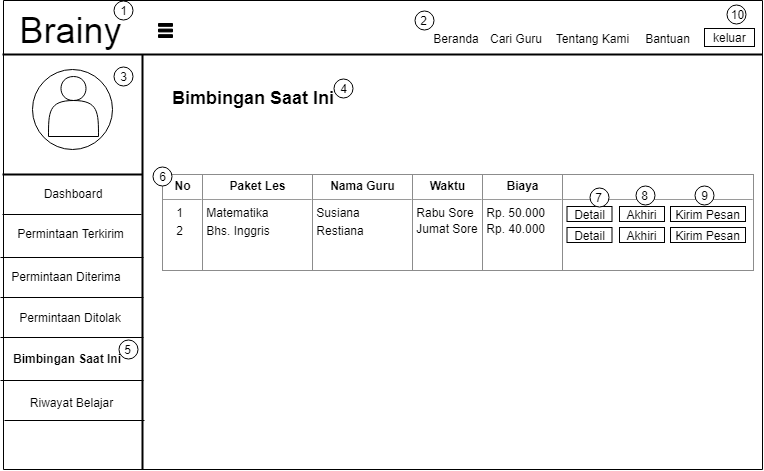
Gambar 5.11 Perancangan Antarmuka Permintaan Ditolak

Table 5.7 Keterangan Halaman Pemintaan Ditolak

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Objek** | **Tipe** | **Keterangan** |
| 1 | Logo | Label | Nama dari sistem |
| 2 | Navbar Menu | Link | Di dalam navbar terdapat menu-menu yang dapat diakses seperti: beranda, cari guru, tentang kami dan bantuan. |
| 3 | Photo Profil | Image | Foto akun pribadi |
| 4 | Judul | Label | Digunakan untuk memberi informasi bahwa halaman tersebut adalah halaman permintaan ditolak |
| 5 | Sidebar Menu Permintaan Ditolak | Link | Menu permintaan ditolak yang ada di sidebar merupakan menu yang digunakan untuk mengakses halaman perminntaan ditolak |
| 6 | Tabel | Table | Tabel yang berisi informasi mengenai pemesanan guru les privat yang telah ditolak. |
| 7 | Detail | Button | Tombol yang digunakan apabila kita ingin melihat detail informasi beserta alasan penolakan. |
| 8 | Keluar | Button | Digunakan untuk keluar dari autorisasi murid. |

* 1. **Perancangan Antarmuka** **Bimbingan Saat Ini**

Pada Gambar tedapat tampilan bimbingan saat ini, dimana untuk melihat bimbingan saat ini yang telah dilakukan. Bimbingan saat ini terdapat fungsi untuk melihat detail berupa form, akhiri untuk mengakhiri kegiatan belajar dan kirim pesan untuk saling berkirim pesan dengan guru les privat.



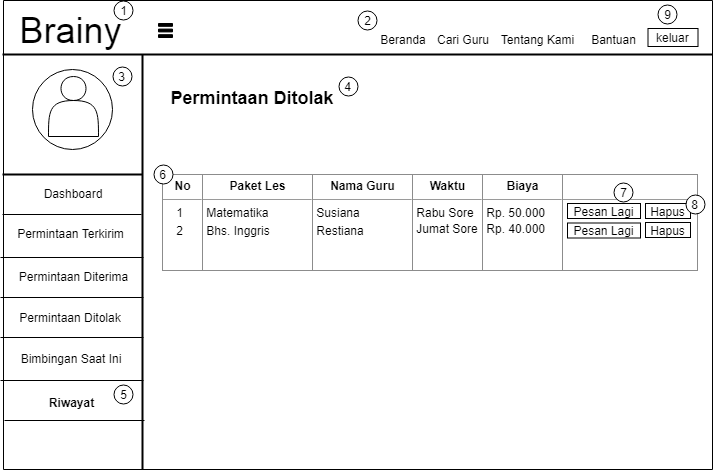
Gambar 5.12 Perancangan Antarmuka Bimbingan Saat Ini

Table 5.8 Keterangan Halaman Bimbingan Saat Ini

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Objek** | **Tipe** | **Keterangan** |
| 1 | Logo | Label | Nama dari sistem |
| 2 | Navbar Menu | Link | Di dalam navbar terdapat menu-menu yang dapat diakses seperti: beranda, cari guru, tentang kami dan bantuan. |
| 3 | Photo Profil | Image | Foto akun pribadi |
| 4 | Judul | Label | Digunakan untuk memberi informasi bahwa di bawah merupakan halaman bimbingan saat ini |
| 5 | Sidebar Menu Bimbingan Saat Ini | Link | Menu bimbingan saat ini yang ada di sidebar merupakan menu yang digunakan untuk mengakses halaman bimbingan saat ini |
| 6 | Tabel | Table | Tabel yang berisi informasi mengenai bimbingan belajar yang saat ini sedang berjalan. |
| 7 | Detail | Button | Tombol yang digunakan apabila kita ingin melihat detail atau form yang telah dikirimkan kepada guru les privat yang dituju. |
| 8 | Akhiri | Button | Tombol yang digunakan untuk mengakhiri kegiatan belajar. |
| 9 | Kirim Pesan | Button | Tombol yang digunakan untuk mengirim pesan kepada guru les privat yang ingin dituju. |
| 10 | Keluar | Button | Digunakan untuk keluar dari autorisasi murid. |

* 1. **Riwayat Belajar**

Pada Gambar merupakan tampilan riwayat belajar yang ada pada dashboar murid. Dan pada riwayat belajar terdapat fitur untuk memesan lagi dan hapus riwayat belajar.



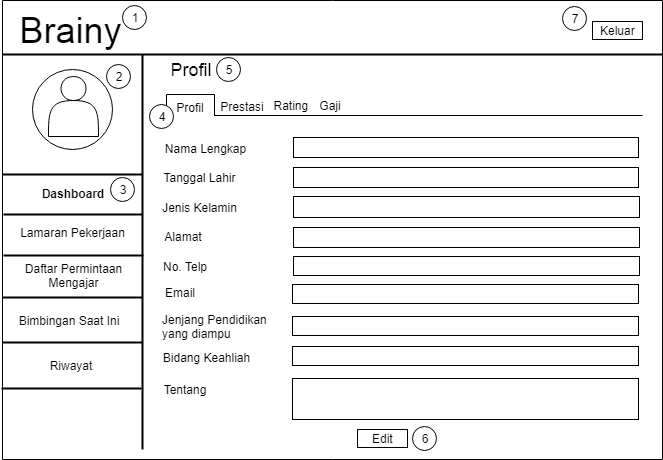
Gambar 5.13 Perancangan Antarmuka Riwayat Belajar

Table 5.9 Keterangan Halaman Riwayat Belajar

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Objek** | **Tipe** | **Keterangan** |
| 1 | Logo | Label | Nama dari sistem |
| 2 | Navbar Menu | Link | Di dalam navbar terdapat menu-menu yang dapat diakses seperti: beranda, cari guru, tentang kami dan bantuan. |
| 3 | Photo Profil | Image | Foto akun pribadi |
| 4 | Judul | Label | Digunakan untuk memberi informasi bahwa di bawah merupakan riwayat belajar |
| 5 | Sidebar Menu Riwayat Belajar | Link | Menu riwayat belajar yang ada di sidebar merupakan menu yang digunakan untuk mengakses halaman riwayat belajar |
| 6 | Tabel | Table | Tabel yang berisi informasi mengenai riwayat bimbingan belajar yang pernah dilakukan |
| 7 | Pesan Lagi | Button | Tombol yang digunakan apabila ingin memesan lagi guru les privat yang sebelumnya pernah belajar bersama. |
| 8 | Hapus | Button | Tombol yang digunakan untuk menghapus riwayat belajar. |
| 9 | Keluar | Button | Digunakan untuk keluar dari autorisasi murid. |

1. **Perancangan Antarmuka** **Dashboard Guru**

Dalam halaman dashboard guru terdapat beberapa menu seperti halaman lamaran pekerjaan, Daftar Permintaan Mengajar, Bimbingan Saat Ini, Riwayat dan Gaji. Serta dapat melihat profil pribadi dan mengedit profil.



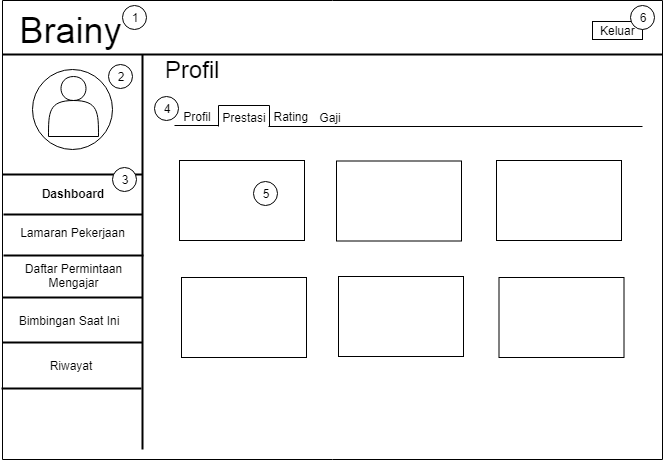
**Gambar 5.14** **Perancangan Antarmuka** **Dashboard Guru**

Table 5.10 Keterangan Halaman Dashboard Guru

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Objek** | **Tipe** | **Keterangan** |
| 1 | Logo | Label | Nama dari sistem |
| 2 | Foto Profil | Image | Ambar foto akun |
| 3 | Sidebar Menu | Link | Merupakan kumpulan menu-menu yang ada dalam sidebar seperti: dashboard, lamaran pekerjaan, daftar permintaan mengajar, bimbingan saat ini dan riwayat. |
| 4 | Menu Tab | Link | Terdapat beberapa menu yang ada dalam menu tab yaitu: profil, prestasi, rating dan gaji. |
| 5 | Judul | Label | Digunakan untuk memberikan informasi bahwa halaman tersebut merupakan halaman profil. |
| 6 | Edit | Button | Tombol yang digunakan untuk mengedit atau mengubah profil pribadi. |
| 7 | Keluar | Button | Tombol yang digunakan untuk keluar dari autorisasi guru les privat. |

* 1. **Perancangan Antarmuka** **Halaman Prestasi**

Pada gambar di bawah ini merupakan dashboard guru les privat yang terdapat pada tab menu prestasi. Dalam halama prestasi tersebut berisi foto-foto pendukung seperti foto sertifikat keahlian atau data pendukung lainnya.

****

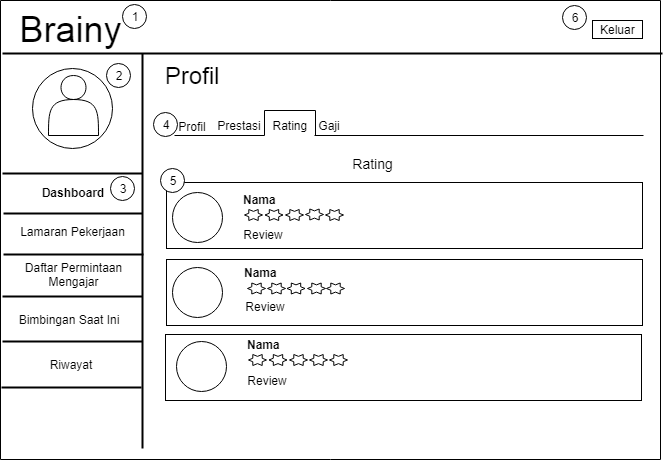
Gambar 5.15 Perancangan Antarmuka Halaman Prestasi

Table 5.11 Keterangan Halaman Prestasi

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Objek** | **Tipe** | **Keterangan** |
| 1 | Logo | Label | Nama dari sistem |
| 2 | Foto Profil | Image | Ambar foto akun |
| 3 | Sidebar Menu | Link | Merupakan kumpulan menu-menu yang ada dalam sidebar seperti: dashboard, lamaran pekerjaan, daftar permintaan mengajar, bimbingan saat ini dan riwayat. |
| 4 | Menu Tab Prestasi | Link | Menu tab prestasi digunakan untuk menampilkan halaman prestasi. |
| 5 | Foto | Image | Halaman prestasi terdapat foto-foto pendukung seperti sertifikat keahlian yang telah diupload oleh guru les privat |
| 6 | Keluar | Button | Tombol yang digunakan untuk keluar dari autorisasi guru les privat. |

* 1. **Perancangan Antarmuka** **Halaman Rating**

Pada gambar 5.16 merupakan halaman rating, dimana dalam halaman rating terdapat daftar masukan berupa rating dan ulasan dari murid yang telah melakukan bimbingan belajar dengan guru les privat sebelumnya.



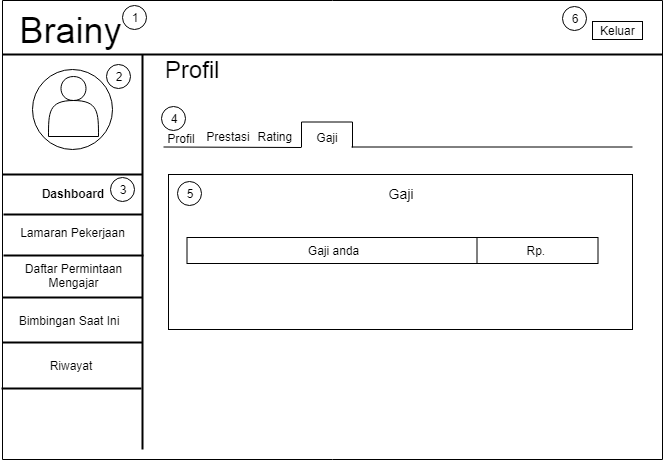
Gambar 5.16 Perancangan Antarmuka Halaman Rating

Table 5.12 Keterangan Halaman Rating

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Objek** | **Tipe** | **Keterangan** |
| 1 | Logo | Label | Nama dari sistem |
| 2 | Foto Profil | Image | Ambar foto akun |
| 3 | Sidebar Menu | Link | Merupakan kumpulan menu-menu yang ada dalam sidebar seperti: dashboard, lamaran pekerjaan, daftar permintaan mengajar, bimbingan saat ini dan riwayat. |
| 4 | Menu Tab Rating | Link | Menu tab rating digunakan untuk menampilkan halaman rating |
| 5 | Ulasan | Text | Pada bagian ulasan terdapat nama pemberi ulasan dan foto serta terdapat rating dan review yang diberikan. |
| 6 | Keluar | Button | Tombol yang digunakan untuk keluar dari autorisasi guru les privat. |

* 1. **Perancangan Antarmuka** **Halaman Gaji**

Pada gambar 5.17 merupakan halaman dashboard di bagian tab menu gaji. Dimana di dalamnya terdapat jumlah gaji yang dimiliki guru les privat.

****

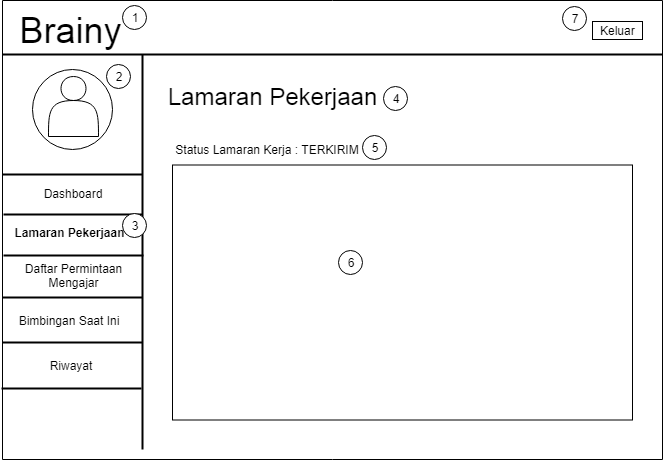
Gambar 5.17 Perancangan Antarmuka Halaman Gaji

Table 5.13 Keterangan Halaman Gaji

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Objek** | **Tipe** | **Keterangan** |
| 1 | Logo | Label | Nama dari sistem |
| 2 | Foto Profil | Image | Ambar foto akun |
| 3 | Sidebar Menu | Link | Merupakan kumpulan menu-menu yang ada dalam sidebar seperti: dashboard, lamaran pekerjaan, daftar permintaan mengajar, bimbingan saat ini dan riwayat. |
| 4 | Menu Tab Rating | Link | Menu tab gaji digunakan untuk menampilkan halaman gaji |
| 5 | Gaji | Text | Pada bagian gaji digunakan untuk melihat jumlah gaji yang telah terkumpul. |
| 6 | Keluar | Button | Tombol yang digunakan untuk keluar dari autorisasi guru les privat. |

* 1. **Perancangan Antarmuka** **Lamaran Pekerjaan**

Gambar dibawah merupakan gambar halaman lamaran pekerjaan guru les privat.



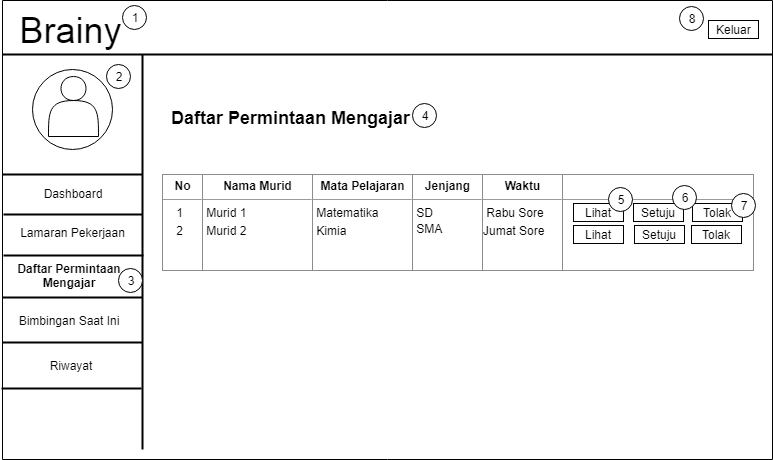
Gambar 5.18 Perancangan Antarmuka Lamaran Pekerjaan

Table 5.14 Keterangan Halaman Lamaran Pekerjaan

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Objek** | **Tipe** | **Keterangan** |
| 1 | Logo | Label | Nama dari sistem |
| 2 | Foto Profil | Image | Ambar foto akun |
| 3 | Sidebar Menu | Link | Merupakan kumpulan menu-menu yang ada dalam sidebar seperti: dashboard, lamaran pekerjaan, daftar permintaan mengajar, bimbingan saat ini dan riwayat. |
| 4 | Judul | Label | Digunakan untuk menunjukkan bahwa halaman yang sedang aktif adalah halaman lamaran pekerjaan |
| 5 | Status Lamaran Kerja | Label | Pada bagian ini menunjukkan status lamaran pekerjaan. Apakah lamaran tersebut terkirim diterima atau ditolak. |
| 6 | Form Lamaran Kerja | Label | File tersebut merupakan file lamaran kerja yang hanya bisa dilihat oleh guru les privat. |
| 7 | Keluar | Button | Tombol yang digunakan untuk keluar dari autorisasi guru les privat. |

* 1. **Perancangan Antarmuka** **Daftar Permintaan Mengajar**

Halaman permintaan mengajar untuk melihat daftar murid yang ingin melakukan bimbingan belajar. Dalam halaman permintaan mengajar terdapat terdapat fitur untuk melihat detail dari pemesanan, menyetujui permintaan belajar dan untuk menolak permintaan.

****

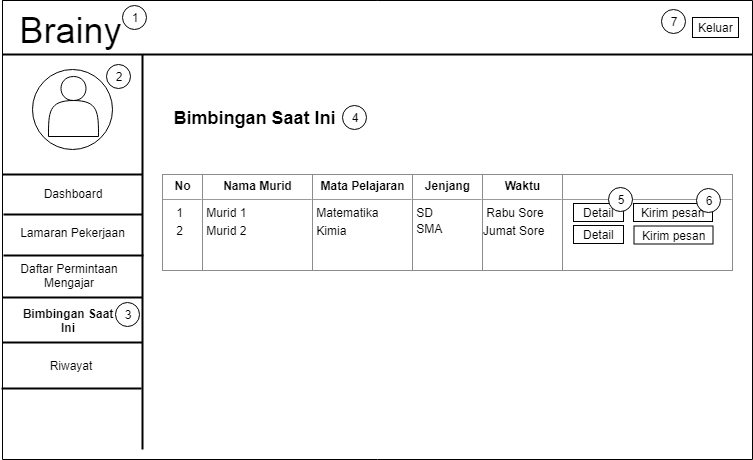
Gambar 5.19 Perancangan Antarmuka Daftar Permintaan Mengajar

Table 5.15 Keterangan Halaman Daftar Permintaan Mengajar

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Objek** | **Tipe** | **Keterangan** |
| 1 | Logo | Label | Nama dari sistem |
| 2 | Foto Profil | Image | Ambar foto akun |
| 3 | Sidebar Menu | Link | Merupakan kumpulan menu-menu yang ada dalam sidebar seperti: dashboard, lamaran pekerjaan, daftar permintaan mengajar, bimbingan saat ini dan riwayat. |
| 4 | Judul | Label | Digunakan untuk menunjukkan bahwa halaman yang sedang aktif adalah halaman daftar permintaan mengajar. |
| 5 | Lihat | Button | Tombol yang digunakan untuk melihat form permintaan mengajar secara lebih detail. |
| 6 | Setuju | Button | Tombol yang digunakan untuk menyetujui permintaan. |
| 7 | Tolak | Button | Tombol yang digunakan untuk menolak permintaan. |
| 8 | Keluar | Button | Tombol yang digunakan untuk keluar dari autorisasi guru les privat. |

* 1. **Perancangan Antarmuka** **Bimbingan Saat Ini**

Pada halaman bimbingan saat ini digunakan untuk melihat daftar bimbingan saat ini dengan murid. Dalam halaman bimbingan terdapat fitur detail untuk melihat lebih detail dan fitur untuk mengirim pesan.

****

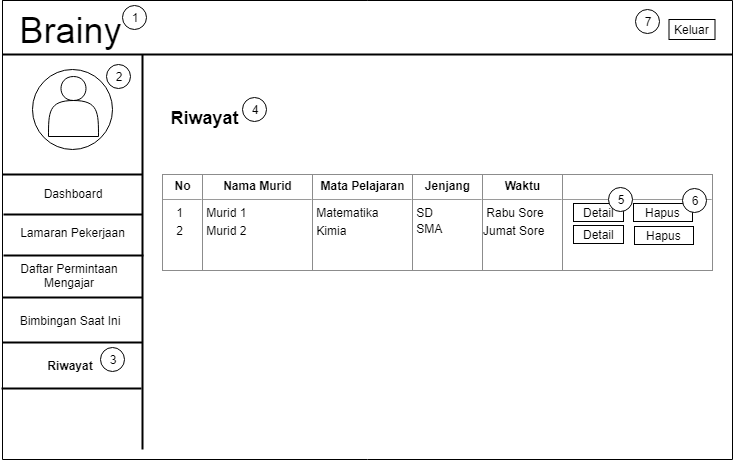
**Gambar 5.20 Perancangan Antarmuka** **Bimbingan Saat Ini**

Table 5.16 Keterangan Halaman Bimbingan Saat Ini

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Objek** | **Tipe** | **Keterangan** |
| 1 | Logo | Label | Nama dari sistem |
| 2 | Foto Profil | Image | Ambar foto akun |
| 3 | Sidebar Menu | Link | Merupakan kumpulan menu-menu yang ada dalam sidebar seperti: dashboard, lamaran pekerjaan, daftar permintaan mengajar, bimbingan saat ini dan riwayat. |
| 4 | Judul | Label | Digunakan untuk menunjukkan bahwa halaman yang sedang aktif adalah halaman bimbingan saat ini. |
| 5 | Detail | Button | Tombol yang digunakan untk melihat lebih detail informasi bimbingan belajar. |
| 6 | Kirim Pesan | Button | Tombol yang digunakan untuk mengirim pesan ke murid yang dituju. |
| 7 | Keluar | Button | Tombol yang digunakan untuk keluar dari autorisasi guru les privat. |

* 1. **Perancangan Antarmuka** **Riwayat**

Pada halaman riwayat digunakan untuk melihat riwayat mengajar. Pada halaman ini terdapat fitur melihat detail dan untuk menghapus riwayat.

****

**Gambar 5.21 Perancangan Antarmuka** **Riwayat**

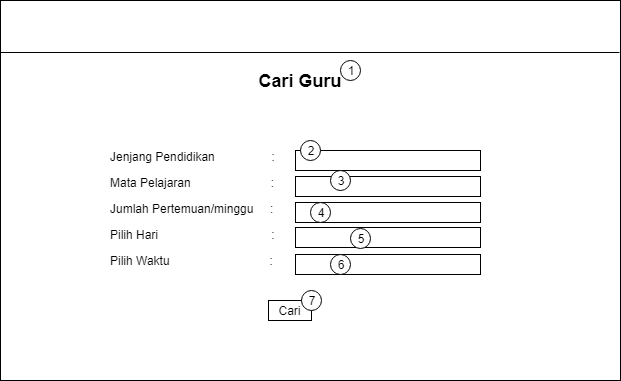
Table 5.17 Keterangan Halaman Riwayat

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Objek** | **Tipe** | **Keterangan** |
| 1 | Logo | Label | Nama dari sistem |
| 2 | Foto Profil | Image | Ambar foto akun |
| 3 | Sidebar Menu | Link | Merupakan kumpulan menu-menu yang ada dalam sidebar seperti: dashboard, lamaran pekerjaan, daftar permintaan mengajar, bimbingan saat ini dan riwayat. |
| 4 | Judul | Label | Digunakan untuk menunjukkan bahwa halaman yang sedang aktif adalah halaman riwayat. |
| 5 | Detail | Button | Tombol yang digunakan untuk melihat lebih detail informasi bimbingan belajar. |
| 6 | Hapus | Button | Tombol yang digunakan untuk menghapus riwayat. |
| 7 | Keluar | Button | Tombol yang digunakan untuk keluar dari autorisasi guru les privat. |

1. **Perancangan Antarmuka** **Cari Guru**

Halaman cari guru digunakan untuk melakukan pencarian guru yang di dalamnya terdapat form yang digunakan untuk memfilter guru dengan kriteria yang

diinginkan.



Gambar 5.22 Perancangan Antarmuka Cari Guru

Table 5.18 Keterangan Halaman Cari Guru

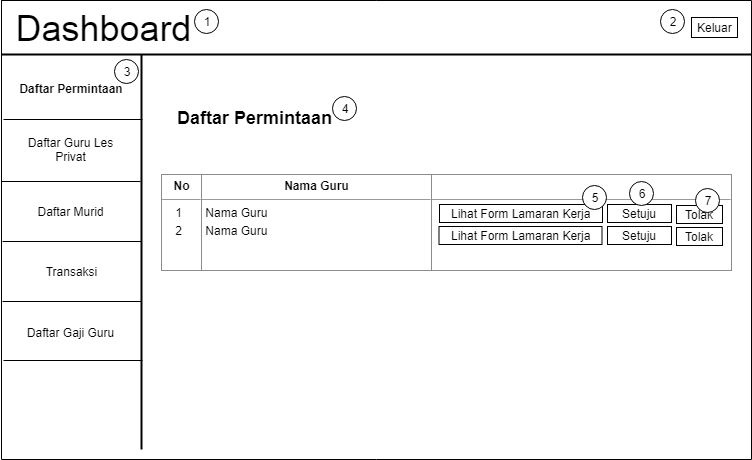
|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Objek** | **Tipe** | **Keterangan** |
| 1 | Judul | Label | Digunakan untuk memberi keterangan bahwa halaman tersebut merupakan halaman untuk melakukan pencarian guru. |
| 2 | Text Field Jenjang Pendidikan | Text Field | Isian yang diberikan berupa jenjang pendidikan |
| 3 | Text Field Mata Pelajaran | Text Field | Isian yang diberikan berupa nama mata pelajaran |
| 4 | Text Field Jumlah Pertemuan | Text Field | Isian yang diberikan berupa angka jumlah pertemuan per minggu |
| 5 | Text Field Pilih Hari | Text Field | Isian yang diberikan berupa hari |
| 6 | Text Field Pilih Waktu | Text Field | Isian yang diberikan berupa waktu yang diinginkan saat melakukan kegiatan belajar. |
| 7 | Cari | Button | Tombol yang digunakan untuk mencari guru les apabila isian telah diisi semua. |

1. **Perancangan Antarmuka** **Dashboard Admin**

Dalam halaman dashboard admin terdapat beberapa menu seperti halaman daftar permintaan, daftar guru les privat, daftar murid, transaksi dan gaji.

* 1. **Perancangan Antarmuka** **Daftar Permintaan**

Halaman daftar permintaan merupakan halaman dari daftar guru yang ingin mengajukan diri sebagai member guru les privat di brainy. Dalam halaman permintaan terdapat fitur untuk melihat dokumen lamaran pekerjaan untuk melihat dokumen lamaran pekerjaan yang telah terkirim, fitur setuju untuk menerima sebagai guru les privat dan fitur tolak untuk menolak.

****

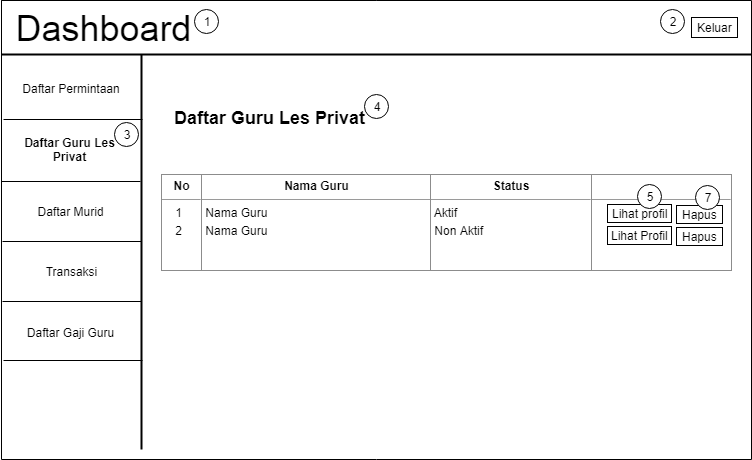
**Gambar 5.23 Perancangan Antarmuka** **Daftar Permintaan**

Table 5.19 Keterangan Halaman Daftar Permintaan

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Objek** | **Tipe** | **Keterangan** |
| 1 | Logo | Label | Digunakan untuk menunjukkan bahwa halaman tersebut merupakan halaman dashboard |
| 2 | Keluar | Button | Digunakan untuk keluar dari autorisasi admin |
| 3 | Sidebar | Link | Dalam side bar terdapat menu-menu seperti: daftar permintaan, daftar guru les privat, daftar murid, transaksi dan daftar gaji guru. |
| 4 | Judul | Label | Digunakan untuk menunjukkan bahwa halaman yang aktif merupakan halaman daftar permintaan. |
| 5 | Lihat Form Lamaran Kerja | Button | Digunakan untuk melihat halaman form lamaran kerja |
| 6 | Setuju | Button | Digunakan untuk menyetujui permintaan guru les privat |
| 7 | Tolak | Button | Digunakan untuk menolak permintaan guru les privat. |

* 1. **Perancangan Antarmuka** **Daftar Guru Les Privat**

Halaman daftar guru les privat menampilkan daftar guru les privat yang telah menjadi member dari brainy. Dalam halaman daftar guru les privat terdapat fitur untuk melihat profil guru dan fitur untuk menghapus guru.

****

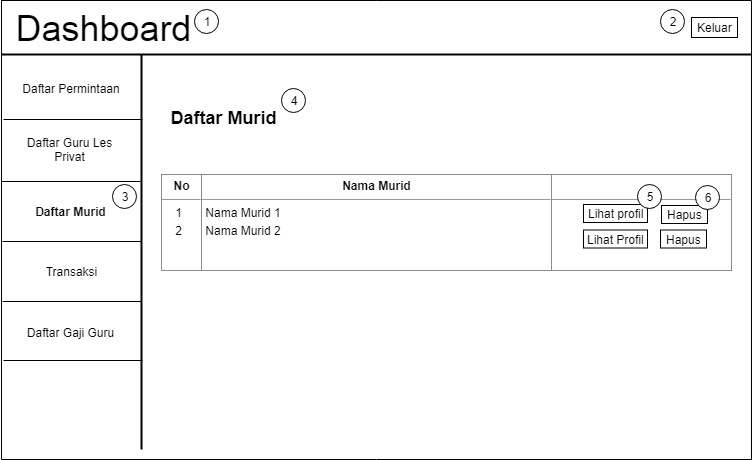
Gambar 5.24 Perancangan Antarmuka Daftar Guru Les Privat

Table 5.20 Keteranga Halaman Daftar Guru Les Privat

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Objek** | **Tipe** | **Keterangan** |
| 1 | Logo | Label | Digunakan untuk menunjukkan bahwa halaman tersebut merupakan halaman dashboard |
| 2 | Keluar | Button | Digunakan untuk keluar dari autorisasi admin |
| 3 | Sidebar | Link | Dalam side bar terdapat menu-menu seperti: daftar permintaan, daftar guru les privat, daftar murid, transaksi dan daftar gaji guru. |
| 4 | Judul | Label | Digunakan untuk menunjukkan bahwa halaman yang aktif merupakan halaman daftar guru les privat. |
| 5 | Lihat Profil | Button | Digunakan untuk melihat halaman profil guru. |
| 6 | Hapus | Button | Digunakan untuk menghapus guru les privat. |

* 1. **Perancangan Antarmuka** **Daftar Murid**

Halaman daftar murid merupakan halaman yang digunakan untuk melihat daftar murid-murid yang telah mendaftarkan diri di brainy. Terdapat fitur lihat profil untuk melihat profil murid dan fitur hapus untuk menghapus murid dari daftar.

****

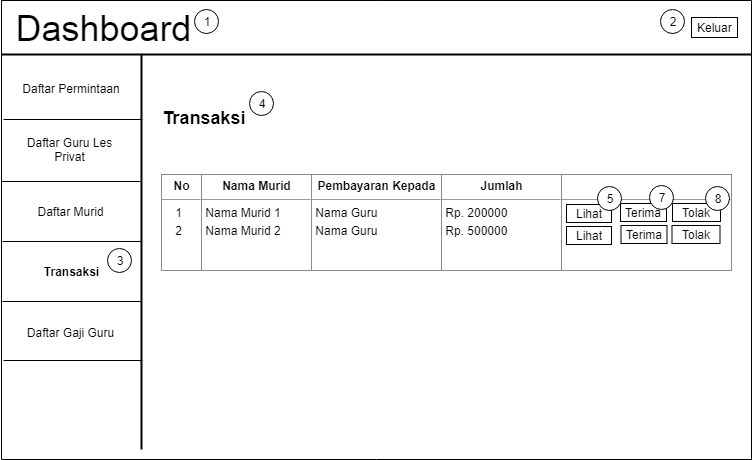
**Gambar 5.25 Perancangan Antarmuka** **Daftar Murid**

Table 5.21 Keterangan Halaman Daftar Murid

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Objek** | **Tipe** | **Keterangan** |
| 1 | Logo | Label | Digunakan untuk menunjukkan bahwa halaman tersebut merupakan halaman dashboard |
| 2 | Keluar | Button | Digunakan untuk keluar dari autorisasi admin |
| 3 | Sidebar | Link | Dalam side bar terdapat menu-menu seperti: daftar permintaan, daftar guru les privat, daftar murid, transaksi dan daftar gaji guru. |
| 4 | Judul | Label | Digunakan untuk menunjukkan bahwa halaman yang aktif merupakan halaman daftar murid yang telah bergabung. |
| 5 | Lihat Profil | Button | Digunakan untuk melihat halaman profil murid. |
| 6 | Hapus | Button | Digunakan untuk menghapus murid. |

1. **Perancangan Antarmuka** **Transaksi**

Halaman transaksi merupakan halaman untuk menampung hasil transaksi yang dilakukan oleh murid kepada guru yang bersangkutan. Terdapat fitur lihat untuk melihat bukti pembayaran yang telah dilakukan oleh murid, fitur terima untuk menerima transaksi dan fitur tolak untuk menolak transaksi karena alasan tertentu.

****

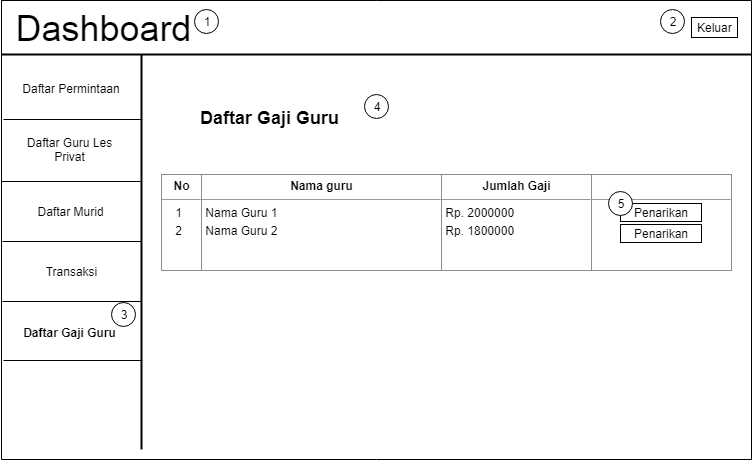
**Gambar 5.26 Perancangan Antarmuka** **Transaksi**

Table 5.22 Keterangan Halaman Transaksi

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Objek** | **Tipe** | **Keterangan** |
| 1 | Logo | Label | Digunakan untuk menunjukkan bahwa halaman tersebut merupakan halaman dashboard |
| 2 | Keluar | Button | Digunakan untuk keluar dari autorisasi admin |
| 3 | Sidebar | Link | Dalam side bar terdapat menu-menu seperti: daftar permintaan, daftar guru les privat, daftar murid, transaksi dan daftar gaji guru. |
| 4 | Judul | Label | Digunakan untuk menunjukkan bahwa halaman yang aktif merupakan halaman transaksi. |
| 5 | Lihat | Button | Digunakan untuk melihat hasil bukti transaksi yang telah diupload leh murid yang ingin melakukan bimbingan belajar atau les. |
| 5 | Terima | Button | Digunakan untuk menerima bukti pembayaran bahwa bukti pembayaran yang diunggah adalah benar. |
| 6 | Tolak | Button | Digunakan untuk menolak transaksi. |

1. **Perancangan Antarmuka** **Daftar Gaji Guru**

Menu daftar gaji guru dalam halaman admin digunakan untuk melihat daftar gaji yang dimiliki oleh guru-guru yang ada. Dan pada halaman gaji terdapat fitur untuk melakukan penarikan gaji apabila guru ingin mengambil gajinya.

****

**Gambar 5.27 Perancangan Antarmuka** **Gaji**

Table 5.23 Keterangan Halaman Gaji

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Objek** | **Tipe** | **Keterangan** |
| 1 | Logo | Label | Digunakan untuk menunjukkan bahwa halaman tersebut merupakan halaman dashboard |
| 2 | Keluar | Button | Digunakan untuk keluar dari autorisasi admin |
| 3 | Sidebar | Link | Dalam side bar terdapat menu-menu seperti: daftar permintaan, daftar guru les privat, daftar murid, transaksi dan daftar gaji guru. |
| 4 | Judul | Label | Digunakan untuk menunjukkan bahwa halaman yang aktif merupakan halaman transaksi. |
| 5 | Penarikan | Button | Apabila guru ingin melakukan penarikan sejumlah uang yang telah terkumpul. |

1. **Implementasi Sistem**

# daftar pustaka

Dwanoko, Y. S. (2016). Implementasi Software Development Life Cycle(SDLC) Dalam Penerapan Pembangunan Aplikasi Perangkat Lunak. *Jurnal Teknologi Informasi*.

Essyana, N., & Qoiriah , A. (2016). 'Go- Teacher' Sistem Pencarian Guru Les Privat Berbasis Web. *Jurnal Manajemen Informasi*.

Fahrurrozi, I., & SN, A. (2012). Poses Pemodelan Software Dengan Metode Waterfall dan Extremme Programming: Studi Perbandingan. *Jurnal ONLINE STMIK EL- Rahma*.

Gandy, E., & Stobart, S. (2005). *JavaScript: Creating Dynamic Web Pages.* Colchester: Lexden Publishing.

Haryana, K. S. (2008). Pengembangan Perangkat Lunak Dengan Menggunakan PHP. *Jurnal Computech & Bisnis*.

Hustinawati, Himawan, A. K., & Latifah. (2014). Performance Analysis Framework Codeigniter anda CakePHP in Website Creation. *International Journal of Computer Application*.

Kristiyanti, L., Sugiharto, A., & W, H. A. (2013). Sistem Pendukung Keputusan Pemilihan Pengajar Les Privat Untuk Siswa Lembaga Bimbingan Belajar Dengan Metode AHP(Studi Kasus LBB Sistem Cerdas). *Jurnal Masyarakat Informatika*.

Meyer, E. A. (2004). *Cascanding Style Sheets: The Definitive Guide.* USA: O'Reilly Media.

Nidhra, S., & Dondeti, J. (2012). Black Box And White Box Testing Techniques - A Literature Review. *International Journal of Embedded Systems and Application(IJESA)*.

Nugroho, A. (2009). *Rekayasa Perangkat Lunak Menggunakan UML dan JAVA.* Yogyakarta: ANDI Yogyakarta.

Oktavian, D. P. (2010). *Menjadi Programmer Jempolan Menggunakan PHP.* Yogyakarta: MediaKom.

Putra, A. K., Nyoto, R. D., & Pratiwi, H. S. (2017). Rancang Bangun Aplikasi Marketplace Penyedia Jasa Les Privat di Kota Pontianak. *Jurnal Sistem dan Teknologi Informasi*.

Simarmata, J. (2010). *Rekayasa Perangkat Lunak.* Yogyakarta: Penerbit ANDI.

Sularso, P., Kokotiasa, W., & Paramedica, D. A. (2015). Pengaruh Bimbingan Belajar Di Luar Sekolah Terhadap Minat Untuk Berprestasi Siswa Kelas VIII MTsN Sidorejo Kabupaten Madiun Tahun Pelajaran 2014/2015. *Citizenship Jurnal Pancasila dan Kewarganegaraan*.

Whitten, J. L., Benley, L. D., & Dittman, K. C. (2001). *Systems Analysis And Design Methods.* New York: David Kendric Brake.

Whitten, J. L., Bentley, L. D., & Dittman, K. C. (2004). *Metode Desain & Analisis Sistem Edisi 6.* Yogyakarta: Penerbit ANDI.