**LAPORAN KERJA PRAKTEK**

**APLIKASI PENGELOLA DATA AKADEMIK MENGGUNAKAN CODEIGNITER VERSI 4.1.1 UNTUK SMA PELITA BUNGA BANGSA ARJASARI**

Diajukan untuk memenuhi persyaratan kelulusan

Matakuliah TIF 335 Kerja Praktek

Oleh:

### **Muhamad Aksal Dzilhamsyah/ 301180052**

Logo

Description automatically generated

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INFORMASIUNIVERSITAS BALE BANDUNG**

**2022**

**LEMBAR PENGESAHAN**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**APLIKASI PENGELOLA DATA AKADEMIK MENGGUNAKAN CODEIGNITER VERSI 4.1.1 UNTUK SMA PELITA BUNGA BANGSA ARJASARI**

Oleh:

Muhamad Aksal Dzilhamsyah/ 301180052

disetujui dan disahkan sebagai

### **LAPORAN KERJA PRAKTEK**

Bandung, 21 Maret 2022

Koordinator Kerja Praktek Program Studi Teknik Informatika

Yusuf Muharam, M.Kom.

NIK: 04104820003

**LEMBAR PENGESAHAN**

**SMA PELITA BUNGA BANGSA ARJASARI**

**APLIKASI PENGELOLA DATA AKADEMIK MENGGUNAKAN CODEIGNITER VERSI 4.1.1 UNTUK SMA PELUTA BUNGA BANGSA ARJASARI**

Oleh:

Muhamad Aksal Dzilhamsyah/ 301180052

disetujui dan disahkan sebagai

### **LAPORAN KERJA PRAKTEK**

Bandung, 21 Maret 2022

Kepala unit pelaksana Teknik Teknologi Informasi

Khoerudin Sidiq, S.T.

NIK+

# ABSTRAK

Kerja Praktek dilaksanakan di SMA Pelita Bunga Bangsa, mulai tanggal 18 November 2021 sampai dengan tanggal 18 Desember 2021. Kerja praktek yang dilakukan adalah Pembuatan Aplikasi Sistem Informas Akademik Menggunakan Codeigniter 4.1.1 Fitur yang ada dalam sistem tersebut antara lain profil sekolah, data guru, data siswa, data kelas, data mata pelajaran,tahun ajaran, login. Selama pembuatan aplikasi, metodologi yang digunakan adalah *Software Development Life Cycle* (SDLC) dengan menggunakan model *Waterfall*. Tahap pertama adalah *Requirements Analysis and Definition* yakni mengumpulkan kebutuhan secara lengkap kemudian di analisis. Tahap kedua adalah *System and Software Design* yakni mengalokasikan kebutuhan-kebutuhan sistem baik perangkat keras maupun perangkat lunak. Tahap ketiga adalah *Implementation* yakni pengkodean dari desain ke dalam suatu bahasa pemrograman. Tahap keempat adalah *Integration and System Testing* yakni program digabung dan diuji sebagai sebuah sistem lengkap untuk memastikan apakah sesuai dengan kebutuhan perangkat lunak atau tidak. Tahap terakhir adalah *Operation and Maintenance* yakni tahapan yang paling panjang, sistem dipasang dan digunakan secara nyata. Pada akhir kerja praktek telah berhasil menciptakan sebuah aplikasi untuk mengelola data akademik dan membantu SMA Pelita Bunga Bangsa Arjasari khususnya dalam proses mengelola data akademik.Kesimpulan dari keseluruhan proses kerja praktek adalah dengan dibuatnya Aplikasi Sistem Informasi Akademik ini dapat mempermudah pendataan dan akses informasi data akademik.

Kata Kunci : *Codeigniter V4.1.1, Data Akademik, Kerja Praktek, SMA Pelita Bunga Bangsa, UML Diagram Waterfall*

# KATA PENGANTAR

Puji syukur alhamdulillah penulis panjatkan kehadirat Allah SWT yang telah melimpahkan rahmat-Nya serta hidayah-Nya sehingga penulis dapat menyelesaikan laporan kerja praktek dengan judul “Aplikasi Sistem Informasi Akademik Menggunakan Codeigniter Versi 4.1.1”.

Laporan ini dibuat dengan berbagai observasi dan beberapa bantuan dari berbagai pihak untuk membantu menyelesaikan tantangan dan hambatan selama mengerjakan laporan ini. Oleh karena itu, penulis mengucapkan terima kasih yang sebesar-besarnya kepada :

1. Bapak Yudi Herdiana, S.T., M.T. selaku Dekan Fakultas Teknologi Informasi sekaligus dosen pembimbing kerja praktek.
2. Bapak Yusuf Muharam, M.Kom. selaku Ketua Program Studi S1 Teknik Informatika Universitas Bale Bandung sekaligus Koordinator Kerja Praktek.
3. Sahabat sekalian mahasiswa angkatan 2018 yang telah memberikan do’a dan mendorong semangat untuk penyusunan laporan ini.
4. Kedua orang tua yang tidak pernah letih memberikan bimbingan, dukungan, kepercayaan, dan do’a serta nasehatnya untuk keberhasilan penulis.

Laporan ini diharapkan bisa menjadi referensi untuk semua pihak serta bermanfaat bagi yang membutuhkannya. Dan juga Penulis mohon maaf apabila terdapat kesalahan-kesalahan serta kekurangannya.

Salam, Penulis

# DAFTAR ISI

[**LEMBAR PENGESAHAN PROGRAM STUDI**  i](#_Toc97595071)

[**LEMBAR PENGESAHAN INSTANSI** ii](#_Toc97595072)

[**ABSTRAK** iii](#_Toc97595073)

[**KATA PENGANTAR** iv](#_Toc97595074)

[**DAFTAR ISI** vii](#_Toc97595075)

[**DAFTAR GAMBAR** vii](#_Toc97595075)

[**DAFTAR TABEL** ix](#_Toc97595076)

[**BAB I PENDAHULUAN** 1](#_Toc97595077)

[I.1 Latar Belakang 1](#_Toc97595078)

[I.2 Lingkup 2](#_Toc97595079)

[I.3 Tujuan 3](#_Toc97595080)

[**BAB II LINGKUP KERJA PRAKTEK** 4](#_Toc97595081)

[II.1 Struktur Organisasi 4](#_Toc97595082)

[II.2 Lingkup Pekerjaan 5](#_Toc97595083)

[II.3 Deskripsi Pekerjaan 5](#_Toc97595084)

[II.4 Jadwal Kerja 6](#_Toc97595085)

[**BAB III TEORI PENUNJANG KERJA PRAKTEK**  8](#_Toc97595086)

[III.1 Teori Penunjang 8](#_Toc97595087)

[III.2 Peralatan Pembuatan Aplikasi Web 17](#_Toc97595088)

[**BAB IV PELAKSANAAN KERJA PRAKTEK 33**](#_Toc97595089)

[IV.1 Input 33](#_Toc97595091)

[IV.2 Proses 34](#_Toc97595092)

[IV.2.1 Eksplorasi 34](#_Toc97595093)

[IV.2.2 Pembuatan Aplikasi Sistem Informasi Akademik Sekolah b Menengah Atas 36](#_Toc97595094)

[IV.2.3 Pelaporan Hasil Kerja Praktek 80](#_Toc97595095)

[IV.3 Pencapaian Hasil 81](#_Toc97595096)

[**BAB V PENUTUL 91**](#_Toc97595097)

[V.1Kesimpulan dan saran mengenai pelaksanaan 91](#_Toc97595099)

[V.1.1.Kesimpulan Pelaksanaan Kerja Praktek 91](#_Toc97595100)

[V.1.2Saran Pelaksanaan KP 91](#_Toc97595101)

[V.2Kesimpulan dan saran mengenai substansi 92](#_Toc97595102)

[V.2.1Kesimpulan Pembuatan Aplikasi Sistem Informasi Akademik ………………………………………………………………………….92](#_Toc97595103)

[V.2.2Saran mengenai Aplikasi Sistem Informasi Akademik 92](#_Toc97595104)

[**DAFRTAR PUSTAKA xi**](#_Toc97595105)

## **DAFTAR GAMBAR**

[Gambar II.1 Struktur Organisasi 5](#_Toc97549278)

[Gambar III. 1 Model Waterfall 10](#_Toc97549279)

[Gambar III. 3 Htdocs 18](#_Toc97549281)

[Gambar III. 4 Phpmyadmin 19](#_Toc97549282)

Gambar III. 5 Control Panel……………………………...………………………...19

[Gambar IV. 1 *Use Case Diagram* 40](#_Toc97549280)

[Gambar IV. 2 *Activity Diagram* Login 43](#_Toc97549281)

[Gambar IV. 3 *Activity Diagram* Profil Sekolah 43](#_Toc97549282)

Gambar IV. 4 *Activity Diagram* tahun ajaran…………….………….………...44

[Gambar IV. 5 *Activity Diagram* Asisten data siswa 45](#_Toc97549280)

[Gambar IV. 6 *Activity Diagram* Data Kelas 46](#_Toc97549281)

[Gambar IV. 7 *Activity Diagram* Dara Rombel 47](#_Toc97549282)

[Gambar IV. 8 *Activity Diagram* Data Wali Kelas 48](#_Toc97549281)

[Gambar IV. 9 *Activity Diagram* Data Guru 49](#_Toc97549282)

[Gambar IV. 10 *Activity Diagram* Mata Pelajaran 50](#_Toc97549280)

[Gambar IV. 6 *Activity Diagram* Data Kelas 51](#_Toc97549281)

[Gambar IV. 11 *Activity Diagram* Data pengguna 52](#_Toc97549282)

[Gambar IV. 12 *Cass Diagram* 53](#_Toc97549281)

[Gambar IV. 13 Desain Tampilan Home 53](#_Toc97549282)

[Gambar IV. 15 Desain Tampilan Profil Sekolah 54](#_Toc97549280)

[Gambar IV. 16 DesainTampilan Data Tahun Ajaran 54](#_Toc97549281)

[Gambar IV. 17 Desain Tampilan Data Siswa 55](#_Toc97549282)

[Gambar IV.18 Desain Tampilan Data Kelas 55](#_Toc97549281)

[Gambar IV.19 Desain Tampilan Data Rombel 56](#_Toc97549282)

[Gambar IV.20 Desain Tampilan Data Wali Kelas 56](#_Toc97549280)

[Gambar IV.21 Desain Tampilan Data Guru 57](#_Toc97549281)

[Gambar IV.22 Desain Tampilan Data Mata Pelajaran 57](#_Toc97549282)

[Gambar IV.23 Desain Tampilan Data Pengguna 58](#_Toc97549281)

[Gambar IV.24 Database login 63](#_Toc97549282)

[Gambar IV.25 Database tahun ajaran 63](#_Toc97549282)

[Gambar IV.26 Database Siswa 64](#_Toc97549280)

[Gambar IV.27 Database mata pelajaran 64](#_Toc97549281)

[Gambar IV.28 Database Guru 65](#_Toc97549282)

[Gambar IV.29 Relasi Database 65](#_Toc97549281)

[Gambar IV.30 Tampilan Dashboard 66](#_Toc97549282)

[Gambar IV.31 Tampilan Profil Sekolah 66](#_Toc97549282)

[Gambar IV.32 Tampilan Login 67](#_Toc97549280)

[Gambar IV.33 Tampilan Tahun Ajaran 67](#_Toc97549281)

[Gambar IV.34 Tampilan Data Siswa 68](#_Toc97549282)

[Gambar IV.35 Tampilan Tambah Data Kelas 68](#_Toc97549281)

[Gambar IV.36 Tampilan Edit Data Rombongan Belajar 69](#_Toc97549282)

[Gambar IV.37 Tampilan Data Wali Kelas 69](#_Toc97549282)

[Gambar IV.38 Tampilan Data Guru 70](#_Toc97549280)

[Gambar IV.39 Tampilan Data Mata Pelajaran 70](#_Toc97549281)

[Gambar IV.40 Tampilan Data Pengajar 71](#_Toc97549282)

[Gambar IV.41 Tampilan Data Pengguna 71](#_Toc97549281)

## **DAFTAR TABEL**

[Tabel III.1 Use Case Diagram 14](#_Toc97549279)

[Tabel III. 2 Activity Diagram 15](#_Toc97549280)

[Tabel IV.1 Perangkat Keras 37](#_Toc97549281)

[Tabel III. 2 Activity Diagram 36](#_Toc97549280)

[Tabel IV.3 Perangkat Lunak 38](#_Toc97549282)

[Tabel IV.4 Skenario Use Case Login 41](#_Toc97549279)

[Tabel IV.5 Skenario Use Case Cek Login 41](#_Toc97549280)

[Tabel IV.6 Skenario Use Case Menampilkan/Melihat Data 41](#_Toc97549281)

[Tabel IV.7 Skenario Use Case Tambah Profil Sekolah 41](#_Toc97549282)

[Tabel IV.8 Skenario Use Case Edit Profil Sekolah 42](#_Toc97549279)

[Tabel IV.9 Skenario Use Case Tambah Tahun Pelajaran 42](#_Toc97549280)

[Tabel IV.10 Skenario Use Case Edit Tahun Pelajaran 43](#_Toc97549281)

[Tabel IV.11 Skenario Use Case Hapus Data Tahun Pelajaran 43](#_Toc97549280)

[Tabel IV.12 Skenario Use Case Tambah Data Siswa 44](#_Toc97549282)

[Tabel IV.13 Skenario Use Case Edit Data Siswa 44](#_Toc97549279)

[Tabel IV.14 Skenario Use Case Hapus Data Siswa 45](#_Toc97549280)

[Tabel IV.15 Skenario Use Case Tambah data Kelas 45](#_Toc97549281)

[Tabel IV.16 Skenario Use Case Edit Data Kelas 46](#_Toc97549282)

[Tabel IV.18 Skenario Use Case Tambah Data Rombel 46](#_Toc97549282)

[Tabel IV.19 Skenario Use Case Edit Data Rombel 47](#_Toc97549279)

[Tabel IV.20 Skenario Use Case Hapus Data Rombel 47](#_Toc97549280)

[Tabel IV.21 Skenario Use Case Tambah Data Wali Kelas 47](#_Toc97549281)

[Tabel IV.22 Skenario Use Case Edit Data Rombel 48](#_Toc97549280)

[Tabel IV.23 Skenario Use Case Hapus Data Wali Kelas 48](#_Toc97549282)

[Tabel IV.24 Skenario Use Case Tambah Data Guru 49](#_Toc97549279)

[Tabel IV.25 Skenario Use Case Edit Data guru 49](#_Toc97549280)

[Tabel IV.26 Skenario Use Case Hapus Data Wali Kelas 49](#_Toc97549281)

[Tabel IV.27 Skenario Use Case Tambah Data Mata Pelajaran 50](#_Toc97549282)

[Tabel IV.28 Skenario Use Case Edit Data Mata Pelajaran 50](#_Toc97549279)

[Tabel IV.29 Skenario Use Case Hapus Data Mata Pelajaran 51](#_Toc97549280)

[Tabel IV.30 Skenario Use Case Tambah Data Pengajar 51](#_Toc97549281)

[Tabel IV.31 Skenario Use Case Edit Data Pengajar 52](#_Toc97549280)

[Tabel IV.32 Skenario Use Case Hapus Data Pengajar 52](#_Toc97549282)

[Tabel IV.33 Skenario Use Case Tambah Data Pengguna 52](#_Toc97549279)

T[abel IV.34 Skenario Use Case Edit Data Pengguna 53](#_Toc97549280)

[Tabel IV.35 Deskripsi mockup antarmuka login 66](#_Toc97549281)

[Tabel IV.36 Deskripsi mockup antarmuka dashboard 68](#_Toc97549282)

[Tabel IV.37 Deskripsi mockup antarmuka profil sekolahl 70](#_Toc97549282)

[Tabel IV.38 Deskripsi mockup antarmuka tahun ajaran 71](#_Toc97549280)

[Tabel IV.39 tabel basis data siswa 77](#_Toc97549282)

[Tabel IV.40 tabel basis data guru 78](#_Toc97549279)

T[abel IV.41 tabel basis data tahun ajaran 79](#_Toc97549280)

[Tabel IV.42 tabel basis data mata pelajaran 79](#_Toc97549281)

Tabel IV.43 tabel basis data rombongan belajar 81

[Tabel IV.44 tabel basis data login 81](#_Toc97549281)

# BAB I

**PENDAHULUAN**

# I.1 Latar Belakang

Pengelolaan data secara digital dalam dunia pendidikan sangat penting karena memberikan kenyamanan bagi user atau pengguna untuk mengakses hal-hal yang mereka inginkan dan memperlancar semua kegiatan yang ada. Hal-hal yang awalnya masih manual seperti pengolahan data akademik, dan penyebaran informasi kini dapat langsung diakses oleh masing-masing pihak. (Recky T. Djaelangkara, Rizal Sengkey, ST., MT, & Oktavian A. LAntang, ST,. MTI., 2015)

Salah satu sekolah menengah pertama yang sudah menggunakan e-learning adalah SMA Pelita Bunga Bangsa Arjasari. SMA Pelita Bunga Bangsa Arjasari merupakan sekolah mengenah atas yang menjadi contoh bagi sekolah lainnya. Tujuan dari adanya e-learning tersebut untuk memudahkan setiap tenaga pengajar serta siswa dalam mengakses informasi-informasi yang terkait dengan kegiatan yang ada di SMA Pelita Bunga Bangsa Arjasari. Namun penggunaanya belum digunakan secara maksimal dalam memberikan informasi-informasi akademika yang terdapat di sekolah. Dan juga pengoperasian program yang sulit untuk digunakan oleh guru.

Maka dibutuhkan adanya aplikasi pengelolaan data akademik ini diharapkan dapat mempermudah pekerjaan guru dalam proses belajar mengajar serta dapat memperkenalkan para siswa dengan pembelajaran yang berbasis teknologi.

Dari jurnal yang ditulis oleh Sarintan Kaharu, dan Oki Sakina, berdasarkan hasil penelitiannya menyatakan bahwa pengelolaan data akademik sekolah secara digital akan lebih memudahkan akses data, dan validasi data yang lebih akurat. (Sarintan Kaharu, 2016)

Oleh karena itulah, berdasarkan permasalahan tersebut dibutuhkan sebuah sistem maupun aplikasi untuk mengatasi permasalahan mengenai pengelolaan data akademik sekolah yang sudah terdigitalisasi agar lebih memudahkan pengelolaan, dan validasi data. Maka dari itu penulis memutuskan untuk merancan sebuah aplikasi pengelola data akademik menggunakan codeigniter versi 4.1.1. Diharapkan aplikasi yang dibuat tersebut dapat digunakan untuk keperluan pengelolaan data akademik sekolah untuk menghindari resiko-resiko yang dapat terjadi, seperti efisiensi waktu saat melakukan pengecekan aset, informasi tidak relevan, kemungkinan adanya duplikasi data, serta ketidaksesuaikan data dengan bukti fisik.

# I.2 Lingkup

Lingkup materi kerja praktek yang dilaksanakan di Sekolah Menengah Atas (SMA) Pelita Bunga Bangsa Arjasari adalah pembuatan aplikasi pengelola data akademik. menangani semua data dan proses yang menyangkut mengenai data informasi akademik adalah sebagai berikut :

1. Data Login Admin

(Username, password),

1. Data Pengguna

(usermame,password,email ,first\_name,last\_name,company, phone**).**

1. Data Siswa

(Nama\_lengkap, nama\_panggilan, nis, nisn, nik, nikk, jenis\_kelamin, tempat\_lahir, tanggal\_lahir, alamat, kelurahan, kecamatan, kota\_kab, provinsi, kode\_pos, agama, ayah, pekerjaan\_ayah, penghasilan\_ayah, ibu, pekerjaan\_ibu, penghasilan\_ibu, foto, tanggal\_masuk, aktif, telp\_siswa, telp\_ayah, telp\_ibu),

1. Data Guru

(Nama\_lengkap, jenis\_kelamin, tempat\_lahir, tanggal\_lahir, nip, nik, nikk, alamat, gelar\_depan, gelar\_belakang, pendidikan\_terakhir, email, telp**),**

1. Data Mata Pelajaran

(Nama, kode, kelompok).

1. Data pengajar

(Id\_guru, id\_kelas, id\_mapel, id\_tahun).

1. Data Kelas

(Nama, kode\_kelas, tingkat).

1. Data tahun ajaran

(tahun, semester, id\_kepsek ,tanggal\_rapor).

1. Data Rombel

(id\_tahun, id\_kelas, id\_siswa).

1. Data wali kelas

(id\_guru, id\_kelas, id\_tahun ).

Aplikasi Pengelola Data Akademik ini merupakan aplikasi yang dibuat di Sekolah Menengah Atas (SMA) Pelita Bunga Bangsar untuk mendukung proses belajar mengajar agar lebih mudah memproses informasi akademik.

# I.3 Tujuan

Tujuan pelaksanaan kerja praktek di Sekolah Menengah Atas (SMA) PBB Arjasari, adalah sebagai berikut :

* Membuat aplikasi Pengelola Data Akademik
* Mempermudah pihak sekolah dalam mengelola data akademik maupun kebutuhan teknologi informasi lainnya

# BAB II

**LINGKUNGAN KERJA PRAKTEK**

Sekolah Menengah Atas (SMA) Pelita Bunga Bangsa Arjasari yang beralamat di Jl. Raya Arjasari No 05 Arjasari ,Kec Arjasari, Kabupaten Bandung Provinsi Jawa Barat, adalah Lembaga yang bergerak dibidang Pendidikan dengan Visi dan Misisebagai berikut :

1. Visi

Membentuk manusia indonesia yang cerdas, trampil, agamis, dan berguna bagi nusa dan bangsa.

2. Misi

a. Membantu pemerintah untuk mensukseskan program pembangunan nasional dalam bidang pendidikan didaerah pedesaan.

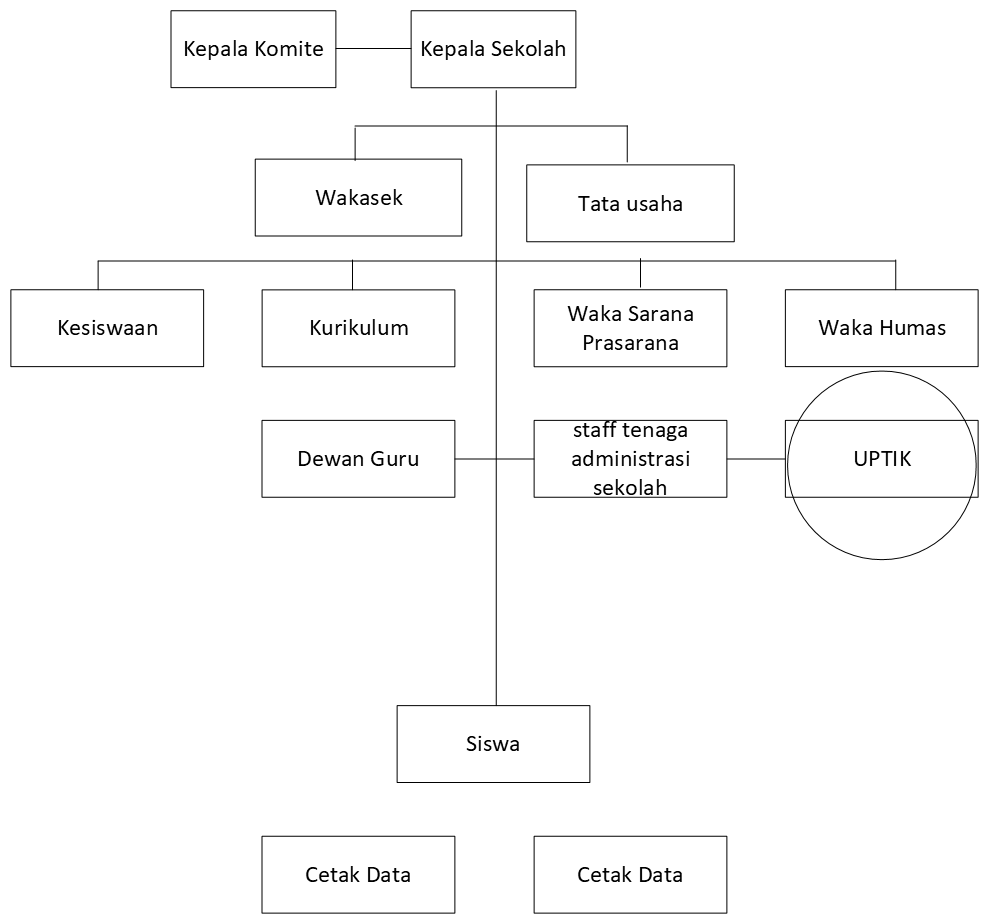
b. berusaha menyediakan sarana prasarana dan kegiatan dalam rangka membangun manusia ummatan wasathon (umat yang sempurna ).

c. berperan serta dalam upaya membentuk sumber daya manusia indonesia yang memiliki ilmu pengetetahuan dan teknologi (iptek ) yang tinggi, iman dan takwa, ( imtaq ) serta akhlak yang mulia.

d.Membentuk Manusia yang cinta lingkungan, hormat terhadap orang tua dan guru.

# II.1 Struktur Organisasi

Struktur Organisasi Sekolah Menengah Atas (SMA) Pelita Bunga Bangsa Arjasari dapat dilihat pada lampiran C. Dalam melaksanakan kerja praktek, didapatkan bimbingan secara langsung dari Bapak Heri Khoerudin Sidiq, S.T. selaku Unit Pelaksana Teknis Teknologi Informasi dan Komunikasi (UPT TIK).



Gambar II.1 Struktur Organisasi

# II.2 Lingkup Pekerjaan

Tempat peserta kerja praktek melaksanakan pekerjaan adalah di bagian staff unit pelaksana teknologi informasi*.* Mengelola sumber daya sistem informasi dan teknologi sesuai kebutuhan sekolah.

# II.3 Deskripsi Pekerjaan

Pekerjaan yang telah dilakukan dapat dibagi dalam 3 tahap :

1.Eksplorasi, baik metodologi pembuatan Sistem Informasi Akademik berbasis web maupun tools software dan hardware yang akan digunakan dalam pembuatan aplikasi tersebut.

2.Pembuatan aplikasi dengan memanfaatkan hasil eksplorasi.

a. Merancang pembuatan website

b. Pembuatan sistem informasi akademik berbasis web dengan menggunakan tools yang dapat dilihat pada Bab III.

c. Pengujian aplikasi sistem informasi akademik berbasis web.

Pelaporan kegiatan dan hasil kerja praktek, baik kepada SMA Pelita Bunga Bangsa Arjasari maupun kepada Departemen Teknik Informatika Universitas Bale Bandung. Pelaporan ini dilakukan baik melalui presentasi maupun pembuatan laporan kerja praktek

# II.4 Jadwal Kerja

Kerja praktek dilaksanakan dari tanggal 19 November 2021 sampai dengan 19 Januari 2021. Waktu kerja praktek adalah dari hari Rabu dan Jumat pukul 08.00 WIB sampai dengan pukul 12.00 WIB .

Secara umum, kegiatan yang dilakukan selama kerja praktek adalah sebagai berikut:

1. Minggu pertama:

* Pengenalan lingkungan kerja praktek
* Eksplorasi tentang Perangkat lunak yang ada di sekolah

2. Minggu Kedua

* Eksplorasi tentang aplikasi Sistem Informasi Akademik Sekolah
* Pengenalan tools yang akan digunakan dalam pembuatan aplikasi

3. Minggu Ketiga

* Installasi tools yang akan digunakan dalam pembuatan aplikasi
* Merancang aplikasi

4. Minggu Keempat

* Melanjutkan proses merancang aplikasi

5. Minggu Kelima

* Eksplorasi Database

6. Minggu Keenam

* Pembuatan Database

7. Minggu Ketujuh

* Melanjutkan proses pembuatan database

8. Minggu Kedelapan

Melakukan proses pelaporan dan Evaluasi kerja praktek Membuat laporan kerja praktek.

Adapun detail kegiatan kerja praktek dalam skala harian dapat dilihat pada lampiran B. secara keseluruhan, realisasi jadwal kerja praktek sesuai dengan rencana yang telah disusun. Selama kerja praktek, pembuatan sistem informasi akademik berbasis web dilakukan oleh penyusun. Dari mulai eksplorasi tools software dan hardware, eksplorasi script, eksplorasi database, serta eksplorasi penulisan laporan kerja praktek.

# BAB III

**TEORI PENUNJANG KERJA PRAKTEK**

# III.1 Teori Penunjang

Selama pelaksanaan kerja praktek di Sekolah Menengah Atas (SMA) Pelita Bunga Bangsa Arjasar, peserta kerja praktek menggunakan pengetahuan yang diperoleh selama masa perkuliahan sebagai landasan teori pembuatan aplikasi Sistem Informasiberbasis web. Pengetahuan dan teori yang digunakan antara lain :

1. Konsep Algoritma Pemrograman

Teori tentang Algoritma Pemrograman dari mata kuliah TIF301

1. Konsep Interaksi Manusia dan Komputer

Teori tentang Konsep Interaksi Manusia dan Komputer diperoleh Di

matakuliah FTI307 yaitu Interaksi Manusia Komputer.

1. Konsep Database Management System

Teori tentang konsep database management diperoleh di matakuliah FTI310 yaitu basis data dan di mata kuliah FTI311 yaitu sistem basis data.

1. Konsep Dasar Web

Konsep dasar web diperoleh di matakuliah FTI319 pemrograman internet.

5.Perancangan Aplikasi berbasis web

Konsep dasar perancangan aplikasi diperoleh pada mata kuliah FTI318 yaitu Manajemen Proyek Perangkat Lunak

Selain dari beberapa mata kuliah, ada beberapa materi yang dijadikan landasan teori pembuatan aplikasipenjadwalan praktikum, adalah:

1. Konsep MVC *(Model, View, Controller*)

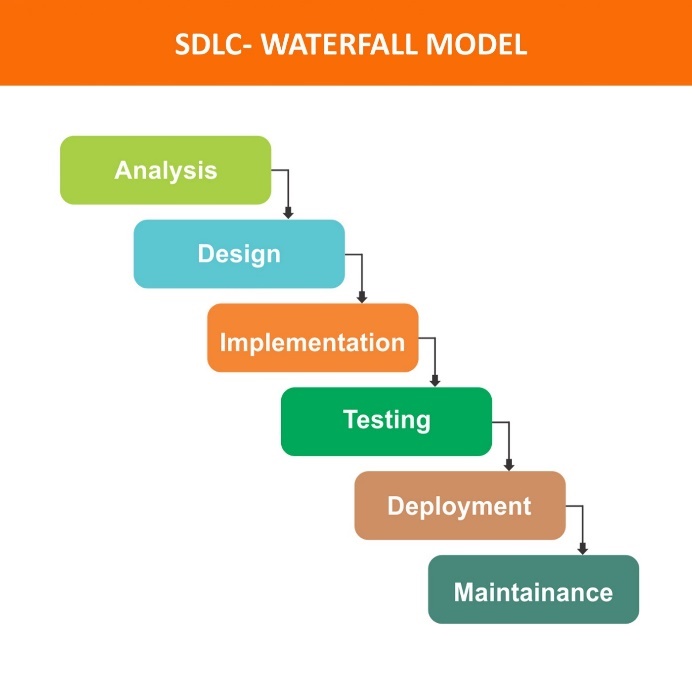
Model View Controller merupakan suatu konsep yang cukup populer dalam pembangunan aplikasi web, berawal pada bahasa pemrograman *Small Talk*, MVC memisahkan pengembangan aplikasi berdasarkan komponen utama yang membangun sebuah aplikasi seperti manipulasi data, antarmuka pengguna, dan bagian yang menjadi kontrol aplikasi. Terdapat 3 jenis komponen yang membangun suatu pola MVC dalam suatu [aplikasi](https://id.wikipedia.org/wiki/Aplikasi) yaitu:

* 1. *View*, merupakan bagian yang menangani logika presentasi. Pada suatu aplikasi web bagian ini biasanya berupa berkas templat HTML, yang diatur oleh *controller*. *View* berfungsi untuk menerima dan merepresentasikan data kepada pengguna. Bagian ini tidak memiliki akses langsung terhadap bagian *model*.
  2. *Model*, biasanya berhubungan langsung dengan pangkalan data untuk memanipulasi data (*insert, update, delete, search*), menangani validasi dari bagian controller, tetapi tidak dapat berhubungan langsung dengan bagian *view*.
  3. *Controller*, merupakan bagian yang mengatur hubungan antara bagian model dan bagian view, *controller* berfungsi untuk menerima permintaan dan data dari pengguna kemudian menentukan apa yang akan diproses oleh aplikasi.

1. Metode SDLC

SDLC (*System Development Life Cycle*) atau Siklus hidup pengembangan sistem adalah proses pembuatan dan pengubahan sistem serta model dan metodologi yang digunakan untuk mengembangkan sistem-sistem tersebut. Metode pengembangan sistem metode SDLC atau sering disebut sebagai pendekatan air terjun(*waterfall*). Metode *waterfall* pertama kali diperkenalkan oleh windows W. Royce pada tahun 1970.

Model waterfall adalah salah satu jenis pengembangan aplikasi dan termasuk dalam siklushidup klasik, dimana kemajuan suatu proses dipandang terus mengalir ke bawah seperti air terjun. Dalam Model Waterfall, setiap tahap harus berurutan, dan tidak dapat melompat ketahap berikutnya, *waterfall* harus menyelesaikan tahap pertama baru lanjut ke tahap ke dua dan seterusnya. (Irnawati, 2020)



Gambar III. 1 Model Waterfall

* 1. Analisis Sistem(*Analyst*)

Analisis system adalah penguraian dari suatu sistem yang utuh kedalam kegiatan-kegiatan komponennya, dengan maksud untuk mengidentifikasikan dan mengevaluasi permasalahan, kesempatan, hambatan yang terjadi dan kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya.

* 1. Desain Sistem (*Design*)

Desain Sistem adalah persiapan rancang bangun implementasi yang menggambarkan bagaimana suatu sistem dibentuk yang berupa penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah kedalam satu kesatuan yang utuh dan berfungsi, menyangkut di dalamnya konfigurasi komponen perangkat lunak dan perangkat keras dari suatu sistem.

* 1. Implementasi (*Implementation*)

Tahap implementasi, yaitu desain program diterjemahkan ke dalam kode-kode instruksi yang akan dijalankan komputer. Menerjemahkan hasil proses perancangan menjadi sebuah bentuk program komputer yang dimengerti oleh mesin komputer. Penerjemahan ke bahasa komputer menggunakan bahasa pemrograman dalam menjalankan instruksi yang disusun. Bahasa komputer itu seperti PHP, Python, C++, JAVA, dan lain sebagainya.

* 1. Uji coba Program (*Testing*)

Uji coba program merupakan elemen yang kritis dari SQA(*Software Quality Assurance*) dan mempresentasikan tinjauan ulang yang menyeluruh terhadap spesifikasi, desain dan pengkodean. Uji coba mempresentasikan ketidak normalan yang terjadi pada pengembangan software. Selama definisi awal dan fase pembangunan, pengembangan berusaha untuk membangun software dari konsep yang abstrak sampai dengan implementasi yang memungkinkan.

* 1. Penyebaran (*Deployment*)

Tahapan Deployment ini seperti perilisan saat produk atau aplikasi dinyatakan fungsional dan bisa digunakan secara langsung oleh pengguna. Tahap deployment dilakukan dengan langkah-langkah sebagai berikut:

1. Menyiapkan Fasilitas Fisik

Fasilitas-fasilitas fisik yang disiapkan antara lain komputer dan peripheralnya, termasuk keamanan fisik untuk menjaga berlangsungnya peralatan dalam jangka waktu yang lama.

1. Menyiapkan Pemakai

Pemakai disiapkan dengan terlebih dahulu yaitu dengan memberikan pelatihan secara prosedural maupun tutorial mengenai sistem informasi sesuai fungsi tugasnya. Tujuannya adalah agar para pemakai mengerti dan mengusai operasi sistem dan cara kerja sistem serta apa saja yang diperoleh dari sistem.

1. Melakukan Simulasi

Kegiatan simulasi berupa pengujian sistem secara nyata yang melibatkan personil yang sesungguhnya.

* 1. Pemeliharaan Sistem (*Maintenance*)

Pada tahap pemeliharaan sistem inisetelah aplikasi berhasil dan siap pakai maka harus melakukan pembenaran dan penyempurnaan software.

* + 1. Untuk membenarkan kesalahan atau kelemahan sistem yang tidak terdeteksi pada saat pengujian,
    2. Untuk membuat sistem up to date,
    3. Untuk meningkatkan kemampuan sistem.

1. Unified *Modeling Language* (UML)

*Unified Modeling Language* (UML) adalah himpunan struktur dan teknik untuk pemodelan desain secara visual yang berfungsi sebagai perancangan sistem berorientasi objek (OOP). UML adalah metodologi untuk mengembangkan sistem OOP dan sekelompok perangkat *tool* untuk mendukung pengembangan sistem tersebut.

Menurut (Rosa-Salahuddin, 2011:113), Unified Modelling Language atau UML merupakan salah satu standar bahasa yang banyak digunakan di dunia industri untuk menggambarkan kebutuhan (requirement), membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek (OOP).

UML mulai diperkenalkan oleh *Object Management Group*, sebuah organisasi yang telah mengembangkan model, teknologi, dan standar OOP sejak tahun 1980-an. Beberapa jenis UML yang digunakan dalam perancangan aplikasi penjadwalan praktikum yaitu:

1. Use Case Diagram

Usecase diagramadalah diagram *usecase* yang digunakan untuk mendeskripsikan suatu urutan interaksi yang saling berkaitan antara aktor dan alur sistem yang dibuat. Diagram *usecase* tidak menjelaskan secara detail tentang penggunaan *usecase*, namun hanya memberi gambaran singkat hubungan antara *usecase*, aktor, dan sistem. Melalui diagram usecase dapat diketahui fungsi-fungsi apa saja yang ada pada sistem (Rosa dan Salahudin, 2013).

Tabel III. 1 *Use Case Diagram*

|  |  |  |  |
| --- | --- | --- | --- |
| No | Simbol | Nama | Keterangan |
| 1 |  | *Actor* | Menspesifikasikan himpuan peran yang pengguna mainkan ketika berinteraksi dengan *use case*. Use case biasanya dinyatakan dengan kata kerja. |
| 2 |  | *Use Case* | Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu *Actor*. |
| 3 |  | *Association* | Apa yang menghubungkan antara objeks atau dengan objek lainnya. |
| 4 |  | *Extend* | Menspesifikasikan bahwa use case target memperluas perilaku dari *use case* sumber pada suatu titik yang diberikan |
| 5 |  | *Generalization* | Hubungan dimana objek anak (*descendent*) berbagi perilaku dan struktur data dari objek yang diatasnya objek induk (ancestor) |
| 6 |  | *Include* | Menspesifikasikan bahwa *use case* sumber secara eksplisit. |

* 1. *Activity Diagram*

*Activity Diagram* adalah diagram yang menggambarkan *worlflow* (aliran kerja) aktivitas dari sebuah sistem atau proses bisnis. Diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem (Rosa dan Salahudin 2013).

Tabel III. 2 *Activity Diagram*

|  |  |  |  |
| --- | --- | --- | --- |
| No | Simbol | Nama | Keterangan |
| 1 |  | Status Awal | Aktivitas yang dilakukan sistem, aktivitas diawalidengan kata kerja |
| 2 | Shape, rectangle  Description automatically generated | Aktivitas | Deskripsi dari urutan aksi- aksi yang yang ditampilkan sistem yang menghasilkan suatu hasi lyang terukur bagi suatu Actor |
| 3 | A picture containing triangle  Description automatically generated | Percabangan | Asosiasi percabangan dimanajika ada pilihan  aktivitas lebih dari satu |
| 4 |  | Penggabungan | Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu |
| 5 |  | Status akhir | Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir. |

* 1. *Class Diagram*

*Class diagram* merupakan alur jalannya database yang saling terhubung pada sebuah sistem. *Class diagram* atau diagram kelas menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. (Rosa A.S dan M.Shalahuddin 2013). Diagram kelas atau class diagram menjelaskan struktursistem dari segi pendefinisian class-class yang akan dibuat untuk membangun sebuah sistem.

Tabel III. 3 *Class Diagram*

|  |  |  |  |
| --- | --- | --- | --- |
| No | Simbol | Nama | Keterangan |
|  | |  | | --- | | Nama\_kelas | | +atribut | | +operasi () | | Kelas | Kelas pada struktur  sistem. |
|  |  | Antarmuka / *interface* | Sama dengan konsep interface dalam pemrograman berorientasi objek |
|  |  | Asosiasi / *association* | Relasi antar class  dengan arti umum |
|  |  | Asosiasi berarah / *directed association* | Relasi antar kelas  dengan makna kelas  yang ataudigunakan  oleh kelasyang lain, asosiasibiasanya juga disertai dengan  multiplicity. |
|  |  | Generalisasi | Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus) |
|  |  | Kebergantungan / dependency | Relasi antar kelas dengan makna kebergantungan antar kelas |
|  |  | Agresi / aggretion | Relasi antar kelas denganmakna semua-bagian (whole-part) |

# III.2 Peralatan Pembuatan Aplikasi Web

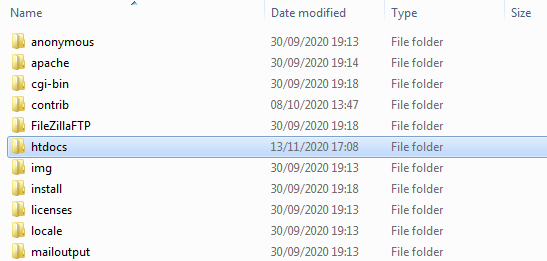
1. XAMPP

XAMPP adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai server yang berdiri sendiri (*localhost*), yang terdiri atas program Apache HTTP Server, MySQL *database*, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama XAMPP merupakan singkatan dari X (tempat sistem operasi apapun), Apache, MySQL, PHP dan Perl. Program ini tersedia dalam GNU *General Public License* dan bebas, merupakan web server yang mudah digunakan yang dapat melayani tampilan halaman web yang dinamis.

* 1. Fungsi XAMPP

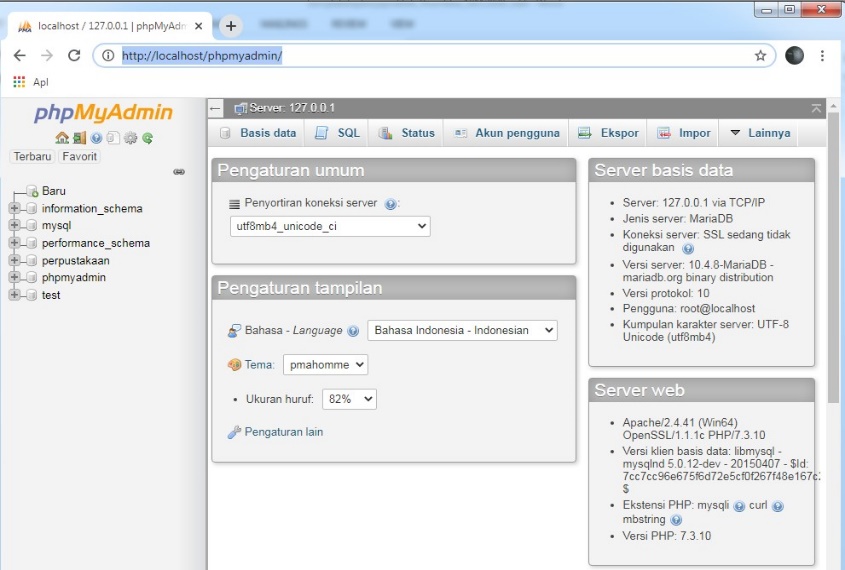
Fungsi XAMPP adalah sebagai sebuah server lokal yang berdiri sendiri (disebut juga *Localhost*) yang terdiri dari program Apache HTTP Server, MySQL *database*, PHP dan Perl.Dalam prakteknya XAMPP berfungsi sebagai "*demo*" dari tampilan halaman website.

* 1. Bagian Penting XAMPP
     1. htdoc adalah folder tempat meletakkan berkas-berkas yang akan dijalankan, seperti berkas PHP, HTML dan skrip lain.



Gambar III. 3 Htdocs

* + 1. phpMyAdmin merupakan bagian untuk mengelola basis data MySQL yang ada dikomputer. Untuk membukanya, buka browser lalu ketikkan alamat [http://localhost/phpMyAdmin,](http://localhost/phpMyAdmin) maka akan muncul halaman phpMyAdmin.



Gambar III. 4 Phpmyadmin

* + 1. Control Panel berfungsi mengelola layanan (*service*) XAMPP. Seperti menghentikan (*stop*) layanan atau memulai (*start*).

Graphical user interface, application

Description automatically generated

Gambar III. 5 Control Panel

* 1. Type data

Type data merupakan jenis data yang digunakan untuk mendefinisikan field atau kolom pada XAMPP, ada pula nantinya pada tabel basisdata kolom index yang unik yaitu *primary key* (PK) dan *foreign key*(FK). Jenis-jenis type data pada MySQL *database*

XAMPP antara lain:

* + 1. Tipe Numerik

Tipe data numerik digunakan untuk menyimpan data numeric (angka). Ciri utama data numeric adalah suatu data yang memungkinkan untuk dikenai operasi aritmatika seperti pertambahan, pengurangan, perkalian dan pembagian. Berikut ini tipe field (kolom) di MySQL yang termasuk ke dalam kelompok tipe numerik:

1. Tinyint

|  |  |  |
| --- | --- | --- |
| Penggunaan | : | digunakan untuk menyimpan data bilangan bulat positif dan negatif |
| Jangkauan | : | -128 s/d 127 |
| Ukuran | : | 1 byte (8 bit) |

1. Smallint

|  |  |  |
| --- | --- | --- |
| Penggunaan | : | digunakan untuk menyimpan data bilangan bulat positif dan negatif |
| Jangkauan | : | -32.768 s/d 32.767 |
| Ukuran | : | 2 byte (16 bit) |

1. Mediumint

|  |  |  |
| --- | --- | --- |
| Penggunaan | : | digunakan untuk menyimpan data bilangan bulat positif dan negatif |
| Jangkauan | : | -8.388.608 s/d 8.388.607 |
| Ukuran | : | 3 byte (24 bit) |

1. Int

|  |  |  |
| --- | --- | --- |
| Penggunaan | : | digunakan untuk menyimpan data bilangan bulat positif dan negatif |
| Jangkauan | : | -2.147.483.648 s/d 2.147.483.647 |
| Ukuran | : | byte (32 bit) |

1. Bigint

|  |  |  |
| --- | --- | --- |
| Penggunaan | : | digunakan untuk menyimpan data bilangan bulat positif dan negatif |
| Jangkauan | : | ± 9,22 x 1018 |
| Ukuran | : | 8 byte (64 bit) |

1. Float

|  |  |  |
| --- | --- | --- |
| Penggunaan | : | digunakan untuk menyimpan data bilangan pecahan positif dan negatif presisi tunggal |
| Jangkauan | : | 3.402823466E+38 s/d -1.175494351E-38, 0, dan 1.175494351E-38 s/d 3.402823466E+38 |
| Ukuran | : | 4 byte (32 bit) |

1. Double/real

|  |  |  |
| --- | --- | --- |
| Penggunaan | : | digunakan untuk menyimpan data bilangan pecahan positif dan negatif presisi ganda |
| Jangkauan | : | -1.79…E+308 s/d -2.22…E-308, 0, dan 2.22…E-308 s/d 1.79…E+308 |
| Ukuran | : | 8 byte (64 bit) |

1. Decimal/numeric

|  |  |  |
| --- | --- | --- |
| Penggunaan | : | digunakan untuk menyimpan data bilangan pecahan positif dan negatif |
| Jangkauan | : | -1.79…E+308 s/d -2.22…E-308, 0, dan 2.22…E-308 s/d 1.79…E+308 |
| Ukuran | : | 8 byte (64 bit) |

* + 1. Type Date dan Time

Tipe data date dan time digunakan untuk menyimpan data tanggal dan waktu. Berikut ini tipe field (kolom) di MySQL yang termasuk ke dalam kelompok tipe date dan time:

1. Date

|  |  |  |
| --- | --- | --- |
| Penggunaan | : | digunakan untuk menyimpan data tanggal |
| Jangkauan | : | 1000-01-01 s/d 9999-12-31 (YYYY-MM-DD) |
| Ukuran | : | 3 byte |

1. Time

|  |  |  |
| --- | --- | --- |
| Penggunaan | : | digunakan untuk menyimpan data waktu |
| Jangkauan | : | -838:59:59 s/d +838:59:59 (HH:MM:SS) |
| Ukuran | : | 3 byte |

1. Datetime

|  |  |  |
| --- | --- | --- |
| Penggunaan | : | digunakan untuk menyimpan data tanggal dan waktu |
| Jangkauan | : | ‘1000-01-01 00:00:00’ s/d ‘9999-12-31 23:59:59’ |
| Ukuran | : | 8 byte |

1. Year

|  |  |  |
| --- | --- | --- |
| Penggunaan | : | digunakan untuk menyimpan data tahun dari tanggal |
| Jangkauan | : | 1900 s/d 2155 |
| Ukuran | : | 1 byte |

* + 1. Tipe String (Text)

Tipe data string digunakan untuk menyimpan data string (text). Ciri utama data string adalah suatu data yang memungkinkan untuk dikenai operasi aritmatika seperti pertambahan, pengurangan, perkalian dan pembagian. Berikut ini tipe field (kolom) di MySQL yang termasuk ke dalam kelompok tipe string:

1. Char

|  |  |  |
| --- | --- | --- |
| Penggunaan | : | digunakan untuk menyimpan data string ukuran tetap |
| Jangkauan | : | 0 s/d 255 karakter |

1. Varchar

|  |  |  |
| --- | --- | --- |
| Penggunaan | : | digunakan untuk menyimpan data string ukuran dinamis |
| Jangkauan | : | 0 s/d 255 karakter (versi 4.1), 0 s/d 65.535 (versi 5.0.3) |

1. Tinytext

|  |  |  |
| --- | --- | --- |
| Penggunaan | : | digunakan untuk menyimpan data text |
| Jangkauan | : | 0 s/d 255 karakter (versi 4.1), 0 s/d 65.535 (versi 5.0.3) |

1. Text

|  |  |  |
| --- | --- | --- |
| Penggunaan | : | digunakan untuk menyimpan data text |
| Jangkauan | : | 0 s/d 65.535 (216 – 1) karakter |

1. Mediumtext

|  |  |  |
| --- | --- | --- |
| Penggunaan | : | digunakan untuk menyimpan data text |
| Jangkauan | : | 0 s/d 224 – 1 karakter |

1. Longtext

|  |  |  |
| --- | --- | --- |
| Penggunaan | : | digunakan untuk menyimpan data text |
| Jangkauan | : | 0 s/d 232 – 1 karakter |

* + 1. Tipe BLOB (Biner)

Tipe data blob digunakan untuk menyimpan data biner. Tipe ini biasanya digunakan untuk menyimpan kode-kode biner dari suatu file atau object. BLOB merupakan singkatan dari Binary Large Object. Berikut ini tipe field (kolom) di MySQL yang termasuk ke dalam kelompok tipe blob:

1. Bit (sejak versi 5.0.3)

|  |  |  |
| --- | --- | --- |
| Penggunaan | : | digunakan untuk menyimpan data biner |
| Jangkauan | : | 64 digit biner |

1. Tinyblob

|  |  |  |
| --- | --- | --- |
| Penggunaan | : | digunakan untuk menyimpan data biner. |
| Jangkauan | : | 255 byte |

1. Blob

|  |  |  |
| --- | --- | --- |
| Penggunaan | : | digunakan untuk menyimpan data biner. |
| Jangkauan | : | 216 – 1 byte |

1. Mediumblob

|  |  |  |
| --- | --- | --- |
| Penggunaan | : | digunakan untuk menyimpan data biner. |
| Jangkauan | : | 224 – 1 byte |

1. Longblob

|  |  |  |
| --- | --- | --- |
| Penggunaan | : | digunakan untuk menyimpan data biner. |
| Jangkauan | : | 232 – 1 byte |

* + 1. Tipe data yang lain

MySQL juga menyediakan tipe data yang lain. Tipe data di MySQL mungkin akan terus bertambah seiring dengan perkembangan versi MySQL. Berikut ini beberapa tipe data tambahan MySQL:

1. Enum

|  |  |  |
| --- | --- | --- |
| Penggunaan | : | Enumerasi (kumpulan data) |
| Jangkauan | : | Sampai dengan 65535 string |

1. Set

|  |  |  |
| --- | --- | --- |
| Penggunaan | : | Combination (himpunan data) |
| Jangkauan | : | Sampai dengan 255 string anggotas |

1. Visual Studio Code

Visual Studio Code adalah Sofware yang sangat ringan, namun kuat editor kode sumbernya yang berjalan dari desktop. Muncul dengan built-in dukungan untuk JavaScript, naskah dan Node.js dan memiliki array beragam ekstensi yang tersedia untuk bahasa lain, termasuk C ++, C# , Python, dan PHP. Editor ini adalah fitur lengkap lingkungan pengembangan terpadu (IDE) dirancang untuk pengembang yang bekerja dengan teknologi cloud yang terbuka. Microsoft Visual Studio Code menggunakan open source NET perkakas untuk memberikan dukungan untuk ASP.NET C# kode, membangun alat pengembang Omnisharp NET dan compiler Roslyn. Antarmuka yang mudah untuk bekerja dengan, karena didasarkan pada gaya explorer umum, dengan panel di sebelah kiri, yang menunjukkan semua file dan folder Anda memiliki akses ke panel editor di sebelah kanan, yang menunjukkan isi dari file yang telah dibuka. Dalam hal ini, editor telah dikembangkan dengan baik, dan menyenangkan pada mata. Ia juga memiliki fungsi yang baik, dengan intellisense dan autocomplete bekerja dengan baik untuk JSON, CSS, HTML, dan Node.js.

Visual Studio Code telah dirancang untuk bekerja dengan alat-alat yang ada, dan Microsoft menyediakan dokumentasi untuk membantu pengembang bersama, dengan bantuan untuk bekerja dengan ASP.NET 5, Node.js, dan Microsoft naskah, serta alat-alat yang dapat digunakan untuk membantu membangun dan mengelola aplikasi Node.js. Visual Studio Code benar-benar sedang ditargetkan pada pengembang JavaScript yang ingin alat pengembangannya lengkap untuk scripting server-side mereka dan yang mungkin ingin usaha dari Node.js untuk kerangka berbasis NET. Visual Studio Code, adalah belum solid, lintas platform kode Editor ringan, yang dapat digunakan oleh siapa saja untuk membangun aplikasi untuk Web.

Kelebihan Visual Studio Code:

1. **Text editor gratis:** Text editor ini gratis tanpa harus bayar.
2. **Sudah terinstall plugin EMMET:** Plugin emmet merupakan plugin yang sangat membantu karena dapat menghemat waktu dan mempercepat dalam pembuatan sebuah web.
3. **Mudah untuk mengelola Extensions:** Pengelolaan extensions yang sangat mudah sehingga mudah dipahami bagiyang baru memakai visual studio code. hanya perlu ke menu ekstensions atau tekan CTRL + SHIFT + X dan pilih ekstensions mana yang mauinstall atau bisa search di kolom search.
4. **Extensions yang banyak:** Banyak extensions visual studio code untuk memudahkan pekerjaan contohnya auto rename tag sangat berguna bagi web developer dan masih banyak lagi ekstensions yang dapat mempermudah pekerjaan.
5. **Kostumisasi tamping :** Di visual studio code juga bias merubah tampilan seperti icon, font atau warna sesukaataubisa menginstall herbage tema seperti material theme atau tema lainnya yangsukai.
6. **Terintegrasi dengan GIT :** Di visual studio code sudah terintegrasi dengan git jadilebih mudah dalam menyelesaikan conflict atau mengetahui mana baris yang berubah atau ditambahkan. Sehingga nantinya akan lebih memudahkan dalam kerja tim.
7. **Snippet :** Di visual studio code bisa membuat snippet sendiri atau bisa menginstall snippet dari menu ekstensions.
8. Dukungan bahasa : Visual studio code sudah mendukung banyak bahasa seperti c,c++,php,javascript dan masih banyak yang lainnya.
9. **Autocomplete :** Di visual studio code sudah disediakan fitur autocomplete tentunya untuk memudahkan pekerjaan bisa memunculkan autocomplete dengan menekan CTRL + SPACE maka autocomplete akan muncul.

Kekurangan Visual Studio Code :

1. Performa : kekurangan visual studio code terletak pada performa karena dibandingkan dengan text editor lain seperti sublime text yang masih lancar di cpu 2 core dengan 2gb ram sedangkan dengan spek yang sama visual studio code kadang suka crash atau lag. Jadi untuk performa yang stabil visual studio code butuh cpu 4 core dan 4gb ram.Dan saya rasamungkin 4gb ram masih belum cukup karena jika sambil buka browser seperti chrome mungkin visual studio code akan sedikit lag.

Belum menghafal shortcut key : Mungkin bagi yang baru sebelumnya memakai sublime text dan ingin mencoba visual studio code mungkinakan sedikit kebingungan dengan shortcut key di visual studio code karena tidak seperti di sublime text untuk melakuakan suatu tindakan menggunakan tombol CTRL sedangkan di visual studio codemenggunakan tombol ALT.

(Gustop, 2021)

1. Google Chrome

Google Chrome adalah peramban web lintas platform yang dikembangkan oleh Google. Peramban ini pertama kali dirilis pada tahun 2008 untuk Microsoft Windows, kemudian diporting ke Linux, macOS, iOS, dan Android yang menjadikannya sebagai peramban bawaan dalam sistem operasi. Peramban ini juga merupakan komponen utama Chrome OS, yang berfungsi sebagai platform untuk aplikasi web.

Google Chrome merupakan mesin pencarian mampu melakukan penelusuran dalam waktu kurang dari beberapa detik dengan perangkat lunak yang telah diinstal ke dalam Sistem Operasi windows untuk memberikan pengguna aksesoris pendukung seperti mediator layanan browser, file manager,downloader dan lain-lain. Sebagai salah satu layanan software yang memungkinkan pengguna website menelusuri informasi, media video dan audio, serta data teknis Google Chrome tersedia dan sangat mendukung untuk semua Operasi Sistem Dektop hingga pengguna smartphone seperti Android dan Apple agar broswer menjadi terkendali untuk diterima, ditelususri, disimpan hingga digunakan sebaik-baiknya dalam dunia maya.

Kelebihan Browser Chrome

* + 1. Tampilan : Pada Interface dari Chrome terlihat bahawa google ingin para penggunanya lebih fokus pada web dan melupakan browser yang digunakan. Ini artinya Google Chrome memiliki tampilan yang tidak mengusik dan nyaman ketika sedang digunakan.
    2. Modus Penyamaran : Pada modus ini memungkinkan para penggunanya dapat mengakses website tanpa meninggalkan jejak. Sehingga cocok digunakan untuk mengkases konten yang berbau porno secara diam-diam he he he
    3. Aplikasi Web : Google memberikan opsi “*Make Application Shortcut*” . Dengan underline ini sebuah aplikasi web seperti GMAIL atau Google Teader dapat dijalankan lewat shrtcut pada Desktop atau Start Menu.
    4. Pengelolaan Memory : Pada setiap TAB yang dibuka di Chrome memiliki proses yang terpisah, sehingga ketika eror/crash pada salah satu ta tidak akan meyebabkan seluruh browser eror. (ini kekurangan yang terdapat pada FireFox dan browser lainnya)
    5. Pencarian : Chrome memiliki fungsi pencarian yang sangat baik. Contohnya, chrome daat mendeteksi ketika pengguna pernah melakukan pencarian di suatu website dan memasukkan website tersebut dalam daftar penyedia pencarian.
    6. Anti Penipuan : Chrome menyediakan kemampuan menebalkan nama domain sebuah website.

Kelemahan Browser Chrome :

* + 1. Privasi : google menyimpan 2% interpretation pencarian pengguna, lengkap dengan alamat ip-a. walaupun dalam beberapa waktu tertentu interpretation ini akan dianonimkan. ini artinya google bisa saja tahu “siapa mencari apa dan dimana”
    2. Lisensi : google sempat mencantumkan pada terms of service mereka, bahwa semua muatan dari pengguna yang hak ciptanya dimiliki oleh pengguna akan diserahkan haknya pada google. tapi indicate ini telah dicabut oleh pihak google.
    3. Celah Keamanan : beberapa pakar confidence menemukan adanya lubang kecil/bugs pada chrome. sehingga ketika membuka suatu halaman website akan membuat browser ini menjadicrash.laluchromejugamemiliki underline [download](https://www.google.com/chrome/index.html?hl=id&brand=CHMB&utm_campaign=id&utm_source=id-ha-apac-id-sk&utm_medium=ha) otomatis yang dikhawatirkan akan disalah gunakan oleh hacker.
    4. *Extensions* **:** pada chrome tidak terdapat extension/plugin/addons yang dapat ditambahkan. tidak seperti firefox yang memiliki banyak aplikasi2 tambahan yang dapat membuat dan meningkatkan kinerja browser.

1. CodeIgniter

CodeIgniter merupakan aplikasi [sumber terbuka](https://id.wikipedia.org/wiki/Sumber_terbuka) yang berupa kerangka kerja PHP dengan model [MVC](https://id.wikipedia.org/wiki/MVC) (*Model, View, Controller*) untuk membangun [situs web](https://id.wikipedia.org/wiki/Situs_web) dinamis dengan menggunakan [PHP](https://id.wikipedia.org/wiki/PHP). CodeIgniter memudahkan pengembang web untuk membuat aplikasi web dengan cepat dan mudah dibandingkan dengan membuatnya dari awal. CodeIgniter dirilis pertama kali pada [28 Februari](https://id.wikipedia.org/wiki/28_Februari) 2006. Versi stabil terakhir adalah versi 4.1.11.

Menurut Supono dan Putratama (2016:109) “Codeigniter adalah aplikasi open source yang berupa framework dengan model MVC (*Model, View, Controller*) untuk membangun website dinamis dengan menggunakan PHP ”.

Sedangkan menurut Purbadian (2016:18) “Codeigniter adalah sebuah *framework* yang dibuat menggunakan bahasa pemrograman PHP yang bertujuan untuk memudahkan para programmer web untuk membuat atau mengembangkan aplikasi berbasis web”.

Berdasarkan pengertian diatas dapat ditarik kesimpulan bahwa Codeigniter adalah aplikasi *open source* yang merupakan framework PHP dengan eksekusi tercepat dan menggunakan model basic MVC (*Model View Controller*) untuk membangun sebuah website yang dinamis dan bertujuan untuk memudahkan para programmer web untuk membuat atau mengembangkan aplikasi berbasis web.

Dengan menggunakan prinsip MVC suatu aplikasi dapat dikembangkan sesuai dengan kemampuan pengembangnya, yaitu programer yang menangani bagian *model* dan *controller*, sedangkan desainer menangani pada bagian view*,* sehingga penggunaan arsitektur MVC dapat meningkatkan pemeliharaan dan pengorganisasian kode. Namun dibutuhkan komunikasi yang baik antara programer dan desainer dalam menangani variabel-variabel yang akan ditampilkan.

Kelebihan CodeIgniter

* + 1. Performa sangat cepat: salah satu alasan tidak menggunakan kerangka kerja adalah karena eksekusinya yang lebih lambat daripada PHP *from the scracth*, tapi CodeIgniter sangat cepat bahkan mungkin bisa dibilang CodeIgniter merupakan kerangka kerja yang paling cepat dibanding kerangka kerja yang lain.
    2. Konfigurasi yang sangat minim (*nearly zero configuration*): tentu saja untuk menyesuaikan dengan pangkalan data dan keleluasaan perutean tetap diizinkan melakukan konfigurasi dengan mengubah beberapa berkas konfigurasi seperti database.php atau autoload.php, tetapi untuk menggunakan CodeIgniter dengan pengaturan yang standar, Anda hanya perlu mengubah sedikit saja berkas pada folder config.
    3. Banyak komunitas: dengan banyaknya komunitas CI ini, memudahkan kita untuk berinteraksi dengan yang lain, baik itu bertanya atau teknologi terbaru. (Idcloudhost, 2021)

1. Balsamiq

Balsamiq merupakan salah satu *tool* yang biasa digunakan oleh para UI designer ataupun UX designer unatuk merancang desain tampilan aplikasi yang akan dibuat. Menurut website resmi Balsamiq https://balsamiq.com/. “Balsamiq Mockups adalah alat wireframing cepat yang membantu Anda bekerja lebih cepat & lebih pintar”. Balsamiq Mockups menciptakan pengalaman sketsa di papan tulis, tetapi menggunakan komputer, membuat mockups menjadi cepat. Menurut (Hanifah, 2015) Balsamiq Mockup adalah salah satu software yang di gunakan dalam pembuatan desain atau prototype.

Aplikasi ini dianggap cukup ramah bagi pemula karena tidak membutuhkan kode untuk bisa mengoperasikannya. Ketika penggunaan tool pada aplikasi ini cukup men-*drag* dan *drop* elemen-elemen desain yang diperlukan. Dengan begitu, proses desain pun bisa menjadi lebih cepat. Balsamiq juga bisa digunakan untuk membuat *prototype* interaktif untuk situs atau aplikasi yang sedang dirancang.

Kelebihan balsamiq mockups :

1. Aplikasi desain yang mudah digunakan karena interface atau tampilan antarmuka yang mudah dimengerti oleh semua kalangan.
2. Elemen-elemen yang diperlukan dalam mendesain tampilan aplikasi cukup lengkap.
3. Kustomiasi elemen yang bisa dilakukan dengan mudah.
4. Fitur yang sangat memudahkan proses kerja dengan Balsamiq adalah fitur kolaborasinya dengan sharing control yang fleksibel dan cepat,
5. Hasil wireframe yang telah dibuat bisa langsung di-export dalam beberapa format, seperti PDF, PNG, dan JSON.

Kekurangan balsamiqs mockups :

* + - 1. Sulitnya men-scroll library elemen UI yang dimilikinya,
      2. pembuatan sitemap di Balsamiq juga cukup terbatas,
      3. Balsamiq hanya bisa memfasilitasi sitemap sederhana,
      4. Jika ingin membuat yang lebih kompleks, butuh *software* atau tool lainnya karena Balsamiq berfokus untuk membuat *wireframe*.

1. Microsoft visio

**Microsoft Office Visio** atau biasa disebut dengan Microsoft Visio merupakan sebuah software komputer yang biasanya **digunakan untuk membuat diagram, diagram alir, brainstorm, UML, dan skema jaringan**. Selain Word, Excel dan PowerPoint, Microsoft Visio juga termasuk dalam paket Microsoft Office.

Dengan software ini **dapat membantu pengguna dalam meningkatkan kinerja,**mulai dari mempersiapkan penggambaran diagram seperti DFD, ERD, UML, Jaringan, Rancangan User Interface dan sejenisnya. Terlebih adanya sejumlah template dan elemen yang disediakan, sehingga dapat memungkinkan pengguna untuk **membuat diagram dengan mudah, intuitif serta profesional.**

**Kelebihan microsoft visio :**

* + - 1. **User friendly**
      2. **Buat diagram profesional dengan mudah**
      3. **Tools yang dipahami**
      4. **Banyak pilihan gambar**
      5. **Tersedia template menarik**
      6. **Fitur kolaborasi**

Kekurangan microsoft visio :

* + - 1. Hanya untuk platform windows
      2. Tidak gratis

# BAB IV

# PELAKSANAAN KERJA PRAKTEK

# IV.1 Input

Dalam mempelajari metodologi pembuatan aplikasi sistem informasi akademik, diberikan informasi mengenai data yang diperlukan dalam pembuatan aplikasi penjadwalan praktikum laboratorium yang berisi keterangan mengenai data akademik sekolah terutama mengenai jadwal pembelajaran.

Aplikasi sistem informasi akademik menangani inputan yang diperlukan sebagai database yang akan menampilkan informasi mengenai jadwal praktikum. Input data yang menyangkut hal berikut:

1. Home berisi foto sekolah dan tombol menuju form login,
2. Dasboard, berisikah tentang (profil singkat sekolah, jumlah siswa ,jumlah guru,jumlah kelas dan rombongan belajar ,jumlah mata pelajaran,profil,tahun ajaran,menu administator).
3. Form Data Login Admin

(Username, password),

1. Form Data Pengguna

(ip\_address, usermame,password,email ,first\_name,last\_name,company, phone**).**

1. Form Data Siswa

(Nama\_lengkap, nama\_panggilan, nis, nisn, nik, nikk, jenis\_kelamin, tempat\_lahir, tanggal\_lahir, alamat, kelurahan, kecamatan, kota\_kab, provinsi, kode\_pos, agama, ayah, pekerjaan\_ayah, penghasilan\_ayah, ibu, pekerjaan\_ibu, penghasilan\_ibu, foto, tanggal\_masuk, aktif, telp\_siswa, telp\_ayah, telp\_ibu),

1. Form Data Guru

(Nama\_lengkap, jenis\_kelamin, tempat\_lahir, tanggal\_lahir, nip, nik, nikk, alamat, gelar\_depan, gelar\_belakang, pendidikan\_terakhir, email, telp**),**

1. Form Data Mata Pelajaran

(nama, kode, kelompok).

1. Form Data pengajar

(Id\_guru, id\_kelas, id\_mapel, id\_tahun).

1. Form Data Kelas

(Nama, kode\_kelas, tingkat).

1. Form Data tahun ajaran

(tahun, semester, id\_kepsek ,tanggal\_rapor ).

1. Form Data Rombel

(id\_tahun, id\_kelas, id\_siswa).

1. Form Data wali kelas

(id\_guru, id\_kelas, id\_tahun ).

1. Logout

# IV.2 Proses

Setelah melakukan observasi pengenalan lingkungan kerja pada pelaksanaan kerja praktek, selanjutnya proses kerja praktek dapat dibagi menjadi beberapa tahap, yaitu eksplorasi dalan pembuatan aplikasi pengelola data akdemik, dan pelaporan hasil kerja praktek.

# IV.2.1 Eksplorasi

Tahap eksplorasi dimulai dengan melakukan eksplorasi mengenai metodologi yang akan digunakan dalam pembuatan aplikasi sistem informasi akademik. Untuk mendukung pelaksanaan metodologi Waterfall, diperlukan pula pengetahuan mengenai pemodelan dengan menggunakan Unified Modelling Language (UML).

Eksplorasi juga dilakukan terhadap teknologi yang akan dipakai dalam pembuatan aplikasi penjadwalan praktikum. Sebagai acuan utama dalam mempelajari pemrograman PHP sebagai bahasa pemrograman dengan menggunakan framework codeigniter dan MySQL sebagai database. Pada tahap ini dilakukan pencarian alternatif teknologi yang akan diterapkan. Dengan demikian, tidak seluruh hasil eksplorasi pada akhirnya diterapkan dalam pembuatan aplikasi penjadwalan praktikum ini. Selama proses eksplorasi ini, dilakukan pula instalasi tools yang diperlukan.

Proses eksplorasi masih berlangsung selama pembuatan aplikasi penjadwalan praktikum. Hal ini dimaksudkan untuk menyelaraskan antara hasil eksplorasi dengan penerapannya pada aplikasi yang sedang dibangun.

1. Analisis Sistem Kebutuhan

Perencanaan dan pembuatan aplikasi penjadwalan praktikum yang dilakukan dimulai dengan analisis kebutuhan perangkat lunak dan perangkat keras yang diperlukan.

1. Kebutuhan Perangkat Keras

Pembuatan aplikasi penjadwalan praktikum ini menggunakan komputer dengan spesifikasi pada Tabel IV. 1 Perangkat Keras

Tabel IV.1 Perangkat Keras

|  |  |
| --- | --- |
| Prosesor | Intel Core i5 10400F |
| Grafik Card | Radeon RX570 (4GB VRAM) |
| Ram | 16 GB |
| SSD | 128 GB |
| HDD | 1000 TB |

1. Minimum Kebutuhan Perangkat Keras

Minimun requirementkomputer yang masih bisa digunakan untuk membuat dan menjalankan aplikasi penjadwalan praktikum adalah:

Tabel IIV.2 Sytem Requirement

|  |  |
| --- | --- |
| Prosesor | AMD C-60 APU with Redon(tm) HD Graphics 1.00 GHz |
| Ram | 2 GB |
| HDD | 128 GB |

1. Kebutuhan Perangkat Lunak

Perangkat lunak adalah perangkat-perangkat tambahan berupa sistem atau tool yang digunakan atau diperlukan untuk menjalankan dan membuat aplikasi penjadwalan praktikum. Berikut adalah Perangkat lunak yang digunakan penulis untuk membuat aplikasi penjadwalan praktikum ini:

Tabel IIIV.3 Perangkat Lunak

|  |  |
| --- | --- |
| Sistem Operasi | Windows 11 |
| Server | XAMPP versi 3.2.3 |
| Aplikasi Pembuatan | Visual Studio Code |
| Browser | Microsoft Edge |
| Bahasa pemrograman | PHP versi 7.3.2 |
| Framework back-end | Codeigniter versi 4.1 |

# IV.2.2 Pembuatan Aplikasi Sistem Informasi Akademik Sekolah Menengah Atas

Pembuatan aplikasi penjadwalan praktikum yang dilakukan dimulai dengan perancangan pembuatan aplikasi sistem informasi akademik. Pembuatan aplikasi ini dilakukan berdasarkan perancangan atau perencanaan seperti yang dituliskan pada bab sebelumnya. Untuk memastikan aplikasi yang dihasilkan sesuai dengan kebutuhan dan berfungsi dengan semestinya, dilakukan beberapa kegiatan pendukung seperti pengujian, *bug fixing*, dan optimasi performansi.

Metode perancangan dalam pembuatan aplikasi sistem informasi akademik yang digunakan adalah Metodologi Waterfall*,* yaitu bertujuan agar memperoleh tahapan perancangan yang lebih baik karena tahapan yang digunakan memiliki proses yang berurut mulai dari analisa hingga *support*, sehingga dalam pembuatannya membutuhkan analisa yang penuh mengenai kebutuhan aplikasi yang akan dirancang. Sehingga selanjutnya bisa dilakukan proses sebagaimana tahap-tahap metode yang digunakan dalam pembuatan aplikasi ini.

Setiap prosesnya juga memiliki spesifikasinya sendiri, sehingga sebuah sistem dapat bekerja sesuai dengan apa yang dikehendaki (tepat sasaran) dan juga setiap prosesnya tidak saling tumpang tindih, dengan itu digunakannya metode waterfall memudahkan dalam pembuatan aplikasi penjadwalan praktikum ini.

1. Analisis Sistem

Analisis sistem dilakukan dengan cara survey dan wawancara kepada pihak staff TU SMA Pelita Bunga Bangsa Arjasari untuk perancangan aplikasi sistem informasi akademik yang akan dibuat. Dilakukan juga pencarian baik dari jurnal maupun dari internet. Setelah melakukan analisa, didapatkan hasil berbagai kebutuhan sistem dan perangkat.

1. Perancangan aplikasi sistem informasi akademik
   1. Perancangan Perangkat

Kemudian pembuatan aplikasi sistem informasi akademik dilakukan berdasarkan perancangan yang sudah di rencanakan. Untuk memastikan sistem informasi yang dihasilkan sesuai dengan kebutuhan dan berfungsi dengan baik.

* 1. Use *Case Diagram*

Pada rancangan *Use Case Diagram* dapat dilihat apa saja yang dapat dilakukan admin untuk mengelola data yang terdapat pada aplikasi penjadwalan praktikum, dirancangan tersebut dijelaskan proses yang dilakukan admin untuk mengelola data mulai dari data kelas, data mata pelajaran, data wali kelas, data kelas praktikum, data guru, data , data jadwal praktikum, data kehadiran, data detail kelas, data perubahan jadwal, data agama dan data login. Sedangkan untuk petugas hampir sama mengelola data seperti yang bagian admin lakukan kecuali bagian data petugas, data agama dan data ruang lab.

A picture containing text, screenshot, computer, monitor

Description automatically generated

Gambar IV. 1 *Use Case Diagram*

*Tabel IV.4 Skenario Use Case Login*

|  |  |  |
| --- | --- | --- |
| Aksi Aktor | | Aksi Sistem |
| Skenario Normal | |  |
| 1. Memasukan username dan password | |  |
|  | | 2.memeriksa valik tidaknya data |
| Skenario Alternatif |  | |
| 1. Memasukan username dan password |  | |
|  | 2.memeriksa valik tidaknya data | |
|  | 3.menampilkan login incorrect | |
| 4.memasukan username dan password yang valid |  | |
|  | 5.memeriksa valid atau tidaknya data | |
|  | 6.masuk ke menu dashboard SIA PBB | |

Tabel IV.5 Skenario Use Case Cek Login

|  |  |
| --- | --- |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal |  |
|  | 1. Memeriksa ke *variabel session* sebagai penanda login apakah *user* sudah login |
|  | 2. Mengembalikan status login apakah sudah login atau belum |

Tabel IV.6 Skenario Use Case Menampilkan/Melihat Data

|  |  |
| --- | --- |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal |  |
|  | 1. Memeriksa status login |
| 2. Memilih menu data yang akan ditampilkan |  |
|  | 3. Menampilkan data yang dipilih (semua kolom) dari menu data yang dipilih |

*Tabel IV.7 Skenario Use Case Tambah Profil Sekolah*

|  |  |
| --- | --- |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal |  |
|  | 1. Memeriksa status login |
| 2. Memilih menu profil sekolah |  |
|  | 3. Melihat profil sekolah |
| 4. klik tombol tambah |  |
|  | 5. Menampilkan tabel data profil sekolah |
| 6. mengisi data profil sekolah |  |
|  | 7. Menyimpan data yang diubah ke *database* |
|  | 8. Menampilkan pesan data sukses disimpan |

Tabel IV.8 Skenario Use Case Edit Profil Sekolah

|  |  |
| --- | --- |
| Skenario Normal |  |
|  | 1. Memeriksa status login |
| 2. Memilih menu profil sekolah |  |
|  | 3. Melihat profil sekolah |
| 4. klik edit data profil sekolah |  |
|  | 5. Menampilkan tabel data profil sekolah |
| 6. Mengubah data yang dipilih |  |
|  | 7. Menyimpan data yang diubah ke *database* |
|  | 8. Menampilkan pesan data sukses disimpan |

Tabel IV.9 Skenario Use Case Tambah Tahun Pelajaran

|  |  |
| --- | --- |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal |  |
|  | 1. Memeriksa status login |
| 2. Memilih menu tahun pelajaran |  |
|  | 3. Melihat table data tahun pelajaran |
| 4. klik tombol tambah tahun pelajaran |  |
|  | 5. Menampilkan tabel data tahun pelajaran |
| 6. mengisi data tahun pelajaran |  |
|  | 7. Menyimpan data yang diubah ke *database* |
|  | 8. Menampilkan pesan data sukses disimpan |

Tabel IV.10 Skenario Use Case Edit Tahun Pelajaran

|  |  |
| --- | --- |
| Skenario Normal |  |
|  | 1. Memeriksa status login |
| 2. Memilih menu tahun pelajaran |  |
|  | 3. Melihat profil sekolah |
| 4. klik edit data tahun pelajaran |  |
|  | 5. Menampilkan tabel data tahun pelajaran |
| 6. Mengubah data yang dipilih |  |
|  | 7. Menyimpan data yang diubah ke *database* |
|  | 8. Menampilkan pesan data sukses disimpan |

Tabel IV.11 Skenario Use Case Hapus Data Tahun Pelajaran

|  |  |
| --- | --- |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal |  |
|  | 1. Memeriksa status login |
| 2. Memilih menu tahun pelajaran |  |
|  | 3. Melihat data tahun pelajaran |
| 4. Memilih data yang akan dihapus |  |
| 6. Mengkonfirmasi penghapusan data |  |
|  | 7. menghapus data dari *database* |
|  | 8. Menampilkan pesan data sukses dihapus |

Tabel IV.12 Skenario Use Case Tambah Data Siswa

|  |  |
| --- | --- |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal |  |
|  | 1. Memeriksa status login |
| 2. Memilih menu data siswa |  |
|  | 3. Melihat table data siswa |
| 4. memilih tombol tambah siswa |  |
|  | 5. Menampilkan tabel data siswa |
| 6. mengisi data siswa |  |
|  | 7. Menyimpan data yang diubah ke *database* |
|  | 8. Menampilkan pesan data sukses disimpan |

Tabel IV.13 Skenario Use Case Edit Data Siswa

|  |  |
| --- | --- |
| Skenario Normal |  |
|  | 1. Memeriksa status login |
| 2. Memilih menu Data Siswa |  |
|  | 3. Melihat Data Siswa |
| 4. Memilih edit data Siswa |  |
|  | 5. Menampilkan tabel data Siswa |
| 6. Mengubah data yang dipilih |  |
|  | 7. Menyimpan data yang diubah ke *database* |
|  | 8. Menampilkan pesan data sukses disimpan |

Tabel IV.14 Skenario Use Case Hapus Data Siswa

|  |  |
| --- | --- |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal |  |
|  | 1. Memeriksa status login |
| 2. Memilih menu Siswa |  |
|  | 3. Melihat data Siswa |
| 4. Memilih data yang akan dihapus |  |
| 6. Mengkonfirmasi penghapusan data |  |
|  | 7. menghapus data dari *database* |
|  | 8. Menampilkan pesan data sukses dihapus |

*Tabel IV.15 Skenario Use Case Tambah data Kelas*

|  |  |
| --- | --- |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal |  |
|  | 1. Memeriksa status login |
| 2. Memilih menu data Kelas |  |
|  | 3. Melihat table data Kelas |
| 4. memilih tombol tambah Kelas |  |
|  | 5. Menampilkan tabel data Kelas |
| 6. mengisi data Kelas |  |
|  | 7. Menyimpan data yang diubah ke *database* |
|  | 8. Menampilkan pesan data sukses disimpan |

Tabel IV.16 Skenario Use Case Edit Data Kelas

|  |  |
| --- | --- |
| Skenario Normal |  |
|  | 1. Memeriksa status login |
| 2. Memilih menu Data Kelas |  |
|  | 3. Melihat Data Kelas |
| 4. Memilih edit data Kelas |  |
|  | 5. Menampilkan tabel data Kelas |
| 6. Mengubah data yang dipilih |  |
|  | 7. Menyimpan data yang diubah ke *database* |
|  | 8. Menampilkan pesan data sukses disimpan |

Tabel IV.17 Skenario Use Case Hapus Data Kelas

|  |  |
| --- | --- |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal |  |
|  | 1. Memeriksa status login |
| 2. Memilih menu Kelas |  |
|  | 3. Melihat data Kelas |
| 4. Memilih data yang akan dihapus |  |
| 6. Mengkonfirmasi penghapusan data |  |
|  | 7. menghapus data dari *database* |
|  | 8. Menampilkan pesan data sukses dihapus |

Tabel IV.18 Skenario Use Case Tambah Data Rombel

|  |  |
| --- | --- |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal |  |
|  | 1. Memeriksa status login |
| 2. Memilih menu data Rombel |  |
|  | 3. Melihat table data Rombel |
| 4. memilih tombol tambah Rombel |  |
|  | 5. Menampilkan tabel data Rombel |
| 6. mengisi data Rombel |  |
|  | 7. Menyimpan data yang diubah ke *database* |
|  | 8. Menampilkan pesan data sukses disimpan |

*Tabel IV.19 Skenario Use Case Edit Data Rombel*

|  |  |
| --- | --- |
| Skenario Normal |  |
|  | 1. Memeriksa status login |
| 2. Memilih menu Data Rombel |  |
|  | 3. Melihat Data Rombel |
| 4. Memilih edit data Rombel |  |
|  | 5. Menampilkan tabel data Rombel |
| 6. Mengubah data yang dipilih |  |
|  | 7. Menyimpan data yang diubah ke *database* |
|  | 8. Menampilkan pesan data sukses disimpan |

Tabel IV.20 Skenario Use Case Hapus Data Rombel

|  |  |
| --- | --- |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal |  |
|  | 1. Memeriksa status login |
| 2. Memilih menu Rombel |  |
|  | 3. Melihat data Rombel |
| 4. Memilih data yang akan dihapus |  |
| 6. Mengkonfirmasi penghapusan data |  |
|  | 7. menghapus data dari *database* |
|  | 8. Menampilkan pesan data sukses dihapus |

Tabel IV.21 Skenario Use Case Tambah Data Wali Kelas

|  |  |
| --- | --- |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal |  |
|  | 1. Memeriksa status login |
| 2. Memilih menu data Wali Kelas |  |
|  | 3. Melihat table data Wali Kelas |
| 4. memilih tombol tambah Wali Kelas |  |
|  | 5. Menampilkan tabel data Rombel |
| 6. mengisi data Wali Kelas |  |
|  | 7. Menyimpan data yang diubah ke *database* |
|  | 8. Menampilkan pesan data sukses disimpan |

Tabel IV.22 Skenario Use Case Edit Data Rombel

|  |  |
| --- | --- |
| Skenario Normal |  |
|  | 1. Memeriksa status login |
| 2. Memilih menu Data Wali Kelas |  |
|  | 3. Melihat Data Wali Kelas |
| 4. Memilih edit data Wali Kelas |  |
|  | 5. Menampilkan tabel data Wali Kelas |
| 6. Mengubah data yang dipilih |  |
|  | 7. Menyimpan data yang diubah ke *database* |
|  | 8. Menampilkan pesan data sukses disimpan |

Tabel IV.23 Skenario Use Case Hapus Data Wali Kelas

|  |  |
| --- | --- |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal |  |
|  | 1. Memeriksa status login |
| 2. Memilih menu Wali Kelas |  |
|  | 3. Melihat data Wali Kelas |
| 4. Memilih data yang akan dihapus |  |
| 6. Mengkonfirmasi penghapusan data |  |
|  | 7. menghapus data dari *database* |
|  | 8. Menampilkan pesan data sukses dihapus |

Tabel IV.24 Skenario Use Case Tambah Data Guru

|  |  |
| --- | --- |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal |  |
|  | 1. Memeriksa status login |
| 2. Memilih menu data Guru |  |
|  | 3. Melihat table data Guru |
| 4. memilih tombol tambah Wali Kelas |  |
|  | 5. Menampilkan tabel data Guru |
| 6. mengisi data Guru |  |
|  | 7. Menyimpan data yang diubah ke *database* |
|  | 8. Menampilkan pesan data sukses disimpan |

Tabel IV.25 Skenario Use Case Edit Data guru

|  |  |
| --- | --- |
| Skenario Normal |  |
|  | 1. Memeriksa status login |
| 2. Memilih menu Data Guru |  |
|  | 3. Melihat Data Guru |
| 4. Memilih edit data Guru |  |
|  | 5. Menampilkan tabel data Guru |
| 6. Mengubah data yang dipilih |  |
|  | 7. Menyimpan data yang diubah ke *database* |
|  | 8. Menampilkan pesan data sukses disimpan |

Tabel IV.26 Skenario Use Case Hapus Data Wali Kelas

|  |  |
| --- | --- |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal |  |
|  | 1. Memeriksa status login |
| 2. Memilih menu Guru |  |
|  | 3. Melihat data Guru |
| 4. Memilih data yang akan dihapus |  |
| 6. Mengkonfirmasi penghapusan data |  |
|  | 7. menghapus data dari *database* |
|  | 8. Menampilkan pesan data sukses dihapus |

Tabel IV.27 Skenario Use Case Tambah Data Mata Pelajaran

|  |  |
| --- | --- |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal |  |
|  | 1. Memeriksa status login |
| 2. Memilih menu data Mata Pelajatan |  |
|  | 3. Melihat table data Mata Pelajaran |
| 4. memilih tombol tambah Mata Pelajaran |  |
|  | 5. Menampilkan tabel data Mata Pelajaran |
| 6. mengisi data Mata Pelajaran |  |
|  | 7. Menyimpan data yang diubah ke *database* |
|  | 8. Menampilkan pesan data sukses disimpan |

Tabel IV.28 Skenario Use Case Edit Data Mata Pelajaran

|  |  |
| --- | --- |
| Skenario Normal |  |
|  | 1. Memeriksa status login |
| 2. Memilih menu Data Mata Pelajaran |  |
|  | 3. Melihat Data Mata Pelajaran |
| 4. Memilih edit data Mata Pelajaran |  |
|  | 5. Menampilkan tabel data Mata Pelajaran |
| 6. Mengubah data yang dipilih |  |
|  | 7. Menyimpan data yang diubah ke *database* |
|  | 8. Menampilkan pesan data sukses disimpan |

Tabel IV.29 Skenario Use Case Hapus Data Mata Pelajaran

|  |  |
| --- | --- |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal |  |
|  | 1. Memeriksa status login |
| 2. Memilih menu Mata Pelajaran |  |
|  | 3. Melihat data Mata Pelajaran |
| 4. Memilih data yang akan dihapus |  |
| 6. Mengkonfirmasi penghapusan data |  |
|  | 7. menghapus data dari *database* |
|  | 8. Menampilkan pesan data sukses dihapus |

Tabel IV.30 Skenario Use Case Tambah Data Pengajar

|  |  |
| --- | --- |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal |  |
|  | 1. Memeriksa status login |
| 2. Memilih menu data Pengajar |  |
|  | 3. Melihat table data Pengajar |
| 4. memilih tombol tambah Pengajar |  |
|  | 5. Menampilkan tabel data Pengajar |
| 6. mengisi data Pengajar |  |
|  | 7. Menyimpan data yang diubah ke *database* |
|  | 8. Menampilkan pesan data sukses disimpan |

*Tabel IV.31 Skenario Use Case Edit Data Pengajar*

|  |  |
| --- | --- |
| Skenario Normal |  |
|  | 1. Memeriksa status login |
| 2. Memilih menu Data Pengajar |  |
|  | 3. Melihat Data Pengajar |
| 4. Memilih edit data Pengajar |  |
|  | 5. Menampilkan tabel data Pengajar |
| 6. Mengubah data yang dipilih |  |
|  | 7. Menyimpan data yang diubah ke *database* |
|  | 8. Menampilkan pesan data sukses disimpan |

Tabel IV.32 Skenario Use Case Hapus Data Pengajar

|  |  |
| --- | --- |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal |  |
|  | 1. Memeriksa status login |
| 2. Memilih menu Pengajar |  |
|  | 3. Melihat data Pengajar |
| 4. Memilih data yang akan dihapus |  |
| 6. Mengkonfirmasi penghapusan data |  |
|  | 7. menghapus data dari *database* |
|  | 8. Menampilkan pesan data sukses dihapus |

Tabel IV.33 Skenario Use Case Tambah Data Pengguna

|  |  |
| --- | --- |
| Aksi Aktor | Reaksi Sistem |
| Skenario Normal |  |
|  | 1. Memeriksa status login |
| 2. Memilih menu data Pengguna |  |
|  | 3. Melihat table data Pengguna |
| 4. memilih tombol tambah Pengguna |  |
|  | 5. Menampilkan tabel data Pengguna |
| 6. mengisi data Pengguna |  |
|  | 7. Menyimpan data yang diubah ke *database* |
|  | 8. Menampilkan pesan data sukses disimpan |

Tabel IV.34 Skenario Use Case Edit Data Pengguna

|  |  |
| --- | --- |
| Skenario Normal |  |
|  | 1. Memeriksa status login |
| 2. Memilih menu Data Pengguna |  |
|  | 3. Melihat Data Pengguna |
| 4. Memilih edit data Pengguna |  |
|  | 5. Menampilkan tabel data Pengguna |
| 6. Mengubah data yang dipilih |  |
|  | 7. Menyimpan data yang diubah ke *database* |
|  | 8. Menampilkan pesan data sukses disimpan |

* 1. *Activity Diagram*

Pada *activity diagram* menjelaskan tentang aktivitas-aktivitas dari *use case diagram*. Pada *activity diagram* ini menjelaskan langkah-langkah aktivitas berdasarkan menu yang ada pada aplikasi sistem informasi akademik. Menu pada aplikasi ini berisi dengan menu home sampai login, data akademik, data pengguna.

A screenshot of a computer

Description automatically generated

Gambar IV. 2 *Activity Diagram* Login

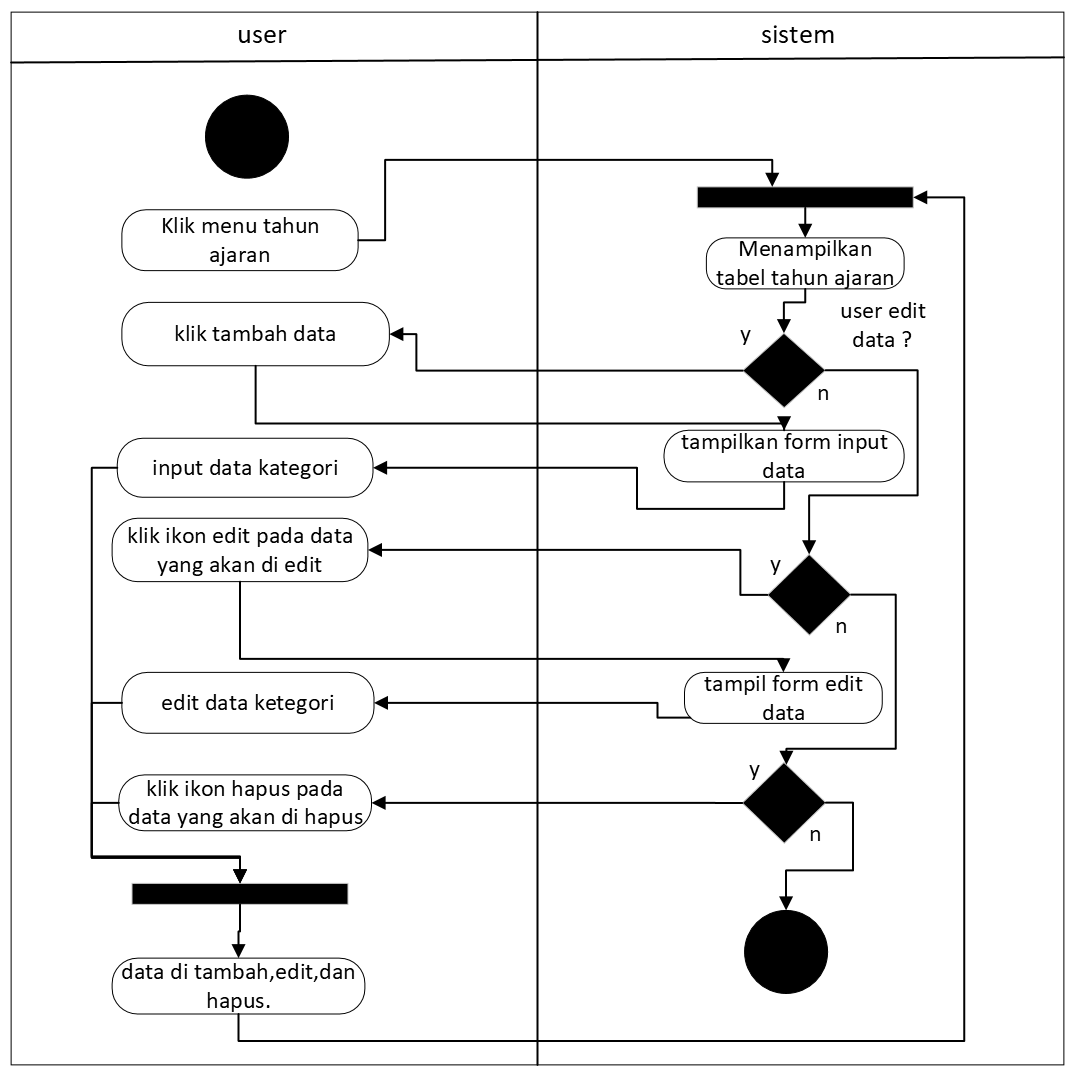
Pada activity diagram ini, menjelaskan alur aplikasi jika *user* mengakses Login. Saat *user* mengakses login, maka akan dialihkan ke halaman form login.

A screenshot of a computer

Description automatically generated

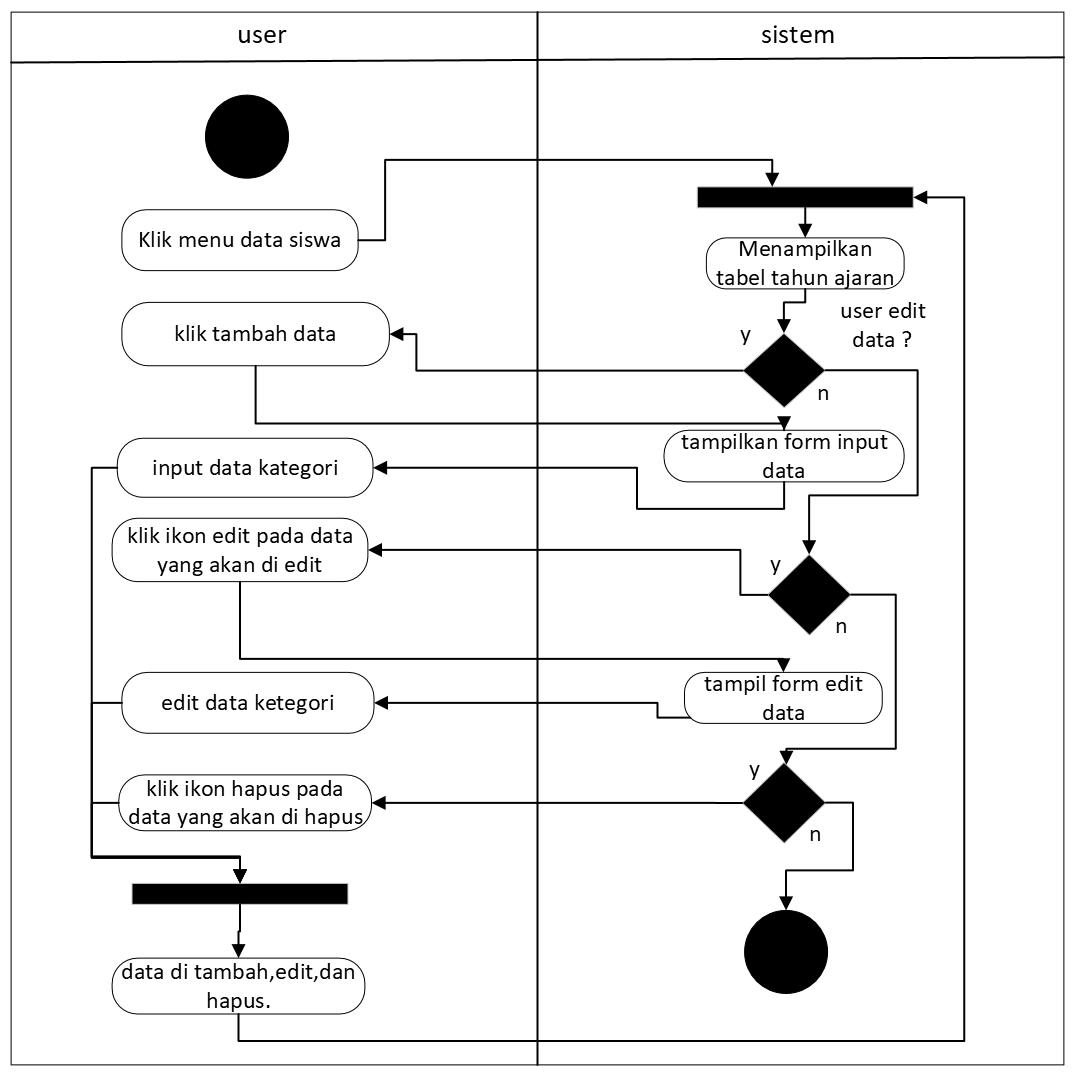
Gambar IV. 3 *Activity Diagram* Profil Sekolah

Pada activity diagram ini, menjelaskan alur aplikasi jika *user* mengakses Profil Sekolah. Saat *user* mengakses login, maka akan dialihkan ke halaman Profil Sekolah.



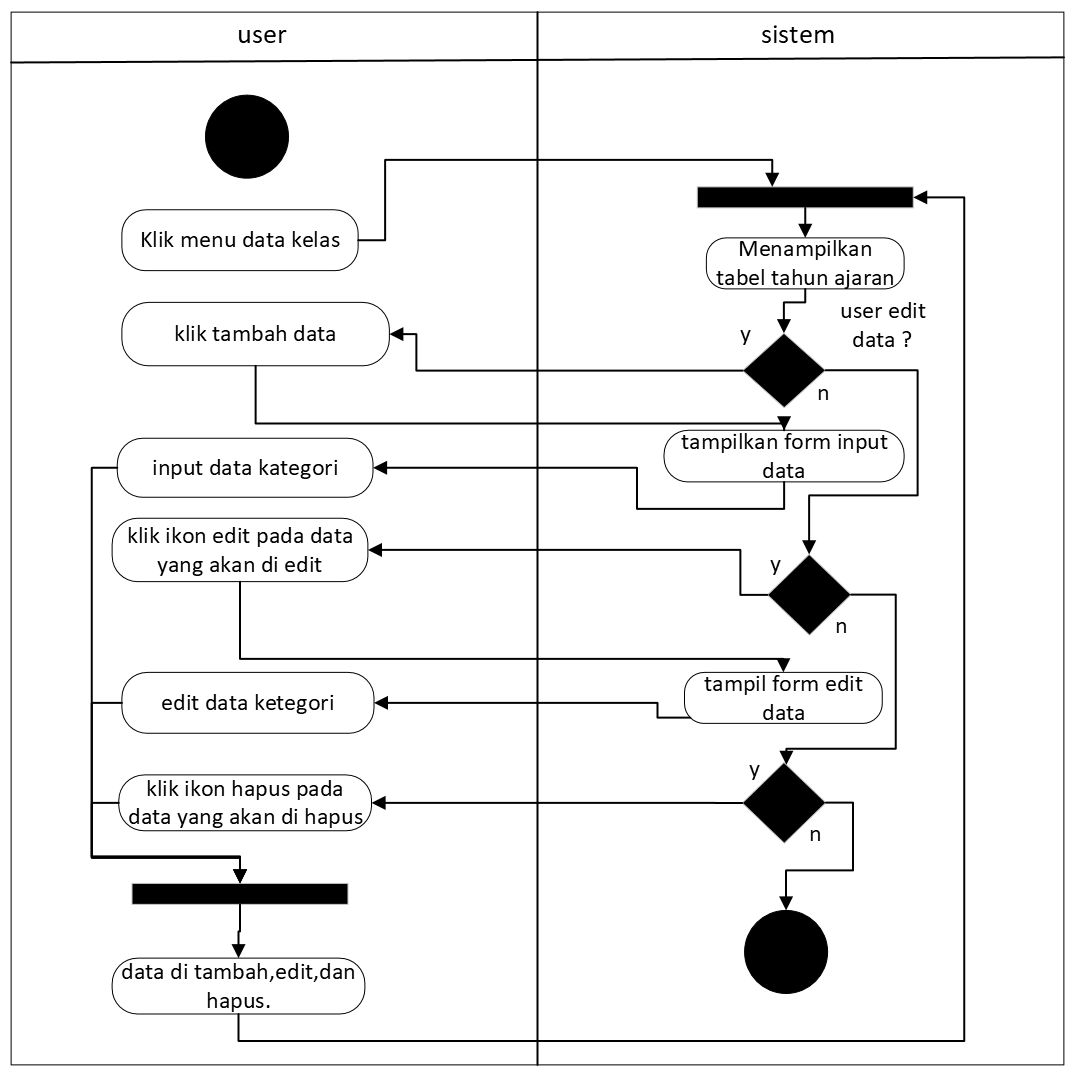
Gambar IV. 4 *Activity Diagram* tahun ajaran

Pada activity diagram peralatan, menjelaskan alur aplikasi jika *user* mengakses menu Tahun Pelajaran. Pada menu peralatan, memiliki 3 *method* yaitu *read, update* dan *delete*.



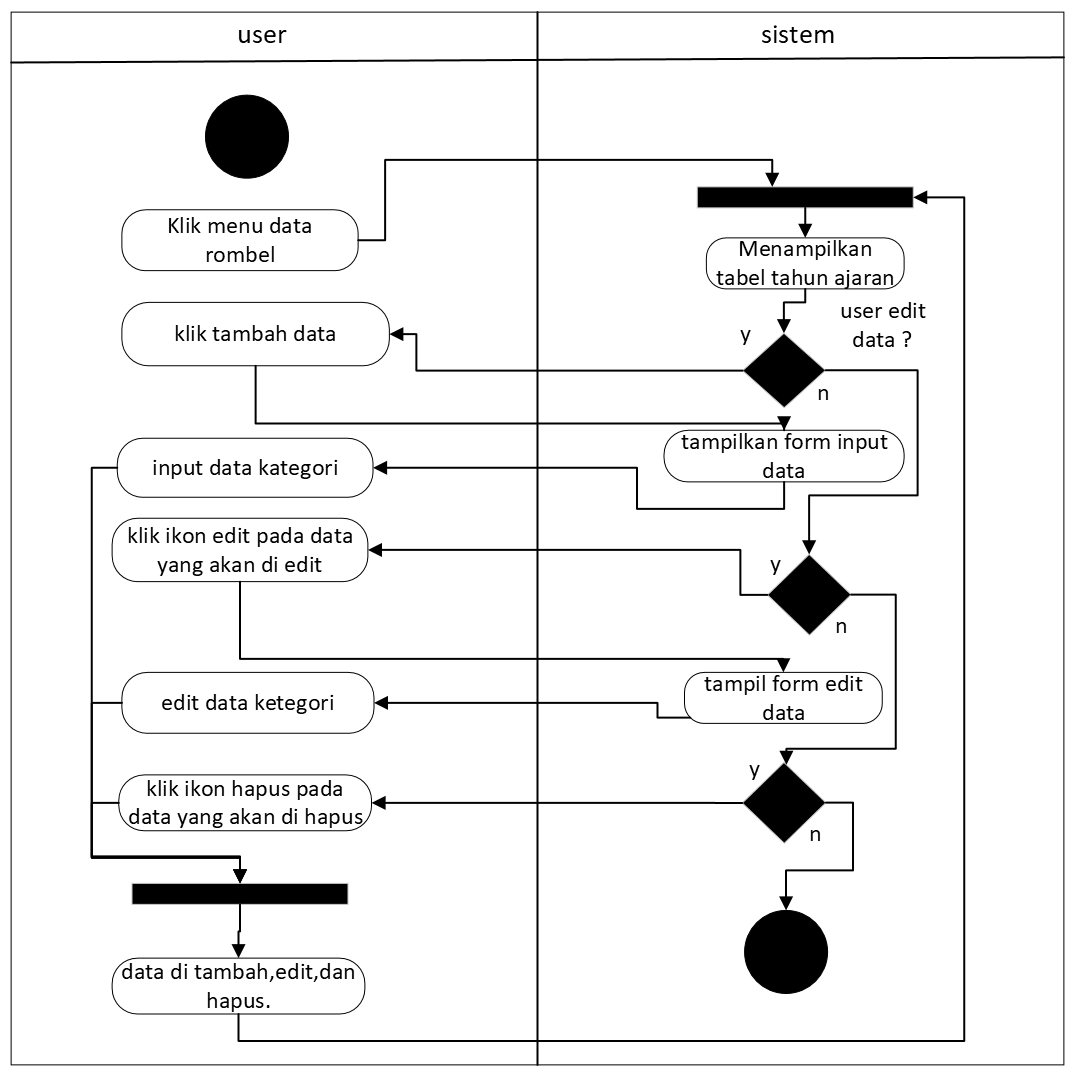
Gambar IV. 5 *Activity Diagram* data siswa

Pada activity diagram data akademik, menjelaskan alur aplikasi jika *user* mengakses menu data akademik. Pada menu peralatan, memiliki 3 *method* yaitu *read, update* dan *delete*.



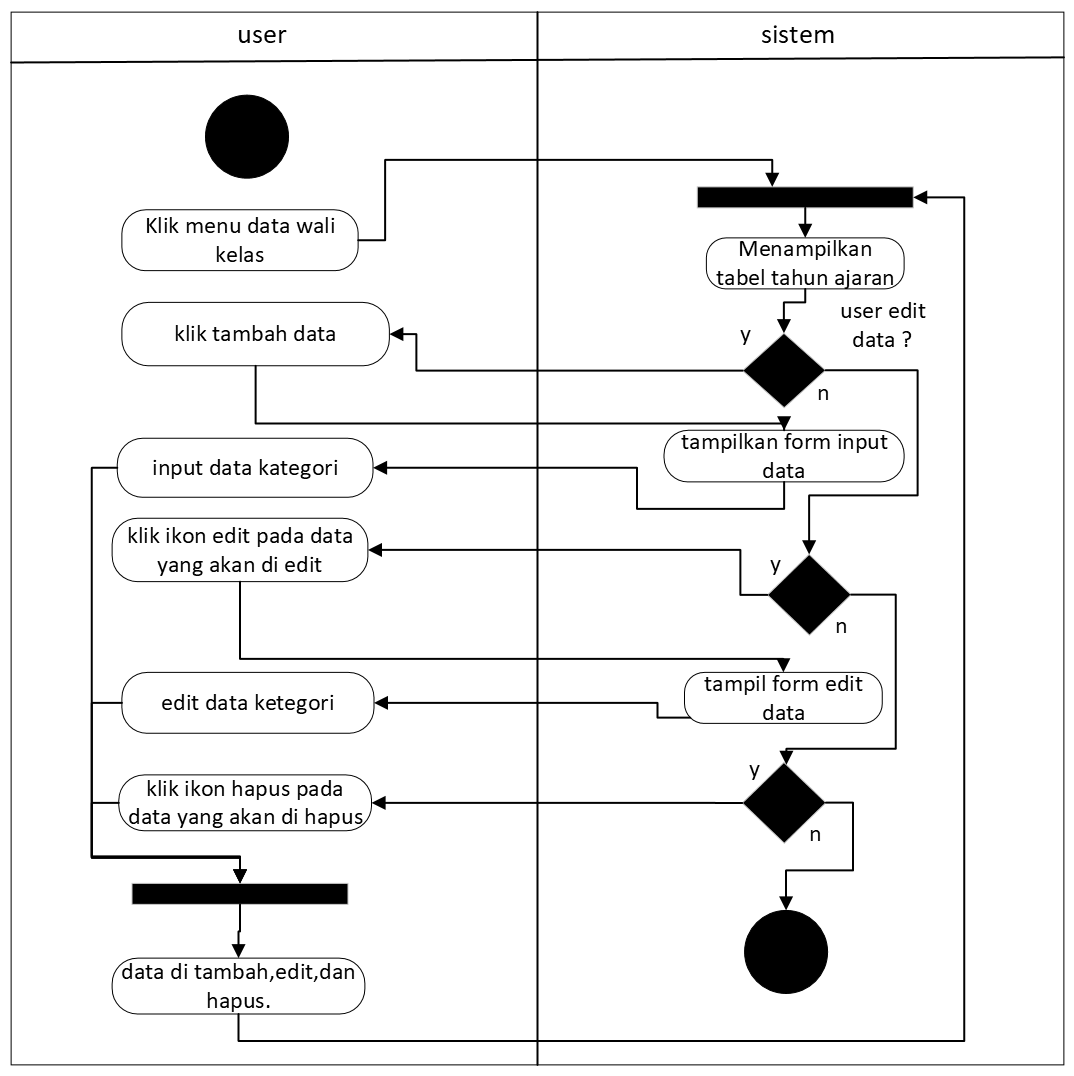
Gambar IV. 6 *Activity Diagram* Data Kelas

Pada Activity Diagram Data Kelas ini memiliki Proses yang sama seperti Activity Diagram sebelumnya



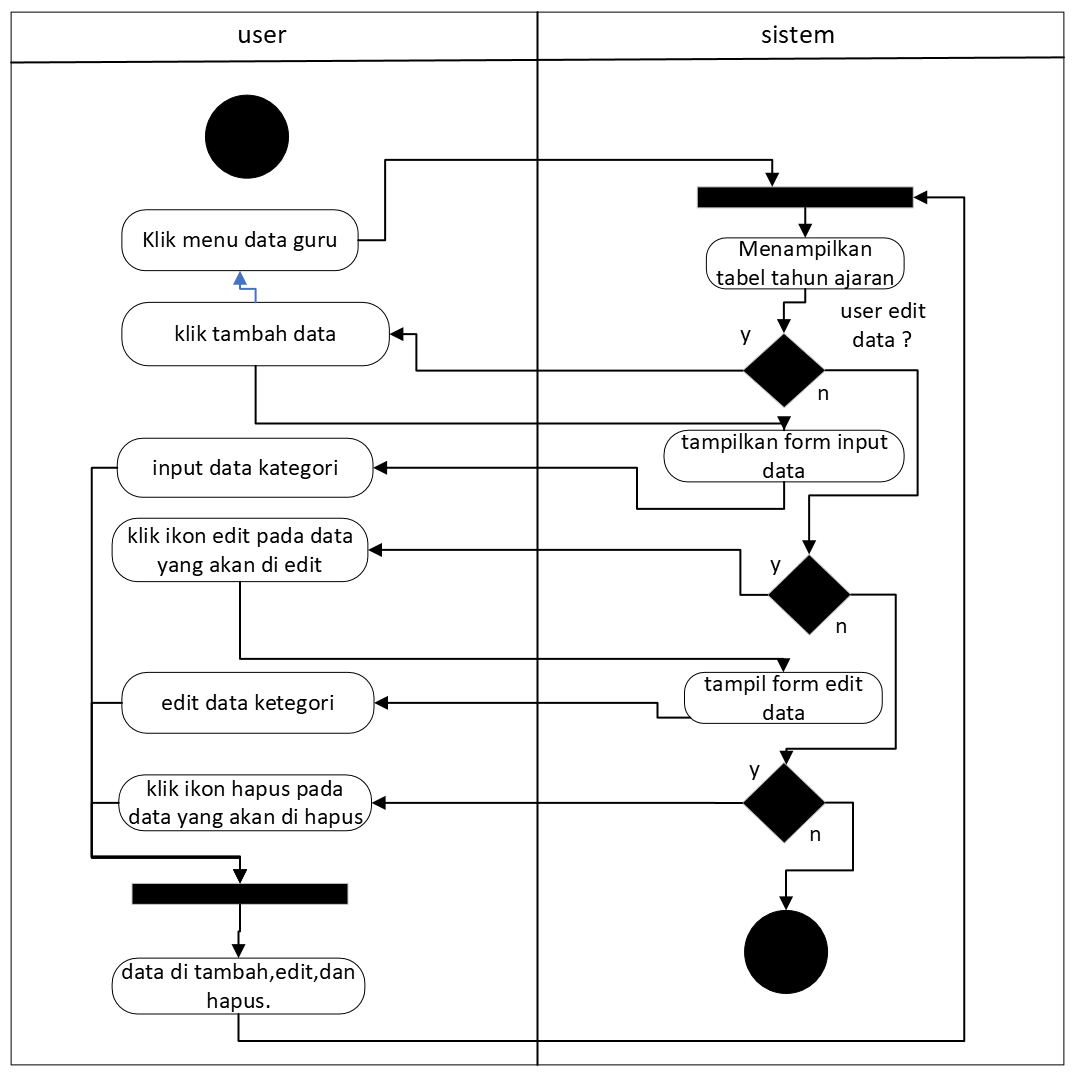
Gambar IV. 7 *Activity Diagram* Dara Rombel

Pada Activity Diagram Data Rombel ini memiliki Proses yang sama seperti Activity Diagram sebelumnya



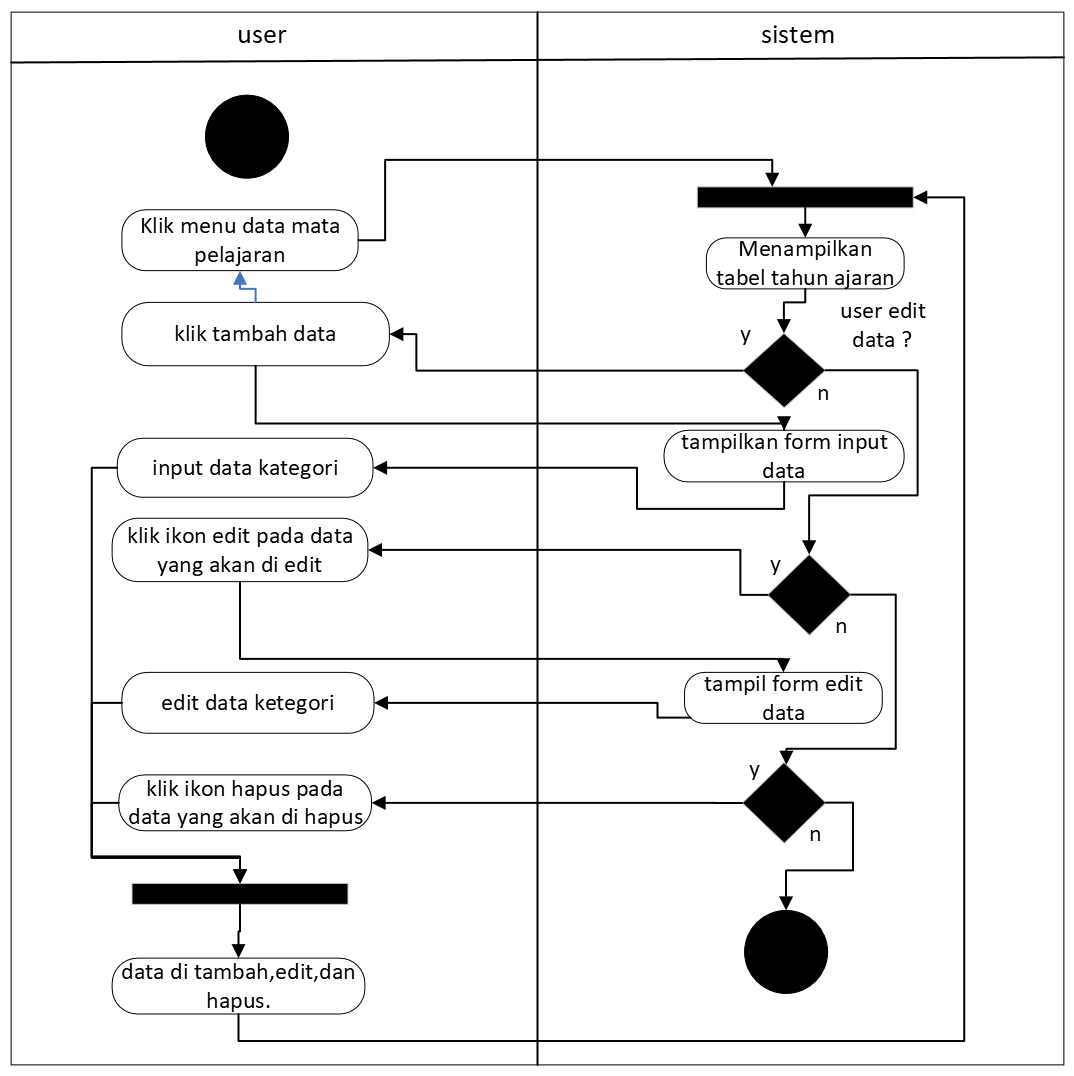
Gambar IV. 8 *Activity Diagram* Data Wali Kelas

Pada Activity Diagram Data Wali Kelas ini memiliki Proses yang sama seperti Activity Diagram sebelumnya



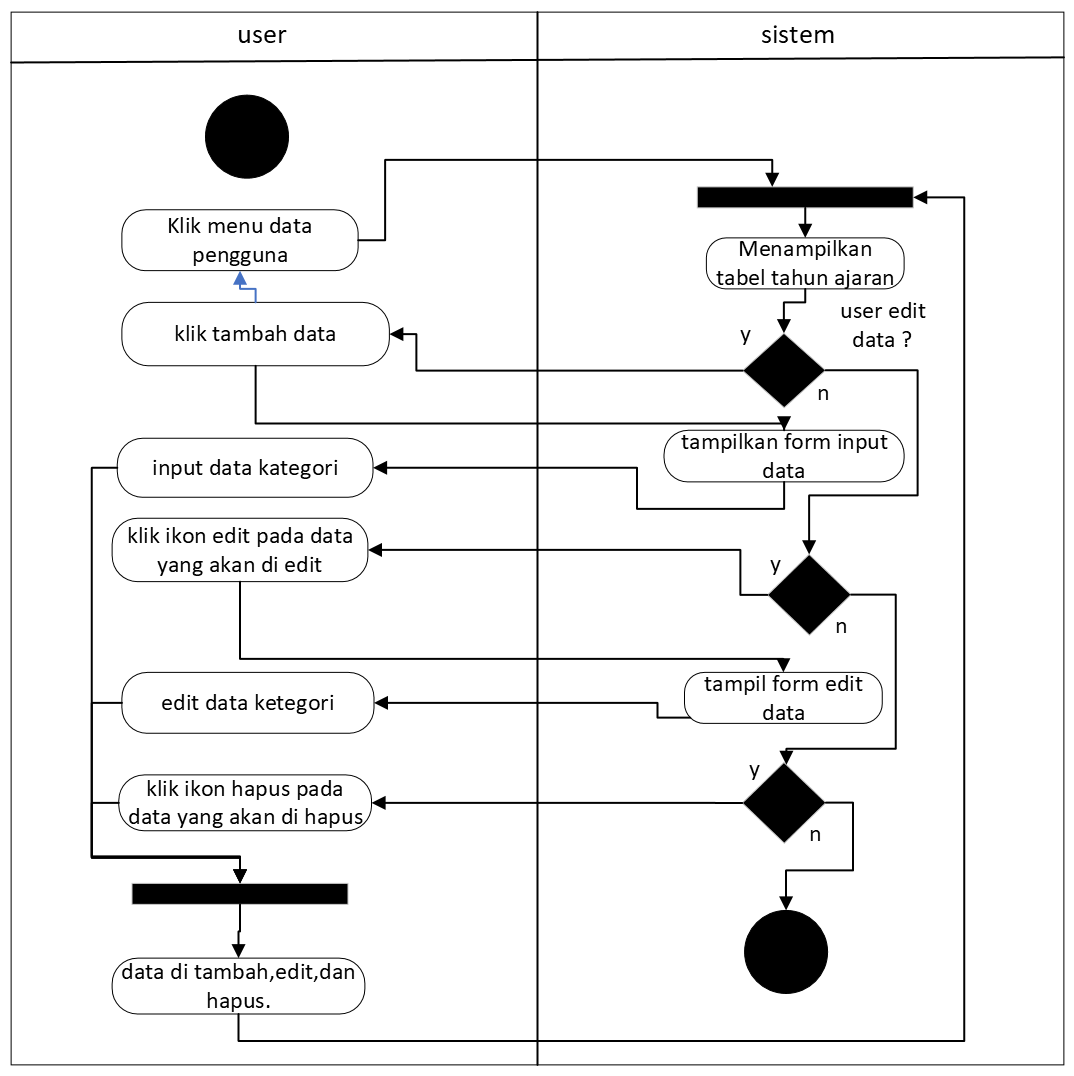
Gambar IV. 9 *Activity Diagram* Data Guru

Pada Activity Diagram Data Guru ini memiliki Proses yang sama seperti Activity Diagram sebelumnya



Gambar IV. 10 *Activity Diagram* Mata Pelajaran

Pada Activity Diagram Data Mata Kelas ini memiliki Proses yang sama seperti Activity Diagram sebelumnya



Gambar IV. 11 *Activity Diagram* Data pengguna

Pada Activity Diagram Data Pengguna ini memiliki Proses yang sama seperti Activity Diagram sebelumnya

* 1. Class Diagram

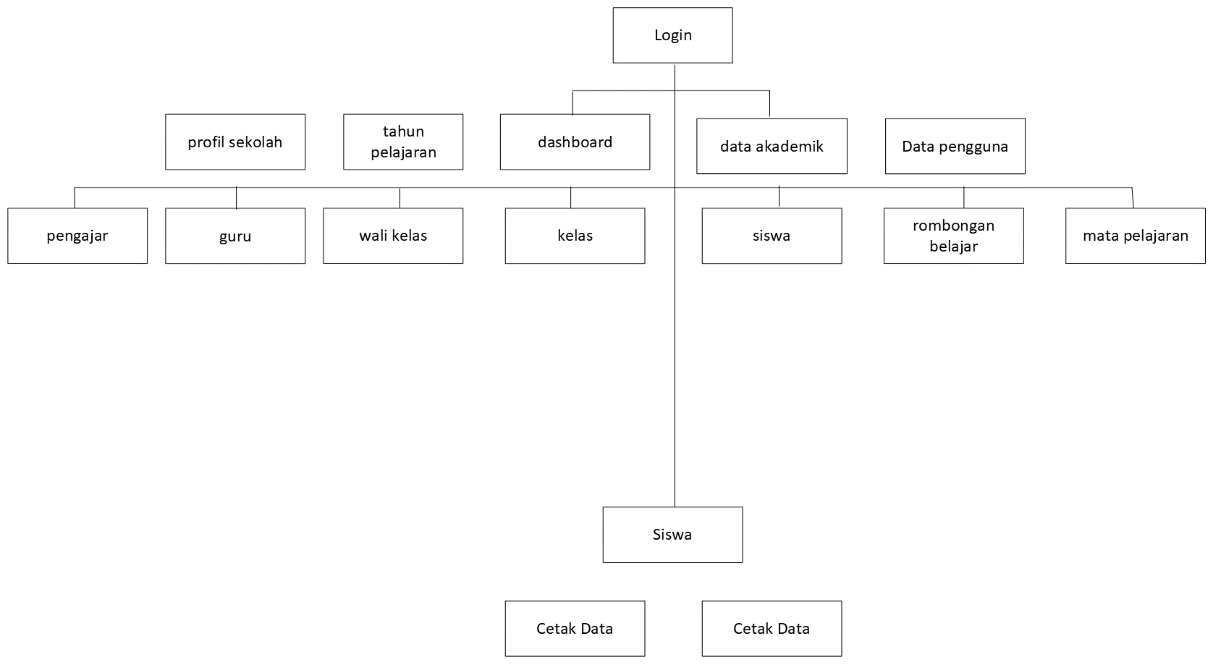
Pada perancangan class diagram menjelaskan mengenai tabel

database yang akan dibuat untuk aplikasi apengelola data akademik komputer. Pada tabel ini akan menunjukan atribut, method dan

relasi penghubung di tiap tabel yang ada.

* 1. Perancangan Tampilan Antarmuka

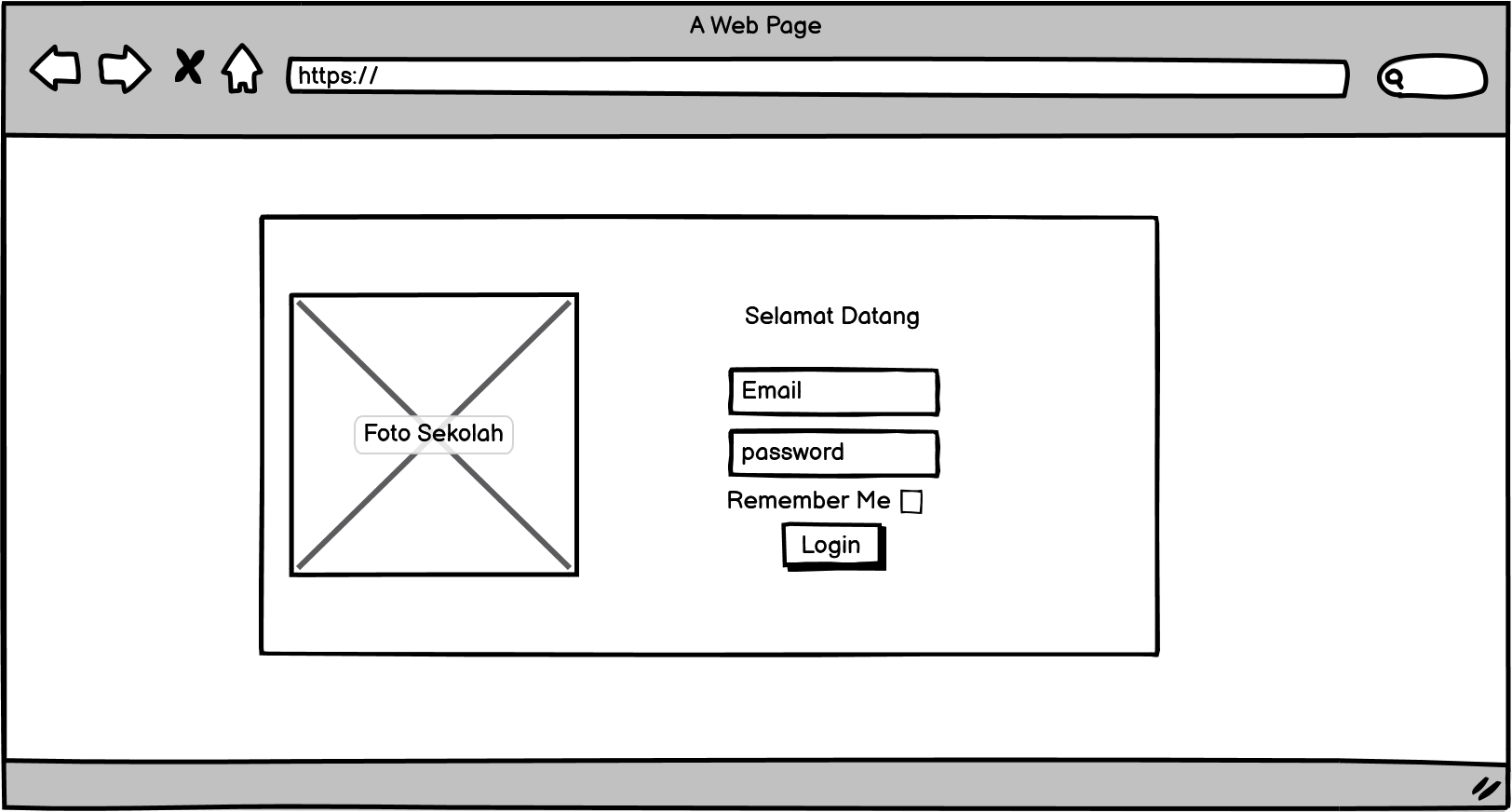
Pada tahapan ini yaitu tahap perancangan tampilan antarmuka aplikasi sistem informasi akademik yang dilakukan dengan memanfaatkan *tools* atau *software* yang sudah dijelaskan dibab sebelumnya, yaitu dengan menggunakan aplikasi Balsamiq Mokups. Perancangan desain tampilan antarmuka ditujukan agar tampilan website yang akan dibuat sudah terancang dengan baik. Ketika akan membuat tampilan website hanya tinggal mengikuti desain yang sudah dibuat dengan aplikasi balsamiq mokups



Gambar IV. 12 *Alur Aplikasi*

Adapun desain tampilan antarmuka dari aplikasi sistem informasi akademik ini sebagai berikut:

* + 1. Desain Tampilan Login



Gambar IV. 13 Desain Tampilan Login

Berikut merupakan deskripsi dari rancangan mockup antarmuka login:

Tabel IV.35 Deskripsi mockup antarmuka login

|  |  |  |
| --- | --- | --- |
| Komponen | Tipe | Fungsi |
| Foto Sekolah | Image | Foto Sekolah |
| Masukkan Akun | Teks | Keterangan untuk memasukkan username dan password |
| Username | Input  Teks | Username akun untuk login |
| Password | Input  Teks | Password akun untuk login |
| Login | Tombol | 1. Konfirmasi username dan password akun untuk login 2. Ke halaman menu dashboard system informasi akademik |

Tabel IV.36 Deskripsi mockup antarmuka dashboard

2.Dashboard

Graphical user interface

Description automatically generated

Gambar IV. 14 Desain Tampilan Menu Dashboard

Berikut merupakan deskripsi dari rancangan mockup antarmuka dashboard:

|  |  |  |
| --- | --- | --- |
| Komponen | Tipe | Fungsi |
| dashboard | Tombol | Ke halaman dashboard/home |
|  | Tombol | Tampilkan/sembunyikan sidebar menu |
| Profil | Tombol | 1. Informasi akun 2. Logout |
| Logo | Image | Logo SMA Pelita Bunga Bangsa |
| Profil Sekolah | Teks | Kategori menu yang berisikan informasi sekolah |
| Tahun Pelajaran | tombol | Kategori menu data yang bersangkutan dengan pengelolaan data tahun ajaran |
| Data akademik | tombol | Kategori menu untuk sub menu data siswa ,guru,mata pelajaran,rombongan belajar,wali kelas, dan pengajar . |
| Data pengguna | Tombol | Kategori menu untuk menambahkan, dan edit akun login |
| Jumlah siswa | text | Menampilkan jumlah siswa aktif yang terdaftar |
| Jumlah guru | text | Menampilkan jumlah guru yang mengajar di sekolah |
| Jumlah kelas | Text | Menampilkan jumlah kelas yang ada di sekolah |
| Jumlah mata pelajaran | Text | Menampilkan jumlah seluruh mata pelajaran yang ada di sekolah |
| Tahun ajaran | Text | Menampilkan informasi tahun ajaran yang sedang berlangsung |
| System informasi akademik sma pelita bunga bangsa | Text | Nama aplikasi |
| Profil singkat sekolah | Text | Menampilkan profil singkat sekolah sma pelita bunga bangsa |

* + 1. Desain Tampilan Profil Sekolah

Graphical user interface, table

Description automatically generated

Gambar IV. 15 Desain Tampilan Profil Sekolah

Berikut merupakan deskripsi dari rancangan mockup antarmuka profil sekolah:

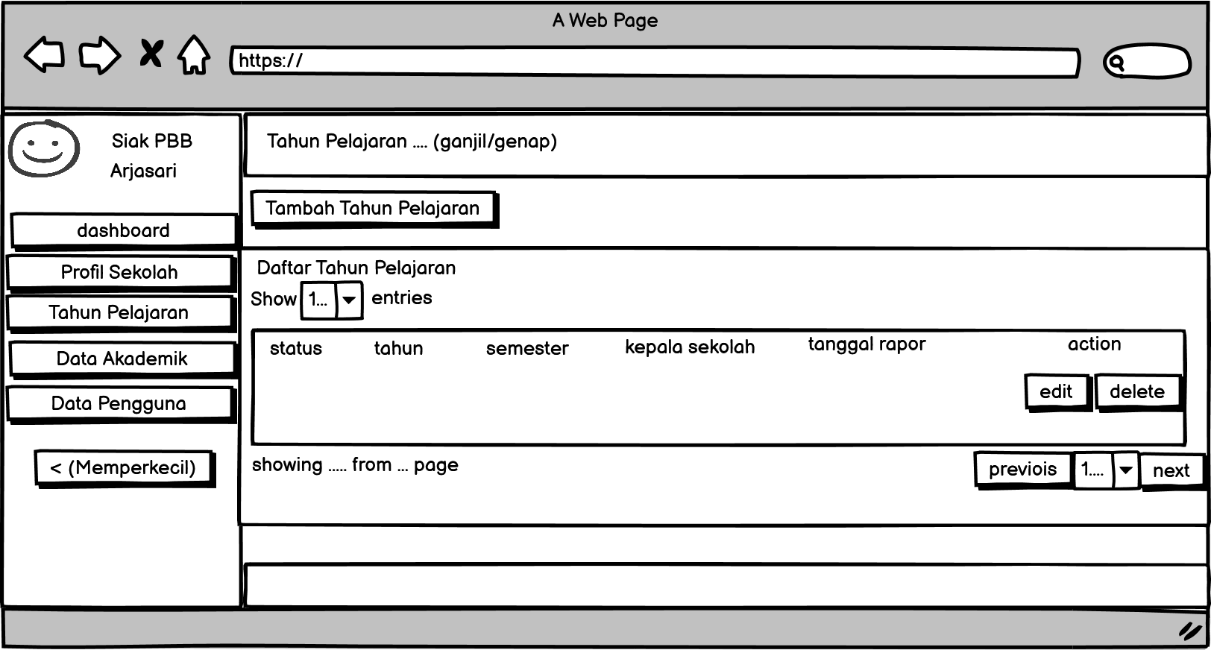
Tabel IV.36 Deskripsi mockup antarmuka dashboard

|  |  |  |
| --- | --- | --- |
| Komponen | Tipe | Fungsi |
| dashboard | Tombol | Ke halaman dashboard/home |
| Profil | Tombol | 1. Informasi akun 2. Logout |
| Logo | Image | Logo SMA Pelita Bunga Bangsa |
| Profil Sekolah | Teks | Kategori menu yang berisikan informasi sekolah |
| Tahun Pelajaran | tombol | Kategori menu data yang bersangkutan dengan pengelolaan data tahun ajaran |
| Data akademik | tombol | Kategori menu untuk sub menu data siswa ,guru,mata pelajaran,rombongan belajar,wali kelas, dan pengajar . |
| Data pengguna | Tombol | Kategori menu untuk menambahkan, dan edit akun login |
| Tahun ajaran | text | Menampilkan informasi tahun ajaran yang sedang berlangsung |
| Profil lengkap sekolah | Text | Menampilkan profil lengkap sma pelita bunga bangsa arjasari |

Tabel IV.37 Deskripsi mockup antarmuka profil sekolah

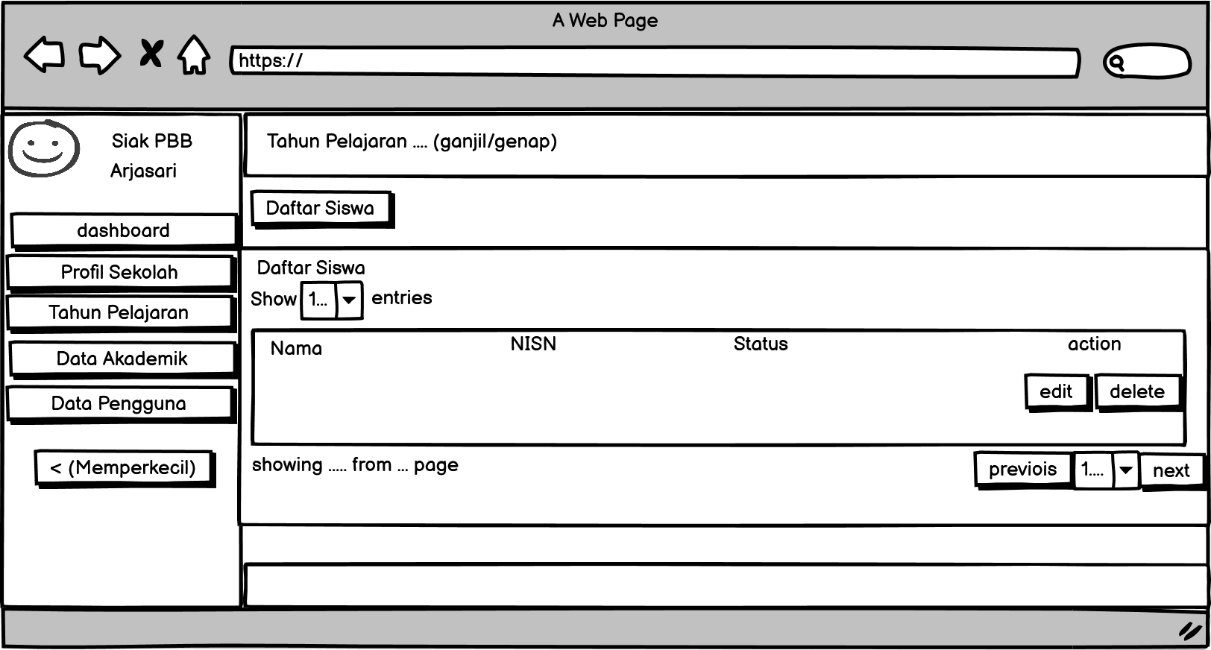
|  |  |  |
| --- | --- | --- |
| Komponen | Tipe | Fungsi |
| dashboard | Tombol | Ke halaman dashboard/home |
| Profil | Tombol | 1. Informasi akun 2. Logout |
| Logo | Image | Logo SMA Pelita Bunga Bangsa |
| Profil Sekolah | Teks | Kategori menu yang berisikan informasi sekolah |
| Tahun Pelajaran | tombol | Kategori menu data yang bersangkutan dengan pengelolaan data tahun ajaran |
| Data akademik | tombol | Kategori menu untuk sub menu data siswa ,guru,mata pelajaran,rombongan belajar,wali kelas, dan pengajar . |
| Data pengguna | Tombol | Kategori menu untuk menambahkan, dan edit akun login |
| Tahun ajaran | text | Menampilkan informasi tahun ajaran yang sedang berlangsung |
| Tambah tahun ajaran | Tombol | Kategori menu untuk menambahkan data tahun ajaran |
| Daftar tahun ajaran | Text | Menampilkan table data tahun ajaran |
| Tabel tahun ajaran | Text | Menampilkan data tahun ajaran yang sedang berlangsung |
| Hapus | Tombol | Tombol utuk menghapus data tahun ajaran |
| Edit | Tombol | Tombol untuk edit data tahun ajaran |
| Previous | Tombol | Tombol untuk Kembali ke slide data sebelumnya |
| Next | Tombol | Tombol untuk maju ke slide data berikutnya |

* + 1. Desain Tampilan Data Tahun Ajaran



Gambar IV. 16 DesainTampilan Data Tahun Ajaran

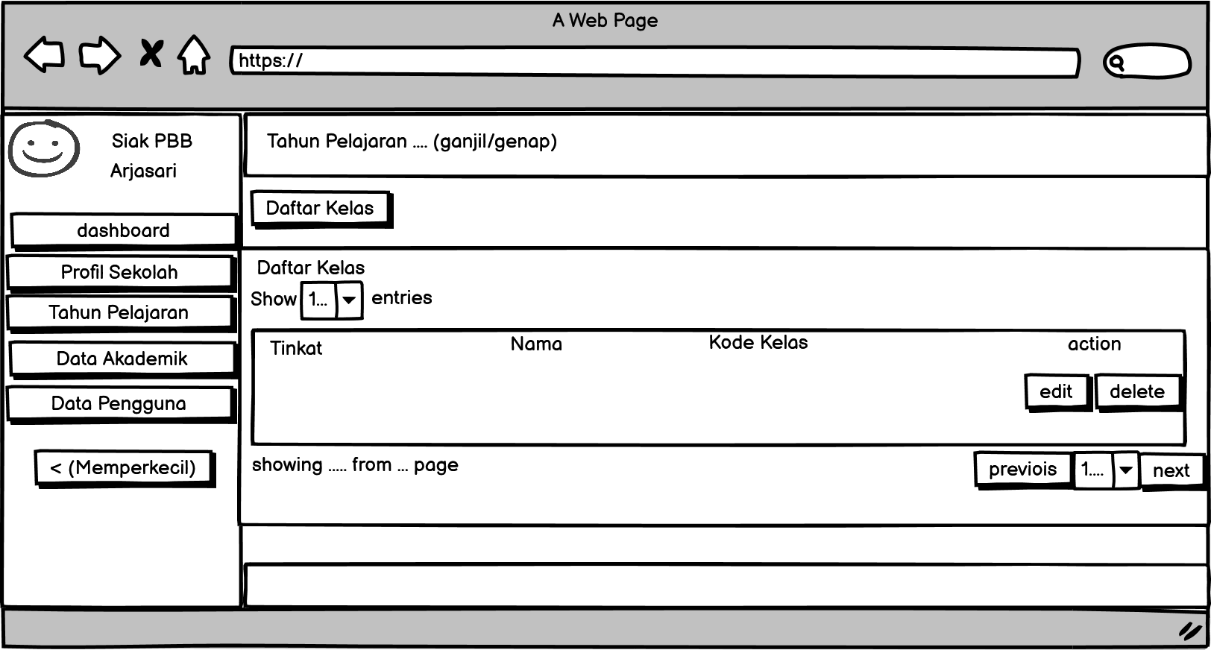
* + 1. Desain Tampilan Data Siswa



Gambar IV. 17 Desain Tampilan Data Siswa

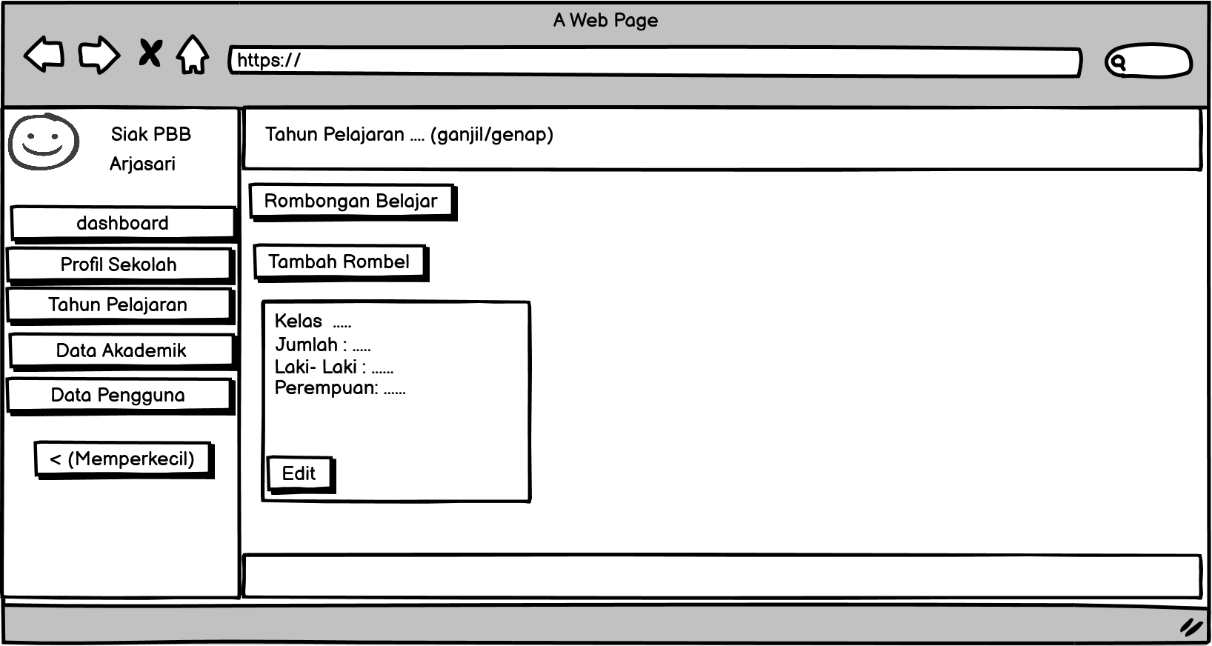
Deskripsi dari rancangan mockup antarmuka data kategori sama saja seperti deskripsi dari rancangan antarmuka sebelumnya. Yang membedakan hanyalah data yang dikelolanya.

* + 1. Desain Tampilan Data Kelas

Gambar IV.18 Desain Tampilan Data Kelas

Deskripsi dari rancangan mockup antarmuka data kategori sama saja seperti deskripsi dari rancangan antarmuka sebelumnya. Yang membedakan hanyalah data yang dikelolanya.

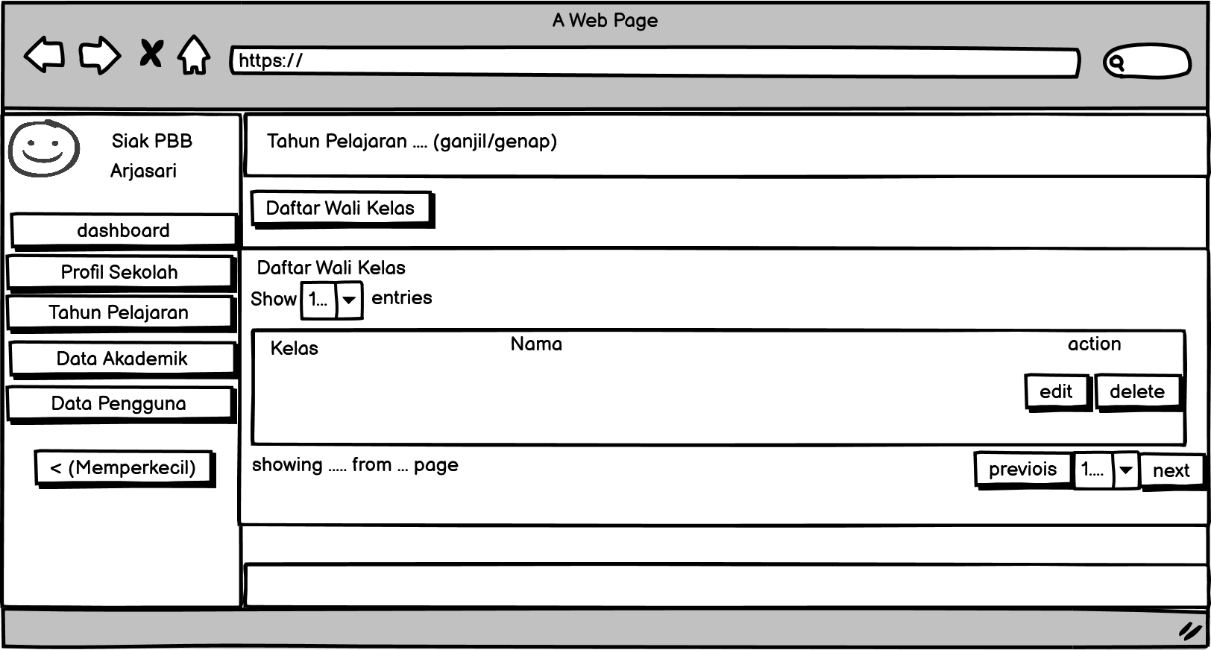
* + 1. Desain Tampilan Tambah Rombel



Gambar IV.19 Desain Tampilan Data Rombel

Deskripsi dari rancangan mockup antarmuka data kategori sama saja seperti deskripsi dari rancangan antarmuka sebelumnya. Yang membedakan hanyalah data yang dikelolanya.

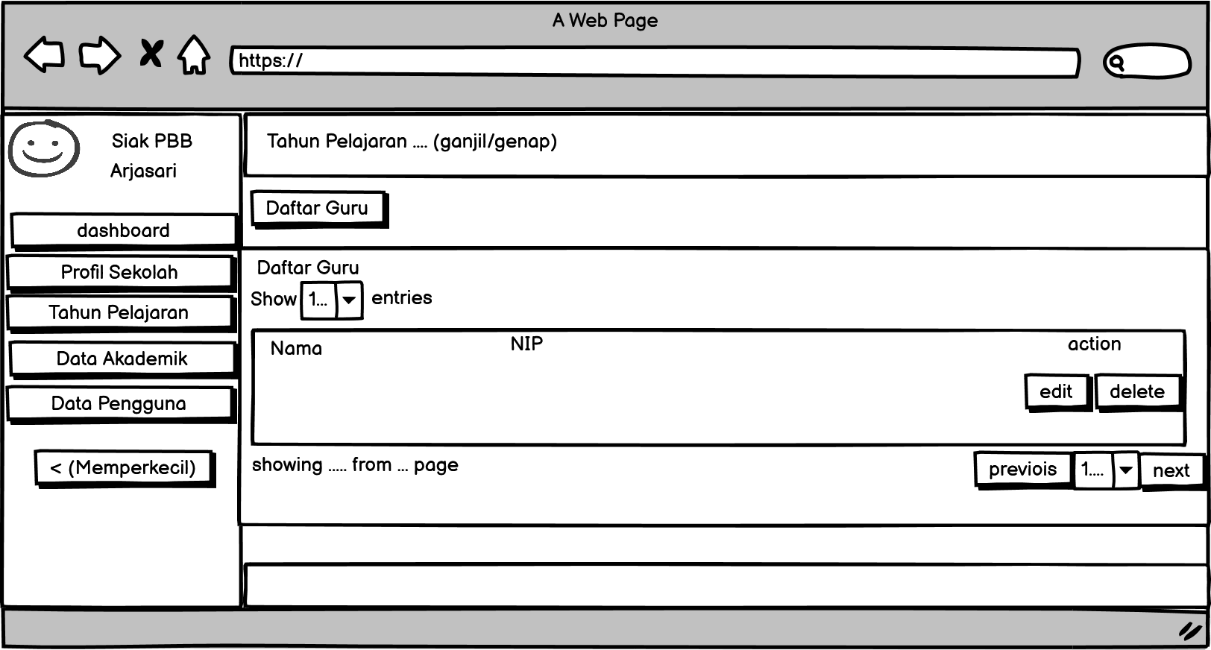
* + 1. Desain Tampilan Data Wali Kelas



Gambar IV.20 Desain Tampilan Data Wali Kelas

Deskripsi dari rancangan mockup antarmuka data kategori sama saja seperti deskripsi dari rancangan antarmuka sebelumnya. Yang membedakan hanyalah data yang dikelolanya.

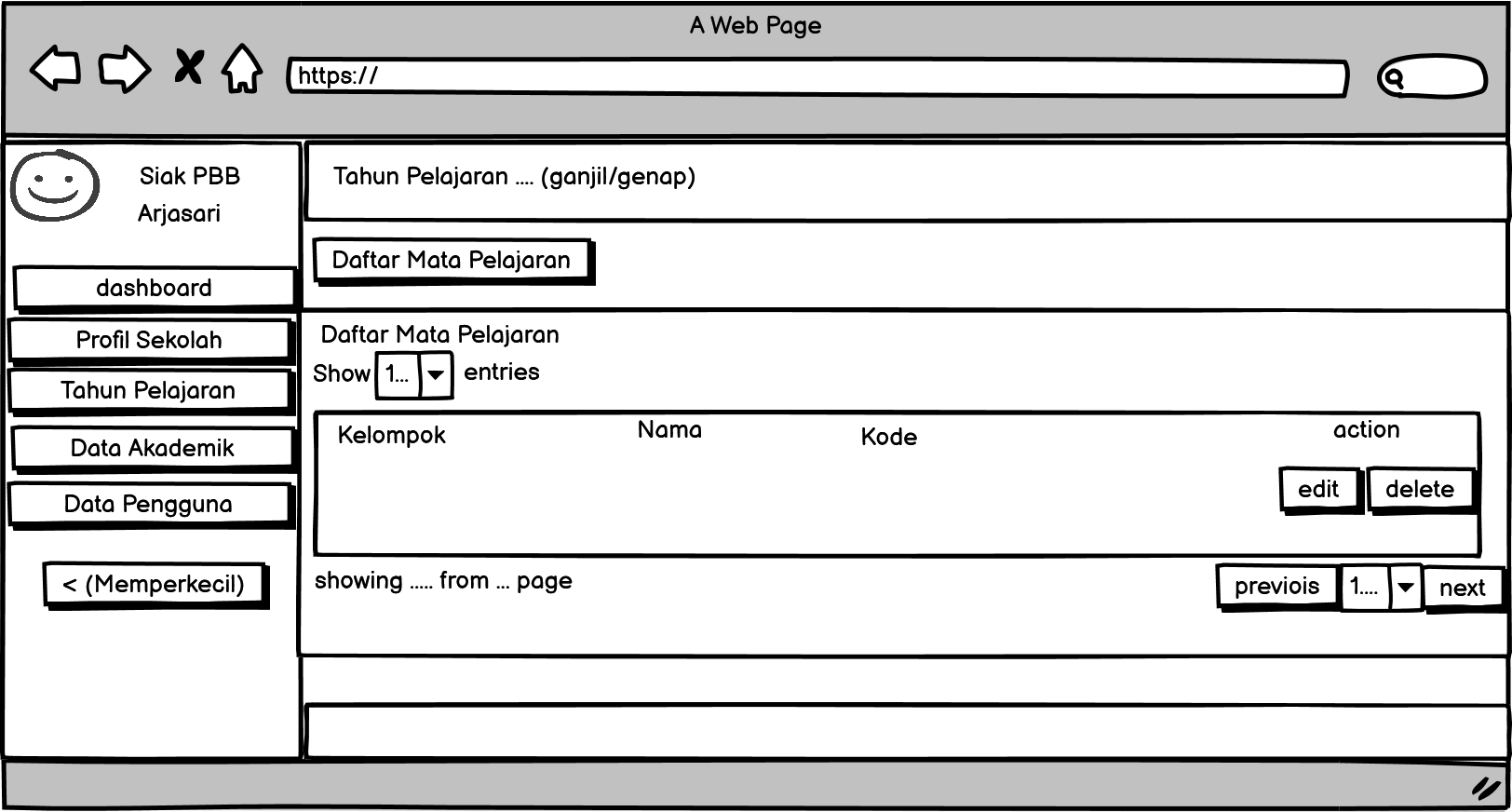
* + 1. Desain Tampilan Data Guru



Gambar IV.21 Desain Tampilan Data Guru

Deskripsi dari rancangan mockup antarmuka data kategori sama saja seperti deskripsi dari rancangan antarmuka sebelumnya. Yang membedakan hanyalah data yang dikelolanya.

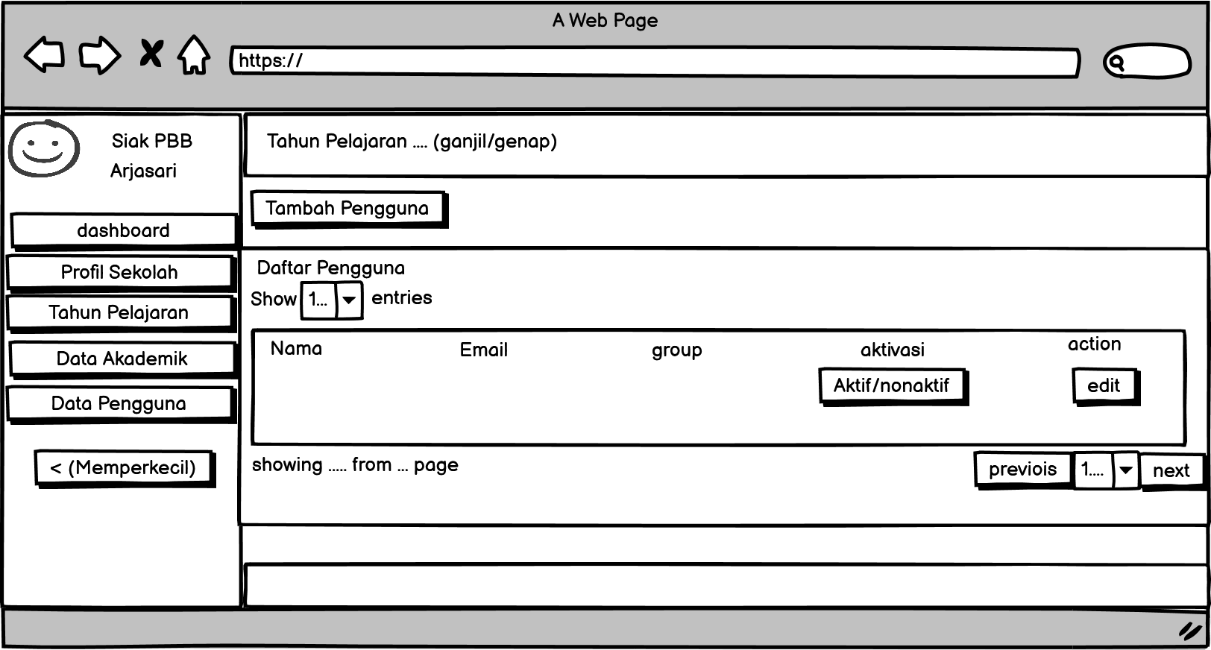
* + 1. Desain Tampilan Data Mata Pelajaran



Gambar IV.22 Desain Tampilan Data Mata Pelajaran

Deskripsi dari rancangan mockup antarmuka data kategori sama saja seperti deskripsi dari rancangan antarmuka sebelumnya. Yang membedakan hanyalah data yang dikelolanya.

* + 1. Desain Tampilan Data Pengguna



Gambar IV.23 Desain Tampilan Data Pengguna

Deskripsi dari rancangan mockup antarmuka data kategori sama saja seperti deskripsi dari rancangan antarmuka sebelumnya. Yang membedakan hanyalah data yang dikelolanya.

* 1. Perancangan Basis Data

Basis data atau *database* dibuat untuk menyimpan semua data untuk dijadikan informasi yang diperlukan petugas sehingga perlunya ditampilkan pada informasi aplikasi. Pada perancangan basis data menunjukan field, type, size, index dan keterangan. Adapun rancangan basis data dari aplikasi penjadwalan praktikum ini sebagai berikut:

Tabel IV.39 tabel basis data siswa

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field | Type | Size | Index | Deskripsi |
| nama\_lengkap | Varchar | 20 | PK | Nama siswa |
| nama\_panggilan | Varchar | 50 |  | Nama panggilan siswa |
| nis | Varchar | 8 |  | Nomor Induk siswa |
| nisn | Varchar | 8 |  | Nomor Induk Siswa Nasional |
| nik | Varchar | 16 | FK | Nomor Induk Keluarga |
| nikk | Varchar | 16 |  |  |
| Jenis\_kelami | Varchar | 1 |  | Jenis Kelamin |
| Tempat\_lahir | Date |  |  | Tempat lahir |
| Tanggal\_lahir | Date |  |  | Tanggal lahir |
| Alamat | Varchar | 70 |  | Alamat |
| Kelurahan | Varchar | 30 |  | Kelurahan |
| Kecamatan | Varchar | 30 |  | Kecamatan |
| Kota\_kab | Varchar | 30 |  | Asal kota |
| Provinsi | Varchar | 30 |  | Asal provinsi |
| Kode\_pos | Varchar | 6 |  | Kode pos |
| agama | Varchar | 20 |  | Agama |
| Ayah | Varchar | 30 |  | Nama ayah |
| Pekerjaan\_ayah | Varchar | 30 |  | Pekerjaan ayah |
| ibu | Varchar | 30 |  | Nama ibu |
| Pekerjaan\_ibu | Varchar | 30 |  | Pekerjaan ibu |
| Penghasilan\_ibu | Varchar | 30 |  | Penghasilan ibu |
| Wali | Varchar | 30 |  | Nama wali |
| Pekerjaan\_wali | Varchar | 30 |  | Pekerjaan wali |
| Foto | Varchar | 255 |  | foto |
| Tanggal\_masuk | Varchar |  |  | Tanggal masuk |
| Aktif | tinyint | 1 |  | Keterangan |
| Telp\_siswa | Varchar | 14 |  | Nomor telepon siswa |
| Telp\_ayah | Varchar | 14 |  | Nomor telepon ayah |
| Telp\_ibu | Varchar | 14 |  | Nomor telepon ibu |

Tabel IV.40 tabel basis data guru

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field | Type | Size | Index | Deskripsi |
| Nama\_lengkap | Varchar | 30 | PK | Kode kelas |
| Jenis\_kelamin | Varchar | 30 |  | Nama kelas |
| Tempat\_lahir | Varchar | 30 |  | Semester |
| Tanggal\_lahir | date | 1 |  | Tanggal lahir guru |
| Nip | Varchar | 9 |  | Nip |
| Nik | Varchar | 16 |  | Nik |
| Nikk | Varchar | 16 |  | nikk |
| alamat | Varchar | 50 |  | Alamat guru |
| Gelar\_depan | Varchar | 6 |  | Gelar depan guru |
| Gelar\_belakang | Varchar | 6 |  | Gelar belakang guru |
| Pendidikan\_terakhir | Varchar | 6 |  | Pendidikan terakhir guru |
| email | Varchar | 50 |  | Email guru |
| telp | Varchar | 14 |  | Telp guru |

Tabel IV.41 tabel basis data tahun ajaran

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field | Type | Size | Index | Deskripsi |
| Tahun | Varchar | 20 | PK | Tahun ajaran |
| Semester | Varchar | 10 |  | Semester |
| Id\_kepsek | Int | 30 |  | Nama kepsek |
| Tanggal\_rapor | varchar | 30 |  | Tanggal pembagian rapor |

Tabel IV.42 tabel basis data profil sekolah

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field | Type | Size | Index | Deskripsi |
| Nama\_lsekolah | Varchar | 50 |  | Nama sekolah |
| Npsn | Varchar | 8 |  | Npsn sekolah |
| Bentuk\_sekolah | varchar | 10 |  | Bentuk sekolah |
| Alamat | Varchar | 50 |  | Alamat sekolah |
| DesaKelurahan | Varchar | 20 |  | Desa tempat sekolah |
| Kecamatan | Varchar | 20 |  | Kecamatan sekolah |
| KabupatenKota | Varchar | 20 |  | Kabupaten sekolah |
| Provinsi | Varchar | 20 |  | Provinsi sekolah |
| Rt | Varchar | 15 |  | Rt sekolah |
| Rw | Varchar | 15 |  | Rw sekolah |
| Dusun | Varchar | 20 |  | Dusun sekolah |
| KodePos | Varchar | 6 |  | Kode pos kecamatan |
| lintang | Varchar | 5 |  | Lintang |
| Bujur | Varchar | 6 |  | Bujur |
| Telp | Varchar | 14 |  | No telp sekolah |
| website | Varchar | 50 |  | Website sekolah |
| Email | Varchar | 50 |  | Email sekolah |
| logo | Varchar | 255 |  | Logo sekolah |

Tabel IV.43 tabel basis data rombongan belajar

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field | Type | Size | Index | Deskripsi |
| Nama | Int | 20 |  | Nama mata pelajaran |
| Kode | Int | 5 |  | Kode mata pelajaran |
| kelompok | int | 20 |  | Kelompok jurusan |

Tabel IV.44 tabel basis data login

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field | Type | Size | Index | Deskripsi |
| Ip\_adress | Varchar | 45 |  | Semester |
| Login | varchar | 100 |  | Nama kepsek |
| time | int | 11 |  | Tanggal pembagian rapor |

Tabel IV.45 tabel basis data kelas

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field | Type | Size | Index | Deskripsi |
| Nama | varchar | 20 |  | Nama |
| Kode\_kelas | varchar | 5 |  | Kode kelas |
| tingkat | varchar | 5 |  | Tingkat kelas |

# IV.2.3 Pelaporan Hasil Kerja Praktek

Proses pelaporan hasil kerja praktek dilakukan pada tahap akhir kerja praktek di Ruang Laboratorium Fakultas Teknologi Informasi Universitas Bale Bandung. Pelaporan hasil kerja praktek dilakukan dengan pembuatan laporan kerja praktek, yang nantinya akan pelaporan hasil kerja praktek akan dipresentasikan kepada penguji kampus.

# IV.3 Pencapaian Hasil

Adapun hasil yang telah dicapai ketika melaksanakan kerja praktek di Laboratorium Fakultas Teknologi Informasi Universitas Bale Bandung yaitu aplikasi penjadwalan praktikum.

Berikut tampilan aplikasi penjadwalan praktikum yang sudah dibuat:

1. Tampilan Database MySQL

Graphical user interface, text, application, email

Description automatically generated

Gambar IV.24 Database login

Graphical user interface, text, application, email

Description automatically generated

Gambar IV.25 Database tahun ajaran

Table

Description automatically generated

Gambar IV.26 Database Siswa

Graphical user interface, text, application, email

Description automatically generated

Gambar IV.27 Database mata pelajaran

A screenshot of a computer

Description automatically generated

Gambar IV.28 Database Guru

Graphical user interface, text, application, email

Description automatically generated

Gambar IV.29 Wali Kelas

A screenshot of a computer

Description automatically generated

Gambar IV.30 Rombongan belajar

1. Relasi Database

Table

Description automatically generated

Gambar IV.29 Relasi Database

1. Tampilan Aplikasi
2. Tampilan Halaman Dashboard

Graphical user interface, text, application

Description automatically generated

Gambar IV. 30 Tampilan Dashboard

1. Tampilan Profil Sekolah

Graphical user interface, application

Description automatically generated

Gambar IV. 31 Tampilan profil sekolah

1. Tampilan Login

Graphical user interface, application

Description automatically generated

Gambar IV. 32 Tampilan Login

1. Tampilan Tahun Ajaran

Graphical user interface, text, application, email

Description automatically generated

Gambar IV. 33 Tampilan Tahun Ajaran

1. Tampilan Data Siswa

Graphical user interface, application

Description automatically generated

Gambar IV.34 Tampilan Data Siswa

1. Tampilan Tambah Data Kelas

Graphical user interface, application

Description automatically generated

Gambar IV.35 Tampilan Tambah Data Kelas

1. Tampilan Edit Data Rombongan belajar

Graphical user interface, application, Word

Description automatically generated

Gambar IV.36 Tampilan Edit Data Rombongan Belajar

1. Tampilan Data Wali Kelas

Graphical user interface, text, application, Teams

Description automatically generated

Gambar IV.37 Tampilan Data Wali Kelas

1. Tampilan Data Guru

Graphical user interface, text, application, email

Description automatically generated

Gambar IV.38 Tampilan Data Guru

1. Tampilan Data Mata Pelajaran

Graphical user interface, application, email

Description automatically generated

Gambar IV.39 Tampilan Data Mata Pelajaran

1. Tampilan Data Pengajar

Graphical user interface, application

Description automatically generated

Gambar IV.40 Tampilan Data Mata Pengajar

1. Tampilan Data Pengguna

Graphical user interface, application

Description automatically generated

Gambar IV.41 Tampilan Data Penggu

# BAB V

# PENUTUP

# Kesimpulan dan saran mengenai pelaksanaan

Berdasarkan Penjelasan pada bab-bab sebelumnya maka dapat ditarik kesimpulan sebagai berikut:

# Kesimpulan Pelaksanaan Kerja Praktek

1. Guru dan Administrasi sekolah dapat menginput data siswa, mata pelajaran,guru,wali kelas, tahun ajaran,dan mata pelajaran dengan lebih mudah dalam satu website yang sama..
2. Sekolah dapat dengan mudah mendeskripsikan profilnya dalam website untuk perkenalan untuk calon siswa baru.
3. Siswa jadi lebih mudah mengakses informasi akademik
4. Kerja praktek dapat melatih mahasiswa untuk bekerja sama dalam suatu tim, baik antar peserta kerja praktek maupun dengan pihak lain di lingkungan pekerjaan
5. Mahasiswa kerja praktek mendapatkan pengalaman untuk melaksanakan pekerjaan pada bidang jurusannya.

# Saran Pelaksanaan KP

Adapun saran mengenai pelaksanaan kerja praktek antara lain:

1. Perlu ditumbuhkan kebiasaan belajar secara mandiri (*self-learning*) di kalangan mahasiswa, khususnya dalam mempelajari teknologi terutama yang diperlukan. Salah satu fasilitas yang tersedia yang mendukung proses pembelajaran secara mandiri ini adalah koneksi internet yang cukup cepat.
2. Perlu adanya dealine untuk bimbingan ,laporan, maupun sidang kerja praktek. Agar mahasiswa tidak kebingungan mengenai jadwal kerja praktek.
3. Perlu adanya pengarahan bagi mahasiswa kerja praktek agar mahasiswa tidak kebingungan saat menentukan perusahaan mana yang akan dituju untuk kerja praktek.
4. Jika memungkinkan, dalam pelaksanaan kerja praktek mahasiswa dapat dilibatkan dalam suatu proyek di mana mahasiswa dapat membantu dan bekerja sama dengan pegawai lain.

# Kesimpulan dan saran mengenai substansi

Berikut kesimpulan dan saran mengenai substansi yang diamati selama kerja praktek di SMA Pelita Bunga Bangsa :

# Kesimpulan Pembuatan Aplikasi Pengelola Data Akademik

Setelah melalui proses pembuatan aplikasi Pengelola Data Akademik, kesimpulan yang didapat sebagai berikut:

1. Hasil kegiatan kerja praktek ini adalah dengan dibuatnya sebuah aplikasi sistem informasi akademik SMA Pelita Bunga Bangsa Arjasari.
2. Dengan adanya aplikasi ini nantinya dapat memberikan informasi mengenai informasi akademik kepada siswa dengan lebih mudah dan akurat.

# Saran Mengenai Aplikasi Pengelola Data Akademik

Berdasarkan hasil pembangunan aplikasi Sistem Informasi Akademik saran yang diajukan adalah sebagai berikut:

1. Perlu adanya fitur tambahan, seperti cetak rapor sementara, cetak nilai ,absensi siswa,user untuk siswa.
2. Perlu adanya penambahan fitur pembayaran SPP agar siswa dapat dengan jelas mengetahui apa saja yang perlu dibayarkan dan kapan waktu harus membayarkan.

# DAFRTAR PUSTAKA

Agus Ramdhani Nugraha, G. P. (2017). SISTEM INFORMASI AKADEMIK SEKOLAH BERBASIS WEB DI SEKOLAH MENENGAH PERTAMA NEGERI 11 TASIKMALAYA. *JUMIKA(JURNAL MANAJEMEN INFORMATIKA)*.

Blog, e. (2017). Mengenal Lebih Dekat Apa itu Sistem Informasi akademik. 1.

Gustop. (2021). Kelebihan dan Kekurangan VS Code. *https://www.plimbi.com/article/176010/kelebihandan-kekurangan-vs-code*.

Idcloudhost, T. (2021). Mengetahui Fungsi, Kelebihan dan Kekurangan Framework CodeIgniter. *Retrieved from idcloudhost.com: https://idcloudhost.com/panduan/mengetahuifungsi-kelebihan-dan-kekurangan-codeigniter/* .

Irnawati, O. &. (2020). Penerapan model waterfall dalam analisis perancangan sistem informasi inventarisasi berbasis web. *Teknologi Dan Sistem Informasi*, 109-116.

Mulyodiputro, M. D. (2018). Perancangan Database Sistem Informasi Apotik Menggunakan MySQL pada Apotik Cemara. *SainsTech Innovation Journal*, 16-19.

Recky T. Djaelangkara, Rizal Sengkey, ST., MT, & Oktavian A. LAntang, ST,. MTI. (2015). Perancangan Sistem Informasi Akademik Sekolah Berbasis Web Studi Kasus Sekolah Menengah Atas Kristen 1 Tohomon. *e-jurnal Teknik Elektro dan Komputer* , 87.

Solahudin, M. (2021). Rancangan Bangun Sistem Informasi Akademik Sekolah (SIAS) Berbasis Web. *Journal of Computer and Information Technology*, 108.

**LAMPIRAN A**

**TOR (Term Of Reference)**

Sebelum melakukan kerja praktek penulis melakukan beberapa metode penelitian yaitu diantaranya adalah observasi, interview dan studi pustaka. Setelah mengamati dan mempelajari lokasi kerja praktek kemudian ditentukan serta disetujui oleh instansi tempat kerja praktek, kemudian penulis dijelaskan bahwa selama kerja praktek memiliki tugas yang harus dikerjakan dilokasi kerja praktek yaitu :

1. Membantu staff dan guru yang ada di SMA PBB ARJASARI

2. Membuat aplikasi pengelola data akademik sekolah

Bandung, Maret 2022

Disetujui Oleh :

|  |  |
| --- | --- |
| Peserta Kerja Praktek | Pembimbing Lapangan |
|  |  |
| Muhamad Aksal Dzilhamsyah | Khoerudin Sidiq, S.T,. |
| NIM : 301180052 | NIK : |

**LAMPIRAN B**

**LOG ACTIVITY**

Berikut adalah Log Activity selama kerja praktek :

|  |  |  |
| --- | --- | --- |
| Minggu tgl/th | Kegiatan | Hasil |
| Minggu ke 1, 3 november 2021 | Survey lingkungan sekolah | Selesai |
| Minggu ke 1, 6 November 2021 | Pembuatan Proposal KP | Selesai |
| Minggu ke 1, 9 Novemver 2021 | Persetujuan Proposal KP | Selesai |
| Minggu ke 2, 11 November 2021 | Pengumpulan data yang diperlukan | Selesai |
| Minggu ke 2, 16 November 2021 | Mempersiapkan software penunjang perancangan aplikasi | Selesai |
| Minggu ke 3, 18 November 2021 | Merancang Aplikasi | Selesai |
| Minggu ke 4, 24 Novemver 2021 | Pembuatan Mockup Aplikasi | Selesai |
| Minggu ke 5, 2 Oktober 2021 | Perancangan Database | Selesai |
| Minggu ke 6, 10 Oktober 2021 | Pembuatan Aplikasi dan Database | Selesai |
| Minggu ke 7, 16 Oktober 2021 | Pengujian Aplikasi | Selesai |
| Minggu ke 8, 24 Oktober 2021 | Penulisan Laporan | Selesai |