

JAWABAN UAS PPL

1. Untuk pengembangan **Sistem Informasi Pendaftaran dan Pengelolaan Data Mahasiswa UKM Badminton Berbasis Web**, bahasa pemrograman yang dipilih dapat terdiri dari beberapa pilihan sesuai dengan kebutuhan proyek tersebut. Berikut adalah penjelasan yang lebih detail:

Bahasa Pemrograman yang Digunakan:

PHP (Hypertext Preprocessor)

Alasan Pemilihan: PHP adalah bahasa pemrograman server-side yang sangat cocok untuk aplikasi berbasis web. PHP banyak digunakan untuk mengembangkan aplikasi web dinamis dan interaktif. Dengan PHP, kita dapat mengelola data dengan mudah melalui database, serta memudahkan integrasi dengan HTML dan CSS untuk tampilan yang menarik.

Keunggulan:

Banyak tersedia framework PHP (seperti Laravel, CodeIgniter) yang mempermudah pengembangan.

Kemampuan untuk menangani request HTTP dengan cepat.

Kompatibilitas yang luas dengan berbagai platform hosting dan database.

PHP juga relatif mudah dipelajari, banyak dokumentasi, serta komunitas besar.

JavaScript (untuk Frontend)

Alasan Pemilihan: JavaScript adalah bahasa pemrograman yang digunakan untuk pengembangan antarmuka pengguna interaktif (frontend) di aplikasi berbasis web. Dengan JavaScript, kita dapat membuat aplikasi yang responsif dan dinamis.

Keunggulan:

Kemampuan untuk mengupdate bagian-bagian halaman tanpa perlu me-refresh seluruh halaman (AJAX).

Memungkinkan penggunaan framework JavaScript seperti React atau Vue.js yang dapat meningkatkan kecepatan dan kemudahan pengembangan aplikasi.

Mendukung berbagai API untuk integrasi dengan berbagai layanan eksternal.

MySQL

Alasan Pemilihan: MySQL adalah sistem manajemen basis data relasional yang banyak digunakan dan kompatibel dengan PHP. MySQL dapat menangani data yang besar dengan baik dan cukup handal dalam menangani operasi CRUD (Create, Read, Update, Delete).

Keunggulan:

Open-source dan gratis digunakan.

Dukungan untuk operasi transaksi yang aman.

Mudah diintegrasikan dengan PHP.

Memiliki kecepatan yang baik dalam pemrosesan query.

Platform Pengembangan:

Web

Sistem ini akan berbasis web, yang berarti dapat diakses melalui browser menggunakan perangkat apapun seperti PC, laptop, maupun perangkat mobile. Platform ini dipilih agar sistem dapat diakses secara mudah oleh berbagai pihak (pengguna, admin, dll.) di mana saja selama terhubung ke internet.

Perangkat Lunak Pendukung:

IDE/Editor Kode:

Visual Studio Code: Editor yang populer dengan berbagai plugin yang mendukung pengembangan PHP, JavaScript, dan HTML/CSS.

PHPStorm: IDE yang khusus dirancang untuk pengembangan PHP dan sangat baik dalam menangani pengelolaan kode dan debugging.

Framework:

Laravel (untuk PHP): Mempermudah pengembangan aplikasi web dengan fitur-fitur seperti routing, middleware, dan ORM untuk interaksi dengan database.

Bootstrap: Framework CSS untuk mempercepat desain antarmuka yang responsif dan modern.

Version Control:

Git: Untuk mengelola versi kode dan kolaborasi antara tim pengembang.

GitHub/GitLab: Platform untuk menyimpan repositori dan kolaborasi tim.

Database Management:

phpMyAdmin: Untuk manajemen database MySQL secara visual.

MySQL Workbench: Alat lain yang sering digunakan untuk desain database dan query.

Kesimpulan:

Bahasa pemrograman PHP dipilih karena fleksibilitasnya dalam pengembangan aplikasi web yang dinamis, ditambah dengan JavaScript untuk front-end dan MySQL untuk penyimpanan data. Platform web memungkinkan akses yang mudah melalui berbagai perangkat, dan perangkat lunak pendukung seperti Laravel dan Git mendukung efisiensi pengembangan dan kolaborasi tim.

2. Permasalahan yang Ingin Diselesaikan dalam Proyek Perangkat Lunak

Judul: *Sistem Informasi Pendaftaran dan Pengelolaan Data Mahasiswa UKM Badminton Berbasis Web*

Permasalahan: Salah satu permasalahan utama yang ingin diselesaikan dalam proyek ini adalah **kurangnya sistem pendaftaran dan pengelolaan data yang efisien** untuk mahasiswa yang bergabung dalam UKM Badminton di sebuah universitas. Saat ini, proses pendaftaran anggota baru dan pengelolaan data mahasiswa UKM dilakukan secara manual, dengan menggunakan formulir fisik, pencatatan dalam buku, atau spreadsheet yang memerlukan waktu dan tenaga yang cukup besar. Selain itu, pencatatan dan pencarian data juga cenderung kurang terstruktur dan rawan kesalahan manusia.

Penyebab Utama Permasalahan:

Proses Manual: Pendaftaran mahasiswa dan pengelolaan data dilakukan secara manual menggunakan formulir kertas atau spreadsheet. Hal ini sangat rentan terhadap kesalahan input data, duplikasi data, atau hilangnya data penting yang tidak tercatat dengan baik.

Kurangnya Otomatisasi: Tanpa adanya sistem otomatisasi, proses pendaftaran, verifikasi, dan pembaruan data mahasiswa UKM menjadi lambat dan tidak efisien. Hal ini mengakibatkan keterlambatan dalam proses administrasi dan menyulitkan pengelola UKM dalam mengelola informasi terkait anggota.

Kesulitan dalam Akses Data: Jika data mahasiswa hanya disimpan dalam bentuk dokumen fisik atau file spreadsheet, akan sulit bagi pengelola untuk mengakses, memperbarui, dan menganalisis data secara cepat dan tepat. Proses pencarian data juga menjadi lebih lambat, terutama jika jumlah anggota UKM bertambah banyak.

Tidak Ada Sistem Pemantauan Keanggotaan dan Kegiatan: Tanpa adanya sistem yang terstruktur, pengelola UKM kesulitan dalam memantau partisipasi mahasiswa dalam kegiatan atau keanggotaan aktif di UKM, sehingga sulit untuk membuat keputusan terkait pengembangan UKM.

Dampak Permasalahan:

Ketidakefisienan: Proses manual yang memakan banyak waktu dan tenaga menyebabkan pengelola UKM kesulitan dalam menyelesaikan pekerjaan administratif secara efisien. Akibatnya, pengelola tidak bisa fokus pada pengembangan kegiatan UKM atau acara lainnya.

Kehilangan Data: Data yang tidak terkelola dengan baik berisiko hilang atau rusak, yang dapat merugikan mahasiswa dan pengelola UKM. Informasi penting seperti status keanggotaan, riwayat keikutsertaan dalam kegiatan, dan data pribadi mahasiswa bisa hilang atau tidak terupdate.

Kesalahan Pengelolaan Data: Sistem manual berisiko tinggi mengalami kesalahan pencatatan, seperti data ganda, data yang salah, atau ketidakakuratan lainnya. Hal ini dapat mengganggu transparansi dan akurasi informasi mengenai anggota UKM.

Ketidakkampuan Memantau Keanggotaan: Tanpa sistem yang baik untuk memantau keanggotaan, pengelola UKM kesulitan mengidentifikasi mahasiswa yang aktif dan yang sudah tidak aktif, sehingga pengelolaan sumber daya dan perencanaan kegiatan menjadi kurang optimal.

Usulan Solusi dengan Teknologi Perangkat Lunak

Untuk mengatasi permasalahan di atas, solusi yang diusulkan adalah mengembangkan **Sistem Informasi Pendaftaran dan Pengelolaan Data Mahasiswa UKM Badminton Berbasis Web**. Sistem ini akan menyediakan solusi digital yang terintegrasi untuk mempermudah pendaftaran, pengelolaan data, dan pemantauan kegiatan mahasiswa UKM Badminton.

Komponen Solusi:

Pendaftaran Online:

Mahasiswa dapat mendaftar ke UKM Badminton secara langsung melalui formulir pendaftaran yang tersedia di situs web.

Sistem akan memvalidasi data mahasiswa secara otomatis untuk memastikan bahwa data yang dimasukkan benar dan lengkap, menghindari kesalahan input.

Pengguna dapat mengakses status pendaftaran mereka dan informasi terkait keanggotaan secara real-time.

Pengelolaan Data Mahasiswa:

Semua data mahasiswa yang terdaftar akan disimpan secara terstruktur dalam database, dengan sistem yang memungkinkan pencarian, pembaruan, dan pengelolaan data secara efisien.

Setiap mahasiswa memiliki profil yang mencakup informasi pribadi, status keanggotaan, serta riwayat partisipasi dalam kegiatan UKM.

Sistem memungkinkan pengelola UKM untuk memperbarui data anggota, memverifikasi keanggotaan, dan menambahkan kegiatan atau acara yang diikuti oleh mahasiswa.

Dashboard untuk Pengelola:

Pengelola UKM akan memiliki akses ke dashboard yang memberikan gambaran umum tentang jumlah anggota, status keanggotaan, dan kegiatan yang telah dilaksanakan.

Sistem juga dapat memberikan laporan terkait statistik keanggotaan, absensi, serta analisis aktivitas mahasiswa dalam UKM.

Pemantauan Kegiatan:

Setiap kegiatan atau event yang diselenggarakan oleh UKM Badminton dapat tercatat dengan rapi. Mahasiswa bisa melakukan pendaftaran kegiatan atau event langsung melalui sistem.

Pengelola dapat melihat data partisipasi mahasiswa, memantau keaktifan mahasiswa, dan mengirimkan pengumuman atau informasi terkait kegiatan.

Keamanan dan Backup Data:

Data mahasiswa dan kegiatan disimpan dengan aman dalam server dan dilindungi menggunakan enkripsi.

Sistem juga memiliki fitur backup otomatis untuk menghindari kehilangan data.

Bagaimana Solusi Ini Memenuhi Kebutuhan Pengguna?

Efisiensi Administrasi: Dengan adanya sistem pendaftaran online dan pengelolaan data yang otomatis, pengelola UKM tidak perlu lagi melakukan pencatatan manual. Semua data akan tersimpan dengan baik dan mudah diakses, menghemat waktu dan tenaga.

Akses Data yang Mudah dan Cepat: Sistem berbasis web memungkinkan pengelola dan mahasiswa untuk mengakses informasi secara cepat dan real-time dari mana saja, asalkan terhubung dengan internet. Ini sangat memudahkan dalam mencari data dan memperbarui informasi.

Keakuratan Data: Karena semua data disimpan dalam sistem terstruktur dan terkelola dengan baik, risiko kesalahan data dapat diminimalisir. Proses verifikasi dan validasi otomatis akan memastikan bahwa data yang dimasukkan adalah akurat.

Peningkatan Keterlibatan Anggota: Mahasiswa dapat dengan mudah mendaftar dan berpartisipasi dalam kegiatan UKM, serta memantau kegiatan mereka. Dengan fitur yang memudahkan pemantauan keanggotaan, pengelola bisa lebih mudah mengetahui siapa yang aktif dan siapa yang tidak, sehingga dapat lebih fokus pada pengembangan UKM.

Laporan dan Analisis: Pengelola UKM dapat dengan mudah menghasilkan laporan mengenai keanggotaan dan kegiatan yang dilaksanakan. Ini akan membantu dalam perencanaan kegiatan di masa depan dan dalam pengambilan keputusan strategis.

Kesimpulan:

Dengan menggunakan **Sistem Informasi Pendaftaran dan Pengelolaan Data Mahasiswa UKM Badminton Berbasis Web**, permasalahan terkait pencatatan data manual yang rentan kesalahan dan memakan waktu dapat diatasi. Solusi ini memberikan kemudahan, efisiensi, dan keamanan dalam pengelolaan data, serta mempermudah pemantauan keanggotaan dan kegiatan. Teknologi perangkat lunak ini dapat memenuhi kebutuhan pengelola UKM dalam mengoptimalkan pengelolaan dan meningkatkan keterlibatan mahasiswa dalam kegiatan UKM.

3. Berikut adalah **rancangan teknis perangkat lunak** untuk **Sistem Informasi Pendaftaran dan Pengelolaan Data Mahasiswa UKM Badminton Berbasis Web**, yang mencakup skema hardware, software, database, dan actor dalam sistem.

a. Skema Hardware

Perangkat Keras yang Dibutuhkan untuk Menjalankan Sistem:

Server Web (Server Utama):

Fungsi: Menjalankan aplikasi web dan menyediakan akses ke data bagi pengguna.

Spesifikasi:

CPU: Intel Core i5 atau yang lebih tinggi.

RAM: 8GB atau lebih.

Storage: SSD 256GB atau lebih.

Koneksi Internet: Minimal 10 Mbps untuk memastikan sistem berjalan lancar.

Sistem Operasi: Linux (Ubuntu/Debian) atau Windows Server (tergantung pada preferensi hosting).

Database Server:

Fungsi: Menyimpan dan mengelola database aplikasi.

Spesifikasi:

CPU: Intel Xeon atau setara.

RAM: 8GB atau lebih.

Storage: SSD 512GB untuk performa database yang optimal.

Sistem Operasi: Linux (Ubuntu/Debian) atau Windows Server.

Client/Perangkat Pengguna:

Fungsi: Akses aplikasi berbasis web oleh mahasiswa dan pengelola UKM.

Spesifikasi:

CPU: Intel Core i3 atau lebih tinggi.

RAM: 4GB atau lebih.

Storage: 50GB (untuk sistem operasi dan aplikasi).

Browser: Chrome, Firefox, atau browser lain yang mendukung HTML5 dan JavaScript.

Perangkat: Laptop, PC, atau perangkat mobile (smartphone/tablet).

b. Skema Software

Spesifikasi Sistem Operasi, Framework, Library, dan Bahasa Pemrograman:

Sistem Operasi:

Server: Linux (Ubuntu/Debian) atau Windows Server (tergantung hosting atau preferensi pengembangan).

Client: Windows, macOS, atau Linux dengan browser web yang mendukung HTML5.

Bahasa Pemrograman:

Backend:

PHP: Digunakan untuk logika server-side dan komunikasi dengan database.

Frontend:

JavaScript: Untuk menangani antarmuka pengguna dinamis dan interaktif.

HTML/CSS: Untuk struktur dan desain tampilan halaman.

Database:

MySQL: Digunakan untuk penyimpanan data terstruktur mengenai mahasiswa, kegiatan, dan keanggotaan.

Framework:

Backend:

Laravel (PHP): Framework PHP untuk pengembangan yang lebih cepat dan aman dengan fitur seperti routing, ORM (Eloquent), middleware, dll.

Frontend:

React.js atau **Vue.js:** Framework JavaScript untuk membuat antarmuka yang lebih dinamis dan responsif.

Bootstrap: Framework CSS untuk membangun tampilan yang responsif dan modern.

Database:

phpMyAdmin: Untuk manajemen dan mengelola database MySQL secara visual.

Library:

jQuery: Library JavaScript untuk mempermudah manipulasi DOM dan AJAX.

Axios: Untuk menangani permintaan HTTP secara asynchronous.

c. Skema Database

Entity Relationship Diagram (ERD):

ERD berikut menggambarkan struktur data yang digunakan dalam sistem pendaftaran dan pengelolaan data mahasiswa UKM Badminton. Tabel-tabel utama meliputi **Mahasiswa**, **Kegiatan**, **Pendaftaran**, dan **Keanggotaan**.

ERD Diagram:

lua

Mahasiswa		Kegiatan		Pendaftaran	
id_mahasiswa	<----->	id_kegiatan	<----->	id_pendaftaran	
nama		nama_kegiatan		id_mahasiswa	
nim		tanggal		id_kegiatan	
email		lokasi		status	
telepon					

Penjelasan:

- **Tabel Mahasiswa:** Menyimpan data mahasiswa seperti nama, NIM, email, dan nomor telepon.
- **Tabel Kegiatan:** Menyimpan informasi tentang kegiatan yang diselenggarakan oleh UKM Badminton, termasuk nama, tanggal, dan lokasi kegiatan.
- **Tabel Pendaftaran:** Menghubungkan mahasiswa dengan kegiatan yang mereka ikuti dan menyimpan status pendaftaran mereka (terdaftar, tidak terdaftar, dll).

Penjelasan:

- **Tabel Mahasiswa:** Menyimpan data mahasiswa seperti nama, NIM, email, dan nomor telepon.
- **Tabel Kegiatan:** Menyimpan informasi tentang kegiatan yang diselenggarakan oleh UKM Badminton, termasuk nama, tanggal, dan lokasi kegiatan.
- **Tabel Pendaftaran:** Menghubungkan mahasiswa dengan kegiatan yang mereka ikuti dan menyimpan status pendaftaran mereka (terdaftar, tidak terdaftar, dll).

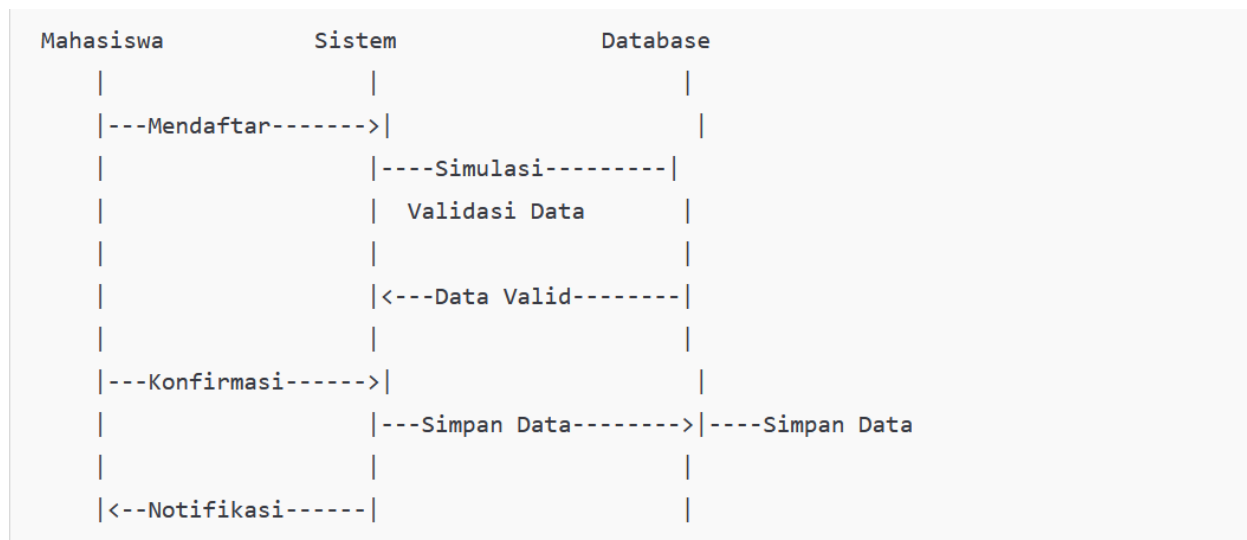
Mahasiswa		Pengelola UKM	
- Mendaftar untuk kegiatan		- Kelola data mahasiswa	
- Melihat status keanggotaan		- Tambah/ubah kegiatan UKI	
- Mengupdate profil		- Lihat daftar pendafatara	
- Lihat pengumuman		- Verifikasi pendaftaran	

Penjelasan:

- **Mahasiswa** dapat melakukan pendaftaran untuk kegiatan, melihat status keanggotaan, dan memperbarui profil mereka.
- **Pengelola UKM** dapat mengelola data mahasiswa, menambah atau mengubah kegiatan UKM, melihat daftar pendaftaran, dan memverifikasi pendaftaran mahasiswa.

Sequence Diagram:

Berikut adalah **Sequence Diagram** untuk kasus penggunaan "Pendaftaran Mahasiswa untuk Kegiatan".



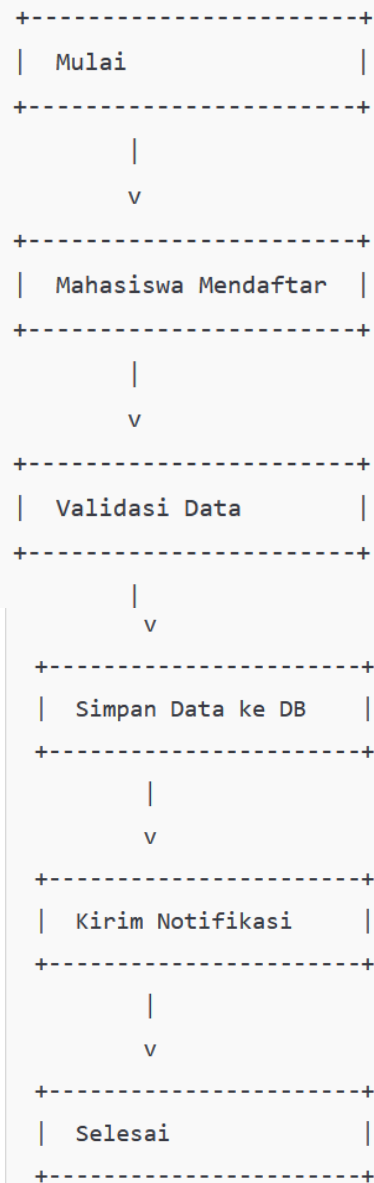
Penjelasan:

1. Mahasiswa mengisi formulir pendaftaran.
2. Sistem memvalidasi data dan mengonfirmasi keanggotaan mahasiswa.
3. Data pendaftaran mahasiswa disimpan ke database.

Mahasiswa menerima notifikasi pendaftaran yang berhasil.

Activity Diagram:

Berikut adalah **Activity Diagram** untuk proses pendaftaran mahasiswa:



4. Proses Implementasi Sistem: Sistem Informasi Pendaftaran dan Pengelolaan Data Mahasiswa UKM Badminton Berbasis Web

Berikut adalah langkah-langkah implementasi sistem yang mencakup persiapan lingkungan kerja, pengembangan perangkat lunak, integrasi antara hardware, software, database, dan antarmuka pengguna, hingga penerapan komponen-komponen perangkat lunak dalam sistem informasi pendaftaran dan pengelolaan data mahasiswa UKM Badminton berbasis web.

1. Persiapan Lingkungan Kerja

Langkah-langkah yang dilakukan:

1. **Pemilihan dan Pengaturan Hardware:**
 - **Server:** Menyediakan server dengan spesifikasi yang cukup untuk menjalankan aplikasi dan database, seperti CPU Intel Core i5, RAM 8GB, dan SSD 256GB.
 - **Koneksi Internet:** Memastikan koneksi internet yang stabil dengan kecepatan minimal 10 Mbps untuk memastikan aplikasi berjalan lancar bagi pengguna.
2. **Instalasi Sistem Operasi:**
 - **Server Web:** Memilih sistem operasi berbasis Linux (Ubuntu/Debian) untuk server web dan database, yang lebih efisien dan memiliki dukungan yang baik untuk pengembangan aplikasi berbasis PHP dan MySQL.
 - **Client:** Memastikan perangkat pengguna (laptop/PC/smartphone) memiliki sistem operasi yang kompatibel dengan browser modern (seperti Windows, macOS, atau Linux).
3. **Instalasi Perangkat Lunak dan Alat Pengembangan:**
 - **Web Server:** Menginstal **Apache** atau **Nginx** untuk menangani permintaan HTTP.
 - **PHP:** Menginstal PHP versi terbaru untuk menjalankan backend.
 - **MySQL:** Menginstal MySQL sebagai database server untuk menyimpan data mahasiswa dan kegiatan UKM.
 - **phpMyAdmin:** Menginstal phpMyAdmin untuk memudahkan pengelolaan database.
 - **IDE (Integrated Development Environment):** Menggunakan editor atau IDE seperti **Visual Studio Code** atau **PHPStorm** untuk menulis kode.
 - **Framework:** Menginstal **Laravel** (untuk backend) dan **Bootstrap** untuk frontend.
4. **Pengaturan Tools Pengembangan:**
 - **Version Control:** Menggunakan **Git** untuk kontrol versi dan **GitHub** untuk repositori kode.
 - **Dependency Management:** Menggunakan **Composer** untuk mengelola dependensi PHP dan **npm** untuk mengelola dependensi JavaScript.

2. Pengembangan Perangkat Lunak

Langkah-langkah Pengembangan:

1. **Desain Database dan Struktur Tabel:**
 - Membuat struktur database sesuai dengan **Entity Relationship Diagram (ERD)** yang telah disiapkan, yang mencakup tabel **Mahasiswa**, **Kegiatan**, **Pendaftaran**, dan **Keanggotaan**.
 - Menggunakan **phpMyAdmin** untuk memudahkan pembuatan dan pengelolaan database, serta memastikan relasi antar tabel sesuai dengan desain.

Contoh Struktur Tabel (MySQL):

```

class MahasiswaController extends Controller {
  public function store(Request $request) {
    $mahasiswa = new Mahasiswa;
    $mahasiswa->nama = $request->nama;
    $mahasiswa->nim = $request->nim;
    $mahasiswa->email = $request->email;
    $mahasiswa->telepon = $request->telepon;
    $mahasiswa->save();

    return response()->json($mahasiswa, 201);
  }
}

```

Pengembangan Frontend:

- Menggunakan **HTML**, **CSS**, dan **JavaScript** untuk antarmuka pengguna (UI).
- Untuk antarmuka dinamis dan interaktif, menggunakan **React.js** atau **Vue.js** untuk membangun aplikasi web berbasis SPA (Single Page Application).
- Menggunakan **Bootstrap** untuk mendesain tampilan responsif yang sesuai dengan berbagai perangkat.

Contoh Komponen React:

```

import React, { useState } from 'react';
import axios from 'axios';

const Pendaftaran = () => {
  const [nama, setNama] = useState('');
  const [nim, setNim] = useState('');

  const handleSubmit = async (event) => {
    event.preventDefault();
    const response = await axios.post('api/mahasiswa', { nama, nim });
  }
}

```

Pengujian dan Debugging:

- Melakukan pengujian unit dan pengujian fungsional untuk memastikan bahwa setiap komponen aplikasi bekerja dengan benar.
- Menggunakan alat seperti **Postman** untuk menguji API backend dan **Jest** atau **Mocha** untuk pengujian frontend.

Integrasi Hardware, Software, dan Database

Langkah-langkah Integrasi:

1. Integrasi Backend dan Database:

- Menghubungkan backend (Laravel) dengan database MySQL melalui konfigurasi di file `.env` di Laravel.
- Menggunakan **Eloquent ORM** untuk mempermudah operasi CRUD (Create, Read, Update, Delete) pada database.

2. Integrasi Frontend dengan Backend:

- Menghubungkan frontend (React.js) dengan backend melalui API menggunakan **Axios** untuk menangani permintaan HTTP asinkron (GET, POST).
- Menampilkan data yang diterima dari backend dalam bentuk yang sesuai di frontend (misalnya, daftar mahasiswa, daftar kegiatan, dll).

3. Testing Integrasi:

- Melakukan pengujian end-to-end untuk memastikan bahwa alur pendaftaran dan pengelolaan data berjalan lancar, mulai dari pengisian formulir pendaftaran oleh mahasiswa hingga penyimpanan data di database dan tampilan data di antarmuka pengguna.

. Penerapan dan Deployment

Langkah-langkah Deployment:

1. Menyiapkan Server Produksi:

- Mengonfigurasi server produksi menggunakan **Apache** atau **Nginx** untuk menyajikan aplikasi web.
- Mengonfigurasi **SSL/TLS** untuk keamanan komunikasi dengan HTTPS.

2. Deployment Aplikasi ke Server:

- Menggunakan **FTP** atau **Git** untuk mentransfer kode aplikasi ke server.
- Menyusun file `.env` untuk pengaturan database dan environment lainnya di server produksi.

3. Mengonfigurasi dan Mengoptimalkan Database:

- Menyesuaikan konfigurasi database di server agar aplikasi dapat berjalan dengan performa yang optimal (indeksasi tabel, optimasi query).

4. Pengujian di Server Produksi:

- Melakukan pengujian akhir setelah aplikasi dipasang di server produksi untuk memastikan semua fitur berjalan dengan baik dan aplikasi stabil.

Monitoring dan Pemeliharaan

Langkah-langkah Pemeliharaan dan Monitoring:

1. Monitoring Kinerja Aplikasi:

- Menggunakan alat seperti **New Relic** atau **Google Analytics** untuk memonitor kinerja aplikasi dan lalu lintas pengguna.

2. Pemeliharaan Berkala:

- Melakukan pembaruan sistem dan memperbaiki bug jika ditemukan masalah atau kesalahan selama operasional aplikasi.

Kesimpulan:

Proses implementasi sistem ini melibatkan beberapa tahap yang dimulai dengan persiapan lingkungan kerja, pengembangan perangkat lunak menggunakan **Laravel**, **React.js**, dan **MySQL**, serta integrasi antara komponen hardware, software, dan database. Setelah itu, sistem diuji dan diterapkan di server produksi untuk memastikan semuanya berfungsi dengan baik. Langkah-langkah pemeliharaan juga disiapkan untuk memastikan sistem berjalan stabil dan optimal setelah penerapan.

5. Metode Pengujian Perangkat Lunak untuk Sistem Informasi Pendaftaran dan Pengelolaan Data Mahasiswa UKM Badminton Berbasis Web

Pengujian perangkat lunak adalah tahap penting dalam pengembangan sistem untuk memastikan bahwa sistem berjalan dengan baik dan memenuhi semua kebutuhan fungsional serta non-fungsional. Untuk **Sistem Informasi Pendaftaran dan Pengelolaan Data Mahasiswa UKM Badminton Berbasis Web**, kami menggunakan **metode pengujian black-box** dan **white-box**. Berikut adalah penjelasan mengenai jenis pengujian yang dilakukan, metode yang digunakan, hasil pengujian, dan kesimpulan dari pengujian tersebut.

1. Metode Pengujian yang Digunakan Pengujian Black-box (Fungsional)

Metode **black-box** testing berfokus pada pengujian fungsionalitas sistem tanpa mengetahui implementasi internalnya. Pengujian ini dilakukan berdasarkan spesifikasi yang telah ditentukan untuk memastikan bahwa semua fitur berjalan sebagaimana mestinya.

Pengujian White-box (Struktural)

Metode **white-box** testing digunakan untuk menguji bagian internal dari aplikasi, seperti kode dan algoritma yang diterapkan. Pengujian ini memastikan bahwa logika dan struktur kode berjalan dengan baik serta dapat menangani berbagai kondisi input.

2. Jenis Pengujian yang Diterapkan

Berikut adalah jenis pengujian yang diterapkan untuk memastikan sistem berjalan dengan baik:

a. Unit Testing

Unit testing bertujuan untuk menguji unit terkecil dari perangkat lunak, yaitu fungsi atau metode dalam aplikasi.

Tujuan: Menguji setiap fungsi dan metode pada backend untuk memastikan mereka bekerja dengan baik.

Alat yang Digunakan: **PHPUnit** (untuk backend Laravel) dan **Jest** atau **Mocha** (untuk frontend React.js).

Contoh Pengujian:

Menguji fungsi pendaftaran mahasiswa untuk memastikan data yang dimasukkan disimpan dengan benar dalam database.

Menguji logika perhitungan status keanggotaan mahasiswa di UKM.

b. Integration Testing

Integration testing digunakan untuk menguji interaksi antar berbagai komponen atau modul dalam sistem.

Tujuan: Menguji komunikasi dan integrasi antara backend (Laravel), frontend (React.js), dan database (MySQL).

Alat yang Digunakan: **Postman** untuk menguji API dan **PHPUnit** untuk pengujian integrasi backend.

Contoh Pengujian:

Menguji API untuk pendaftaran mahasiswa yang menghubungkan frontend dan backend.

Menguji data yang disimpan di database dan diproses oleh aplikasi.

c. System Testing

6. System testing menguji keseluruhan sistem untuk memastikan semua komponen bekerja bersama-sama dengan baik.

Tujuan: Memastikan sistem berfungsi sesuai dengan kebutuhan pengguna dan spesifikasi teknis.

Alat yang Digunakan: Alat pengujian manual dan otomatis (seperti **Selenium** untuk pengujian UI).

Contoh Pengujian:

Menguji seluruh alur pendaftaran dari awal hingga akhir, mulai dari pengisian formulir hingga data tersimpan dan notifikasi terkirim.

Menguji pengelolaan data mahasiswa dan kegiatan, serta apakah pengelola dapat melakukan perubahan data dan melihat pendaftaran.

d. User Acceptance Testing (UAT)

UAT dilakukan untuk memastikan bahwa sistem sesuai dengan kebutuhan dan harapan pengguna akhir.

Tujuan: Memastikan sistem sesuai dengan harapan pengguna dan dapat diterima oleh pengguna.

Alat yang Digunakan: Pengujian manual oleh pengguna akhir (mahasiswa dan pengelola UKM).

Contoh Pengujian:

Pengguna (mahasiswa) mengisi dan mengirimkan formulir pendaftaran untuk memastikan bahwa UI/UX sistem mudah digunakan.

Pengelola UKM mengelola data mahasiswa dan kegiatan untuk memastikan mereka dapat melakukan tugas mereka tanpa kesulitan.

3. Hasil Pengujian

Berikut adalah hasil pengujian yang diperoleh selama proses pengujian:

a. Hasil Unit Testing

Fungsi Pendaftaran Mahasiswa: Semua fungsi terkait pendaftaran mahasiswa (misalnya, menyimpan data mahasiswa, memeriksa keunikan NIM, dan mengirim notifikasi) berfungsi dengan baik.

Status Keanggotaan: Fungsi untuk menghitung status keanggotaan mahasiswa di UKM berjalan sesuai dengan yang diharapkan, berdasarkan status pendaftaran dan keanggotaan yang ditentukan.

b. Hasil Integration Testing

Integrasi Frontend dan Backend: Komunikasi antara frontend React.js dan backend Laravel berjalan lancar. API berhasil mengirim dan menerima data dengan benar.

Database: Data mahasiswa dan kegiatan berhasil disimpan dengan benar dalam database MySQL, dan perubahan data dapat tercermin dengan cepat di antarmuka pengguna.

c. Hasil System Testing

Alur Pendaftaran: Sistem pendaftaran berjalan tanpa masalah. Pengguna dapat mengisi formulir, mengirimkan data, dan mendapatkan konfirmasi bahwa pendaftaran berhasil.

Pengelolaan Data Mahasiswa dan Kegiatan: Pengelola UKM dapat dengan mudah menambah, mengedit, dan menghapus data mahasiswa dan kegiatan. Sistem juga dapat menampilkan laporan dan status pendaftaran dengan akurat.

d. Hasil User Acceptance Testing (UAT)

Pengalaman Pengguna: Pengguna (mahasiswa dan pengelola UKM) merasa nyaman dengan antarmuka yang disediakan. Pengguna tidak mengalami kesulitan dalam melakukan pendaftaran atau mengelola data.

Kepuasan Pengguna: Mayoritas pengguna memberikan umpan balik positif mengenai kemudahan penggunaan dan kecepatan sistem.

4. Kesimpulan dari Pengujian

Dari hasil pengujian yang dilakukan, dapat disimpulkan bahwa **Sistem Informasi Pendaftaran dan Pengelolaan Data Mahasiswa UKM Badminton Berbasis Web** berfungsi dengan baik dan sesuai dengan spesifikasi yang diharapkan. Berikut adalah kesimpulan utama:

Fungsionalitas Sistem: Semua fitur utama, seperti pendaftaran mahasiswa, pengelolaan kegiatan, dan pengelolaan data mahasiswa, berjalan dengan baik tanpa kesalahan yang signifikan.

Kinerja Sistem: Sistem dapat menangani sejumlah pengguna dengan baik dan mengelola data mahasiswa serta kegiatan dengan efisien.

Pengalaman Pengguna: Antarmuka pengguna mudah dipahami dan digunakan, baik oleh mahasiswa maupun pengelola UKM.

Stabilitas dan Keamanan: Sistem stabil dan aman selama pengujian, dan tidak ditemukan kerentanannya dalam integrasi database dan backend.

Namun, ada beberapa saran untuk peningkatan sistem:

Peningkatan UI/UX: Beberapa pengguna memberikan umpan balik bahwa tampilan pada perangkat mobile bisa lebih responsif dan mudah digunakan.

Pengujian Beban (Load Testing): Pengujian lebih lanjut perlu dilakukan untuk memastikan sistem dapat menangani jumlah pengguna yang lebih besar dalam kondisi nyata.

Secara keseluruhan, sistem telah siap untuk diterapkan dalam lingkungan produksi dan memenuhi kebutuhan fungsional yang ditetapkan.

