

PENERAPAN *SINGLE PAGE APPLICATION* PADA PENGEMBANGAN APLIKASI *E-LEARNING* NUSABOT

Salwa Nur Atifah¹, Aji Primajaya², Dadang Yusup³

Universitas Singaperbangsa Karawang¹²³

Jl. HS. Ronggowaluyo Telukjambe Timur Karawang 41361

E-mail : salwanurs09@gmail.com¹, aji.primajaya@staff.unsika.ac.id², dadang.dyf@staff.unsika.ac.id³

ABSTRAK

Nusabot merupakan aplikasi *e-learning* yang menyediakan berbagai program kelas online berbayar dengan fokus di bidang Internet of Things (IoT). Instruktur dapat membuat kelas yang diinginkan dengan mengikuti proses yang ada termasuk pemeriksaan yang dilakukan oleh admin. Saat ini interaksi antara admin dan instruktur masih manual dimulai dari penambahan kelas dan kurikulum, pengajuan kelas, dan persetujuan atau penolakan kelas. Hanya admin yang diberikan hak akses untuk pengelolaan kelas terutama penambahan dan publikasi kelas. Proses penambahan kelas dan kurikulum juga masih belum efisien dikarenakan form penambahan kurikulum berada di halaman yang berbeda-beda sehingga *browser* akan memuat ulang halaman setiap terjadi penambahan data. Untuk itu, sistem dikembangkan dengan menerapkan *Single Page Application* terutama pada pembuatan akses instruktur dalam penyusunan kurikulum kelas. Metode yang digunakan dalam pengembangan yaitu model *Prototype* yang memfokuskan pengguna dalam pengembangan. Sistem ini dikembangkan menggunakan bahasa pemrograman PHP dengan *framework* Laravel sebagai *back-end* dan API services, React JS sebagai *front-end*, Inertia JS sebagai penghubung antara Laravel dengan React JS, dan MySQL sebagai *database*. Hasil dari pengembangan sistem ini membuat interaksi antara admin dan instruktur menjadi lebih efektif dan efisien.

Kata kunci : *Single Page Application*, Laravel, React JS

ABSTRACTS

Nusabot is an e-learning application that provides various paid online class programs with a focus on the Internet of Things (IoT). Instructors can hold the desired class by following the existing flow including checks carried out by the admin. Currently, the interaction between the admin and the instructor is still manual, starting from class and curriculum preparation, class submissions, and class approval or rejection. Only admins are given access rights for class management, especially adding and publishing classes and their curriculum. The process of adding classes and curriculum is also still inefficient because the addition forms are on different pages so the browser will reload the page every time data is added. For this reason, the system was developed by implementing a Single Page Application, especially in making access to instructors in preparing class curriculum. The method used in development is the Prototype model which focuses on users in development. This system uses the PHP programming language with the Laravel framework as the back-end and API services, React JS as the front-end, Inertia JS as the liaison between Laravel and React JS, and MySQL as the database. The results of this system development make classroom management more effective and efficient.

Keywords: *Single Page Application*, Laravel, React JS

1. PENDAHULUAN

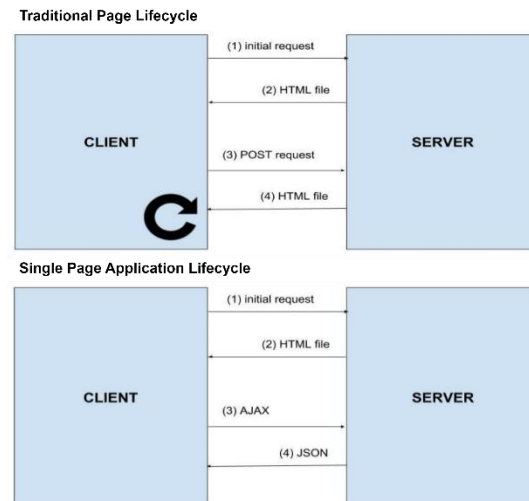
Teknologi yang terus berkembang memiliki pengaruh yang besar terlebih di era sekarang penggunaan internet sangat mudah digunakan. Hal ini membuat banyak permintaan kebutuhan salah satunya *e-learning*. *e-learning* merupakan metode

pembelajaran berbasis digital. Saat ini *e-learning* banyak digunakan karena proses belajar mengajar lebih mudah, cepat dan efisien. Tentunya *e-learning* memuat banyak data yang akan ditampilkan sehingga mempengaruhi kecepatan saat *website* diakses. Hal itu juga perlu diperhatikan agar pengguna mendapatkan pengalaman yang lebih baik.

Nusabot merupakan aplikasi *e-learning* yang menyediakan berbagai program kelas online berbayar dengan fokus di bidang *Internet of Things* (IoT). Didampingi oleh instruktur, peserta dapat menerapkan inovasi teknologi IoT. Namun, aplikasi Nusabot belum memiliki hak akses untuk instruktur sehingga dalam proses pengajuan kelasnya instruktur harus menyampaikan kepada admin secara manual untuk diperiksa, dan selanjutnya penambahan dan penerbitan kelas dilakukan oleh admin. Admin memiliki akses untuk mengelola kelas beserta kurikulumnya, yang juga terdapat kendala yaitu penambahan kurikulum kelas seperti topik, materi, tugas, kuis dan lain-lain dilakukan di halaman yang berbeda dan dikirim ke *database* secara satu per satu yang dimana *browser* akan memuat ulang (*reload*) halaman setiap terjadi perubahan data. Hal ini dikarenakan aplikasi Nusabot menggunakan konsep MPA (*Multi Page Application*) yang ada pada *framework* Laravel. *Framework* Laravel memberikan standar kebutuhan untuk membangun *back-end* dan RESTful infrastruktur [1]. Laravel dibangun dengan Bahasa pemrograman PHP yang memberikan kemudahan dan fleksibilitas desain. Laravel bisa digunakan untuk sisi *back-end* dan *API services*.

Dalam pengembangan web terdapat dua *design pattern* utama yaitu MPA dan SPA (*Single Page Application*) [2]. MPA merupakan konsep tradisional yang sudah lama digunakan. Selama perpindahan halaman, *browser* memuat ulang konten halaman sepenuhnya [2]. Sedangkan *Single Page Application* (SPA) adalah teknologi yang bekerja di *browser* tanpa perlu *reload* halaman ketika berpindah halaman. Teknologi ini dapat meningkatkan kecepatan *website* karena memproses seluruh data (HTML, CSS, JavaScript) satu kali saat *website* diakses dan selanjutnya data diproses hanya sesuai dengan *request* [3]. Pengguna tidak akan berpindah halaman dengan melakukan *request* kepada *server* setiap terjadi interaksi pada aplikasi. SPA hanya memuat satu halaman dari *server* dan mekanisme *routing* dilakukan oleh *client* [4]. Hal ini mempercepat waktu yang dibutuhkan untuk menerima informasi tentang data yang diakses dan bahkan jumlah waktu yang diperlukan untuk mengakses fitur tertentu [5]. Sebagai *library* JavaScript, React JS menangani komponen *view* berbasis *Single Page Application* menggunakan *data-driven* UI yang efisien dan dinamis [6].

Saat ini Laravel dapat diintegrasikan dengan *front-end* melalui Inertia JS. Inertia JS merupakan monolith modern sebagai penghubung antara *front-end* dan *back-end*. Inertia dapat membuat aplikasi satu halaman yang sepenuhnya dimuat di sisi klien tanpa kerumitan yang disertakan dengan SPA.



Gambar 1 Traditional & Single Page Lifecycle

Berdasarkan permasalahan tersebut, maka penelitian ini akan menerapkan *Single Page Application* pada pengembangan aplikasi *e-learning* Nusabot menggunakan React JS sebagai *front-end* dan Laravel sebagai *back-end* dan *API services*. Sistem dikembangkan dengan model *Prototype*. Diterapkannya SPA pada sistem diharapkan dapat membuat proses pengelolaan dan pengajuan kelas lebih efisien. Instruktur akan dapat menyusun dan mengajukan kelas beserta kurikulumnya melalui aplikasi dan admin dapat menyetujui atau menolak kelas yang diajukan. Sistem akan mengirim data kelas dan kurikulum yang dimasukkan oleh instruktur secara bersamaan ke *database* untuk disimpan tanpa perpindahan dan pemuatan ulang halaman. Fitur lainnya pada akses instruktur yaitu instruktur dapat melihat pendapatan, siswa yang mengikuti kelas, dan ulasan kelas.

2. METODE PENELITIAN

A. Teknik Pengumpulan Data

Hal pertama yang dilakukan dalam analisis sistem adalah mengumpulkan data, yaitu sebagai berikut [7]:

a. Observasi

Observasi dilakukan dengan cara pengamatan langsung bagaimana sistem yang sedang berjalan. Dalam hal ini, penulis melakukan penelitian langsung tentang proses pengelolaan data kelas dan kuis yang dilakukan oleh admin.

b. Wawancara

Wawancara dilakukan kepada pemilik sistem yaitu Direktur dan CTO (*Chief Technology Officer*)

Nusabot mengenai permasalahan pada sistem yang sedang berjalan.

c. Studi Literatur

Studi literatur dilakukan dengan mengumpulkan referensi dari berbagai sumber seperti jurnal, *thesis*, buku dan sumber lainnya.

B. Metode Pengembangan Perangkat Lunak

Prototype merupakan model di dalam SDLC yang berfokus terhadap kebutuhan pengguna dimana pengguna yang langsung mengevaluasi kebutuhan dalam pengembangan perangkat lunak. *Prototype* sangat dinamis, efisien, dan responsif terhadap kebutuhan pengguna [8]. Tahapan dalam *Prototype* yaitu, *communication*, *quick plan*, *modelling quick design*, *construction of Prototype*, *deployment delivery & feedback* [9].

Adapun tahapan pengembangan model *Prototype* ini sebagai berikut:

a. Komunikasi dengan Pengguna (Communication)

Tahap pertama dimulai dari *communication* yaitu proses komunikasi antara pengembang dan pengguna untuk menentukan tujuan dari perangkat lunak, mengidentifikasi persyaratan, dan gambaran garis besar mengenai kebutuhan pengguna pada sistem.

b. Perencanaan (Quick Plan)

Tahap ini merupakan perencanaan dari hasil pengumpulan data kebutuhan.

c. Pemodelan Desain (Modelling Quick Design)

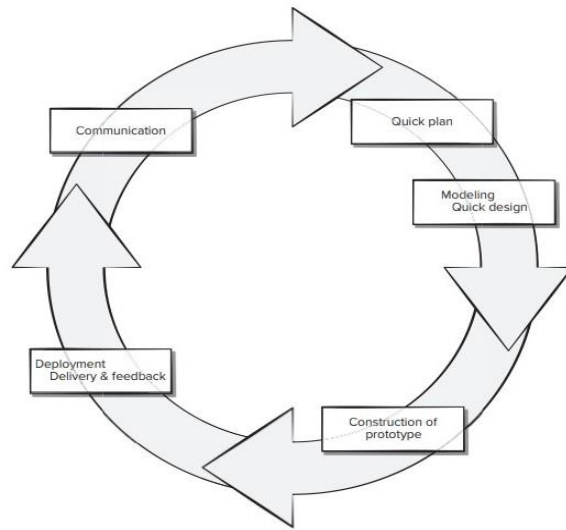
Tahap ini merupakan perancangan desain sistem sebelum pembuatan kode program yang berfokus pada struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengkodean.

d. Konstruksi Prototipe (Construction of Prototype)

Tahap ini membuat kode program sesuai rancangan kebutuhan pengguna. Sistem dibuat dengan Bahasa pemrograman PHP dengan *framework* Laravel dan React JS sebagai *front-end*.

e. Penyampaian Hasil Implementasi & Evaluasi (Deployment Delivery & Feedback)

Sistem yang sudah jadi langsung diberikan kepada pengguna untuk dilakukan evaluasi atau pengujian. Pengujian terbagi berdasarkan sisi klien dan sisi *server* menggunakan *Black Box Testing*. Jika pengguna belum puas atau masih ada kebutuhan maka dilakukan perbaikan dimulai dari tahap pemodelan desain.

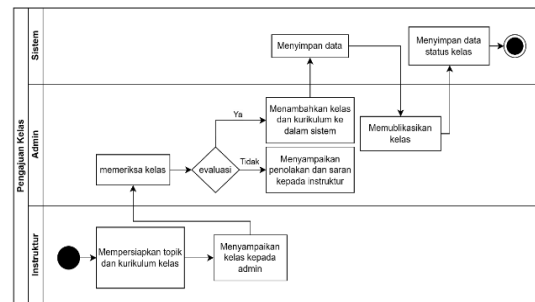


Gambar 2 Model Prototype

3. HASIL DAN PEMBAHASAN

3.1 Alur Sistem yang Sedang Berjalan

Hasil pengamatan sistem yang sedang berjalan sesuai dengan ruang lingkup penelitian ini yaitu proses pengajuan dan penerbitan kelas oleh admin dan instruktur digambarkan menggunakan *swimlane* diagram sebagai berikut :



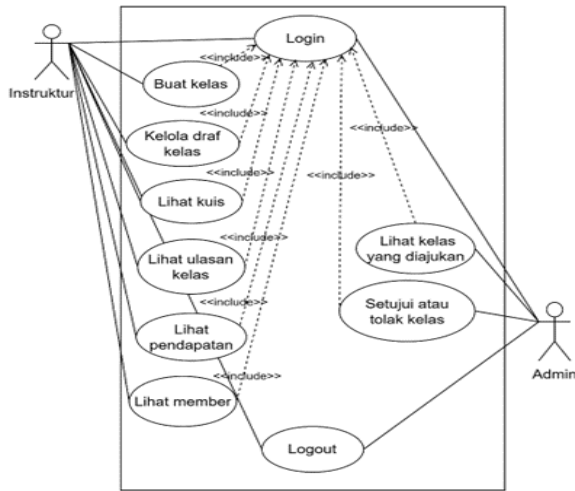
Gambar 3 Proses Pengajuan Kelas

Proses ini dimulai dari instruktur yang ingin mengadakan kelas, instruktur harus menentukan judul kelas serta menyusun kurikulum kelas berupa topik dan video materi yang selanjutnya diserahkan kepada admin secara manual untuk diperiksa. Jika program kelas disetujui maka kelas akan ditambahkan ke dalam sistem dan dipublikasikan oleh admin, jika tidak disetujui maka admin akan menyampaikan penolakan kepada instruktur dan kelas tidak dipublikasikan.

3.2 Hasil Perancangan Sistem

a. Use Case Diagram

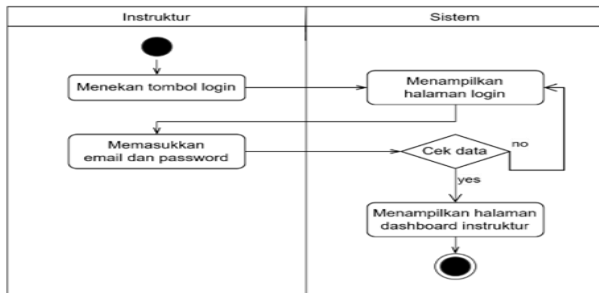
Use case mendeskripsikan sebuah interaksi antara



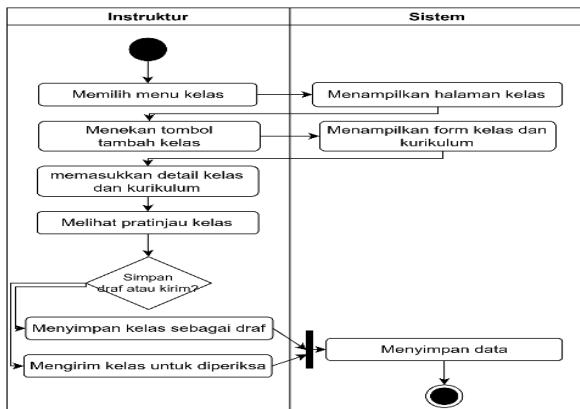
Gambar 4. Use Case Diagram

b. Activity Diagram

Activity Diagram menggambarkan alir aktivitas dari sebuah sistem yang ada pada perangkat lunak [7].

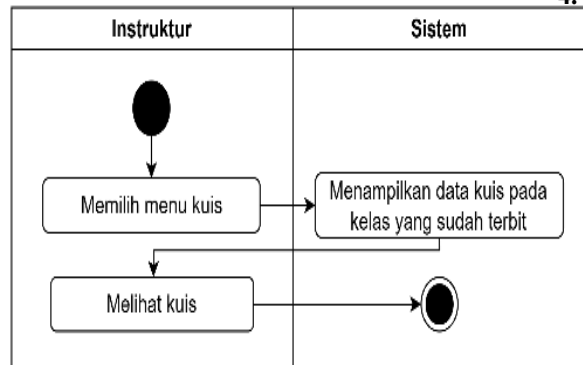


Gambar 5. Activity Diagram Login

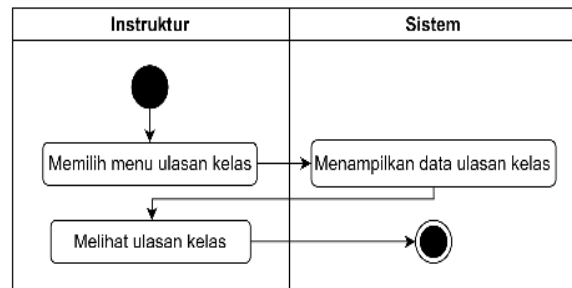


Gambar 6. Activity Diagram Mengajukan Kelas

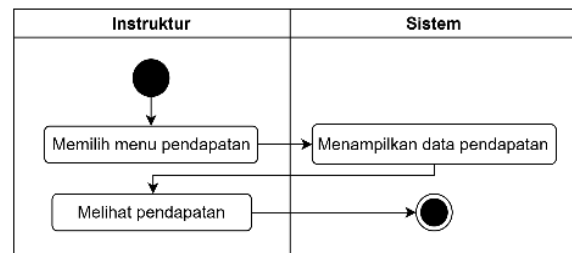
satu atau lebih aktor dengan sistem yang dibuat [7].



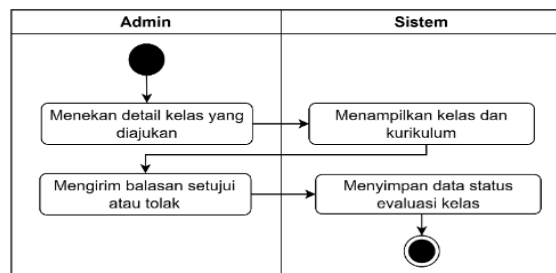
Gambar 7. Activity Diagram Melihat Kuis



Gambar 8 Activity Diagram Melihat Ulasan Kelas



Gambar 9 Activity Diagram Melihat Pendapatan

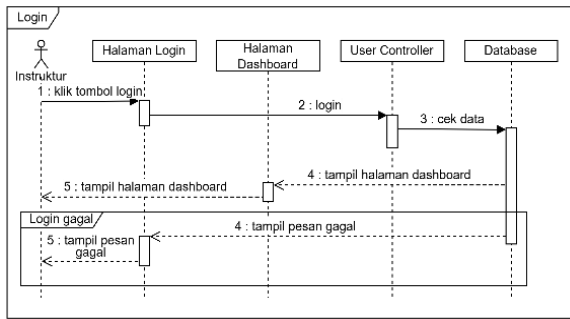


Gambar 10 Activity Diagram Menyetujui/Menolak Kelas

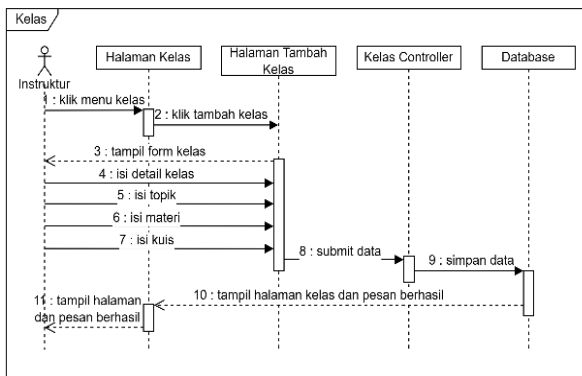
c. Sequence Diagram

Sequence diagram menggambarkan perilaku (behavior) objek pada use case. Sequence diagram bermanfaat untuk menggambarkan metode-metode

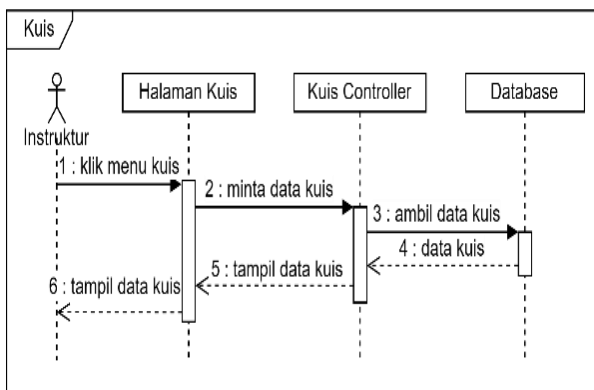
yang dimiliki kelas dalam sebuah use case [10].



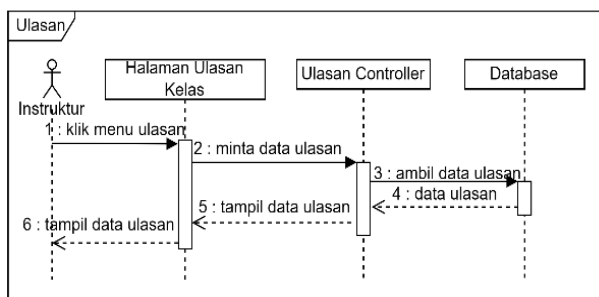
Gambar 11 Sequence Diagram Login



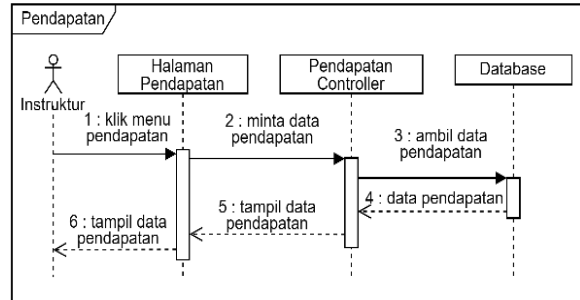
Gambar 12 Sequence Diagram Mengajukan Kelas



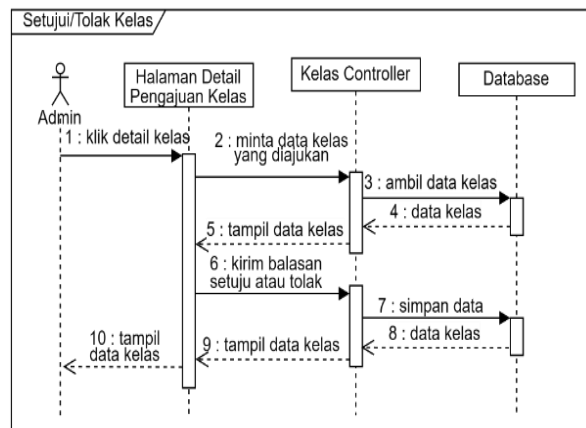
Gambar 13 Sequence Diagram Melihat Kuis



Gambar 14 Sequence Diagram Melihat Ulasan Kelas



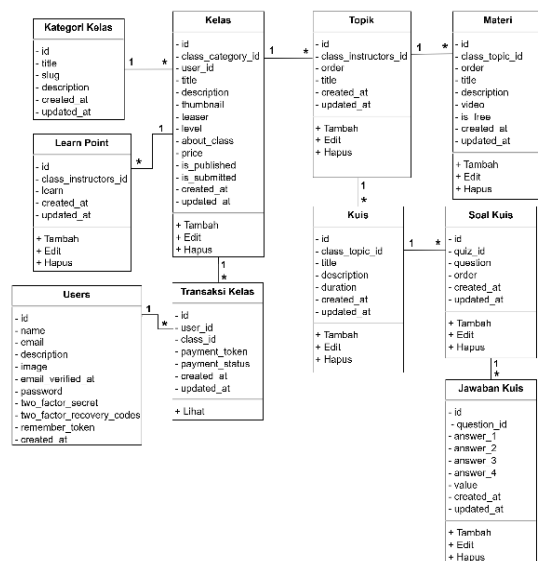
Gambar 15 Sequence Diagram Melihat Pendapatan



Gambar 16 Sequence Diagram Menyetujui/Menolak Kelas

d. Class Diagram

Diagram kelas dibuat agar pembuat sistem membuat kelas-kelas sesuai rancangan di dalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sinkron [7].



Gambar 17 Class Diagram

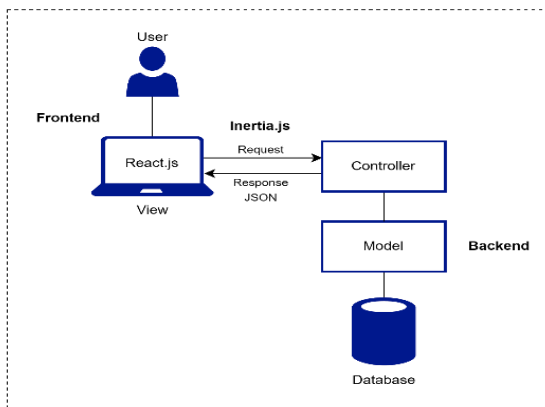
3.3 Hasil Antarmuka Sistem

a. Penerapan SPA

Penerapan SPA pada aplikasi Nusabot untuk akses instruktur menggunakan React JS dan Laravel yang dihubungkan dengan Inertia JS, HeadlessUI sebagai komponen UI, dan Tailwind sebagai *framework* CSS.

Dalam proses penyusunan kurikulum digunakan fitur dari React JS yaitu React Hooks yaitu `useState` untuk menyimpan *state form* penambahan kurikulum. Semua *state* yang berisi data kurikulum disimpan di `localStorage` sehingga setiap penambahan kurikulum sistem akan tetap berada di halaman yang sama. Data akan dikirim dengan membuat *request* menggunakan *Inertia method post* ke sisi *server*.

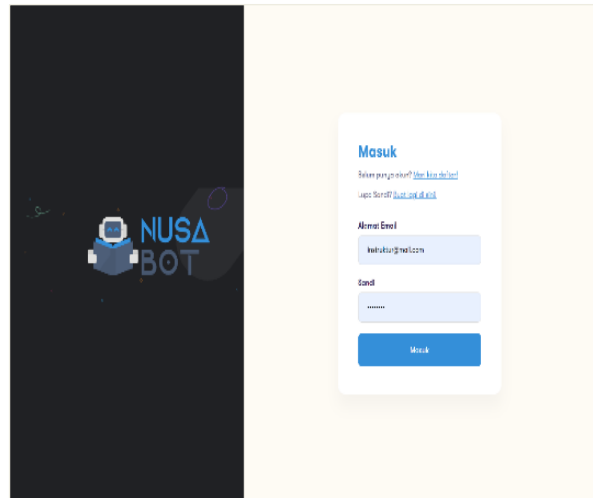
Ketika pengguna mengirim *request* ke *server* melalui *Inertia*, *server* akan menerima dan meneruskannya ke *PHP engine*. Inisialisasi dilakukan sebelum permintaan dikirim ke *middleware* untuk menyaring *request* seperti autentikasi pengguna dan membatasi akses *routes* berdasarkan *role* pengguna [1]. Setelah melewati *routes*, *controller* menerima dan mengirim *request* serta data ke model. Model mengelola data dalam *database* yang digunakan yaitu MySQL dan *phpMyAdmin*. *Responses* data yang diterima dari *server* dalam format *JSON (JavaScript Object Notation)* tidak akan ditampilkan di *view blade*, melainkan dikirim ke sisi klien sebagai *response* [1]. Selanjutnya, panggil *method* *Inertia render* di dalam *controller* dengan menyebutkan nama komponen halaman yang ingin dirender beserta *props* data.



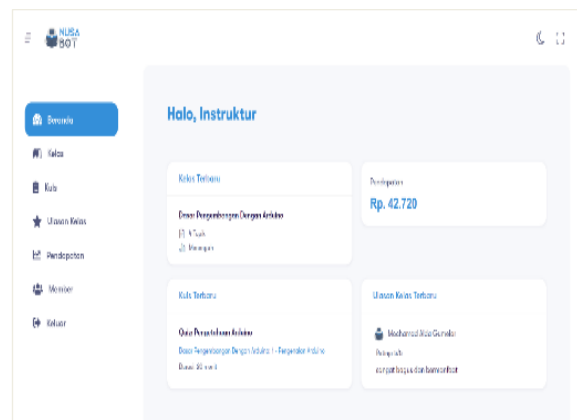
Gambar 18 Arsitektur SPA pada Laravel

b. Antarmuka

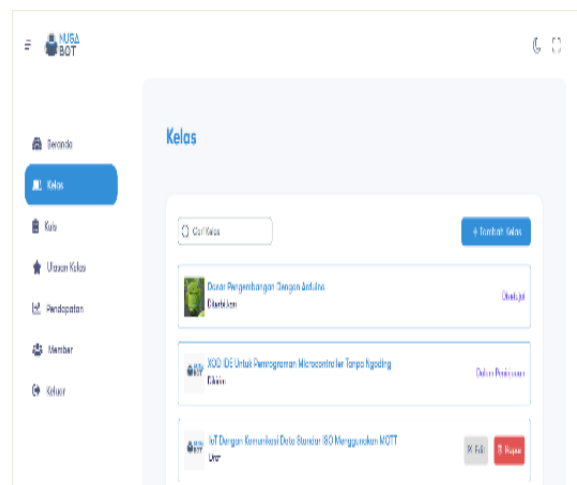
Tampilan antarmuka aplikasi Nusabot berbasis web sebagai berikut:



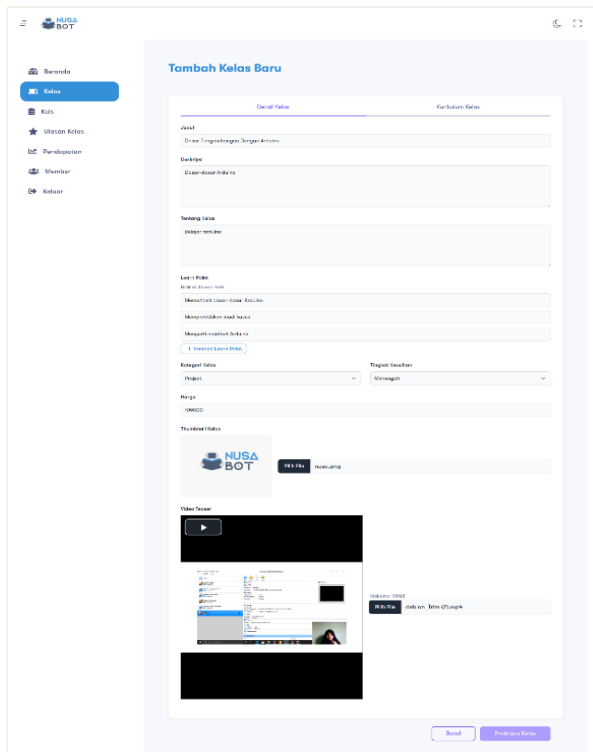
Gambar 19 Halaman Login



Gambar 20 Dashboard Instruktur



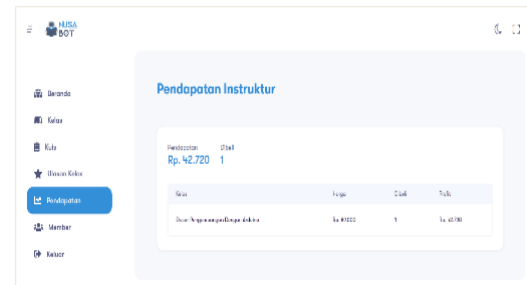
Gambar 21 Halaman Kelas



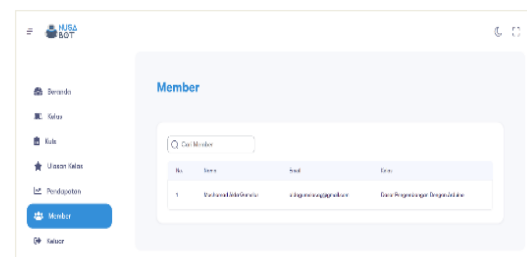
Gambar 22 Halaman Buat Kelas



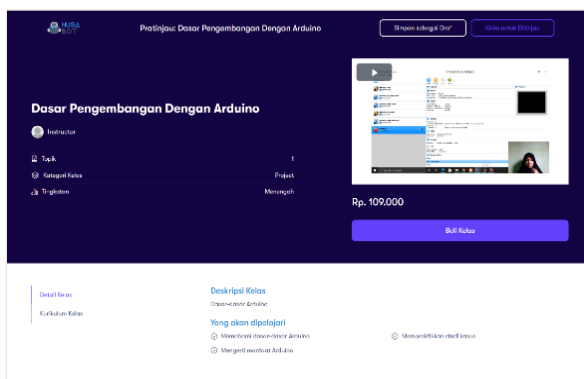
Gambar 25 Halaman Ulasan Kelas



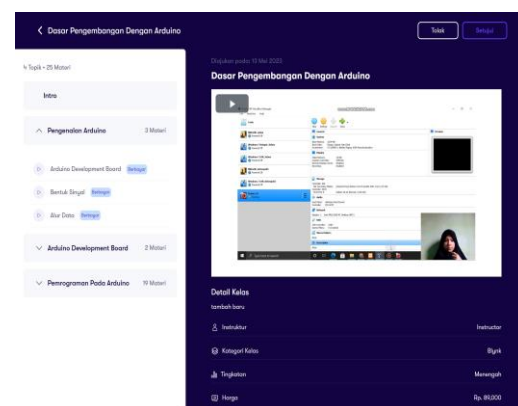
Gambar 26 Halaman Pendapatan



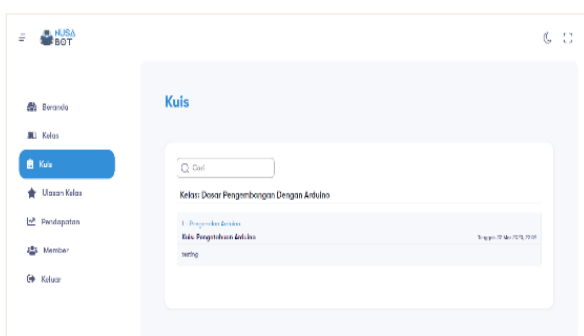
Gambar 27 Halaman Member



Gambar 23 Halaman Pratinjau Kelas



Gambar 28 Halaman Detail Pengajaran Kelas



Gambar 24 Halaman Kuis

3.4 Hasil Pengujian

Pengujian yang digunakan menggunakan pengujian *Black Box* untuk menguji fungsionalitas sistem di sisi klien dan *server*.

Tabel 1. Pengujian *Black Box* Sisi Klien

Pengujian	Hasil yang diharapkan	Hasil
Login	Jika user memasukkan email dan password valid maka sistem menampilkan halaman dashboard instruktur, jika tidak valid maka menampilkan pesan kesalahan	Sesuai
Halaman Kelas	Menampilkan halaman kelas	Sesuai
Menambah kelas	Menambahkan detail kelas, topik, materi video, kuis, foto thumbnail, dan melihat pratinjau kelas sebelum dikirim atau disimpan.	Sesuai
Mengubah kelas yang berstatus draf	Mengubah detail kelas, topik, video materi, kuis, foto thumbnail, dan dapat melihat pratinjau kelas sebelum dikirim atau disimpan.	Sesuai
Menghapus kelas yang berstatus draf	Menghapus kelas beserta kurikulumnya	Sesuai
Halaman Kuis	Menampilkan halaman kuis dan fitur pencarian	Sesuai
Halaman Ulasan Kelas	Menampilkan halaman ulasan dan filter rating	Sesuai
Halaman pendapatan	Menampilkan halaman pendapatan dengan tabel beserta jumlah pendapatan dan terjual	Sesuai
Halaman member	Menampilkan halaman member dengan tabel dan fitur pencarian	Sesuai

Tabel 2. Pengujian Black Box Sisi Server

Pengujian	Hasil yang diharapkan	Hasil
Login	Jika user memasukkan email dan password valid maka sistem menampilkan halaman dashboard instruktur, jika tidak valid maka menampilkan pesan kesalahan	Sesuai
Halaman Kelas	Menampilkan data kelas	Sesuai
Menambah kelas	Mengirim atau menyimpan draf kelas beserta kurikulum	Sesuai
Mengubah kelas yang berstatus draf	Mengirim atau menyimpan perubahan draf kelas	Sesuai
Menghapus kelas yang berstatus draf	Menghapus kelas beserta kurikulumnya	Sesuai
Halaman	Menampilkan data kuis	Sesuai

Kuis		
Halaman Ulasan Kelas	Menampilkan data ulasan	Sesuai
Halaman pendapatan	Menampilkan data pendapatan	Sesuai
Halaman member	Menampilkan data member yang mengikuti kelas	Sesuai

4. KESIMPULAN DAN SARAN

Berdasarkan hasil penelitian tentang Penerapan Single Page Application pada Pengembangan Aplikasi Nusabot, maka dapat disimpulkan bahwa aplikasi berhasil dikembangkan dengan menerapkan SPA menggunakan React JS dan Laravel. Dengan dikembangkannya sistem, instruktur dapat menyusun dan mengirim kelas beserta kurikulum serta admin dapat menyetujui atau menolak kelas melalui aplikasi tanpa perlu menyampaikannya secara manual. Diterapkannya SPA juga membuat instruktur lebih efisien dalam menyusun kurikulum di satu halaman tanpa adanya reload setiap penambahan data. Admin juga tidak perlu menambahkan data kelas yang ingin dipublikasikan karena data yang diajukan instruktur sudah disimpan dalam database, admin bisa langsung memublikasikan kelas yang diajukan.

DAFTAR PUSTAKA

- [1]. Seputra, K. A., & Dewi, L. J. E. (2022, March 30). *A Design of Patient Registration Apps using Flutter, Laravel and, Vue JS*. <https://doi.org/10.4108/eai.27-11-2021.2315532>.
- [2]. Solovei, V., Olshevska, O., & Bortsova, Y. (2017). *The Difference Between Developing Single Page Application And Traditional Web Application Based On Mechatronics Robot Laboratory Onaft Application*. <http://www.atbp.onaft.edu.ua/4http://creativecommons.org/licenses/by/4.0/>
- [3]. Santoso, M. F. (2021). Teknik Single Page Application (Spa) Layout Web Dengan Menggunakan React Js Dan Bootstrap. *Jurnal Khatulistiwa Informatika*, 9(2). <https://doi.org/10.31294/jki.v9i2.11357>.
- [4]. Wibowo, A. T., & Wiguna, A. S. (2019). Pemanfaatan Teknologi Single Page

- Application (SPA) dalam Pembuatan Aplikasi Feedback Dosen dari Mahasiswa Sebagai Bentuk Pengawasan Lembaga Terhadap Kinerja Dosen di Bidang Pengajaran. *SMARTICS Journal*, 5(1). <https://doi.org/10.21067/smartics.v5i1.3327>.
- [5]. Nasution, & Iswari, L. (2021). Penerapan React JS Pada Pengembangan FrontEnd Aplikasi Startup Ubaform. *Jurnal Automata - UII*, 2(2).
- [6]. Bustamin, S. (2021). Aplikasi dekstop multi platform untuk redis client framework Electron JS dan React JS. *Dewantara Journal of Technology*, 2(1).
- [7]. Sukamto, R. A., & Salahuddin, M. (2018). Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek. In *Informatika Bandung*.
- [8]. Nursaid, F. F., Hendra Brata, A., & Kharisma, A. P. (2020). Pengembangan Sistem Informasi Pengelolaan Persediaan Barang Dengan ReactJS Dan React Native Menggunakan Prototype (Studi Kasus: Toko Uda Fajri). *J-Ptiik.Ub.Ac.Id*, 4(1).
- [9]. Pressman, R. S., & Maxim, B. R. (2010). *Software Engineering*.
- [10]. Nugroho, A. (2014). Rekayasa Perangkat Lunak Menggunakan UML & Java. In *Rekayasa Perangkat Lunak Menggunakan UML & Java*.