

I. PENDAHULUAN

1.1 Latar Belakang

Backend development adalah proses desain sistem yang terjadi di balik layar situs *web* atau aplikasi. Ada beberapa bahasa pemrograman untuk *backend development* diantaranya adalah *javascript*, *php*, *ruby*, *python*, dan lain-lain. Ini semua adalah sistem yang bekerja di belakang layar dan tentunya tidak terlihat secara langsung oleh pengguna, namun dalam mendesain sebuah *web* atau aplikasi, posisi *backend development* memberikan kekuatan pada apa yang terjadi, terutama dalam mengolah data yang akan ditampilkan (Hasanuddin dkk., 2022). Pengembangan *backend Development* berfokus pada *database*, skrip, dan arsitektur situs *web* atau aplikasi. *Backend* sendiri dibuat dengan menggunakan kode pemrograman, sehingga dapat mengelola data, informasi yang akan diberikan kepada *User interface (UI)* yang disediakan oleh *frontend*.

Politeknik Negeri Lampung (POLINELA) merupakan perguruan tinggi negeri yang menyelenggarakan pendidikan vokasi. POLINELA resmi menyelenggarakan pendidikan tinggi secara berdikari & sebagai satu bentuk Perguruan Tinggi Negeri (PTN) pada Provinsi Lampung semenjak lepas 7 April 2001 menurut SK. Menteri Pendidikan Nasional No. 036/O/2001. POLINELA memiliki Unit Kegiatan Mahasiswa (UKM) dalam upaya mengembangkan minat bakat, karakter, profesionalisme, kepemimpinan, kerohanian serta kewirausahaan (*entrepreneurship*) mahasiswa.

UKM sendiri merupakan wadah bagi mahasiswa untuk mengembangkan diri, mahasiswa dapat mengasah *hard skill* dan *softskill* sesuai dengan bakat dan minatnya (Saudah, 2018). POLINELA sudah memiliki beberapa UKM dan Organisasi Mahasiswa (Ormawa) yang aktif dan melaksanakan kegiatan sesuai dengan agenda yang telah dijadwalkan. Kegiatan ini dilaksanakan setiap tahunnya dan menjadi daya tarik bagi calon mahasiswa yang akan mendaftar untuk melanjutkan pendidikan di POLINELA. Namun untuk mempublikasikan kegiatan dan informasi kepada mahasiswa dan calon mahasiswa. UKM tersebut masih menggunakan *platform* sosial media seperti *Instagram*, *Twitter*, dan *Youtube* hal

tersebut masih dinilai belum cukup untuk mencakup mahasiswa secara luas untuk mengetahui kegiatan dan informasi mengenai UKM di POLINELA.

Saat ini UKM POLINELA sendiri sudah memiliki *website* yang terhubung dengan *web* resmi milik POLINELA. Berdasarkan hasil penelitian kepada delapan UKM yang ada di POLINELA, *web* tersebut dinilai masih belum memenuhi kebutuhan dari setiap UKM POLINELA, seperti belum memiliki halaman untuk informasi mengenai pendaftaran dan pengumuman informasi dari setiap UKM, tampilan yang sangat sederhana sehingga mahasiswa kurang tertarik untuk membaca informasi berita yang di terbitkan oleh setiap UKM, dan beberapa fitur yang dirasa kurang oleh pengurus dan anggota dari setiap UKM.

Berdasarkan latar belakang dan perumusan masalah diatas, maka perlu dibuat sebuah pengembangan sistem *backend* berbasis *Application Programming Interface (API)* yang dapat di implementasikan untuk membuat *web* berbasis *web*, *Mobile* dan berbagai *platform* lainnya. Maka dapat di hasilkan sistem berbasis *API* dengan arsitektur *Representational State Transfer (REST)* dalam pengembangan *backend* untuk mempermudah pembuatan *web frontend* membuat tampilan *UI* sehingga dapat memenuhi kebutuhan dari setiap UKM POLINELA.

Pemanfaatan *web service* turut membantu dalam pembuatan *backend*, *web service* memiliki manfaat dapat mengatasi *interoperability* dan mengintegrasikan sistem berbeda. Selain itu *web service* juga dapat memberikan efek internet murah menggunakan mekanisme *Simple Object Access Protocol (SOAP)* pada protokol di *Hypertext Transfer Protocol (HTTP)*.

1.2 Tujuan

Tujuan dari pembuatan tugas akhir ini adalah menghasilkan rancang bangun *web service API* dan dokumentasi *REST API web* portal unit kegiatan mahasiswa di Politeknik Negeri Lampung serta membangun dokumentasi data pada *backend*.

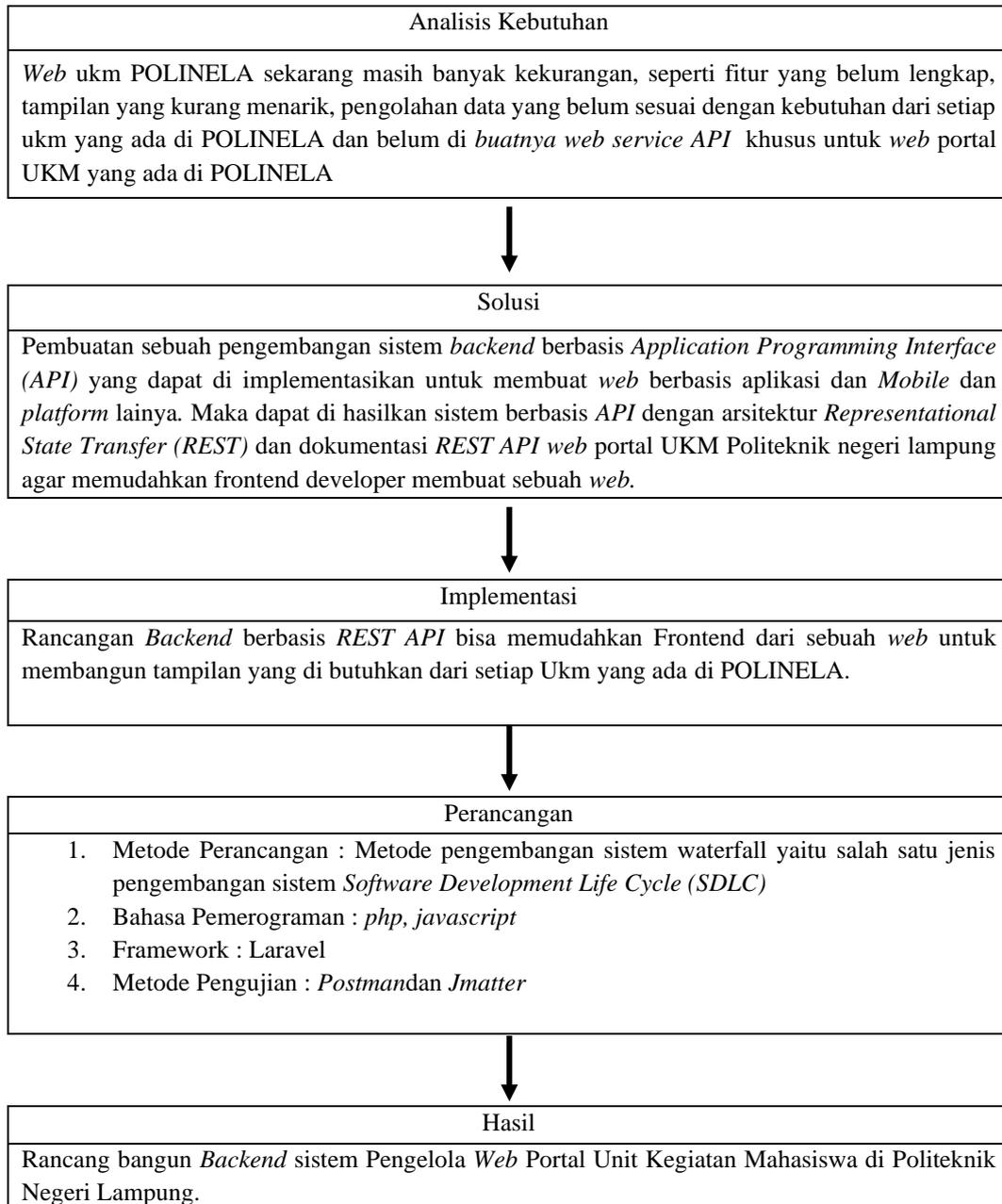
1.3 Kerangka Pemikiran

Setiap kegiatan yang di laksanakan oleh UKM yang ada di POLINELA selalu di publikasikan sehingga mahasiswa dan calon mahasiswa selalu mendapatkan informasi. UKM POLINELA sendiri mempublikasikan kegiatan dan informasi masih melalui *platform* sosial media seperti *Instagram*, *Twitter*, dan

Youtube hal tersebut masih dinilai belum cukup untuk mencakup mahasiswa secara luas untuk mengetahui kegiatan dan informasi mengenai UKM di POLINELA.

Saat ini UKM POLINELA sendiri sudah memiliki *website* yang terhubung dengan *web* resmi milik POLINELA. Sehingga belum adanya *API* khusus untuk UKM yang ada di POLINELA. Selain itu berdasarkan hasil penelitian kepada 8 UKM yang ada di POLINELA, *web* tersebut dinilai masih belum memenuhi kebutuhan dari setiap UKM POLINELA, seperti belum memiliki halaman untuk informasi mengenai pendaftaran dan pengumuman informasi dari setiap UKM, tampilan yang sangat sederhana sehingga mahasiswa kurang tertarik untuk membaca informasi berita yang di terbitkan oleh setiap UKM, dan beberapa fitur yang dirasa kurang oleh pengurus dan anggota dari setiap UKM.

Berdasarkan permasalahan tersebut, dibutuhkan pembuatan *web service API* dan dokumentasi *REST API web* portal unit kegiatan mahasiswa di POLINELA supaya dapat memudahkan *frontend* dalam membangun aplikasi yang memiliki fitur yang lengkap dan sebuah tampilan *web* yang menarik sehingga mahasiswa tertarik untuk mengunjungi *Web* UKM POLINELA. Solusi yang dapat dilakukan adalah dengan membuat "Rancang Bangun *Web Service API* dan Dokumentasi *REST API Web* Portal Unit Kegiatan Mahasiswa di Politeknik Negeri Lampung". Bagan kerangka pemikiran disajikan pada Gambar 1.



Gambar 1. Kerangka Pemikiran

1.4 Kontribusi

Dengan dibuatnya Rancang Bangun *Web Service API* dan Dokumentasi *REST API Web* Portal Unit Kegiatan Mahasiswa di Politeknik Negeri Lampung. diharapkan dapat memberikan kontribusi sebagai berikut :

1. Unit Kegiatan Mahasiswa Politeknik Negeri Lampung
 - a. Mempermudah untuk memberikan informasi kepada Mahasiswa Politeknik Negeri Lampung

- b. Mempermudah mengelola data pada *Web* Politeknik Negeri Lampung
 - c. Mempermudah mengelola *Web* Politeknik Negeri Lampung
2. Mahasiswa
- a. Mempermudah menerima informasi mengenai UKM di Politeknik Negeri Lampung
 - b. Mempermudah untuk melakukan komunikasi secara cepat dan mudah kepada UKM yang ada di Politeknik Negeri Lampung.
3. Politeknik Negeri Lampung
- a. Mempermudah untuk mengontrol setiap kegiatan yang diadakan oleh setiap UKM yang ada di Politeknik Negeri Lampung
 - b. Menyumbangkan karya tulisan kepada Politeknik Negeri Lampung
4. *Frontend Developer*
- a. Membantu dalam pembuatan *Frontend web* Portal UKM Politeknik Negeri Lampung dengan pembuatan dokumentasi *API*
 - b. Membantu dalam mengelola data *web* yang telah di bangun

II. TINJAUAN PUSTAKA

2.1 *Backend Development*

Backend development adalah proses dimana sebuah sistem dirancang secara fokus pada *database*, *scripting*, dan arsitektur melalui *web* atau aplikasi yang dilakukan dibelakang layar sehingga tidak bisa dilihat secara langsung oleh penggunanya, meskipun bekerja dibelakang layar *Backend development* memberikan kekuatan pada pengolahan data yang ditampilkan.

Backend development dibuat menggunakan *javascript*, *php*, *ruby*, dan *python* sebagai bahasa pemrograman. Kode dari *backend development* akan membantu *User* untuk mengkomunikasikan informasi data yang terdapat pada *database* melalui perantara tampilan *User Interface (UI)* yang disediakan oleh *frontend* (Hasanuddin dkk, 2022) .

2.2 *Application Programming Interface (API)*

API ialah antarmuka yang digunakan untuk mengakses aplikasi atau layanan secara terprogram. Pengembang *API* dapat menggunakan fungsionalitas yang ada dari aplikasi lain tanpa harus mengembangkan dari awal. Pada konteks *web*, *API* adalah panggilan fungsi melalui *Hyper Text Transfer Protocol (HTTP)* dan mendapatkan *respons* berupa *Extensible Markup Language (XML)* atau *JavaScript Object Notation (JSON)* (Hasanuddin dkk, 2022).

API dapat memfasilitasi pengembangan *web* atau aplikasi secara fungsional, Itu juga memudahkan untuk memperluas pengembangan dengan *API* dan menyimpan data di *server*. Dengan demikian, ini dikembangkan dengan tujuan mempercepat transfer data dan membuat lebih banyak situs *web* atau aplikasi yang kaya fitur.

2.3 *Representational State Transfer (REST)*

Representational State Transfer merupakan arsitektur berbasis *web* baku yang memakai protokol *HTTP* guna mengkomunikasikan data. Berkomunikasi data yang dimaksud merupakan salah satu bentuk implementasi dari *web service* yang digunakan Sebagai standar untuk pertukaran data antar aplikasi. Karena aplikasi tersebut menggunakan bahasa pemrograman yang berbeda untuk berkomunikasi

satu sama lain harus dibutuhkan *web service*. Sebagai protocol untuk komunikasi data *REST* pertama kali diperkenalkan oleh Roy Fielding pada tahun 2000.

Pada arsitektur *REST*, *server* menyediakan *resources* (sumber data) dan *REST client* akan mengakses dan menampilkan sumber data tersebut untuk penggunaan selanjutnya (Feridi, 2019).

2.4 Framework Laravel

Web framework merupakan kerangka kerja aplikasi yg didesain buat mendukung pengembangan aplikasi *web* termasuk *web service*, *web resources*, dan *web API*. *Web framework* menyediakan cara standar untuk membuat dan mengembangkan aplikasi *web*, memudahkan pekerjaan pengembang (Saputra, 2018).

Menurut *web* resmi laravel *The php Framework for Web Artisans*, *Laravel* adalah *framework* aplikasi *web* yang dapat menggunakan syntax yang *expressive* dan *elegant*. Tim *Laravel* telah membuat fondasi yang dibutuhkan untuk membebaskan penggunanya untuk berkreasi tanpa perlu tau memikirkan hal-hal kecil.

Framework laravel sekarang ini aktif dikembangkan, setiap enam bulan rilis major dirilis untuk publik, menyediakan pilihan yang menarik untuk pengembang aplikasi *web* yang menginginkan dukungan yang baik untuk framework aplikasi *web* yang digunakan (Andrew Fernando Pakpahan dkk, 2020).

2.5 JSON Web Token (JWT)

JSON web token ialah sebuah token yang dapat digunakan oleh lebih dari 1 pihak untuk mengirimkan data terverifikasi, kegunaan *JWT* ini seperti *Password* jadi ketika seseorang berhasil melakukan login maka *server* akan memberikan sebuah token yang akan disimpan pada *Local Storage*, sehingga jika *Users* ingin mengakses halaman halaman tertentu harus menyertakan token tersebut (Hadyan, 2021).

JWT memiliki tiga struktur dasar dimana terdapat *header*, *payload*, dan *verify signature*. Bagian *header* disini berisi informasi mengenai algoritma dan jenis *token* yang digunakan, bagian *payload* sebagai sebuah informasi data yang akan dikirim

melalui *token*, dan bagian terakhir *verify signature* merupakan hasil dari *hash* gabungan *header* yang telah ditambah dengan kode *secret*.

2.6 Web Service

Menurut *World Wide Web Consortium (W3C)*. *Web service* merupakan suatu program *software* yang dirancang untuk membantu interoperabilitas dan interaksi antar program pada suatu jaringan. Penggunaan *web service* sebagai suatu fasilitas yang menyediakan layanan dalam bentuk informasi atau data kepada sistem lain, sehingga bisa berinteraksi dengan sistem tersebut melalui layanan-layanan yang disediakan. Data pada *web service* dapat diakses oleh sistem lain walaupun berbeda *platform*, sistem operasi, dan bahasa pemrograman karna *web service* menyimpan data informasi dalam format *JSON* atau *XML* (Zaman, 2017).

2.7 Metode Pengembangan Sistem

Metode pengembangan sistem *waterfall* yaitu salah satu jenis pengembangan sistem *Software Development Life Cycle (SDLC)*. *SDLC* adalah sebuah metode atau proses pengembangan sistem agar pekerjaan berjalan dengan terstruktur, efektif dan mencapai tujuan dengan sempurna.

Dalam metode *Waterfall*, ide kebutuhan spesifikasi dimulai pengguna kemudian melanjutkan melalui tahapan perencanaan (*planning*), pemodelan (*modeling*), konstruksi (*construction*), dan penyebaran sistem ke para pelanggan/pengguna (*deployment*), yang berakhir dengan dukungan pada perangkat lunak hasil lengkap (Supandi dkk, 2019).

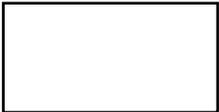
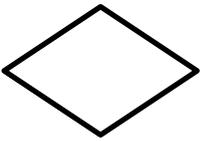
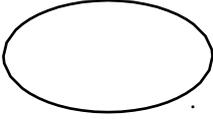
2.8 Entity Relationship Diagram (ERD)

ERD adalah grafik yang mewakili tabel dalam *database*. *ERD* menjelaskan tabel dan menjelaskan hubungan antara tabel *database*. *ERD* menjelaskan struktur data dan hubungan antara data menggunakan notasi dan simbol.

Model *ERD* berisi komponen entitas dan set hubungan yang masing-masing datang dengan atribut yang menyajikan seluruh fakta bahwa ditinjau agar hubungan dapat diketahui antara entitas yang ada dan atributnya. Selain itu juga bisa menggambarkan hubungan yang ada dalam pengolahan data, seperti hubungan

many to many, one to many, one to one (Sukrianto, 2017). Simbol-simbol ERD ditunjukkan pada Tabel 1.

Tabel 1. Simbol - simbol ERD

Simbol	Nama	Keterangan
(1)	(2)	(3)
	Entitas	Kumpulan objek yang dapat diidentifikasi secara unik
	Relasi	Sebuah hubungan yang ada antara satu atau lebih entitas. Jenis-jenis hubungan tersebut adalah : <i>one to one</i> , <i>one to many</i> , dan <i>many to many</i>
	Atribut	Karakteristik entitas atau hubungan yang merupakan deskripsi rinci entitas
	Garis relasi	Hubungan antara entitas dengan atribut dan himpunan entitas dengan himpunan relasi lainnya

Sumber : (Alamsyah, 2019).

2.9 Postman

Postman adalah aplikasi komputer yang digunakan untuk menguji *API*, cara yang dilakukan *Postman* dalam menguji *API* adalah dengan cara *Postman* akan mengirim permintaan ke *server web* dan kemudian akan menerima *respons*. Menggunakan antarmuka yang sederhana dan ramah pengguna. *user* dengan mudah dapat mengirim permintaan dengan mengisi data yang diperlukan saja kemudian memilih metode *HTTP* dan tekan tombol kirim.

Postman memiliki dua penawaran yaitu gratis dan berbayar, dengan penawaran berbayar *postman* dapat digunakan dengan fitur yang lebih pro seperti dapat diakses oleh banyak pengguna, kontrol, *otentikasi SSO* dll. *Postman* juga dapat diunduh menjadi lintas *platform* untuk *windows, Linux & Mac* dan untuk membuat pengguna lebih mudah kini *Postman* juga hadir dalam versi *web* (Fathurrahman, 2021).

2.10 Swagger

Swagger merupakan tempat dokumentasi *API*, di dalam *Swagger* kita bisa melakukan desain ulang ketika desain ulang yang dilakukan mengalami *error* akan ada sebuah log yang diberitahukan secara *Real-time*. *Swagger* berguna untuk membuat kode *API* baru dengan merancang, membangun, mengakses dan mendokumentasikan. Dokumentasi menjadi salah satu hal yang penting karena code akan diperlihatkan secara setumpuk baris sehingga akan membuat *clien* lebih produktif dan dapat berkomunikasi antara *back-end developer* dan *front-end developer*.

Swagger juga dapat melakukan *Request* dari *API* sehingga akan menjadikan *swagger* sebagai *framework* komplit untuk *project API*. beberapa urutan cara dalam membuat *swagger* dengan menggunakan *format file Json* dan *yaml* dengan version 2.0 dan menggunakan alat yang bernama *text editor* untuk membuat file spesifikasi (Edward, 2020).

2.11 Jmeter

Jmeter digunakan untuk *load* dan *performance testing tool*. Pengujian dilakukan dengan simulasi pengguna yang akan mengakses *server REST API* yang telah di buat dengan jumlah pengguna dan waktu *Request* yang berbeda-beda. JMeter dapat melakukan *load/stress testing web Application, FTP Application* dan *Database server test* (Sonia Ginasari dkk, 2021).