

PERANCANGAN APPLICATION PROGRAMMING INTERFACE (API) BERBASIS NODE JS MENGGUNAKAN GAYA ARSITEKTUR REPRESENTATIONAL STATE TRANSFER (REST) UNTUK PENGEMBANGAN APLIKASI EDUPLACE DI PT. ERA SOLUSI DATA

DESIGN OF APPLICATION PROGRAMMING INTERFACE (API) BASED ON NODE JS USING REPRESENTATIONAL STATE TRANSFER (REST) ARCHITECTURE FOR EDUPLACE APPLICATION DEVELOPMENT AT PT. ERA SOLUSI DATA

Galang Pratama Sukma Putra¹, Renna Yanwastika Ariyana²

^{1,2} Program Studi Informatika, Fakultas Teknologi Industri, Institut Sains & Teknologi AKPRIND Yogyakarta
Jl. Kalisahak No. 28 Komplek Balapan 55222

ABSTRAK

Pada tahun 2020, pandemi covid-19 memaksa pembelajaran dilakukan secara jarak jauh. Hal ini menjadi peluang untuk menciptakan platform pembelajaran daring. Eduplace menjadi alternatif platform pembelajaran daring yang menyediakan materi pembelajaran mandiri yang dapat diakses di mana saja dan kapan saja melalui perangkat ponsel cerdas. Eduplace dibuat dengan memanfaatkan teknologi Javascript. Pada aplikasi Eduplace, bagian front-end dan database membutuhkan penghubung sehingga dibutuhkan sistem yang dapat mendefinisikan interaksi antara beberapa perangkat lunak, mulai dari permintaan yang dapat dibuat, bagaimana membuatnya, format data yang harus digunakan, ketentuan yang harus diikuti, dan sebagainya. Semua hal tersebut dapat diakomodasi oleh Application Programming Interface (API), sehingga dapat menghubungkan server dan client dengan baik. Oleh karena itu perlu dibuat perangkat lunak pendukung untuk Eduplace di sisi server. Salah satu perangkat lunak yang populer digunakan adalah Node Js karena mampu menyediakan API menggunakan arsitektur REST dengan cepat. Selain itu, Node Js ditulis dalam bahasa pemrograman Javascript. Eduplace menggunakan Javascript di sisi pengguna, sehingga penggunaan Node Js akan melengkapi kebutuhan di sisi server dengan menggunakan bahasa pemrograman yang sama. Dengan bahasa pemrograman yang sama akan mempermudah developer dalam membuat platform tersebut

Kata Kunci: *Application Programming Interface, API, Representational State Transfer, REST*

ABSTRACT

In 2020, the Covid-19 pandemic forces learning to be carried out remotely. This is an opportunity to create an online learning platform. Eduplace is an alternative online learning platform that provides independent learning materials that can be accessed anywhere and anytime via smartphone devices. Eduplace is made using Javascript technology. In the Eduplace application, the front-end and database parts need a connector, so a system is needed that can define interactions between several software, starting from the requests that can be made, how to make them, the data format to use, the conditions that must be followed, and so on. All of these things can be accommodated by the Application Programming Interface (API), so that it can connect the server and client properly. Therefore, it is necessary to make supporting software for Eduplace on the server side. One of the popular software used is Node Js because it is able to quickly provide an API using the REST architecture. In addition, Node Js is written in the Javascript programming language. Eduplace uses Javascript on the user side, so using Node Js will complement the needs on the server side by using the same programming language. With the same programming language, it will make it easier for developers to create the platform.

Keywords: *Application Programming Interface, API, Representational State Transfer, REST*

1. PENDAHULUAN

Awal tahun 2020, Indonesia dihebak dengan adanya bencana pandemi COVID-19 yang menyebar luas dimasyarakat (Bina & Ramadhani, 2021). COVID-19 atau Corona Virus pertamakali muncul di kota Wuhan, Provinsi Hubai, Cina pada bulan Desember 2019. COVID-19 merupakan virus zoonins yang diduga berasal dari kelelawar yang menulari manusia (Shereen, Khan, Kazmi, Bashir, & Siddique, 2020). Bencana pandemi COVID-19 mengakibatkan perubahan di seluruh tatanan kehidupan manusia, tak terkecuali dunia pendidikan. Dimana dunia Pendidikan dihadapkan pada permasalahan yang cukup komplek. Banyak negara harus menghentikan penyelenggaraan Pendidikan secara tatap muka langsung termasuk Indonesia. Tercatat hingga 1 april 2020 terdapat 188 negara ikut terdampak akibat pandemi COVID-19 dengan jumlah setidaknya 1,5 miliar peserta didik dan 60 juta diantaranya adalah peserta didik Indonesia (Puspitorini, 2020).

United Nations Educational, Scientific and Cultural Organization (UNESCO) merupakan badan khusus perserikatan bangsa-bangsa (PBB) (Wikipedia, 2019). melalui, *International Institute of Educational Planning (IIEP)* didedikasi untuk dapat merencanakan, membuat kebijakan dan mengkoordinasikan materi dan sumberdaya yang relevan untuk mengatasi semua tantangan terkait dengan penyelenggaraan pendidikan selama masa pandemi (iiep, 2020). Sebagai negara yang ikut tergabung dalam PBB, Indonesia tentu turut menyepakati kebijakan tersebut dengan dikeluarkannya surat edaran dari Menteri Pendidikan dan Kebudayaan Nomor 36962/MPK.A/HK/2020 tanggal 13 Maret 2020 terkait pembelajaran secara daring dan bekerja dari rumah dalam rangka pencegahan COVID-19 (Hidayah, Adawiyah, & Mahanani, 2020). Hal ini merupakan bentuk respon pemerintah terhadap upaya pencegahan, maka pemerintah Indonesia resmi mengambil kebijakan agar masyarakat bekerja di rumah, belajar di rumah dan beribadah di rumah (Firyal, 2020).

Peluang untuk menciptakan platform pembelajaran daring. Adanya platform Eduplace menjadi alternatif yang dapat digunakan untuk pembelajaran daring yang menyediakan materi pembelajaran mandiri yang dapat diakses di mana saja dan kapan saja melalui perangkat ponsel cerdas. Eduplace dibuat dengan memanfaatkan teknologi Javascript.

Pada aplikasi Eduplace, bagian *front-end* dan *database* membutuhkan penghubung sehingga dibutuhkan sistem yang dapat mendefinisikan interaksi antara beberapa perangkat lunak, mulai dari permintaan yang dapat dibuat, bagaimana membuatnya, format data yang harus digunakan, ketentuan yang harus diikuti, dan sebagainya. Semua hal tersebut dapat diakomodasi oleh *Application Programming Interface (API)*, sehingga dapat menghubungkan server dan client dengan baik.

Penerapan API menggunakan standar arsitektur *Representational State Transfer (REST)* membuat sistem lebih sederhana dan ringkas. REST disisi server menyediakan sumber daya berupa data dan REST di sisi pengguna dapat mengakses dan menampilkan sumber daya tersebut untuk penggunaan selanjutnya.

Sesuai dengan kebutuhan yang telah diuraikan, dalam penelitian ini adalah membuat perangkat lunak pendukung untuk Eduplace di sisi server. Salah satu perangkat lunak yang populer digunakan adalah Node Js karena mampu menyediakan API menggunakan arsitektur REST dengan cepat. Selain itu, Node Js ditulis dalam bahasa pemrograman Javascript. Eduplace menggunakan Javascript di sisi pengguna, sehingga penggunaan Node Js akan melengkapi kebutuhan di sisi server dengan menggunakan bahasa pemrograman yang sama. Dengan bahasa pemrograman yang sama akan mempermudah developer dalam membuat platform tersebut.

2. TINJAUAN PUSTAKA

Penelitian tentang perancangan API dengan menerapkan standar arsitektur REST ini mengacu pada artikel penelitian yang relevan, di antaranya yang penelitian dilakukan oleh (Putra & Susetyo, 2020), (Arhandi, 2016), dan (Pranata, Hijriani, & Junaidi, 2018).

Dalam penelitian yang dilakukan oleh (Putra & Susetyo, 2020) Susetyo berjudul "Implementasi API Master Store Menggunakan FLASK, REST, dan ORM di PT XYZ" dipaparkan bahwa sistem Master Store masih berjalan dan bekerja dengan arsitektur Monolithic yang mengharuskan setiap aplikasi baru maupun lama untuk melakukan duplikasi *database* Master Store, hal ini dapat menyebabkan pelemahan dan penurunan performa *database* seiring berjalannya waktu. Untuk mengatasi permasalahan tersebut, peneliti membuat API Master Store yang mengubah

arsitektur saat ini menjadi *Microservices* di mana sistem Master Store dapat digunakan secara terus-menerus oleh aplikasi lain dan mempermudah developer untuk membuat aplikasi baru dengan menggunakan *database* Master Store tanpa penurunan terhadap *database* Master Store.

Penelitian API REST juga (Arhandi, 2016) yang berjudul “Pengembangan Sistem Informasi Perijinan Tenaga Kesehatan Dengan Menggunakan Metode *Back End* dan *Front End*” memaparkan tentang Dinas Kesehatan Kabupaten Pasuruan yang ingin mengembangkan sebuah sistem informasi yang dapat melayani proses pembuatan izin praktik dan izin kerja dengan cepat dan efisien. Dengan menggunakan *front-end* yang berupa aplikasi pada perangkat yang terpisah dengan *back-end* sehingga developer dapat mengembangkan berbagai macam bentuk aplikasi di berbagai macam *device* dan platform yang berbeda. *Back-end* dikembangkan dengan menggunakan bahasa pemrograman PHP dan menggunakan *framework* Laravel dengan memanfaatkan REST API, sedangkan *front-end* dikembangkan dengan bahasa pemrograman Javascript dengan *framework* Angular Js. Peneliti berharap dengan penelitian dengan metode *front-end* dan *back-end* yang baik dan benar dapat menjadi landasan pengembangan sistem informasi pada masa yang akan datang dengan lebih baik.

Penelitian lain yang dilakukan oleh (Pranata, Hijriani, & Junaidi, 2018) dengan judul “Perancangan *Application Programming Interface* (API) Berbasis Web Menggunakan Gaya Arsitektur *Representational State Transfer* (REST) Untuk Pengembangan Sistem Informasi Administrasi Pasien Klinik Perawatan Kulit”. Penelitian ini memaparkan bahwa dalam bidang ilmu kesehatan, pemanfaatan komputer untuk mendukung prosedur dan cara-cara dalam penyelenggaraan pelayanan kesehatan yang dikenal dengan istilah komputerisasi administrasi. Dukungan komputer pada administrasi pelayanan kesehatan yang difasilitasi oleh pihak konsultan teknologi informasi umumnya berbentuk sistem informasi. Dalam pengembangannya, konsultan teknologi informasi sebaiknya memperhitungkan bahwa kasus yang dihadapi kliennya akan terus berkembang. Dalam pengembangannya, API dengan pedoman arsitektur REST digunakan sebagai *back-end* sistem informasi administrasi pasien klinik perawatan kulit. API yang dikembangkan menggunakan *Javascript Object*

Notation (JSON) sebagai standar format dalam komunikasi data serta *JSON Web Token* (JWT) sebagai kode otentikasi pengguna sistem.

Berdasarkan penelitian yang telah dilakukan peneliti sebelumnya yang memaparkan mengenai kelebihan penggunaan API dengan standar arsitektur REST, maka dalam penelitian ini juga menerapkan API REST namun dengan menggunakan teknologi Node Js, Express Js, dan MongoDB sebagai penyimpanan basis datanya.

3. METODOLOGI PENELITIAN

Pada penelitian di PT. Era Solusi Data sebagai *back-end developer* tentu membutuhkan data-data yang terkait pembuatan model data dan struktur API yang nantinya akan dibuat menggunakan *Node Js*. Adapun tahapan yang dilakukan dalam penelitian yang dilakukan di PT. Era Solusi data adalah sebagai berikut:

- 1) Study Literatur dilakukan dengan mengumpulkan data dari internet, buku-buku maupun sumber referensi lain yang berkaitan dengan pembuatan model data dan struktur API yang dapat digunakan pada *Node Js*.
- 2) Observasi dan Wawancara, dilakukan dengan datang ke PT. Era Solusi Data untuk mengamati, dan mendokumentasikan secara langsung model data yang dibutuhkan, serta struktur API yang nantinya akan digunakan untuk merancang layanan server pada PT. Era Solusi Data dengan menerapkan standar arsitektur REST. Proses wawancara dilakukan dengan bapak Meychel Danius Fedrix, Ishac Dainurry, dan Kurnia Setyo Himawan sebagai pihak yang paham tentang kebutuhan yang diinginkan dalam merancang layanan server menggunakan *Node Js* sebagai pendukung untuk aplikasi eduplace pada PT. Era Solusi data.

4. PEMBAHASAN

Dalam kegiatan penelitian di PT. Era Solusi Data sebagai *back-end developer* meliputi pembuatan model data dan struktur API yang kemudian dibuat menggunakan *Node Js*. Pengujian fungsional sistem juga dilakukan untuk memastikan API dapat digunakan.

Model Data

Model data merupakan objek yang digunakan untuk menyimpan data, berikut merupakan model yang digunakan:

1) Model Pengguna

Dalam model pengguna menyimpan berbagai data pengguna sebagaimana ditunjukkan pada tabel 1.

Tabel 1. Model Pengguna

Key	Type	Required	Trim	Min. Length
isAdmin	Boolean	true	true	-
name	String	true	true	3
email	String	true	true	3
password	String	true	true	3
photoUrl	String	true	true	-
coursePurchased	Array	-	-	-
courseOwned	Array	-	-	-
cart	Array	-	-	-

2) Model Pelajaran

Dalam model pelajaran menyimpan berbagai data pelajaran ditunjukkan pada tabel 2 berikut.

Tabel 2. Model Pelajaran

Key	Type	Required	Trim
title	String	true	-
description	String	true	-
category	Array	-	-
rating	Array	-	-
document	Array	-	-
posterUrl	String	-	true
price	Number	-	-

3) Model Transaksi

Dalam model transaksi menyimpan berbagai data transaksi seperti ditunjukkan pada tabel 3 berikut ini.

Tabel 1. Model Transaksi

Key	Type	Required	Trim
courseId	String	true	true
accountId	String	true	true
price	Number	true	-
status	Boolean	true	-

4) Model File

Dalam model file menyimpan berbagai data file pengguna ditunjukkan pada tabel 4 dibawah ini.

Tabel 2. Model File

Key	Type	Required	Trim
meta_data	Object	-	-

Struktur API

Pada penelitian ini, sumberdaya diklasifikasi secara terpola berdasarkan *route*, *path*, dan *query string*. Sumberdaya yang telah diklasifikasi berdasarkan pola adalah sebagai berikut:

1) Route User

Dalam *route user* terdapat pola *path* dan *query string* sebagai berikut:

- Titik akses */users* yang digunakan untuk mendapatkan seluruh data publik pengguna.
- Titik akses */users/:id* yang digunakan untuk mendapatkan data publik pengguna berdasarkan *id* yang diberikan.
- Titik akses */users/login* yang digunakan untuk memverifikasi data *login* yang dikirimkan menggunakan metode *post*.
- Titik akses */users/add* yang digunakan untuk menambahkan pengguna baru berdasarkan data pengguna yang dikirimkan menggunakan metode *post*.
- Titik akses */users/update/:id* yang digunakan untuk memperbaharui data pengguna yang dikirimkan menggunakan metode *post* berdasarkan *id* yang diberikan.
- Titik akses */users/delete/:id* yang digunakan untuk menghapus data pengguna berdasarkan *id* yang diberikan.
- Titik akses */users/upload/image* yang digunakan untuk mengunggah gambar pengguna yang dikirimkan menggunakan metode *post*.

- h) Titik akses `/users/image/:refName` yang digunakan untuk mendapatkan gambar pengguna berdasarkan `refName` yang diberikan.

2) Route Course

Dalam *route course* terdapat pola *path* dan *query string* sebagai berikut:

- Titik akses `/course` yang digunakan untuk mendapatkan seluruh data pelajaran.
- Titik akses `/course/add` yang digunakan untuk menambahkan pelajaran berdasarkan data pelajaran yang dikirimkan menggunakan metode *post*.
- Titik akses `/course/update/:id` yang digunakan untuk memperbaharui data pelajaran yang dikirimkan menggunakan metode *post* berdasarkan `id` yang diberikan.
- Titik akses `/course/delete/:id` yang digunakan untuk menghapus data pelajaran berdasarkan `id` yang diberikan.
- Titik akses `/course/upload/image` yang digunakan untuk mengunggah gambar pelajaran yang dikirimkan menggunakan metode *post*.
- Titik akses `/course/image/:refName` yang digunakan untuk mendapatkan gambar pelajaran berdasarkan `refName` yang diberikan.
- Titik akses `/course/upload/document` yang digunakan untuk mengunggah dokumen pelajaran yang dikirimkan menggunakan metode *post*.
- Titik akses `/course/document/:refName` yang digunakan untuk mendapatkan dokumen pelajaran berdasarkan `refName` yang diberikan.

3) Route Transaction

Dalam *route transaction* terdapat pola *path* dan *query string* sebagai berikut:

- Titik akses `/transaction` yang digunakan untuk mendapatkan seluruh data transaksi.
- Titik akses `/transaction/add` yang digunakan untuk menambahkan transaksi berdasarkan data transaksi yang dikirimkan menggunakan metode *post*.
- Titik akses `/transaction/update/:id` yang digunakan untuk memperbaharui data transaksi yang dikirimkan menggunakan metode *post* berdasarkan `id` yang diberikan.

- Titik akses `/transaction/delete/:id` yang digunakan untuk menghapus data transaksi berdasarkan `id` yang diberikan.

Pengujian Fungsional Sistem

Pengujian fungsional sistem REST API dilakukan secara keseluruhan. Pengujian dilakukan berdasarkan pola *route* dengan melakukan uji coba pengaksesan API, sehingga mendapatkan hasil pengujian sebagai berikut:

1) Route User

Dalam pengujian *route user* terdapat pola *path* dan *query string* sebagai berikut:

- Pengujian titik akses `/users` untuk mendapatkan seluruh data publik pengguna, sehingga mendapatkan hasil yang dapat dilihat pada gambar 1 berikut.

```
1 {
2   {
3     "coursePurchased": [
4       "5f865895bc0d10263440ae16",
5       "5f86672dbc0d10263440ae18",
6       "5f86672dbc0d10263440ae17",
7       "5f8fdb7e4af5f01e1815ad92"
8     ],
9     "courseOwned": [],
10    "cart": [
11      "5f62e8fa5671f93a79a050a1"
12    ],
13    "_id": "5f640e111302dc1bed68961e",
14    "isAdmin": false,
15    "name": "user",
16    "email": "user"
```

Gambar 1. Pengujian Titik Akses User

- Pengujian titik akses `/users/:id` untuk mendapatkan seluruh data 91ublic pengguna berdasarkan `id` 5f8d3604b7919c111096ebfd, sehingga mendapatkan hasil yang dapat dilihat pada gambar 2.

```
1 {
2   {
3     "coursePurchased": [
4       "5f8d3763b7919c111096ebfe",
5       "5f8d3838b7919c111096ebff",
6       "5f8fe2d34af5f01e1815ad98",
7       "5f8fe3964af5f01e1815ad99",
8       "5f8fe3ca4af5f01e1815ad9a",
9       "5f8fe4eb4af5f01e1815ad9c",
10      "5f8fe4eb4af5f01e1815ad9b",
11      "5f8fe5bd4af5f01e1815ad9d",
12      "5f8fe6094af5f01e1815ad9f",
13      "5f8fe6394af5f01e1815ada0",
14      "5f9d81d67f661225607bf77d"
15    ],
16    "courseOwned": [],
17    "cart": [
18      "5f62e8d05671f93a79a0509e",
19      "5f62e8fa5671f93a79a050a1"
20    ],
21    "_id": "5f8d3604b7919c111096ebfd"
```

Gambar 2. Pengujian Titik Akses User Id

- Pengujian titik akses `/users/login` yang digunakan untuk memverifikasi data login yang dikirimkan menggunakan metode *post*, sehingga mendapatkan hasil yang dapat dilihat pada gambar 3.


```
1 {
2   "status": true,
3   "_id": "5f8d3604b7919c11096ebfd",
4   "isAdmin": false,
5   "name": "Nana Syahrma",
6   "email": "nana@mail.com",
7   "photoUrl": "IMG-1603091222347.jpg",
8   "coursePurchased": [
9     "5f8d3763b7919c11096ebfe",
10    "5f8d3838b7919c11096ebff",
11    "5f8fe2d34af5f01e1815ad98",
12    "5f8fe3964af5f01e1815ad99",
```

Gambar 3. Pengujian Titik Akses User Login

- d) Pengujian titik akses `/users/add` yang digunakan untuk menambahkan pengguna baru berdasarkan data pengguna yang dikirimkan menggunakan metode `post`, sehingga mendapatkan hasil yang dapat dilihat pada gambar 4.

```
1 {
2   "isAdmin": false,
3   "name": "Galang Pratama",
4   "email": "galang.sukma21@mail.com",
5   "password": "galang",
6   "photoUrl": "/src/photo/dummy.jpg",
7   "coursePurchased": [],
8   "courseOwned": [],
9   "cart": []
```

Gambar 4. Pengujian Titik Akses User Add

- e) Pengujian titik akses `/users/update/:id` yang digunakan untuk memperbaharui data pengguna yang dikirimkan menggunakan metode `post` berdasarkan `id 5f914f6f6d8d6b2264a34b5e`, sehingga mendapatkan hasil yang dapat dilihat pada gambar 5.

```
1 {
2   "isAdmin": false,
3   "name": "Galang Pratama Sukma Putra",
4   "email": "galang.sukma21@mail.com",
5   "password": "galang",
6   "photoUrl": "/src/photo/dummy.jpg",
7   "coursePurchased": [],
8   "courseOwned": [],
9   "cart": []
```

Gambar 5. Pengujian Titik Akses User Update

- e) Pengujian titik akses `/users/delete/:id` yang digunakan untuk menghapus data pengguna berdasarkan `id 5f914f6f6d8d6b2264a34b5e`, sehingga mendapatkan hasil yang dapat dilihat pada gambar 6.

```
Body  Cookies  Headers (8)  Test Results
Pretty  Raw  Preview  Visualize  JSON
1  "User deleted!"
```

Gambar 6. Pengujian Titik Akses User Delete

- f) Pengujian titik akses `/users/upload/image` yang digunakan untuk mengunggah gambar

pengguna yang dikirimkan menggunakan metode `post` tidak dapat diuji melalui Postman, namun berhasil dilakukan melalui aplikasi Eduplace.

- g) Pengujian titik akses `/users/image/:refName` yang digunakan untuk mendapatkan gambar pengguna berdasarkan `refName logo-akprind.png`, sehingga mendapatkan hasil yang dapat dilihat pada gambar 7.



Gambar 7. Pengujian Titik Akses Image Refname

2) Route Course

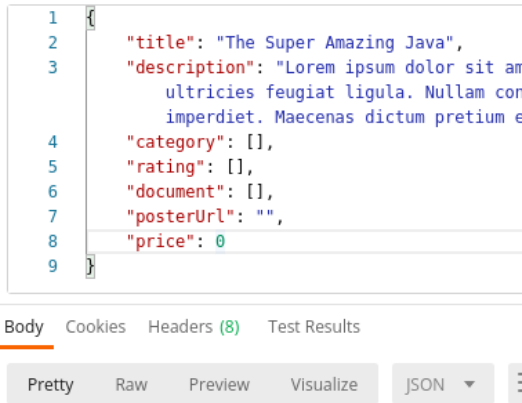
Dalam pengujian `route course` terdapat pola `path` dan `query string` sebagai berikut:

- a) Pengujian titik akses `/course` yang digunakan untuk mendapatkan seluruh data pelajaran, sehingga mendapatkan hasil yang dapat dilihat pada gambar 8.

```
4 {
5   "category": [
6     {
7       "_id": "5f895329cd7ca7723222a35d",
8       "label": "Programming",
9       "value": "Programming"
10    },
11  ],
12  "title": "The Super Amazing Java New",
13  "description": "Lorem ipsum dolor sit amet, consectetur",
14  "rating": [
15    {
16      "_id": "5f62ea096140c33b56259e3f",
17      "rate": 5,
18      "userId": "5f59a1d23bd4f72ccffdd7a07"
```

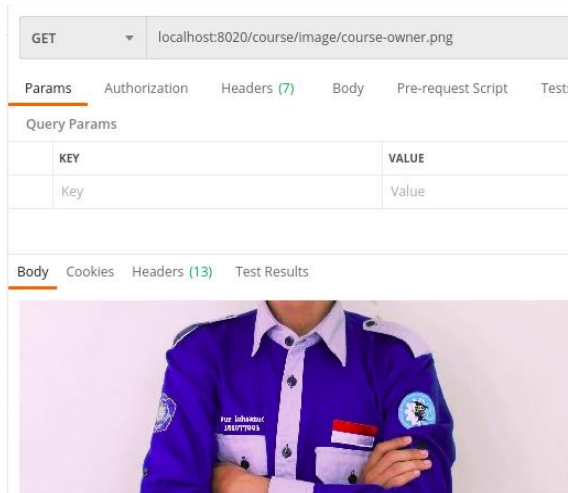
Gambar 8. Pengujian Titik Akses Course

- b) Pengujian titik akses `/course/add` yang digunakan untuk menambahkan pelajaran berdasarkan data pelajaran yang dikirimkan menggunakan metode `post`, sehingga mendapatkan hasil yang dapat dilihat pada gambar 9.



Gambar 9. Pengujian Titik Akses Course Add

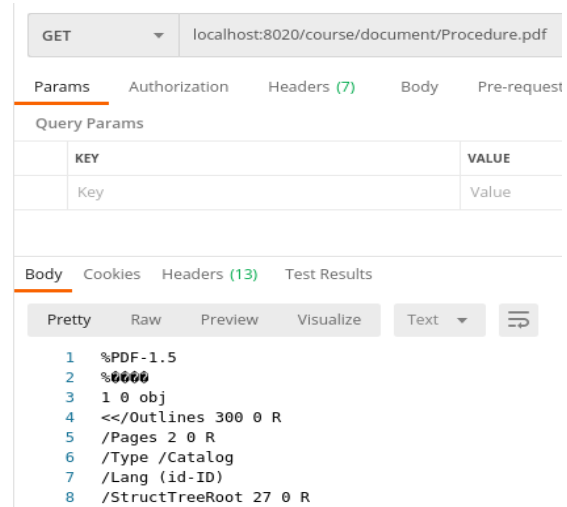
- c) Pengujian titik akses `/course/upload/image` yang digunakan untuk mengunggah gambar pelajaran yang dikirimkan menggunakan metode `post` tidak dapat diuji melalui Postman, namun berhasil dilakukan melalui aplikasi Eduplace.
- d) Pengujian titik akses `/course/image/:refName` yang digunakan untuk mendapatkan gambar pelajaran berdasarkan `refName` `course-owner.png`, sehingga mendapatkan hasil yang dapat dilihat pada gambar 10.



Gambar10. Pengujian Titik Akses Course Image Refname

- e) Pengujian titik akses `/course/upload/document` yang digunakan untuk mengunggah dokumen pelajaran yang dikirimkan menggunakan metode `post` tidak dapat diuji melalui Postman, namun berhasil dilakukan melalui aplikasi Eduplace.
- f) Pengujian titik akses `/course/document/:refName` yang digunakan untuk mendapatkan dokumen pelajaran berdasarkan `refName` `course-owner.png`,

sehingga mendapatkan hasil yang dapat dilihat pada gambar 11.

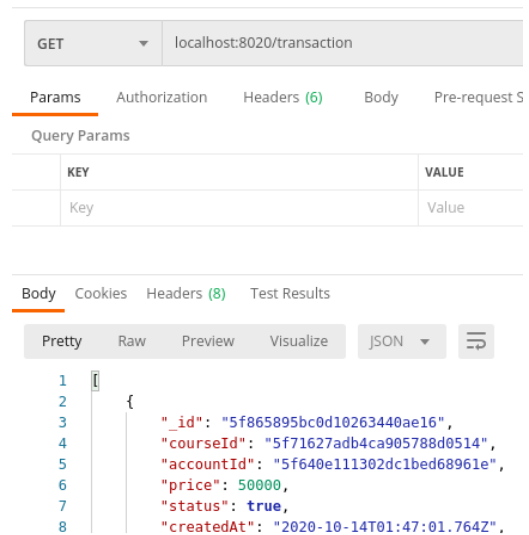


Gambar 11. Pengujian Titik Akses Course Document Refname

3) Route Transaction

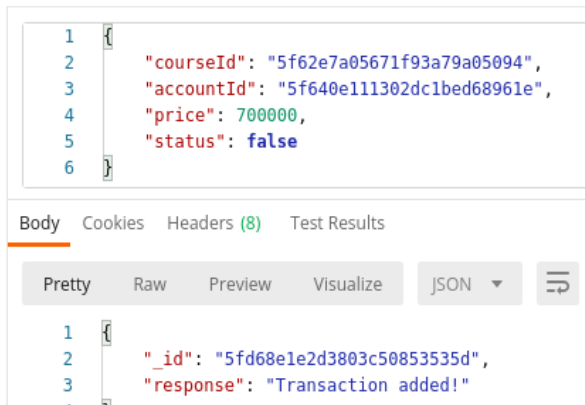
Dalam pengujian `route user` terdapat pola `path` dan `query string` sebagai berikut:

- a) Pengujian titik akses `/transaction` yang digunakan untuk mendapatkan seluruh data transaksi, sehingga mendapatkan hasil yang dapat dilihat pada gambar 12.



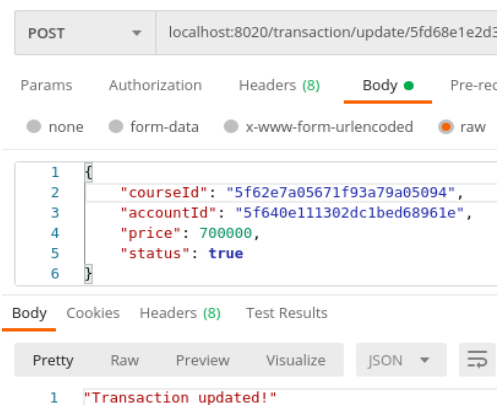
Gambar 12. Pengujian Titik Akses Transaction

- b) Pengujian titik akses `/transaction/add` yang digunakan untuk menambahkan transaksi berdasarkan data transaksi yang dikirimkan menggunakan metode `post`, sehingga mendapatkan hasil yang dapat dilihat pada gambar 13.



Gambar 13. Pengujian Titik Akses Transaction Add

- c) Pengujian titik akses `/transaction/update/:id` yang digunakan untuk memperbaharui data transaksi yang dikirimkan menggunakan metode `post` berdasarkan `id` `"5fd68e1e2d3803c50853535d"`, sehingga mendapatkan hasil yang dapat dilihat pada gambar 14.



Gambar 14. Pengujian Titik Akses Transaction Update

5. KESIMPULAN

Perancangan REST API dalam kegiatan penelitian berhasil diimplementasikan pada sistem Eduplace. Dengan menggunakan gaya arsitektur REST menjadikan sistem sederhana dan ringkas, sehingga dapat memudahkan *developer* untuk menambah maupun meningkatkan sistem dengan mudah. Sistem mampu mengakomodasi seluruh kebutuhan *back-end* dari aplikasi Eduplace. Berdasarkan pengujian yang telah dilakukan, sistem berhasil menjalankan seluruh API dengan baik dan cepat.

Hasil perancangan ini dibutuhkan pengembangan lebih lanjut sehingga menjadi sistem yang lebih *solid* dan mampu memenuhi kebutuhan pertukaran data antara *front-end* dengan *back-end* dan server. Dalam pengembangan REST API akan lebih baik apabila API dapat dibuat dengan lebih

terstruktur dan menggunakan format yang konsisten sehingga dapat memudahkan *developer* lain dalam menggunakan API.

6. UCAPAN TERIMAKASIH

Penelitian ini dapat terlaksana dengan baik berkat bantuan dari berbagai pihak, untuk itu penulis mengucapkan terimakasih kepada PT. Era Solusi data Institut Sains & Teknologi AKPRIND Yogyakarta, khususnya Lembaga Penelitian dan Pengabdian pada Masyarakat, Program Studi Informatika atas kerjasama dan *supportnya*, sehingga penelitian ini dapat diselesaikan tepat waktu

DAFTAR PUSTAKA

- Arhandi, P. P. (2016). Pengembangan Sistem Informasi Perijinan Tenaga Kesehatan Dengan Menggunakan Metode Back-end dan Front-end. *Jurnal Teknologi Informasi*, 7, 39-48.
- Bina, N. S., & Ramadhani, R. (2021). GOOGLE CLASSROOM AS THE LATEST LEARNING MEDIA IN THE COVID-19 PANDEMIC PERIOD. *JCES (Journal of Character Education Society)*, 4(3), 104-113. doi:https://doi.org/10.31764/jces.v3i1.4497
- Firyal, R. A. (2020). Pembelajaran Daring dan Kebijakan New Normal Pemerintah. *LawArXiv*, 1-7. doi:10.31228/osf.io/yt6qs
- Hidayah, A. A., Adawiyah, R. A., & Mahanani, P. A. (2020, September). Efektivitas Pembelajaran Daring di Masa Pandemi Covid-19. *Sosial : Jurnal Penelitian Ilmu-Ilmu Sosial*, 21(2), 53-56.
- iiiep. (2020, March 25). *Three ways to plan for equity during the coronavirus school closures*. Diambil kembali dari iiiep: <http://www.iiiep.unesco.org/en/three-ways-plan-equity-during-coronavirus-school-closures-13365>
- Pranata, B. A., Hijriani, A., & Junaidi, A. (2018). Perancangan Application Programming Interface (API) Berbasis Web Menggunakan Gaya Arsitektur Representational State Transfer (REST) Untuk Pengembangan Sistem Informasi Administrasi Pasien Klinik Perawatan Kulit. 6, 33-42.
- Puspitorini, F. (2020, Juli). Strategi Pembelajaran Di Perguruan Tinggi Pada Masa Pandemi Covid-19. *Jurnal Kajian Ilmiah (JKI)*, 99-106. Diambil kembali dari <http://ejurnal.ubharajaya.ac.id/index.php/JKI/article/view/274/232>

- Putra, B. P., & Susetyo, Y. A. (2020). Implementasi API Master Store Menggunakan FLASK, REST, dan ORM di PT XYZ. *S ISTEMASI : Jurnal Sistem Informasi*, 9, 543-556.
- Shereen, M. A., Khan, S., Kazmi, A., Bashir, N., & Siddique, R. (2020). COVID-19 infection: Origin, transmission, and characteristics of human coronaviruses. *Journal of Advanced Research*, 91-98. doi:<https://doi.org/10.1016/j.jare.2020.03.005>
- Wikipedia. (2019, Agustus 12). *Organisasi Pendidikan, Keilmuan, dan Kebudayaan Perserikatan Bangsa-Bangsa*. Diambil kembali dari Wikipedia: https://id.wikipedia.org/wiki/Organisasi_Pendidikan,_Keilmuan,_dan_Kebudayaan_Perserikatan_Bangsa-Bangsa