

LAPORAN PROYEK UTS

**“ Layanan Web Volunteer Pengembangan Desa Tertinggal (PDT) Unit
PMKL Polstat STIS”**



Dosen Pengampu:

Ibnu Santoso, S.S.T., M.T

Disusun oleh:

Blessy Munthia Purba (222212537) (3SI2)

Kamis, 14 November 2024

**PROGRAM STUDI D-IV KOMPUTASI STATISTIK
POLITEKNIK STATISTIKA STIS
2024**

Tata Tertib Ujian:

1. Waktu pengerjaan dimulai dari dikirimkannya soal ini hingga akhir sesuai dengan jadwal yang ditetapkan BAAK.
2. Proyek dikumpulkan di Google Classroom. Anda tetap harus datang ke BAAK untuk menandatangani daftar hadir ujian pada waktu ujian.
3. Dalam mengerjakan soal, tidak diperbolehkan bekerja sama, memberikan, atau menerima bantuan dalam bentuk apapun dengan cara apapun. Apabila Anda terbukti melakukan salah satu hal tersebut maka akan diberikan nilai 0 untuk semua pihak yang terlibat.

Soal:

Buatlah layanan web (web service) dengan ketentuan sebagai berikut:

1. Layanan web berhubungan dengan proses bisnis di Polstat-STIS atau di BPS. Setiap mahasiswa tidak boleh mengerjakan satu topik atau proses bisnis yang sama. (Buat sheet topik untuk kelas 3SI1 dan 3SI2 dan bagikan linknya ke dosen lewat stream classroom)
2. Layanan web yang dibuat harus memuat manajemen pengguna yang terdiri dari registrasi, login, profil, edit profil, ganti password dan hapus akun.
3. Layanan web harus memuat bisnis proses utama sesuai topik yang dipilih. Misal: layanan perpustakaan harus ada manajemen koleksi buku dan pencarian buku.
4. Setiap endpoint layanan (kecuali login dan registrasi) harus terotentikasi sesuai dengan pengguna yang login menggunakan token-based authentication.
5. Layanan web berupa RESTful atau GraphQL dan dibuat dengan bahasa pemrograman Java.

Pengumpulan Pekerjaan:

1. Buat laporan yang memuat penjelasan proses bisnis, penjelasan kode program layanan utama, dokumentasi atau cara penggunaan setiap endpoint, contoh pengujian dan hasilnya, serta link kode program (<https://git.stis.ac.id/>).
2. Kumpulkan laporan dalam format PDF pada Google Classroom yang disediakan.

Gambaran Umum Proses Bisnis Proyek:

Saya mengambil topik layanan web service **Volunteer Pengembangan Desa Tertinggal**, disingkat PDT. Pengembangan Desa Tertinggal merupakan salah satu program kerja unit/komunitas PMKL, organisasi yang berada di bawah BEM Polstat STIS. Yang menjadi **latar belakang** pembuatan web service ini adalah untuk mempermudah mahasiswa yang ingin mendaftarkan diri menjadi volunteer dan pengurus proker ini dapat melihat siapa saja yang sudah mendaftar menjadi volunteer. Role yang tersedia di web ini ada 3, yaitu:

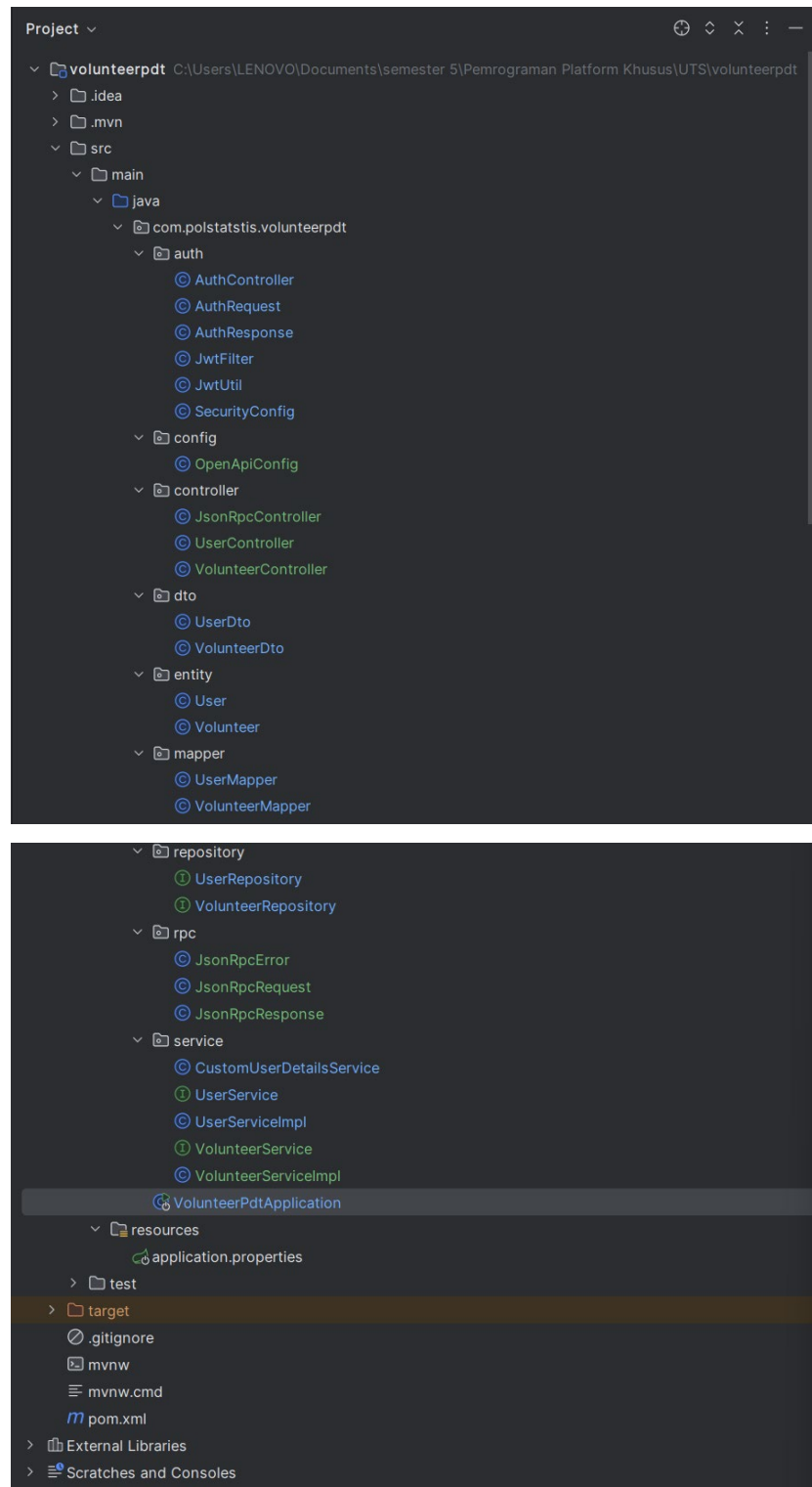
1. mahasiswa yang mendaftar menjadi volunteer, selain registrasi dan login juga bisa melakukan edit data profilnya seperti ganti password, kelas, usia, dll,
2. pengurus PDT yang bisa melakukan penghapusan status dan akun volunteer (masih aktif atau tidak), serta bisa melakukan seluruh endpoint yang tersedia,
3. pengunjung, mahasiswa yang hanya ingin tahu informasi tentang daftar volunteer PDT yang ada.

Alur proses bisnis utama dalam layanan web service ini adalah sebagai berikut:

- Mahasiswa yang sebagai pengurus melakukan registrasi dengan role Pengurus PDT,
- Mahasiswa yang berminat menjadi volunteer PDT dapat registrasi,
- Mahasiswa yang ingin berkunjung pada layanan web juga dapat registrasi dengan role sebagai pengunjung,
- Pengurus PDT dapat menambahkan Volunteer (yang lulus seleksi),
- Mahasiswa melakukan login sesuai akun yang didaftar,
- Mahasiswa baik Pengurus PDT, Pengunjung, maupun Volunteer dapat melihat data profil saat sudah login dan mengedit data mereka,
- Mahasiswa sebagai role Pengurus PDT bisa melakukan tambah dan edit volunteer,
- Mahasiswa baik Pengurus PDT, Pengunjung, maupun Volunteer dapat menghapus akun miliknya.
- Pengurus PDT dapat menghapus akun volunteer jika statusnya sebagai volunteer sudah tidak aktif atau sudah keluar,
- Mahasiswa, Pengurus PDT, maupun Pengunjung dapat melihat seluruh volunteer dan melakukan pencarian volunteer yang ingin diketahui.

Penjelasan Singkat Kode Program:

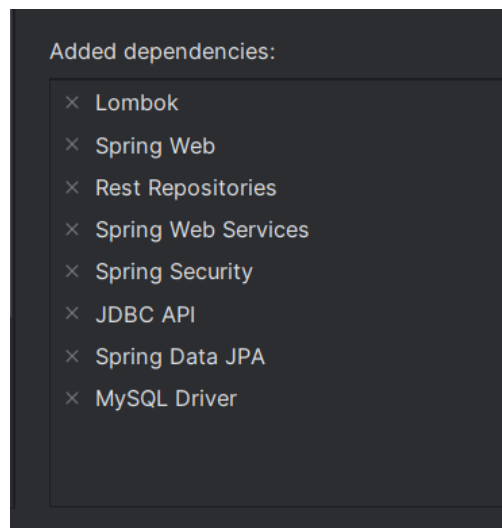
Berikut packages dari source code volunteer_pdt:



Package auth berisi berbagai hal yang berkaitan dengan autentikasi meliputi JWT serta pengaturan web security. Package config merupakan akses untuk dokumentasi API ke swagger.

Package controller berisi sekumpulan Controller yang mengatur path secara manual untuk melakukan request api. Package dto berisi UserDto dan VolunteerDto yang berguna untuk pertukaran data. Package entity berisi struktur tabel yang akan dibuat dalam sistem ke database. Package mapper berisi UserMapper dan VolunteerMapper yang berfungsi untuk mengubah, menyambungkan, atau mentranslasikan dari entity user ke DTO atau sebaliknya. Package repository disini rata rata sudah merupakan restRepository yang sudah mengeset path untuk melakukan request. Package repository ini umumnya digunakan untuk melakukan save ke database atau menemukan entity berdasarkan atribut variabelnya. Package service berisi layanan-layanan yang dibutuhkan dalam sistem ini.

Berikut dependencies yang dipakai:



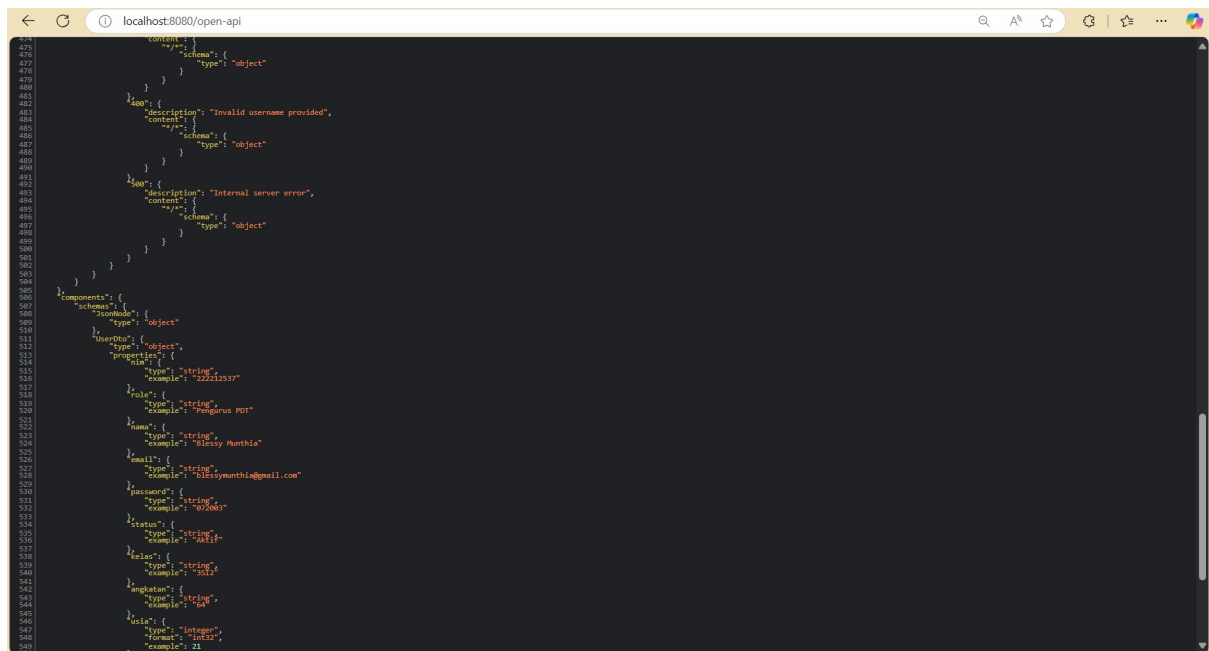
Untuk kode lengkap tersedia di link github berikut:

https://git.stis.ac.id/222212537/uts-ppk_222212537_3SI2_Blessy-Munthia-Purba

Dokumentasi API:

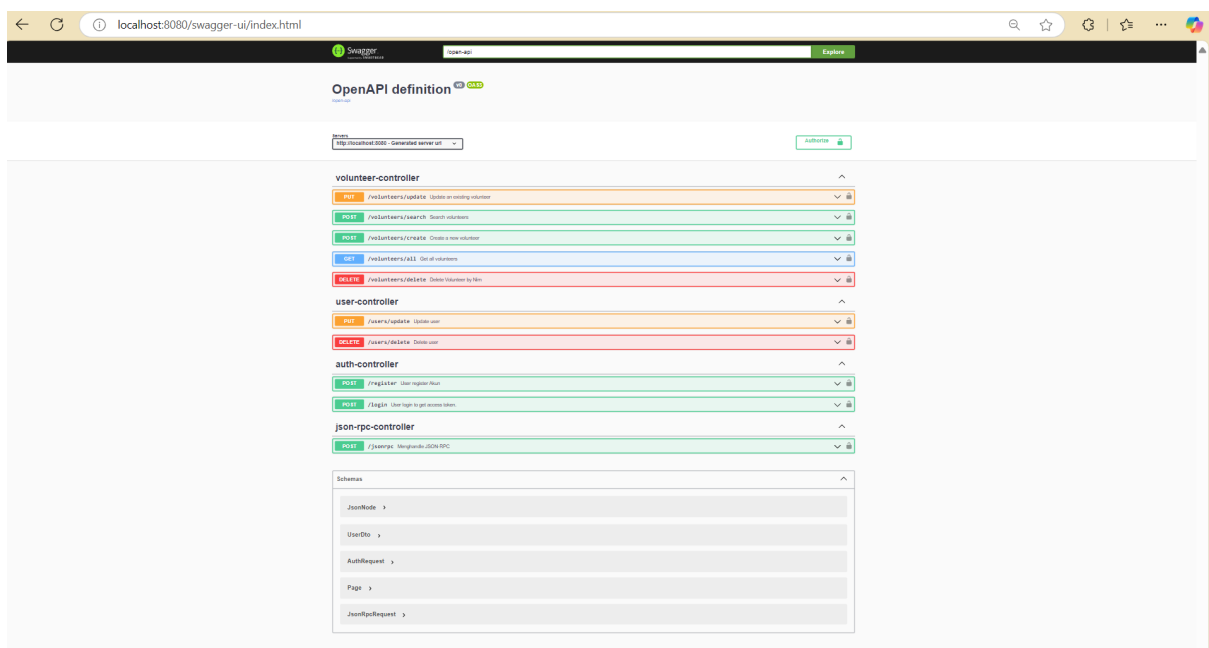
Berikut link untuk melihat dokumentasi:

1. localhost:8080/open-api



```
474      "content": {
475        "schema": {
476          "type": "object"
477        }
478      }
479    }
480  },
481  {
482    "description": "Invalid username provided",
483    "content": {
484      "schema": {
485        "type": "object"
486      }
487    }
488  },
489  {
490    "description": "Internal server error",
491    "content": {
492      "schema": {
493        "type": "object"
494      }
495    }
496  }
497 }
498 }
499 }
500 }
501 }
502 }
503 }
504 }
505 }
506 }
507 }
508 }
509 }
510 }
511 }
512 }
513 }
514 }
515 }
516 }
517 }
518 }
519 }
520 }
521 }
522 }
523 }
524 }
525 }
526 }
527 }
528 }
529 }
530 }
531 }
532 }
533 }
534 }
535 }
536 }
537 }
538 }
539 }
540 }
541 }
542 }
543 }
544 }
545 }
546 }
547 }
548 }
549 }
550 }
```

2. <http://localhost:8080/swagger-ui/index.html>



Pada dokumentasi juga disediakan schema di setiap request method-method nya sehingga dapat lebih mudah dicerna dan dimengerti. Berikut akan ditampilkan:

1. updateVolunteer

volunteer-controller

PUT /volunteers/update Update an existing volunteer

Updates an existing volunteer's details

Parameters Try it out

Name	Description
requestId * required	Example : request-67890
string (query)	<input type="text" value="request-67890"/>

Request body required application/json

Example Value | **Schema**

```
{
  "id": "22212537",
  "name": "Updated Name",
  "status": "Inactive"
}
```

Responses

Code	Description	Links
200	Volunteer updated successfully	No links

Media type application/json

Controls Accept header.

Example Value | **Schema**

```
{
  "jsonrpc": "2.0",
  "result": "updated",
  "id": "request-67890"
}
```

| 400 | Invalid request data | No links |

Media type application/json

Example Value | **Schema**

```
{
  "jsonrpc": "2.0",
  "error": {
    "code": -32602,
    "message": "Invalid parameters"
  },
  "id": "request-67890"
}
```

2. searchVolunteers

POST

/volunteers/search

Search volunteers

Searches for volunteers based on a search term.

Parameters

Try it out

Name	Description
requestId <small>* required</small>	Example : request-55667
string (query)	<input type="text" value="request-55667"/>

Request body required

application/json

Example Value | Schema

```
{
  "search term": "Blessy"
}
```

Responses

Code	Description	Links
200	Successfully retrieved search results	No links
400	Invalid search parameters	No links
500	Internal server error	No links

Media type

application/json

Controls Accept header.

Example Value | Schema

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "id": "222112537",
      "name": "Blessy Muthia",
      "status": "Active"
    }
  ],
  "id": "request-55667"
}
```

Media type

application/json

Example Value | Schema

```
{
  "jsonrpc": "2.0",
  "error": {
    "code": -32602,
    "message": "Invalid parameters"
  },
  "id": "request-55667"
}
```

Media type

application/json

Example Value | Schema

```
{
  "jsonrpc": "2.0",
  "error": {
    "code": -32603,
    "message": "Internal server error"
  },
  "id": "request-55667"
}
```


3. createVolunteer

POST

/volunteers/create

Create a new volunteer

Creates a new volunteer with provided parameters

Parameters

Try it out

Name	Description
requestid <small>* required</small> string (query)	Example : request-12345 <input type="text" value="request-12345"/>

Request body required

application/json

Example Value | Schema

```
{
  "nid": "222212537",
  "name": "Blessy Munthia",
  "status": "Aktif"
}
```

Responses

Code	Description	Links
200	Volunteer created successfully <div><div>Media type</div><div>application/json</div><div>Controls Accept header.</div><div>Example Value Schema</div><div><pre>{ "jsonrpc": "2.0", "result": "created", "id": "request-12345" }</pre></div></div>	No links
400	Invalid request data <div><div>Media type</div><div>application/json</div><div>Example Value Schema</div><div><pre>{ "jsonrpc": "2.0", "error": { "code": -32602, "message": "Invalid parameters" }, "id": "request-12345" }</pre></div></div>	No links

4. getVolunteers

GET

/volunteers/all

Get all volunteers

Retrieves a list of all volunteers.

Parameters

Try it out

Name	Description
requestId <small>required</small> string (query)	<i>Example</i> : request-33445 <input type="text" value="request-33445"/>

Responses

Code	Description	Links
200	Successfully retrieved the list of volunteers <div>Media type <div>application/json</div><div>Controls Accept header.</div><div>Example Value Schema</div><pre>{ "jsonrpc": "2.0", "result": [{ "nim": "222212537", "nama": "Blessy Munthia", "status": "Aktif" }, { "nim": "222212538", "nama": "John Doe", "status": "Inactive" }], "id": "request-33445" }</pre></div>	No links
500	Internal server error <div>Media type <div>application/json</div><div>Example Value Schema</div><pre>{ "jsonrpc": "2.0", "error": { "code": -32603, "message": "Internal server error" }, "id": "request-33445" }</pre></div>	No links

5. deleteVolunteer

DELETE

/volunteers/delete

Delete Volunteer by Nim

Deletes a volunteer based on the provided Nim.

Parameters

Try it out

Name	Description
requestId <small>* required</small>	<i>Example : request-11223</i>
string (query)	<input type="text" value="request-11223"/>

Request body required

application/json

Example Value

Schema

```
{
  "nim": "222212537"
}
```

Responses

Code	Description	Links
200	Volunteer deleted successfully	No links
<div>Media type</div> <div>application/json</div> <div>Controls Accept header.</div> <div>Example Value</div> <div>Schema</div> <div><pre>{ "jsonrpc": "2.0", "result": "deleted", "id": "request-11223" }</pre></div>		
400	Invalid request data	No links
<div>Media type</div> <div>application/json</div> <div>Example Value</div> <div>Schema</div> <div><pre>{ "jsonrpc": "2.0", "error": { "code": -32602, "message": "Invalid parameters" }, "id": "request-11223" }</pre></div>		

6. updateUser

PUT

/users/update

Update user

Updates the details of an existing user.

Parameters

Try it out

Name	Description
requestId <small>* required</small>	<i>Example : request-12345</i>
string (query)	<input type="text" value="request-12345"/>

Request body required

application/json

Example Value

Schema

```
{
  "nim": "222212537",
  "email": "blessymunthia@gmail.com",
  "role": "Dangerous POT",
  "name": "Blessy Munthia",
  "status": "Active",
  "holist": "Holist",
  "negotiate": "ed",
  "usia": 21
}
```

Responses		
Code	Description	Links
200	Successfully updated the user	No links
Media type <input type="text" value="application/json"/> Controls Accept header. Example Value Schema		
<pre>{ "jsonrpc": "2.0", "result": "user updated", "id": "request-12345" }</pre>		
400	Invalid user parameters	No links
Media type <input type="text" value="application/json"/> Example Value Schema		
<pre>{ "jsonrpc": "2.0", "error": { "code": -32602, "message": "Invalid parameters" }, "id": "request-12345" }</pre>		
500	Internal server error	No links
Media type <input type="text" value="application/json"/> Example Value Schema		
<pre>{ "jsonrpc": "2.0", "error": { "code": -32603, "message": "Internal server error" }, "id": "request-12345" }</pre>		

7. deleteUser

DELETE /users/delete Delete user		
Deletes an existing user based on the provided username.		
Parameters		Try it out
Name	Description	
requestId * required string (query)	Example: request-67890	
<input type="text" value="request-67890"/>		
Request body * required		<input type="text" value="application/json"/>
Example Value Schema		
<pre>{ "username": "blessy_munthia" }</pre>		
Responses		
Code	Description	Links
200	Successfully deleted the user	No links
Media type <input type="text" value="application/json"/> Controls Accept header. Example Value Schema		
<pre>{ "jsonrpc": "2.0", "result": "user deleted", "id": "request-67890" }</pre>		

400	Invalid username provided	No links
<div>Media type</div> <div>application/json</div> <div>Example Value Schema</div> <pre>{ "jsonrpc": "2.0", "error": { "code": -32002, "message": "Invalid parameters" }, "id": "request-67890" }</pre>		
500	Internal server error	No links
<div>Media type</div> <div>application/json</div> <div>Example Value Schema</div> <pre>{ "jsonrpc": "2.0", "error": { "code": -32003, "message": "Internal server error" }, "id": "request-67890" }</pre>		

8. register

<div>POST /register User register Akun</div> <div>Try it out</div>		
Parameters		
No parameters		
Request body <small>required</small>		
<div>application/json</div> <div>Example Value Schema</div> <pre>{ "nim": "222212537", "role": "Pengurus POT", "nama": "Blessy Munthia", "email": "blessymunthia@gmail.com", "password": "072003", "status": "Aktif", "kelas": "3512", "angkatan": "64", "usia": 21 }</pre>		
Responses		
Code	Description	Links
200	Register succesful.	No links
<div>Media type</div> <div>application/json</div> <div>Controls Accept header.</div> <div>Example Value Schema</div> <pre>{ "nim": "222212537", "role": "Pengurus POT", "nama": "Blessy Munthia", "email": "blessymunthia@gmail.com", "password": "072003", "status": "Aktif", "kelas": "3512", "angkatan": "64", "usia": 21 }</pre>		
401	Invalid credentials	No links

9. login

POST /login User login to get access token.

Parameters

Try it out

No parameters

Request body *required*

application/json

Example Value | Schema

```
{  "nim": "string",  "password": "string"}
```

Responses

Code	Description	Links
200	Nim and access token <div>Media type<div>application/json</div>Controls Accept header.<div>Example Value Schema<pre>{ "size": 0, "number": 0, "first": true, "maxResults": 0, "firstResult": 0}</pre></div></div>	No links
401	Invalid credentials	No links

10. jsonrpc

json-rpc-controller

POST /jsonrpc Menghandle JSON-RPC

Endpoint to handle JSON-RPC requests for book operations. Supports methods: createVolunteer, getVolunteer, searchVolunteer.

Parameters

Try it out

No parameters

Request body *required*

application/json

Example Value | Schema

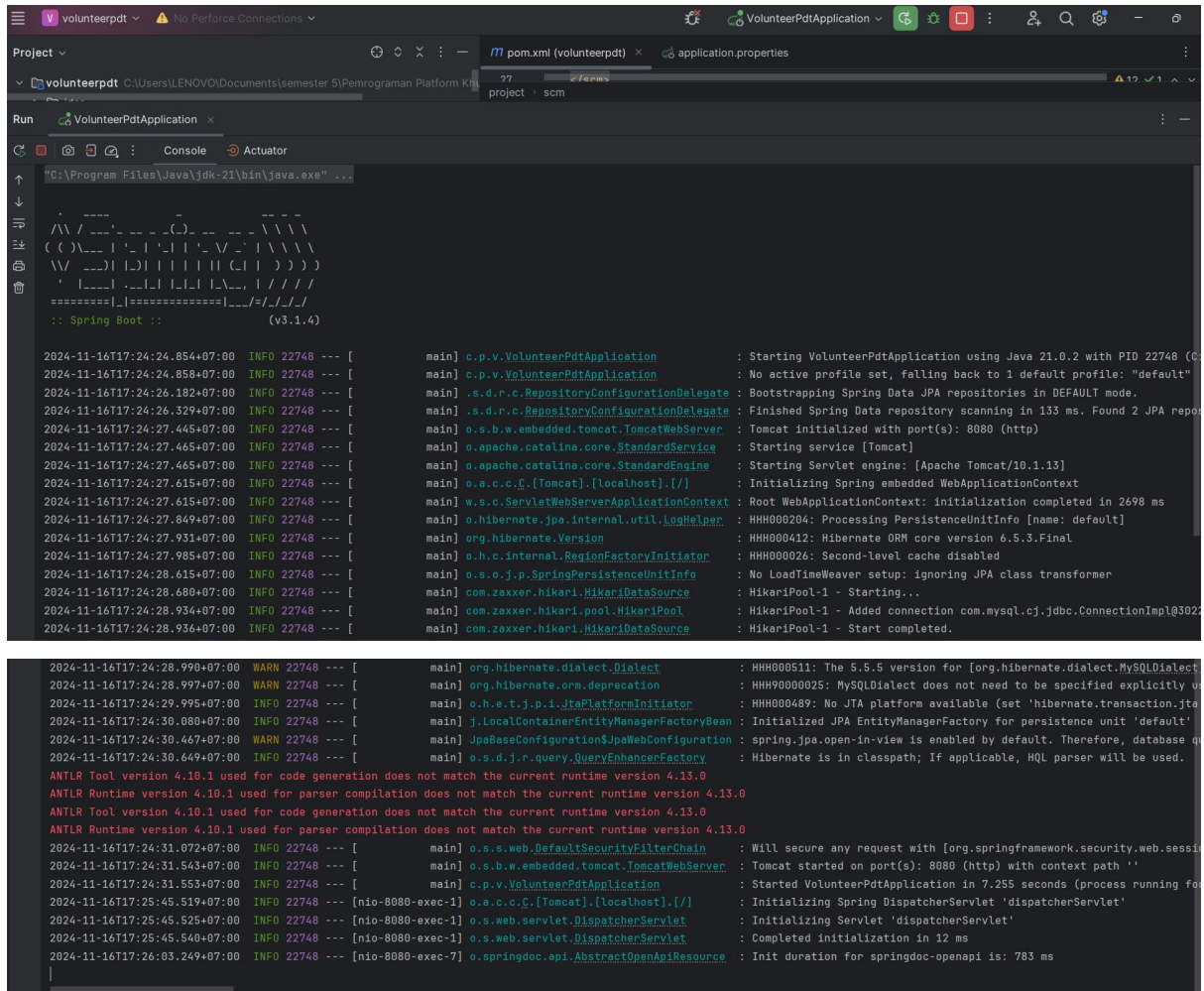
```
{  "jsonrpc": "string",  "method": "string",  "params": {},  "id": "string"}
```

Responses

Code	Description	Links
200	Successful operation <div>Media type<div>application/json</div>Controls Accept header.</div>	No links
400	Bad request invalid JSON-RPC request <div>Media type<div>application/json</div></div>	No links

Pengujian Layanan Web Service:

Menjalankan kode program:



```
Project: volunteerpdt
pom.xml (volunteerpdt)
application.properties

Run: VolunteerPdtApplication

Console:
"\"C:\\Program Files\\Java\\jdk-21\\bin\\java.exe\" ...

:: Spring Boot :: (v3.1.4)

2024-11-16T17:24:24.854+07:00 INFO 22748 --- [main] c.p.v.VolunteerPdtApplication : Starting VolunteerPdtApplication using Java 21.0.2 with PID 22748 (C:\Users\LENOVO\Documents\semester 5\Pemrograman Platform K...
2024-11-16T17:24:24.858+07:00 INFO 22748 --- [main] c.p.v.VolunteerPdtApplication : No active profile set, falling back to 1 default profile: "default"
2024-11-16T17:24:26.182+07:00 INFO 22748 --- [main] s.d.r.c.RepositoryConfigurationDelegate : Bootstrapping Spring Data JPA repositories in DEFAULT mode.
2024-11-16T17:24:26.329+07:00 INFO 22748 --- [main] s.d.r.c.RepositoryConfigurationDelegate : Finished Spring Data repository scanning in 133 ms. Found 2 JPA repositories
2024-11-16T17:24:27.445+07:00 INFO 22748 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 8080 (http)
2024-11-16T17:24:27.465+07:00 INFO 22748 --- [main] o.apache.catalina.core.StandardService : Starting service [Tomcat]
2024-11-16T17:24:27.465+07:00 INFO 22748 --- [main] o.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/10.1.13]
2024-11-16T17:24:27.615+07:00 INFO 22748 --- [main] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext
2024-11-16T17:24:27.615+07:00 INFO 22748 --- [main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 2698 ms
2024-11-16T17:24:27.849+07:00 INFO 22748 --- [main] o.hibernate.jpa.internal.util.LogHelper : HHH000204: Processing PersistenceUnitInfo [name: default]
2024-11-16T17:24:27.931+07:00 INFO 22748 --- [main] org.hibernate.Version : HHH000412: Hibernate ORM core version 6.5.3.Final
2024-11-16T17:24:27.985+07:00 INFO 22748 --- [main] o.h.c.internal.RegionFactoryInitiator : HHH000026: Second-level cache disabled
2024-11-16T17:24:28.615+07:00 INFO 22748 --- [main] o.s.o.j.p.SpringPersistenceUnitInfo : No LoadTimeWeaver setup: ignoring JPA class transformer
2024-11-16T17:24:28.680+07:00 INFO 22748 --- [main] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Starting...
2024-11-16T17:24:28.934+07:00 INFO 22748 --- [main] com.zaxxer.hikari.pool.HikariPool : HikariPool-1 - Added connection com.mysql.cj.jdbc.ConnectionImpl@3022
2024-11-16T17:24:28.936+07:00 INFO 22748 --- [main] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Start completed.

2024-11-16T17:24:28.990+07:00 WARN 22748 --- [main] org.hibernate.dialect.Dialect : HHH000511: The 5.5.S version for [org.hibernate.dialect.MySQLDialect]
2024-11-16T17:24:28.997+07:00 WARN 22748 --- [main] org.hibernate.orm.deprecation : HHH9000025: MySQLDialect does not need to be specified explicitly u
2024-11-16T17:24:29.995+07:00 INFO 22748 --- [main] o.h.e.t.j.p.i.JtaPlatformInitiator : HHH000489: No JTA platform available (set 'hibernate.transaction.jta
2024-11-16T17:24:30.080+07:00 INFO 22748 --- [main] j.LocalContainerEntityManagerFactoryBean : Initialized JPA EntityManagerFactory for persistence unit 'default'
2024-11-16T17:24:30.467+07:00 WARN 22748 --- [main] JpaBaseConfiguration$JpaWebConfiguration : spring.jpa.open-in-view is enabled by default. Therefore, database q
2024-11-16T17:24:30.649+07:00 INFO 22748 --- [main] o.s.d.j.r.query.QueryEnhancerFactory : Hibernate is in classpath; If applicable, HQL parser will be used.

ANTLR Tool version 4.10.1 used for code generation does not match the current runtime version 4.13.0
ANTLR Runtime version 4.10.1 used for parser compilation does not match the current runtime version 4.13.0
ANTLR Tool version 4.10.1 used for code generation does not match the current runtime version 4.13.0
ANTLR Runtime version 4.10.1 used for parser compilation does not match the current runtime version 4.13.0

2024-11-16T17:24:31.072+07:00 INFO 22748 --- [main] o.s.s.web.DefaultSecurityFilterChain : Will secure any request with [org.springframework.security.web.session
2024-11-16T17:24:31.543+07:00 INFO 22748 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 8080 (http) with context path ''
2024-11-16T17:24:31.553+07:00 INFO 22748 --- [main] c.p.v.VolunteerPdtApplication : Started VolunteerPdtApplication in 7.255 seconds (process running fo
2024-11-16T17:25:45.519+07:00 INFO 22748 --- [nio-8080-exec-1] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring DispatcherServlet 'dispatcherServlet'
2024-11-16T17:25:45.525+07:00 INFO 22748 --- [nio-8080-exec-1] o.s.web.servlet.DispatcherServlet : Initializing Servlet 'dispatcherServlet'
2024-11-16T17:25:45.540+07:00 INFO 22748 --- [nio-8080-exec-1] o.s.web.servlet.DispatcherServlet : Completed initialization in 12 ms
2024-11-16T17:26:03.249+07:00 INFO 22748 --- [nio-8080-exec-7] o.springdoc.api.AbstractOpenApiResource : Init duration for springdoc-openapi is: 783 ms
```

Daftar tabel users pada Database volunteer_pdt:

Server: 127.0.0.1 » Database: volunteer_pdt » Table: users

Showing rows 0 - 7 (8 total, Query took 0.0004 seconds.)

`SELECT * FROM `users``

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

				id	angkatan	email	kelas	nama	nim	password	role	usia	status
<input type="checkbox"/>	Edit	Copy	Delete	1	64	blessymunthia@gmail.com	3SI2	Blessy Munthia	222212537	072003	Pengurus PDT	21	Aktif
<input type="checkbox"/>	Edit	Copy	Delete	2	64	raarani@gmail.com	3SI2	Aura Putri	222212538	raa2003	Volunteer	21	Aktif
<input type="checkbox"/>	Edit	Copy	Delete	3	65	imelda65@gmail.com	2ST3	Imelda Manik	222312540	imell2005	Volunteer	19	Aktif
<input type="checkbox"/>	Edit	Copy	Delete	4	64	nansleey@gmail.com	3SI2	Shafnanda Aulia	222212541	nanda2004	Volunteer	20	Aktif
<input type="checkbox"/>	Edit	Copy	Delete	5	64	dantt@gmail.com	3D32	Danti Kim	222212542	danti2003	Volunteer	21	Tidak Aktif
<input type="checkbox"/>	Edit	Copy	Delete	6	64	myurani@gmail.com	3SI1	Meuthia Maharani	222212543	meuth2004	Pengurus PDT	20	Aktif
<input type="checkbox"/>	Edit	Copy	Delete	8	64	putlina@gmail.com	3D31	Putri Rehulina	222212539	putss2004	Volunteer	20	Tidak Aktif
<input type="checkbox"/>	Edit	Copy	Delete	77	65	elsaanna@gmail.com	2D32	Elsa Simbolon	222312544	elsaa2005	Volunteer	19	Aktif

☐ Check all | With selected: Edit Copy Delete Export

Daftar tabel volunteer pada Database volunteer_pdt:

Server: 127.0.0.1 » Database: volunteer_pdt » Table: volunteers

Showing rows 0 - 5 (6 total, Query took 0.0004 seconds.)

`SELECT * FROM `volunteers``

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

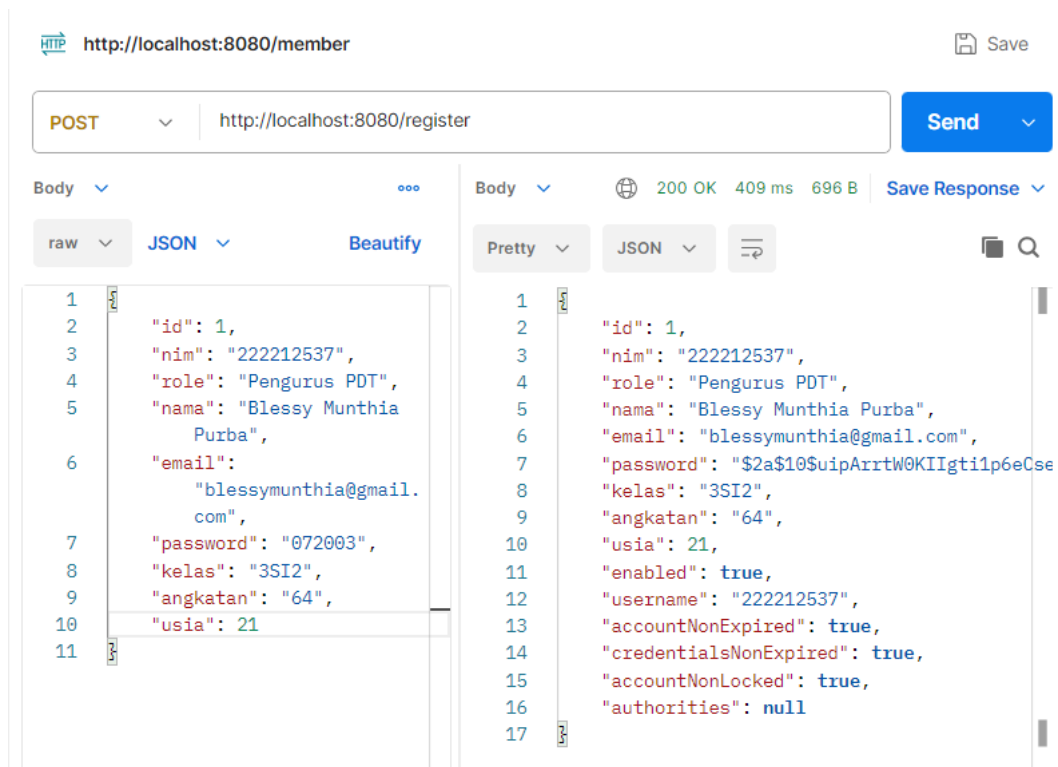
☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

				id	nama	nim	status
<input type="checkbox"/>	Edit	Copy	Delete	1	Aura Putri	222212538	Aktif
<input type="checkbox"/>	Edit	Copy	Delete	2	Imelda Manik	222312540	Aktif
<input type="checkbox"/>	Edit	Copy	Delete	3	Shafnanda Aulia	222212541	Aktif
<input type="checkbox"/>	Edit	Copy	Delete	4	Danti Kim	222212542	Tidak Aktif
<input type="checkbox"/>	Edit	Copy	Delete	5	Elsa Simbolon	222312544	Aktif
<input type="checkbox"/>	Edit	Copy	Delete	6	Putri Rehulina	222212539	Tidak Aktif

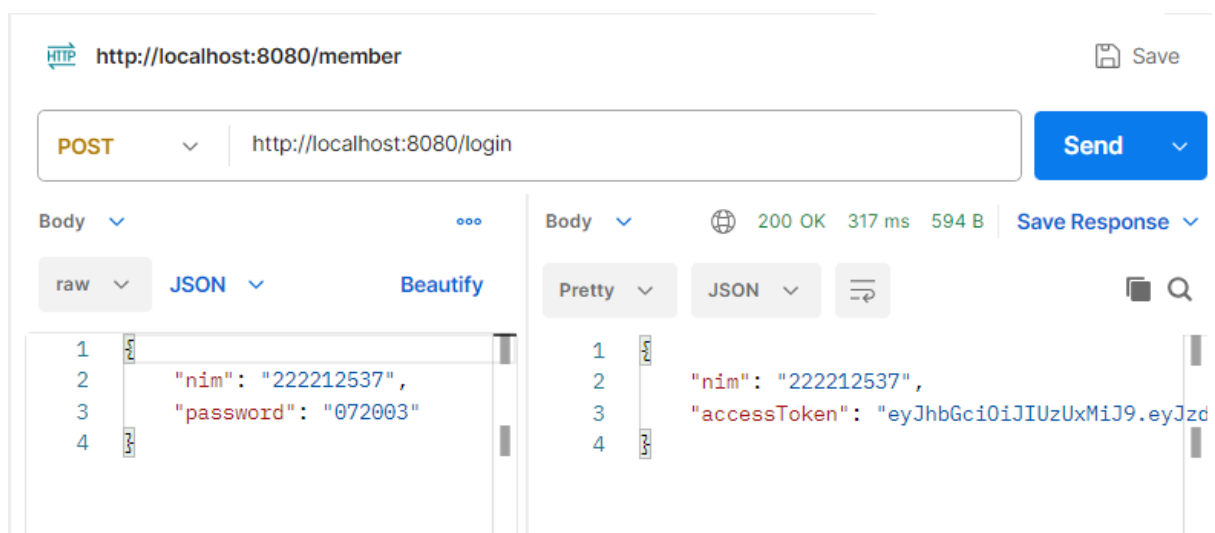
☐ Check all | With selected: Edit Copy Delete Export

1. Register



Semua mahasiswa dapat mendaftar di layanan web service berikut dengan pilihan role sebagai "Pengurus PDT", "Volunteer", dan "Pengunjung".

2. Login



Saat mahasiswa melakukan aksi login, maka layanan akan memberikan accessToken untuk mengautentikasikan seluruh method yang ada selain register dan login.

3. Change Pasword

The screenshot shows a Postman interface for a POST request to `http://localhost:8080/changePassword`. The request body is a JSON object with `"nim": "222212537"` and `"password": "310703"`. The response is a 200 OK status with a JSON body containing user details: `"id": 1`, `"nim": "222212537"`, `"role": "Pengurus PDT"`, `"nama": "Blessy Munthia Purba"`, `"email": "blessymunthia@gmail.com"`, `"password": "$2a$10$zoBPwfXE4tDln.YnwVn60u"`, `"kelas": "3SI2"`, `"angkatan": "64"`, `"usia": 21`, `"enabled": true`, `"username": "222212537"`, `"accountNonExpired": true`, `"credentialsNonExpired": true`, `"accountNonLocked": true`, and `"authorities": null`.

```
POST http://localhost:8080/changePassword
```

```
{
  "nim": "222212537",
  "password": "310703"
}
```

```
{
  "id": 1,
  "nim": "222212537",
  "role": "Pengurus PDT",
  "nama": "Blessy Munthia Purba",
  "email": "blessymunthia@gmail.com",
  "password": "$2a$10$zoBPwfXE4tDln.YnwVn60u",
  "kelas": "3SI2",
  "angkatan": "64",
  "usia": 21,
  "enabled": true,
  "username": "222212537",
  "accountNonExpired": true,
  "credentialsNonExpired": true,
  "accountNonLocked": true,
  "authorities": null
}
```

Dan lain-lain. Tidak tahu kenapa di postmant response nya masih 403 Forbidden, seperti ini:
(masih berusaha memperbaiki program tapi takut kelewat deadline, maaf tidak lengkap pak).

The screenshot shows a Postman interface for a POST request to `http://localhost:8080/volunteers/create`. The request body is a JSON object with `"nim": "222312545"`, `"nama": "Sophia Sinaga"`, and `"status": "Aktif"`. The response is a 403 Forbidden status with an empty body.

```
POST http://localhost:8080/volunteers/create
```

```
{
  "nim": "222312545",
  "nama": "Sophia Sinaga",
  "status": "Aktif"
}
```