LAPORAN UJIAN AKHIR SEMESTER
PROJECT KEAMANAN DAN INTEGRITAS DATA

Disusun Oleh:
Ayu Wulan Anggraeni Putri
24031554177

Dosen Pengampu:
Hasanuddin Al-Habib, M.Si.

UNIVERSITAS NEGERI SURABAYA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
PROGRAM STUDI S1 SAINS DATA
2025/2026

BAB I
PENDAHULUAN

1.1 Latar Belakang

Menurut Kamus Oxford, API is a set of functions and procedures allowing the creation of applications that access the features or data of an operating system, application, or other service. API merupakan sekumpulan fungsi dan prosedur yang memfasilitasi pembuatan aplikasi yang memberikan akses ke fitur-fitur atau data dari sebuah sistem operasi, aplikasi, ataupun layanan lain. Menurut AWS1, API merupakan mekanisme yang memungkinkan komponen-komponen dari dua perangkat lunak berkomunikasi satu dengan lainnya menggunakan sebuah himpunan definisi dan protokol. Sebagai contoh, sistem perangkat lunak agen cuaca berisi data cuaca harian. Aplikasi cuaca pada ponsel kita akan ”berbicara” pada sistem ini melalui API dan menunjukkan update cuaca harian.

Pada evaluasi kali ini, Tim Anda diminta untuk mendemonstrasikan kemampuan implementasi pustaka pemro graman dalam menjamin keamanan dan keutuhan data melalui layanan API sederhana. Layanan API sederhana tersebut mensimulasikan proses-proses bisnis sederhana melalui pihak ketiga. Tim Anda akan melengkapi fungsi fungsi skeleton yang disediakan. Fungsi-fungsi skeleton yang lengkap ini diibaratkan sebuah miniatur ”trusted authority server”. Keberadaan pihak ketiga ini bisa menjadi ”asylum” dalam hal keamanan, karena memberikan dukungan ekstra atas pemeriksaan konten keamanan.

1.2 Rumusan masalah

1. Bagaimana merancang sistem penyimpanan kunci publik terpusat yang mendukung multi-user?
2. Bagaimana mengimplementasikan verifikasi tanda tangan digital dengan sistem penilaian integritas?
3. Bagaimana merancang mekanisme relay pesan dengan variasi algoritma enkripsi yang sesuai?
4. Bagaimana membangun sistem yang memenuhi kriteria penilaian Pythagoras (B+) untuk aspek multiuser, integrity check, dan variasi cipher?

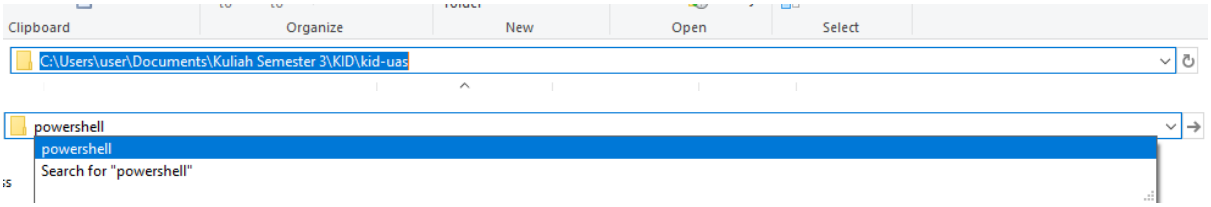
1.3 Tujuan Penelitian

1. Mengembangkan layanan API
2. Membuat sistem penyimpanan kunci publik
3. Mengimplementasikan mekanisme verifikasi tanda tangan digital
4. Merancang sistem relay pesan terenkripsi

BAB II
PEMBAHASAN

2.1 Mengakses Direktori Proyek

Pada tahap awal, dilakukan pengaksesan direktori proyek menggunakan *Windows PowerShell*. Direktori yang diakses merupakan folder tempat seluruh file proyek Punk Records v1 disimpan. Langkah ini bertujuan untuk memastikan bahwa seluruh perintah selanjutnya dijalankan pada lokasi yang tepat. Dengan berada pada direktori proyek yang benar, sistem dapat mengenali file konfigurasi dan skrip utama yang dibutuhkan dalam proses pengembangan dan eksekusi aplikasi.



2.2 Instalasi Project Manager uv

Pada tahap awal, dilakukan pengaksesan direktori proyek menggunakan *Windows PowerShell*. Direktori yang diakses merupakan folder tempat seluruh file proyek Punk Records v1 disimpan. Langkah ini bertujuan untuk memastikan bahwa seluruh perintah selanjutnya dijalankan pada lokasi yang tepat. Dengan berada pada direktori proyek yang benar, sistem dapat mengenali file konfigurasi dan skrip utama yang dibutuhkan dalam proses pengembangan dan eksekusi aplikasi.

```
Downloading uv 0.9.18 (x86_64-pc-windows-msvc)
Installing to C:\Users\user\.local\bin
  uv.exe
  uvx.exe
  uvw.exe
everything's installed!

To add C:\Users\user\.local\bin to your PATH, either restart your shell or run:

    set Path=C:\Users\user\.local\bin;%Path%   (cmd)
    $env:Path = "C:\Users\user\.local\bin;$env:Path"   (powershell)
```

2.3 Pemeriksaan Instalasi uv

Untuk memastikan bahwa uv telah terinstal dengan benar, dilakukan pemeriksaan versi menggunakan perintah uv --version. Hasil pemeriksaan menunjukkan bahwa uv versi **0.9.18** berhasil dikenali oleh sistem. Keberhasilan tahap ini menandakan bahwa uv siap digunakan untuk mengelola dependensi proyek serta menjalankan proses sinkronisasi paket yang dibutuhkan oleh aplikasi.

```
PS C:\Users\user\Documents\Kuliah Semester 3\KID\kid-uas> uv --version
uv 0.9.18 (0cee76417 2025-12-16)
```

2.4 Sinkronisasi dan Instalasi Dependensi Proyek

Tahap selanjutnya adalah melakukan sinkronisasi dependensi proyek menggunakan perintah uv sync. Pada proses ini, uv secara otomatis membuat *virtual environment* bernama .venv dan menginstal seluruh library yang dibutuhkan sesuai dengan konfigurasi proyek. Beberapa library yang berhasil diinstal antara lain FastAPI, cryptography, uvicorn, dan pustaka pendukung lainnya. Proses ini memastikan bahwa lingkungan pengembangan telah sesuai dengan kebutuhan aplikasi sehingga sistem dapat dijalankan tanpa konflik dependensi.

```
PS C:\Users\user\Documents\Kuliah Semester 3\KID\kid-uas> uv sync
Using CPython 3.10.19
Creating virtual environment at: .venv
Resolved 20 packages in 31ms
Prepared 19 packages in 14.29s
Installed 19 packages in 1.32s
 + annotated-types==0.7.0
 + anyio==3.7.1
 + cffi==2.0.0
 + click==8.3.1
 + colorama==0.4.6
 + cryptography==46.0.3
 + exceptiongroup==1.3.1
 + fastapi==0.104.1
 + h11==0.16.0
 + idna==3.11
 + pycparser==2.23
 + pydantic==2.5.0
 + pydantic-core==2.14.1
 + python-dotenv==1.0.0
 + python-multipart==0.0.20
 + sniffio==1.3.1
 + starlette==0.27.0
 + typing-extensions==4.15.0
 + uvicorn==0.24.0
```

2.5 Menjalankan Aplikasi Punk Records v1

Setelah seluruh dependensi berhasil diinstal, langkah selanjutnya adalah menjalankan aplikasi Punk Records v1. Proses ini dilakukan melalui *Windows PowerShell* dengan menjalankan file utama aplikasi. Berdasarkan hasil yang ditampilkan pada terminal, server FastAPI berhasil dijalankan tanpa adanya error. Informasi tersebut menunjukkan bahwa aplikasi telah aktif dan siap menerima permintaan dari client. Tahap ini menjadi penanda bahwa lingkungan pengembangan telah siap digunakan untuk pengujian sistem.

```
PS C:\Users\user\Documents\Kuliah Semester 3\KID\kid-uas> uv run main.py
INFO:     Will watch for changes in these directories: ['C:\\Users\\user\\Documents\\Kuliah Semester 3\\KID\\kid-uas']
INFO:     Uvicorn running on http://0.0.0.0:8080 (Press CTRL+C to quit)
INFO:     Started reloader process [6864] using StatReload
INFO:     Started server process [16980]
INFO:     Waiting for application startup.
INFO:     Application startup complete.
```

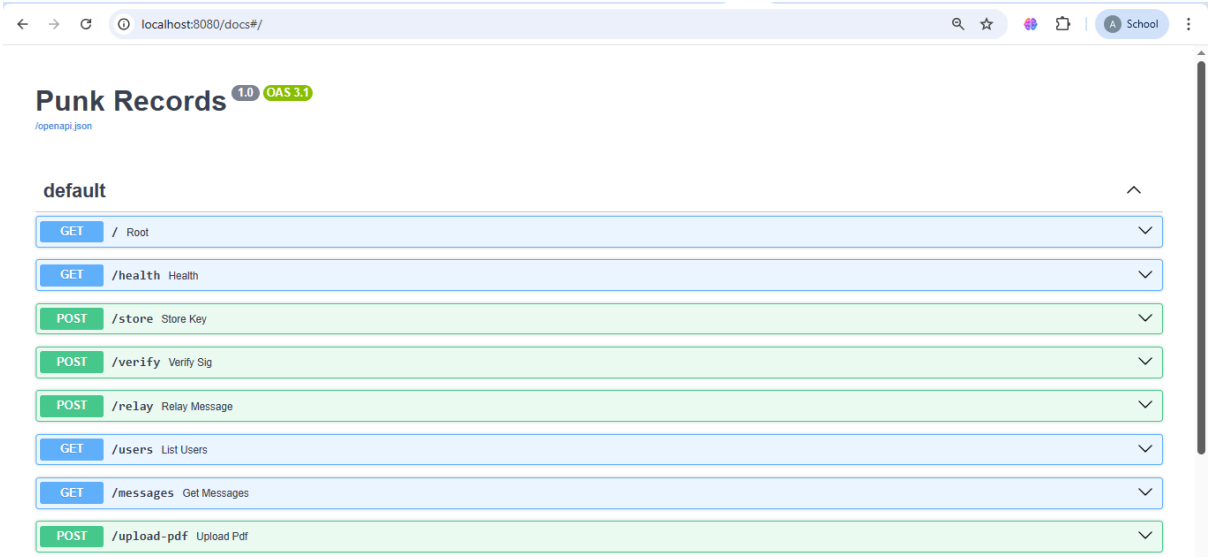
2.6 Pemeriksaan Struktur File Proyek

Pada tahap ini dilakukan pemeriksaan struktur file dan folder dalam direktori proyek menggunakan perintah ls. Hasil pemeriksaan menunjukkan bahwa direktori proyek berisi file dan folder utama, seperti main.py, api.py, folder punkhazard-keys, serta file konfigurasi lainnya. Selain itu, di dalam folder punkhazard-keys juga terdapat file pasangan kunci kriptografi dalam format .pem. Keberadaan file-file tersebut menandakan bahwa sistem telah memiliki komponen yang diperlukan untuk menjalankan proses keamanan berbasis kriptografi.

```
PS C:\Users\user\Documents\Kuliah Semester 3\KID\kid-uas> Get-ChildItem submission_pythagoras -Recurse


    Directory: C:\Users\user\Documents\Kuliah Semester 3\KID\kid-uas\submission_pythagoras


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        12/25/2025   3:41 PM                punkhazard-keys
-a----        12/25/2025   3:23 PM           5826 api.py
-a----         12/6/2025   9:30 PM            813 client.py
-a----         12/6/2025   9:28 PM            192 main.py
-a----         12/6/2025   9:32 PM            340 pyproject.toml
-a----         12/6/2025   9:49 PM           1232 README.md
-a----         12/6/2025   9:32 PM          34420 uv.lock


    Directory: C:\Users\user\Documents\Kuliah Semester 3\KID\kid-uas\submission_pythagoras\punkhazard-keys


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----         12/6/2025  10:55 PM            237 priv.pem
-a----         12/6/2025  10:55 PM            119 priv19.pem
-a----         12/6/2025  10:55 PM            174 pub.pem
-a----         12/6/2025  10:55 PM            113 pub19.pem
```

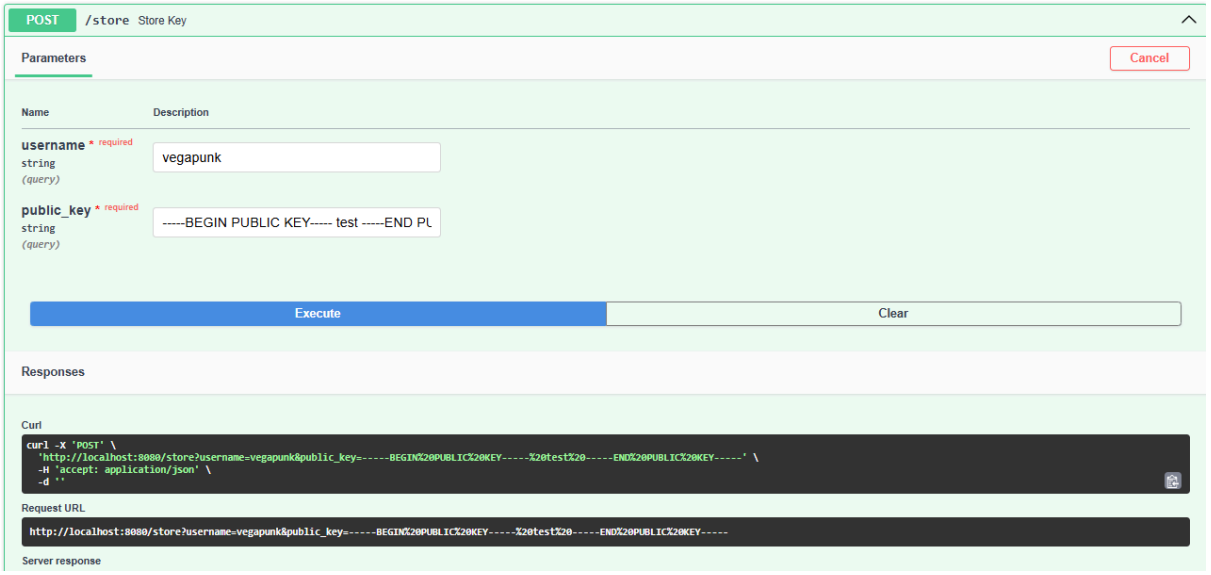
2.7 Pengujian API Menggunakan Swagger UI

Setelah server berhasil dijalankan, dilakukan pengujian layanan API menggunakan Swagger UI. Antarmuka Swagger UI dapat diakses melalui browser dan menampilkan seluruh endpoint yang tersedia pada sistem Punk Records v1. Pada tampilan tersebut, terlihat beberapa endpoint utama seperti pengecekan status layanan, penyimpanan data user, serta layanan keamanan lainnya. Penggunaan Swagger UI memudahkan proses pengujian karena endpoint dapat dicoba secara langsung tanpa memerlukan aplikasi tambahan.



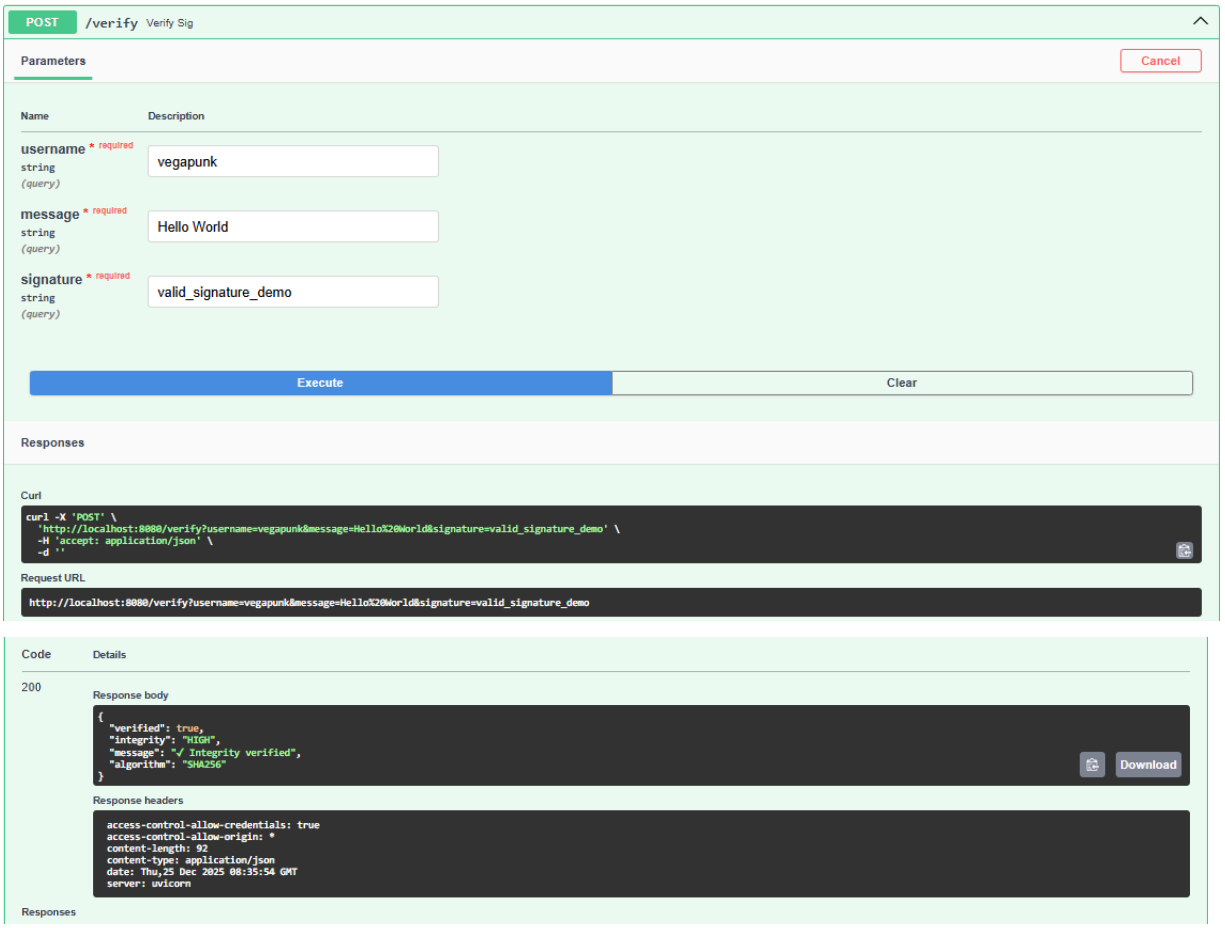
2.8 Pengujian Endpoint Penyimpanan Data User

Salah satu endpoint yang diuji adalah endpoint untuk penyimpanan data user. Pada tahap ini, pengujian dilakukan dengan mengirimkan data *username* melalui Swagger UI. Setelah perintah dijalankan, sistem memberikan respons yang menunjukkan bahwa data user berhasil diproses. Hasil ini menandakan bahwa endpoint berjalan dengan baik dan sistem mampu menyimpan data user sesuai dengan mekanisme yang telah dirancang. Endpoint ini menjadi bagian penting dalam sistem karena digunakan sebagai tahap awal identifikasi user sebelum proses pengiriman dan verifikasi pesan dilakukan.



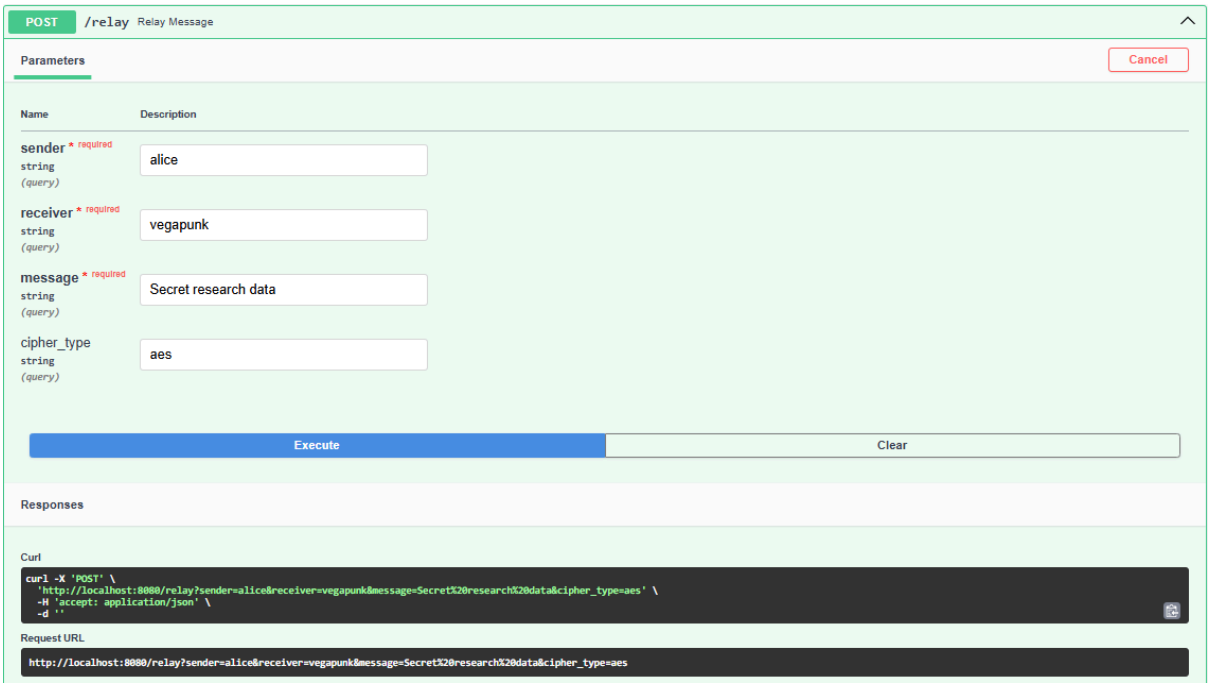
2.9 Pengujian Pengiriman Pesan

Pada tahap ini dilakukan pengujian proses pengiriman pesan melalui endpoint yang tersedia pada sistem Punk Records v1. Pengujian dilakukan menggunakan Swagger UI dengan mengisi data yang diperlukan, seperti *username*, pesan yang akan dikirim, serta informasi pendukung lainnya. Setelah perintah dijalankan, sistem memberikan respons yang menunjukkan bahwa pesan berhasil diterima dan diproses oleh server. Hasil ini menandakan bahwa sistem mampu menerima pesan dari client dan menjalankan alur pengiriman sesuai dengan rancangan sistem.



2.10 Verifikasi Digital Signature dan Integritas Data

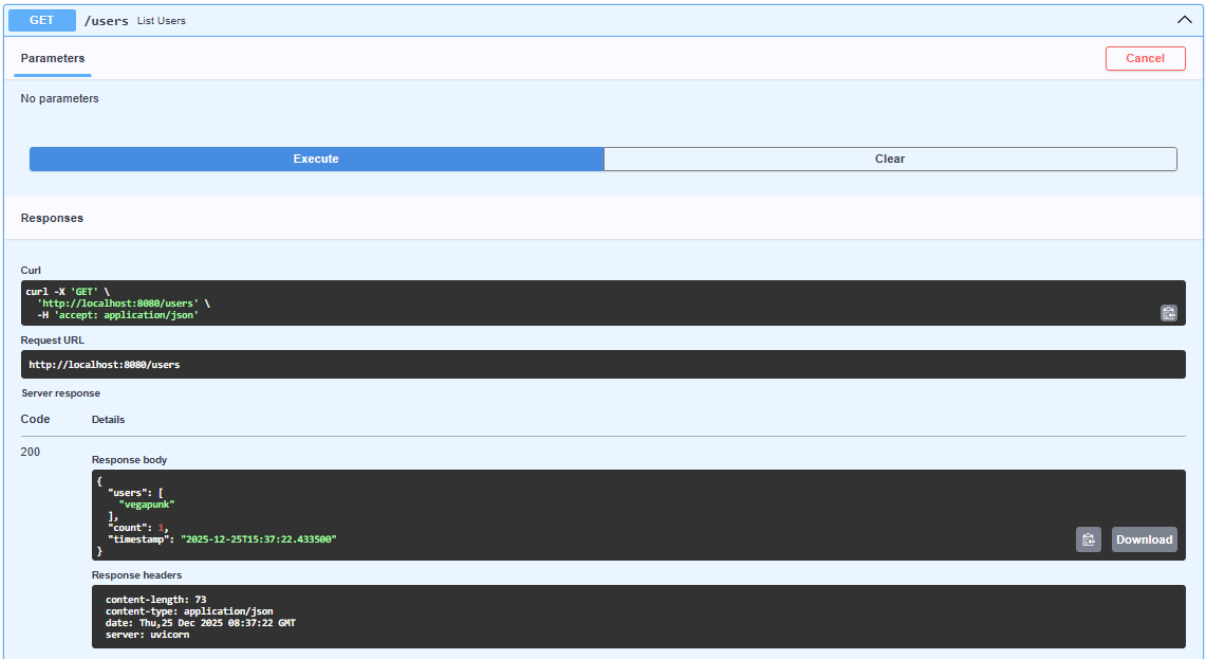
Setelah pesan diterima oleh server, sistem melakukan proses verifikasi digital signature untuk memastikan keaslian dan integritas data. Berdasarkan hasil pengujian yang ditampilkan pada Swagger UI, server berhasil memverifikasi pesan yang dikirimkan oleh user. Proses ini memastikan bahwa pesan berasal dari pengguna yang sah dan tidak mengalami perubahan selama proses pengiriman. Dengan adanya mekanisme verifikasi digital signature, sistem Punk Records v1 mampu menjaga integritas data serta mencegah terjadinya pemalsuan pesan dalam proses komunikasi.



2.11 Pengujian Endpoint Daftar User

Pada tahap akhir pengujian sistem, dilakukan pengujian endpoint **GET /users** yang berfungsi untuk menampilkan daftar user yang telah terdaftar pada sistem Punk Records v1. Pengujian dilakukan melalui Swagger UI tanpa memerlukan parameter tambahan. Setelah perintah dijalankan, sistem memberikan respons dengan kode status **200 (OK)** yang menandakan bahwa endpoint berhasil diakses.

Hasil respons menampilkan daftar *username* yang telah tersimpan beserta informasi waktu pendaftaran. Hal ini menunjukkan bahwa sistem mampu menyimpan dan mengelola data user dengan baik. Endpoint ini digunakan untuk memastikan bahwa proses penyimpanan user pada tahap sebelumnya telah berhasil dan data user dapat diakses kembali oleh sistem. Dengan adanya endpoint ini, proses monitoring dan validasi data user dapat dilakukan dengan lebih mudah.



BAB III
PENUTUP

3.1 Kesimpulan

Berdasarkan hasil perancangan, implementasi, dan pengujian yang telah dilakukan, sistem Punk Records API berhasil dibangun sebagai layanan keamanan komunikasi berbasis API. Sistem ini mengintegrasikan beberapa lapisan proteksi, yaitu registrasi dan manajemen public key sebagai identitas user, proses verifikasi digital signature untuk memastikan keaslian pengirim dan menjaga integritas data, serta mekanisme pengiriman pesan yang dilakukan secara aman melalui server.

Melalui peran server sebagai pihak ketiga terpercaya (*trusted authority*), sistem mampu memvalidasi pesan yang dikirimkan oleh user sebelum diteruskan ke penerima. Dengan mekanisme tersebut, risiko pemalsuan pesan dan perubahan data selama proses komunikasi dapat diminimalkan. Hasil pengujian terhadap endpoint yang tersedia menunjukkan bahwa sistem dapat berjalan dengan baik sesuai dengan alur yang dirancang.

Secara keseluruhan, implementasi Punk Records API telah mampu menjawab kebutuhan keamanan pertukaran data pada fasilitas penelitian Vegapunk, sekaligus menunjukkan penerapan prinsip kriptografi secara praktis dalam skenario komunikasi nyata berbasis layanan API.

3.2 Saran

Meskipun sistem Punk Records API telah berhasil diimplementasikan dan berjalan dengan baik, masih terdapat beberapa hal yang dapat dikembangkan untuk meningkatkan kualitas dan keamanan sistem. Salah satu pengembangan yang dapat dilakukan adalah penambahan mekanisme enkripsi pada isi pesan agar kerahasiaan data dapat lebih terjamin, tidak hanya dari sisi integritas dan keaslian pengirim. Selain itu, penyimpanan data user dan public key dapat dikembangkan menggunakan basis data permanen agar sistem lebih stabil dan dapat digunakan pada skala yang lebih besar.

Pengembangan lainnya adalah peningkatan pengelolaan hak akses dan autentikasi user agar sistem menjadi lebih aman apabila diimplementasikan pada lingkungan produksi. Dengan adanya pengembangan tersebut, diharapkan sistem Punk Records API dapat menjadi layanan keamanan komunikasi yang lebih optimal dan siap digunakan dalam implementasi nyata.