**PENERAPAN APLIKASI PYCHARM DENGAN DJANGO DAN PYTHON TERHADAP ALGORITMA CAESAR CHIPER**

**Dosen Pengampu :**

**Muhammad Khudzaifah, M.Si**

**OLEH:**

**RIZA IVANIA INDAH R. (210601110073)**

**EMILIA FITRIANA (210601110036)**

**DWIPA OKI D. (210601110095)**



**PROGRAM STUDI MATEMATIKA**

**FAKULTAS SAINS DAN TEKNLOGI**

**UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM MALANG**

**2024**

# **BAB I**

# **PENDAHULUAN**

## Latar Belakang

Perkembangan saat ini berakibat juga terhadap perkembangan teknologi yang sangat pesat. Perkembangan teknlogi memudahkan seseorang untuk berkomunikasi jarak jauh, memudahkan berbagai pekerjaan, namun perkembangan teknologi juga membuat banyak kebocoran data yang harus segera diatasi. Cara mengatasi kebocoran data ini dapat diatasi dengan ilmu kriptografi. Ilmu kriptografi adalah ilmu dan seni menjaga kerahasiaan informasi dengan cara mengubahnya menjadi format yang tidak dapat dibaca oleh pihak yang tidak bertanggungjawab.

Kriptografi memiliki beberapa algoritma, salah satunya adalah Caesar Cipher. Algoritma ini merupakan algoritma kriptografi yang paling tua dan sederhana dengan cara substitusi sandi yang ditemukan oleh Julius Caesar. Dalam Caesar Cipher, setiap huruf dalam teks asli digantikan oleh huruf lain yang berada pada posisi tertentu di alfabet. Misalnya, dengan menggunakan pergeseran tiga tempat, huruf A akan menjadi D, B akan menjadi E, dan seterusnya. Meskipun sederhana, metode ini cukup efektif untuk menjaga pesan dari orang lain.

Kriptografi juga dapat dipromgramkan menggunakan python ataupun django. Pemrogaraman ini memudahkan orang yang tidak mengenal terlalu dalam dengan kriptografi dapat tetap menggunakannya. Algoritma kriptografi dapat tetap digunakan dengan membuat website yang dapat diakses oleh orang yang memiliki link. Oleh karena itu, penulisan makalah ini bertujuan untuk memberitahukan kepada orang yang tidak memahami terlalu dalam kriptografi untuk menggunakan algoritma Caesar Cipher dalam persandian yang akan dilakukan. Selain itu, makalah ini juga membahas tentang langkah-langkah pemrograman yang akan penulis lakukan dalam membuat pemrograman menggunakan aplikasi pycharm.

## Rumusan Masalah

Berdasarkan latar belakang di atas makapenelitian ini mempunyai rumusan masalah sebagai berikut:

1. Bagaimana langkah-langkah pemrograman python dan django terhadap pemrograman Caesar Cipher menggunakan aplikasi pycharm?
2. Bagaimana penerapan algoritma Caesar Cipher pada pemrograman python dan django menggunakan aplikasi pycharm?

# **BAB 2**

# **KAJIAN TEORI**

## 2.1 Pengertian

Bab ini akan menguraikan berbagai teori yang berkaitan dengan implementasi aplikasi PyCharm yang dikombinasikan dengan Django dan Python dalam konteks penerapan algoritma Caesar Cipher. Pembahasan akan mencakup pengenalan mendalam tentang algoritma Caesar Cipher, dasar-dasar pemrograman dengan Python, framework Django, serta integrasi keduanya dengan lingkungan pengembangan PyCharm.

## 2.2 Algoritma Caesar Cipher

### 2.2.1 Definisi Caesar Cipher

Caesar Cipher adalah salah satu metode enkripsi tertua dan paling sederhana yang telah dikenal dalam sejarah kriptografi. Algoritma ini dinamai sesuai dengan Julius Caesar, seorang jenderal Romawi yang menggunakan teknik ini untuk menjaga kerahasiaan pesan militer. Caesar Cipher termasuk dalam kategori cipher substitusi, di mana setiap huruf dalam teks asli digantikan dengan huruf lain yang berada pada posisi tertentu dalam alfabet.

### 2.2.2 Cara Kerja Caesar Cipher

Prinsip kerja dari Caesar Cipher adalah sebagai berikut:

1. Enkripsi : Setiap huruf dalam teks asli (plaintext) digeser sejumlah posisi tertentu ke kanan dalam urutan alfabet.

2. Dekripsi : Untuk mengembalikan teks yang telah dienkripsi (ciphertext) ke bentuk aslinya, setiap huruf digeser sejumlah posisi yang sama ke kiri dalam urutan alfabet.

Sebagai ilustrasi, dengan pergeseran sebanyak 3 posisi, huruf 'A' akan menjadi 'D', huruf 'B' akan menjadi 'E', dan seterusnya. Rumus matematis yang digunakan untuk proses enkripsi adalah:

dan untuk dekripsi:

Dimana:

* adalah huruf cipher.
* P adalah huruf plaintext
* adalah jumlah pergeseran.

## 2.3 Dasar-Dasar Python

Python adalah bahasa pemrograman tingkat tinggi yang terkenal karena kemudahannya dalam penulisan dan pembacaan kode. Python sering digunakan dalam berbagai bidang, termasuk pengembangan web, analisis data, otomatisasi tugas, dan banyak lagi.

### 2.3.1 Sintaks Dasar

Python menggunakan indentasi untuk menandai blok kode, berbeda dengan banyak bahasa pemrograman lain yang menggunakan tanda kurung atau kata kunci tertentu. Beberapa fitur dasar Python meliputi:

* 1. Variabel dan Tipe Data : Python mendukung berbagai tipe data, seperti integer, float, string, dan list, yang memungkinkan fleksibilitas dalam pemrograman.
  2. Kontrol Aliran : Pernyataan seperti if, for, dan while digunakan untuk mengontrol jalannya eksekusi program berdasarkan kondisi tertentu.
  3. Fungsi : Fungsi dalam Python didefinisikan menggunakan kata kunci `def` dan memungkinkan pengelompokan kode yang dapat digunakan kembali.

### 2.3.2 Perpustakaan Standar

Python dilengkapi dengan perpustakaan standar yang sangat kaya, mencakup modul-modul untuk berbagai kebutuhan, seperti perhitungan matematika, manipulasi string, dan akses file, yang membuat pengembangan program menjadi lebih efisien.

## 2.4 Framework Django

Django adalah framework web yang berbasis pada bahasa pemrograman Python, dirancang untuk membuat pengembangan aplikasi web menjadi lebih cepat dan mudah. Django mengusung prinsip "Don't Repeat Yourself" (DRY) yang bertujuan untuk mengurangi duplikasi kode dan meningkatkan efisiensi.

### 2.4.1 Arsitektur Django

Django menggunakan arsitektur Model-View-Template (MVT):

* 1. Model : Komponen ini berhubungan dengan basis data dan mengelola data aplikasi.
  2. View : Bagian ini mengandung logika bisnis yang mengatur bagaimana data diproses dan ditampilkan.
  3. Template : Komponen ini berhubungan dengan tampilan aplikasi, memungkinkan pembuatan antarmuka pengguna yang dinamis dan responsif.

### 2.4.2 Fitur Utama

Beberapa fitur utama yang menjadikan Django pilihan populer di kalangan pengembang web meliputi:

* 1. ORM (Object-Relational Mapping) : Django menyediakan ORM yang memudahkan interaksi dengan basis data, memungkinkan pengembang untuk bekerja dengan basis data menggunakan kode Python daripada SQL.
  2. Admin Interface : Django secara otomatis menghasilkan antarmuka administrasi untuk mengelola konten aplikasi, menghemat waktu pengembangan.
  3. Keamanan : Django memiliki fitur keamanan bawaan yang menangani berbagai isu keamanan umum, seperti SQL injection, cross-site scripting (XSS), dan lain-lain, sehingga aplikasi lebih terlindungi.

## 2.5 Integrasi dengan PyCharm

PyCharm adalah sebuah Integrated Development Environment (IDE) yang sangat kuat dan dirancang khusus untuk pengembangan aplikasi menggunakan Python. PyCharm menyediakan berbagai fitur yang memudahkan proses pengembangan aplikasi dengan framework Django.

### 2.5.1 Fitur PyCharm

Beberapa fitur utama PyCharm yang sangat bermanfaat dalam pengembangan aplikasi Django adalah:

* + - * 1. Ode Lengkap Otomatis : Fitur ini membantu pengembang menulis kode lebih cepat dengan memberikan saran dan melengkapi kode secara otomatis.
        2. Debugger : Memudahkan proses debug dengan menyediakan alat untuk memeriksa dan memodifikasi variabel secara real-time.
        3. Manajemen Proyek : PyCharm memudahkan pengorganisasian file dan direktori proyek, memungkinkan pengembang untuk bekerja lebih efisien dan terstruktur

### 2.5.2 Pengaturan Proyek Django di PyCharm

PyCharm menyederhanakan proses pembuatan dan pengelolaan proyek Django. Langkah-langkah untuk memulai proyek Django di PyCharm meliputi:

1. Membuat proyek baru menggunakan template Django yang disediakan oleh PyCharm.

2. Mengonfigurasi pengaturan proyek sesuai kebutuhan.

3. Menggunakan fitur bawaan PyCharm untuk manajemen basis data, migrasi skema, dan tugas-tugas lain yang berhubungan dengan pengembangan aplikasi web.

## 2.6 Penerapan Caesar Cipher dengan Django dan Python

Implementasi algoritma Caesar Cipher dalam sebuah aplikasi web menggunakan Django dan Python dapat dilakukan melalui beberapa langkah berikut:

1. Membuat Model : Mendefinisikan model untuk menyimpan teks asli dan teks yang telah dienkripsi di basis data.

2. Membuat View : Menyusun logika bisnis untuk mengenkripsi dan mendekripsi teks menggunakan algoritma Caesar Cipher, yang akan dipanggil oleh template.

3. Membuat Template : Merancang antarmuka pengguna yang memungkinkan input teks dari pengguna dan menampilkan hasil enkripsi atau dekripsi.

4. Mengintegrasikan dengan PyCharm : Menggunakan berbagai fitur PyCharm untuk menulis, menguji, dan debug kode secara efektif, sehingga aplikasi berjalan dengan lancar dan sesuai harapan.

Bab ini telah menguraikan berbagai teori yang relevan dengan penerapan aplikasi PyCharm yang dikombinasikan dengan Django dan Python dalam konteks implementasi algoritma Caesar Cipher. Dengan memahami dasar-dasar algoritma Caesar Cipher, dasar-dasar pemrograman Python, framework Django, serta integrasinya dengan lingkungan pengembangan PyCharm, kita dapat mengembangkan aplikasi web yang mampu mengenkripsi dan mendekripsi teks menggunakan algoritma Caesar Cipher dengan efisien dan efektif. Bab selanjutnya akan membahas implementasi praktis dari teori-teori ini dalam bentuk proyek aplikasi web yang konkret.

# **BAB 3**

# **METODE PENELITIAN**

* 1. **Jenis Penelitian**

Jenis penelitian ini adalah penelitian kualitatif, artinya penelitian ini berdasarkan atas studi pustaka. Studi pustaka adalah proses penyelidikan dan penyusunan ringkasan, analisis, dan sintesis terhadap literatur, artikel ilmiah, buku, dan sumber-sumber lain yang relevan dengan penerapan aplikasi PyCharm dengan Django dan Python terhadap algoritma Caesar Chiper.

* 1. **Pra Penelitian**

Pra penelitian yang dilakukan adalah mengumpulkan beberapa artikel serta buku-buku yang berkaitan dengan algoritma Caesar Chiper. Juga melakukan tinjauan literatur untuk memahami konsep-konsep kunci yang terkait dengan PyCharm, Django, Python, dan algoritma Caesar Chiper. Hal ini dapat dilakukan dengan membaca buku, jurnal ilmiah, dan artikel online.

* 1. **Tahapan Penelitian**

Beberapa tahapan penelitian yang akan dilakukan antara lain:

* 1. Pengembangan Aplikasi: Mengembangkan aplikasi PyCharm dengan Django dan Python untuk mengimplementasikan algoritma Caesar Chiper. Aplikasi ini harus dirancang dengan baik, mudah digunakan, dan memiliki fungsionalitas yang sesuai dengan kebutuhan penelitian.
  2. Pengujian Aplikasi: Menguji aplikasi yang telah dikembangkan untuk memastikan bahwa aplikasi tersebut berfungsi dengan benar dan tidak ada bug atau kesalahan. Pengujian dapat dilakukan secara manual atau menggunakan alat bantu pengujian.

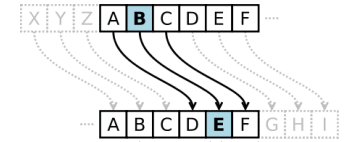
**BAB IV**

**HASIL DAN PEMBAHASAN**

* 1. **Algoritma Caesar Cipher**

Caesar Cipher adalah salah satu yang tertua dan paling dikenal dalam perkembangan kriptografi. Caesar cipher merupakan jenis cipher substitusi yang membentuk cipher dengan cara menukar karakter pada plaintext menjadi tepat satu karakter pada ciphertext. Teknik ini juga dikenal sebagai alfabet sandi tunggal. Simetri dan digunakan jauh sebelum sistem kriptografi public key ditemukan, kriptografi klasik yang ada dan beberapa bentuk algoritma klasik dianggap optimah karena mudha dipecahkan. Inti dari algoritma kriptografi ini adalah pergeseran ke arah semua karakter dalam plaintext dengan nilai pergeseran yang sama. Langkah-langkah yang diambil untuk membangun ciphertext dengan Caesar Cipher adalah:

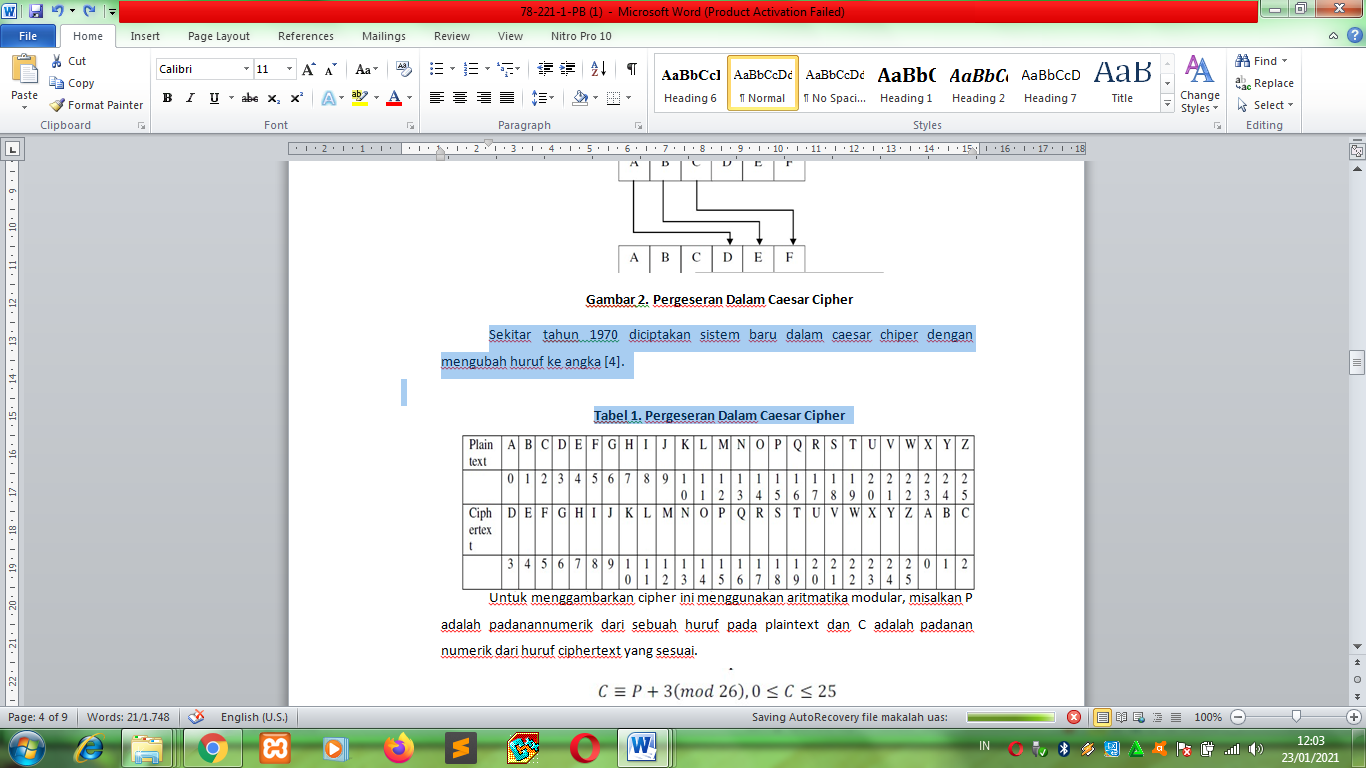
1. Menentukan besarnya pergeseran karakter yang digunakan dalam membentuk cipherteks menjadi plainteks.
2. Menukarkan karakter-karakter pada plaintext menjadi ciphertext dengan berdasarkan pergeseran yang telah ditentukan.



**Gambar 1 Skema Caesar Cipher**

Gambar 1 menunjukkan contoh pergeseran yang sudah ditentukan sebelumnya. Pada gambar tersebut, kita dapat melihat setiap karakter digeser tiga kali dari titik awal. Pada contoh tersebut, karakter B, posisi kedua dalam tabel karakter dipindahkan ke karakter berikutnya. B berpindah ke E. Jika kita menemukan karakter ini dalam proses enkripsi, karakter tersebut secara otomatis berubah menjadi karakter baru yang ditempatkan tiga karakter berikutnya dari karakter pertama.

Teknik ini juga dikenal sebagai single cipher alphabet.Caesar chiper pertama kali digunakan oleh Julius Caesar. Caesar mengkodekan informasi dengan menggubah setiap huruf dalam informasi menjadi tiga huruf di setelah informasi asli dalam urutan alfabet. Algoritma kriptografi Caesar Cipher sangat mudah digunakan. Inti dari algoritma kriptografi ini menggeser semua karakter dalam plaintext dengan nilai pergeseran yang sama.



**Gambar 2 Pergeseran Dalam Caesar**

Untuk menggambarkan cipher ini menggunakan aritmatika modular, misalkan P adalah padanannumerik dari sebuah huruf pada plaintext dan C adalah padanan numerik dari huruf ciphertext yang sesuai.



Penerima pesan dapat mendekripsi dengan langkah



Setelah ciphertext kembali ke padanan numerik plaintext, penerima pesan dapat mengubah kembali pesan angka ke huruf.

* 1. **Penerapan Algoritma Caesar Cipher pada Pemrograman Python dan Django**

Algoritma Caesar Cipher adalah algoritma yang paling sederhana dalam kriptografi. Algortima ini cara penggunaannya adalah mengganti setiap huruf dalam sebuah teks dengan mengganti huruf yang telah digeser. Misalnya, dengan pergeseran 3, 'A' menjadi 'D', 'B' menjadi 'E', dan seterusnya. Untuk lebih jelasnya, maka akan dirincikan dalam perhitungan secara matematis serta menggunakan aplikasi Pycharm.

**4.2.1 Perhitugan Secara Matematis**

Perhitungan menggunakan algortima Caeshar Cipher dapat dilakukan dengan menggeser setiap huruf dalam teks asli dengan posisi yang ditentukan dalam alfabet. Misalnya, mengambil teks asli “SEMUA AMAN DENGAN KRIPTOGRAFI”. Setelah mendapatkan teks asli, maka ditentukan pergeseran alfabet yang akan dilakukan. Sebagai contoh, teks asli akan digeser sebanyak 6 kali, maka :

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Plainteks | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z |
|  | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 |
| Cipherteks | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | A | B | C | D | E | F |
|  | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 0 | 1 | 2 | 3 | 4 | 5 |

Pada tabel diatas, dapat dilihat bahwa pergesaran setiap huruf dengan menggeser teks asli sebanyak 6, misalnya A menjadi G, B menjadi H, dst. Maka, dengan tabel diatas dapat dihitung secara manual, enkripsi dan dekripsi dari teks asli yang telah diperoleh.

* Teks asli : “SEMUA AMAN DENGAN KRIPTOGRAFI”
* Enkripsi : Proses enkripsi dilakukan dengan menkonversi plainteks ke cipherteks dengan pergeseran yang telah ditentukan yaitu 6, dan dapat dilihat dari tabel.

S = Y

E = K

M = S

U = A

A = G

N = T

D = J

G = M

K = Q

R = X

I = O

P = V

T = Z

O = U

F = L

Maka, Cipherteks yang dihasilkan adalah “YKSAG GSGT JKTMGT QYXOVZUMXGLO”

* Dekripsi : Untuk mengembalikan cipherteks ke plainteks kembali, maka geser cipherteks ke kiri sebanyak 6 kali. misal G menjadi A, H menjadi B, dst. Hal ini juga dapat dilihat dalam tabel dibawah ini dengan menggeser chiperteks.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Cipherteks | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | A | B | C | D | E | F |
|  | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 0 | 1 | 2 | 3 | 4 | 5 |
| Plainteks | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z |
|  | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 |

Maka, Plainteks yang dihasilkan dari Cipherteks “YKSAG GSGT JKTMGT QYOVZUMYGLO” adalah “SEMUA AMAN DENGAN KRIPTOGRAFI” teks akan kembali kepada plainteks awal.

**4.2.2 Simulasi Coding**

Setelah mengetahui pengoperasian algoritma caesar cipher dengan perhitungan yang matematis, maka akan ditunjukkan program yang diperlukan dalam membuat program caesar cipher dengan pemrograman python dan django dalam aplikasi pycharm :

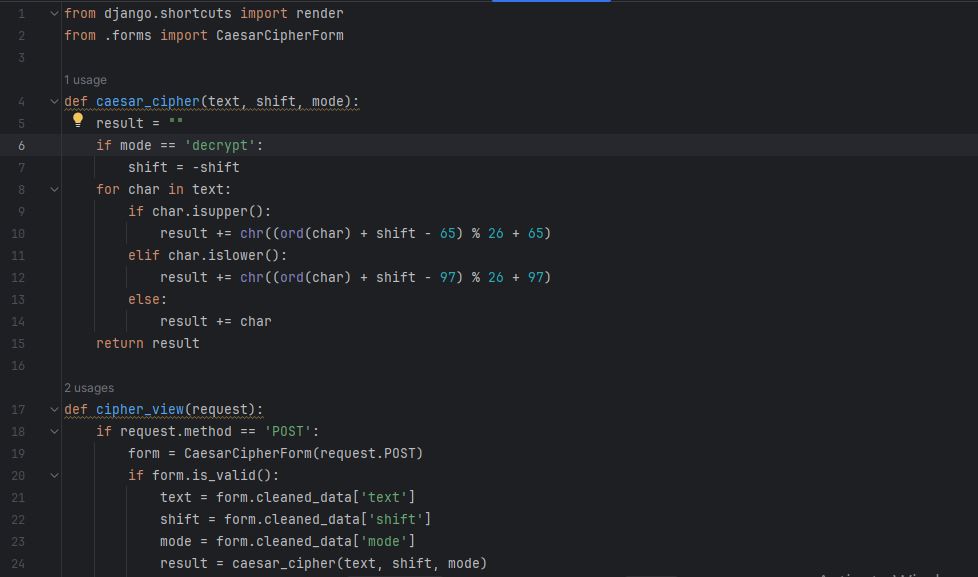
* **Install django terlebih dahulu pada terminal**

**“pip install django”**

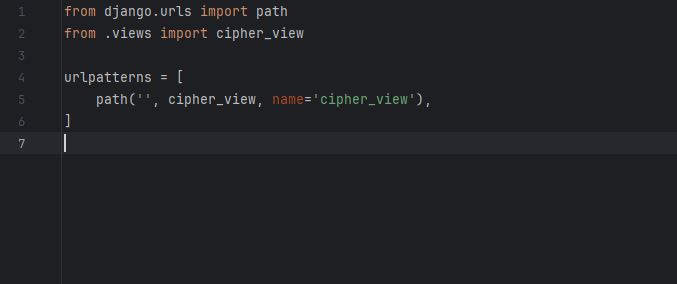
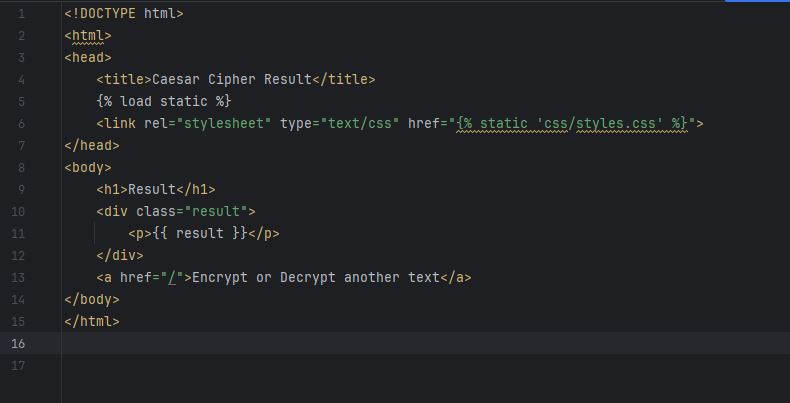
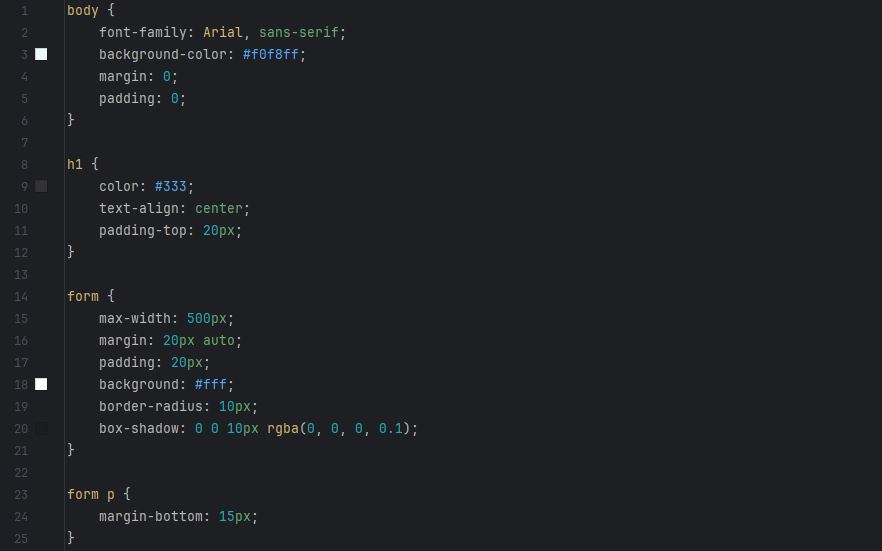
* **Buat proyek django dengan menuliskan ke terminal**

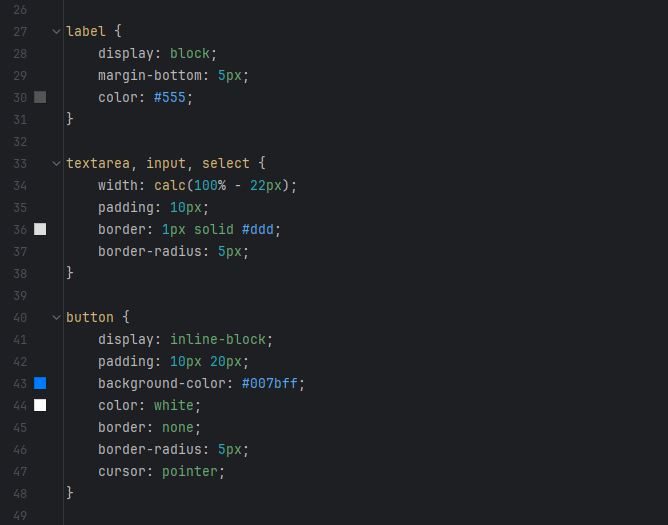
**“django-admin startproject caesar\_cipher\_project” dan**

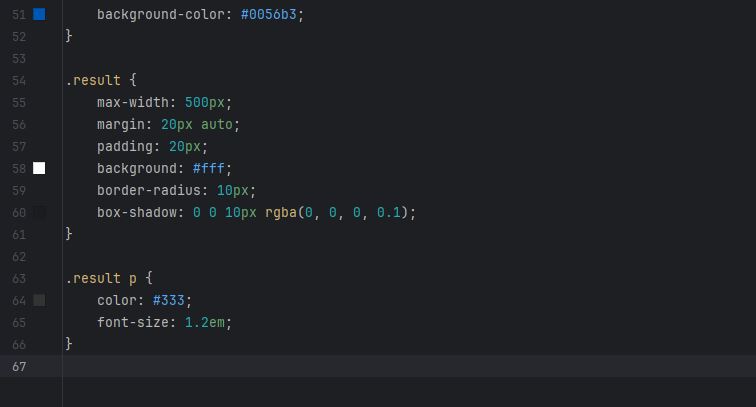
**“cd caesar\_cipher\_project”**

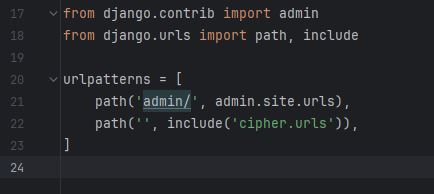
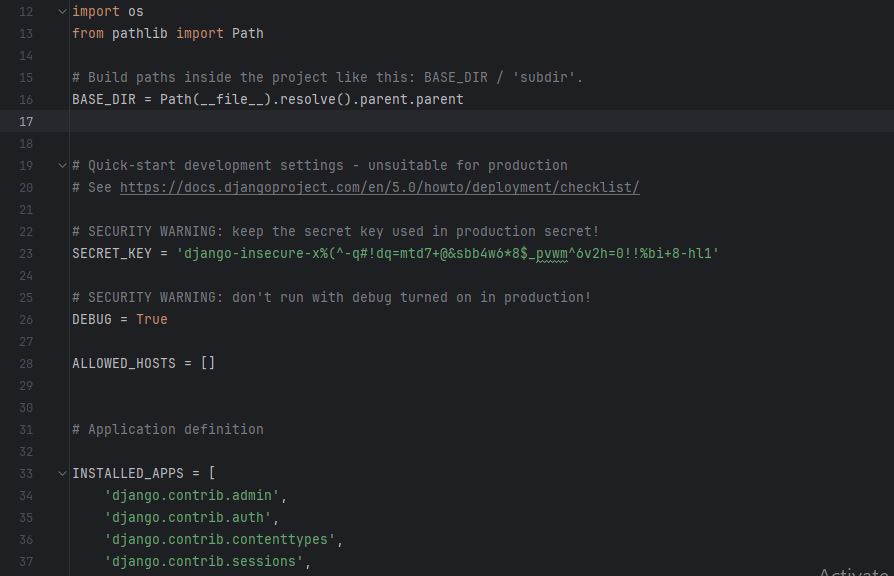
* **views.py** 



* **urls.py dalam folder cipher**
* **index.html dalam folder template**
* **result.html dalam folder template**
* **styles.css dalam folder static untuk mendesain web menjadi yang diinginkan**

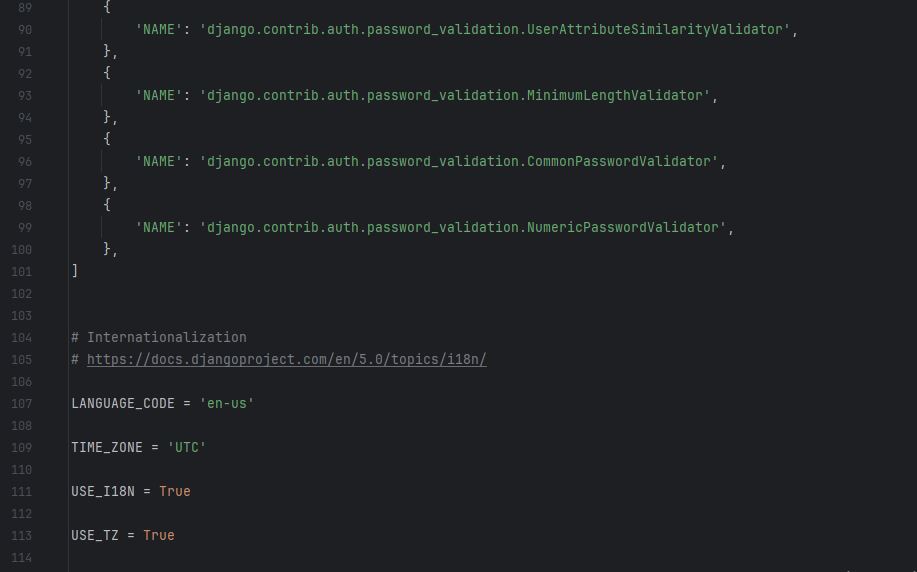




* **urls.py dalam folder caesar\_cipher\_project**
* **setting.py dalam folder caesar\_cipher\_project**

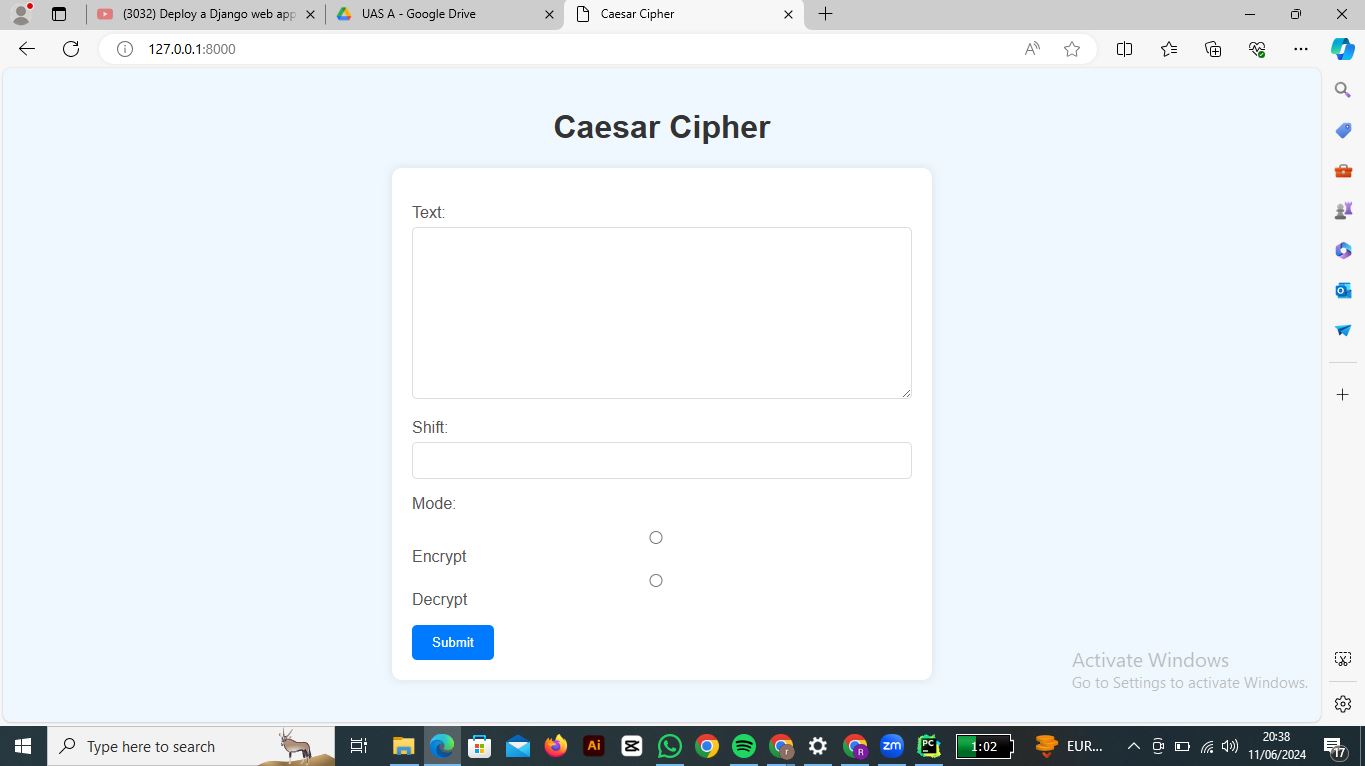


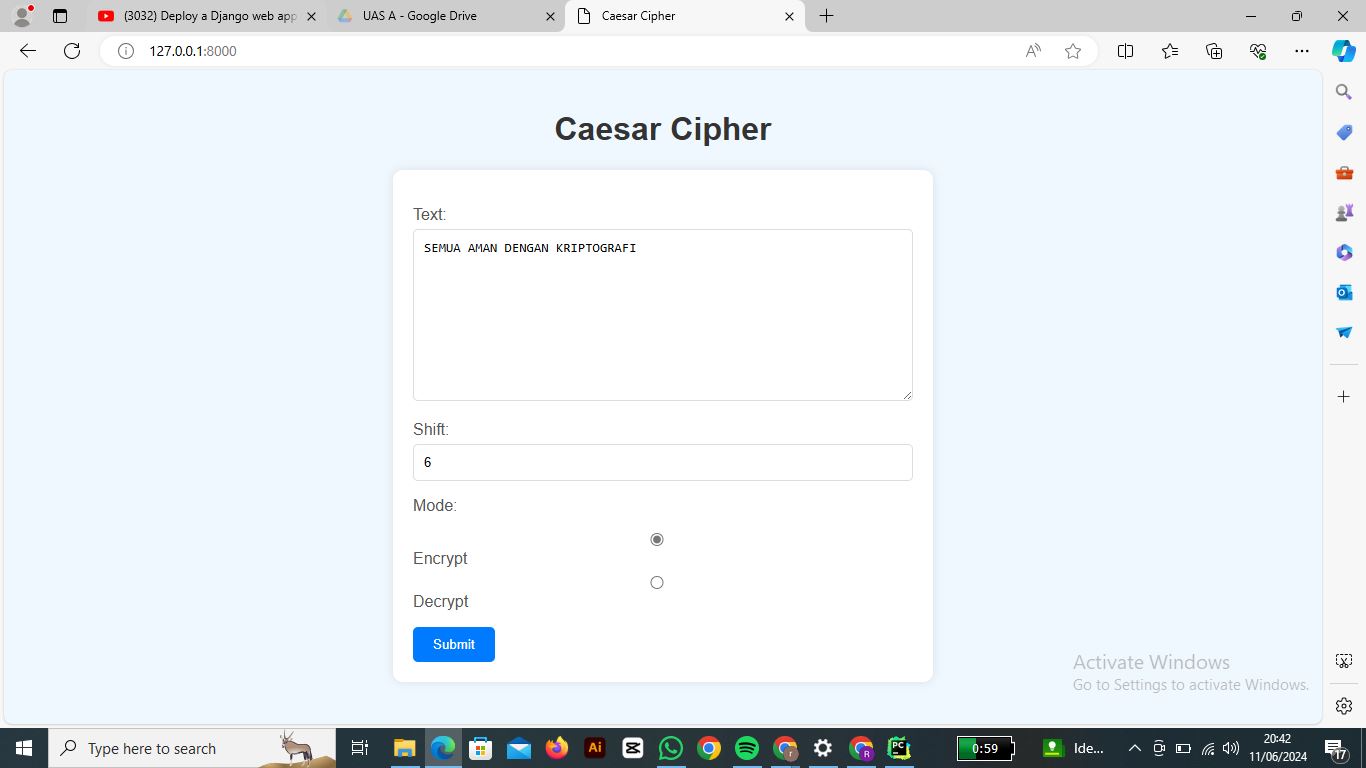


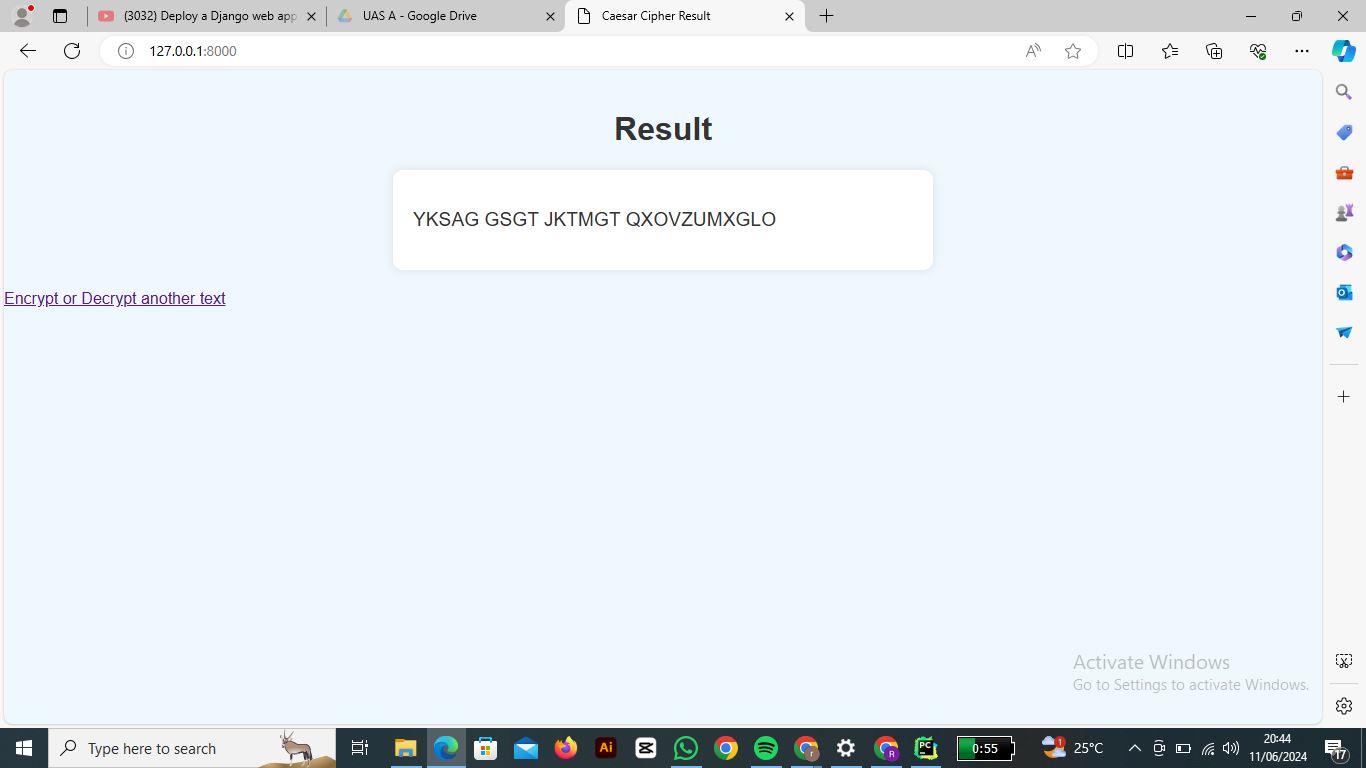
****

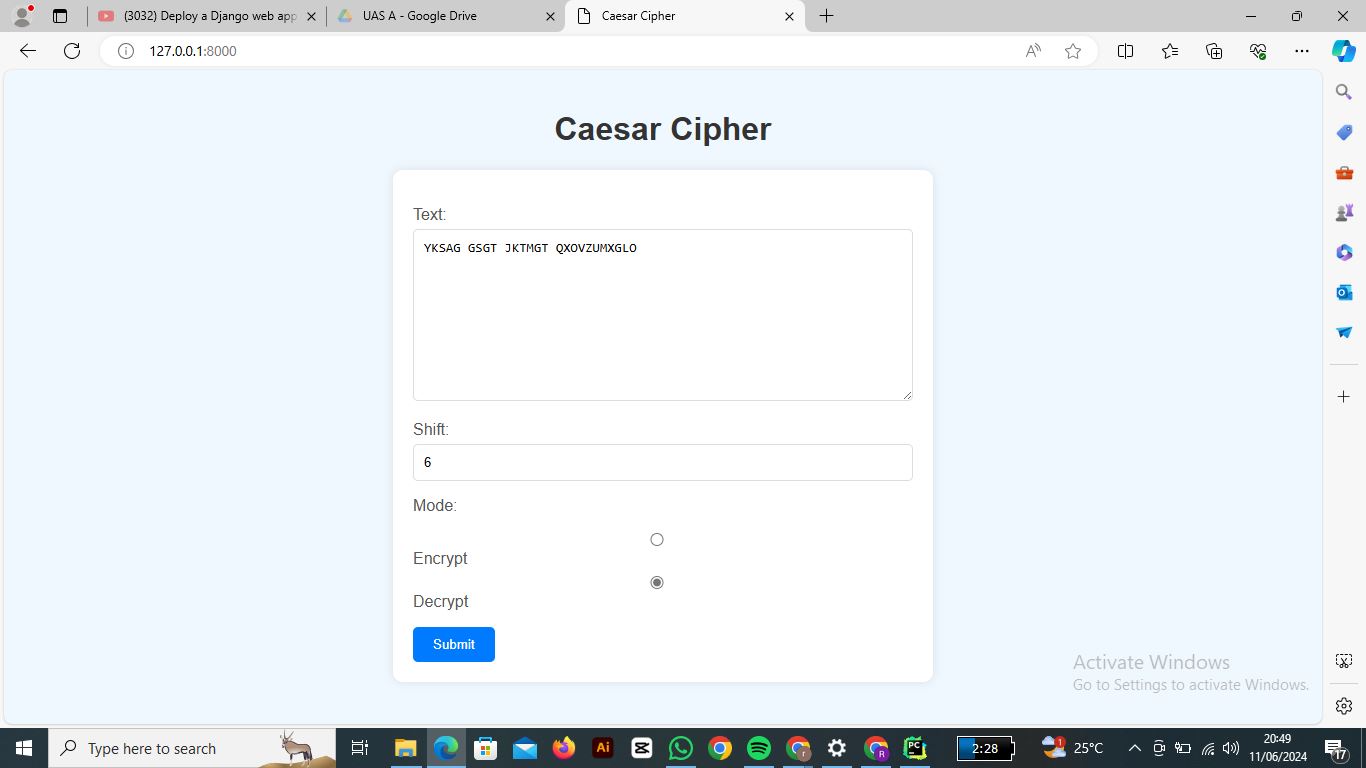


Setelah program dibuat dengan benar dengan folder yang sesuai, maka jalankan program dengan mengetik “python manage.py runserver”. Ketika sudah dijalankan akan muncul link web yang sudah kita buat programnya. Berikut adalah simulasi penggunaan web :

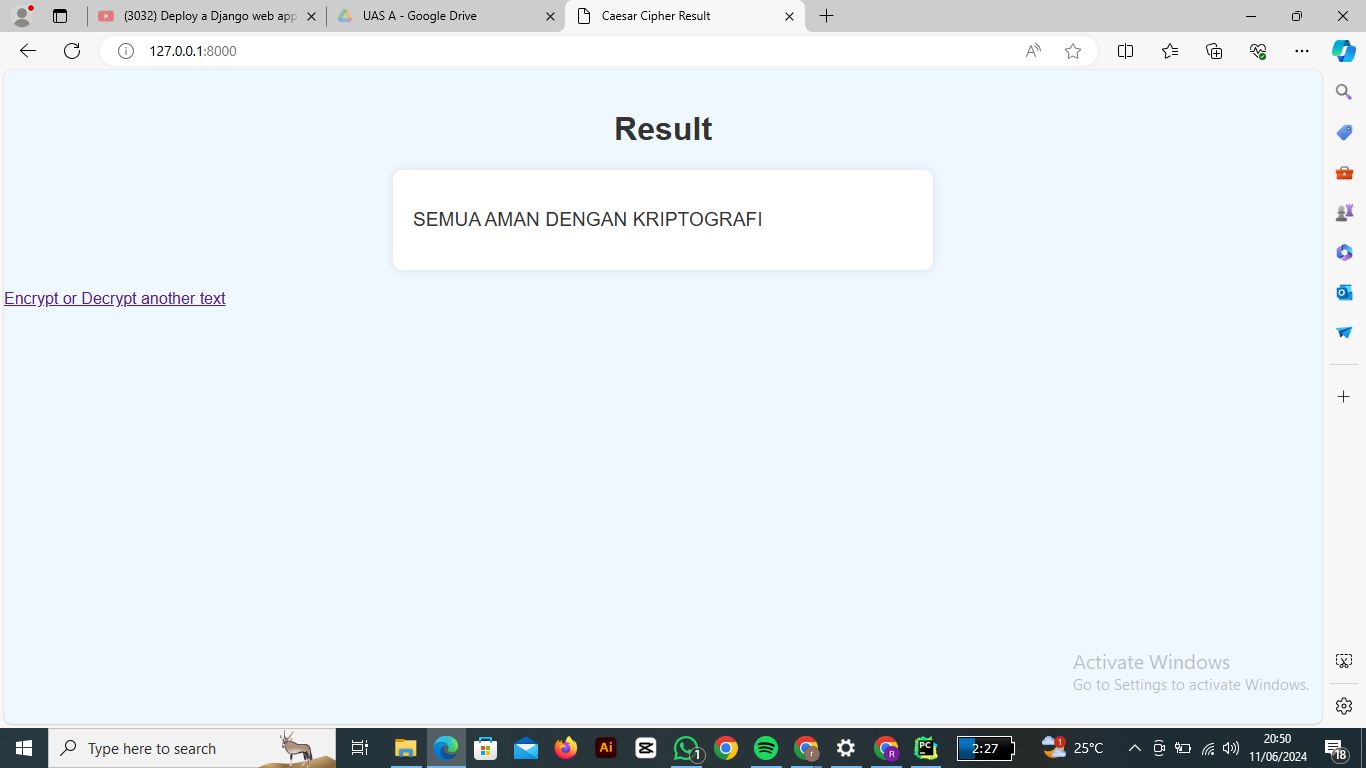
* Halaman web
* Proses Enkripsi

Masukkan plainteks yang akan di enkripsi > Masukkan pergeseran yang ditentukan > pilih mode enkripsi > submit

* Hasil Proses Enkripsi
* Proses Dekripsi

Masukkan cipherteks yang akan di dekripsikan > Masukkan pergeseran yang ditentukan > pilih mode dekripsi > submit

* Hasil Dekripsi



Dari hasil pada web, dapat dilihat bahwa plainteks dan cipherteks dengan proses enkripsi dan dekripsi hasilnya sama, dan saat cipherteks didekripsi akan kembali ke plainteks atau teks asli.

### 4.3. Langkah-langkah Meng-upload Aplikasi ke PythonAnywhere

### 4.3.1. Login ke Akun PythonAnywhere

1. Buka [PythonAnywhere](https://www.pythonanywhere.com) di browser Anda.
2. Klik Log in di pojok kanan atas halaman.

### 4.3.2. Membuat Aplikasi Web Baru

1. Setelah login, masuk ke dashboard PythonAnywhere.
2. Klik tab Web di bagian atas halaman.
3. Klik tombol Add a new web app.

### 4.3.3. Memilih Framework dan Versi Python

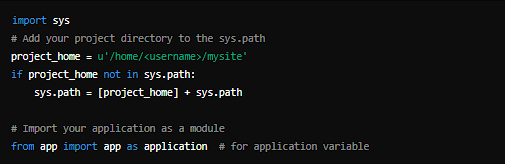
1. Pilih Manual configuration untuk konfigurasi manual, kecuali jika aplikasi Anda menggunakan framework tertentu seperti Django atau Flask, maka pilihlah sesuai framework yang digunakan.
2. Pilih versi Python yang sesuai dengan aplikasi Anda, lalu klik Next.

### 4.3.4. Meng-upload Kode Sumber Aplikasi

1. Kembali ke dashboard, klik tab Files.
2. Navigasikan ke direktori home/<username>/mysite/ (atau direktori lain yang sesuai dengan aplikasi Anda).
3. Klik tombol Upload a file dan pilih file atau folder yang ingin Anda upload dari komputer Anda. Ulangi proses ini hingga semua file aplikasi Anda ter-upload.

### 4.3.5. Mengkonfigurasi Aplikasi Web

1. Kembali ke tab Web.
2. Di bawah bagian Code, pastikan WSGI configuration file menunjuk ke file WSGI aplikasi Anda. Biasanya ini adalah file mysite\_wsgi.py.
3. Klik nama file WSGI untuk mengeditnya, lalu pastikan isi file sesuai dengan konfigurasi aplikasi Anda. Contoh isi file WSGI untuk aplikasi Flask:



### 4.3.6. Mengatur Virtualenv (Opsional)

1. Jika aplikasi Anda memerlukan virtual environment, pastikan virtualenv sudah ter-upload ke direktori home Anda.
2. Di tab Web, di bagian Virtualenv, masukkan path ke virtualenv Anda. Misalnya /home/<username>/myvirtualenv/.

### 4.3.7. Menginstal Paket yang Diperlukan

1. Buka konsol bash dengan klik tab Consoles dan pilih Bash.
2. Aktifkan virtualenv Anda (jika ada) dengan perintah
3. Install paket yang diperlukan menggunakan pip



### 4.3.8. Menjalankan Aplikasi

1. Kembali ke tab Web.
2. Klik tombol Reload <username>.pythonanywhere.com.

## 4.4. Penutup

Selamat, aplikasi sekarang telah berhasil di-upload dan dijalankan di PythonAnywhere. Anda dapat mengaksesnya melalui URL yang diberikan di tab Web.