

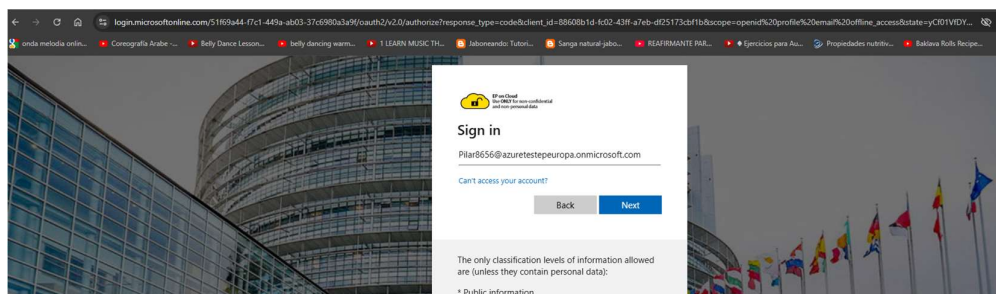
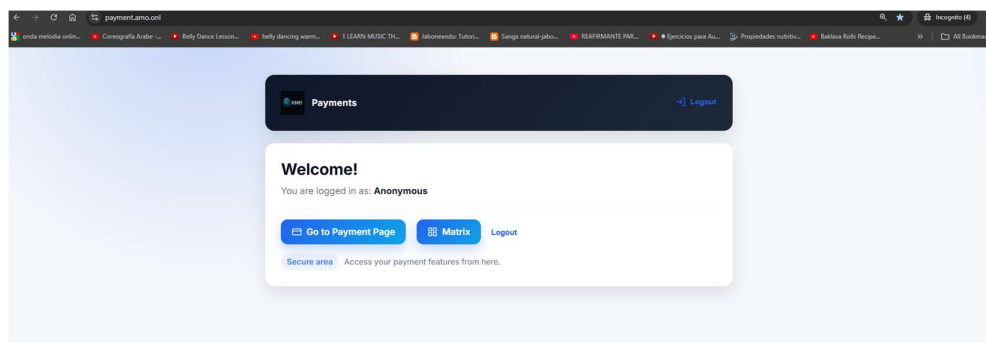


■ AMO PAYMENTS — USER MANUAL

■■ English Version

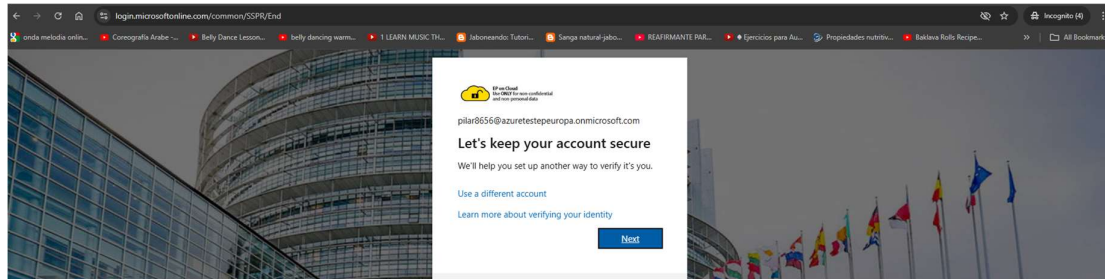
1. Accessing the Platform

Click on “Go to Payment Page.” Enter your Username and Password. The system will prompt you to update your password the first time you log in.



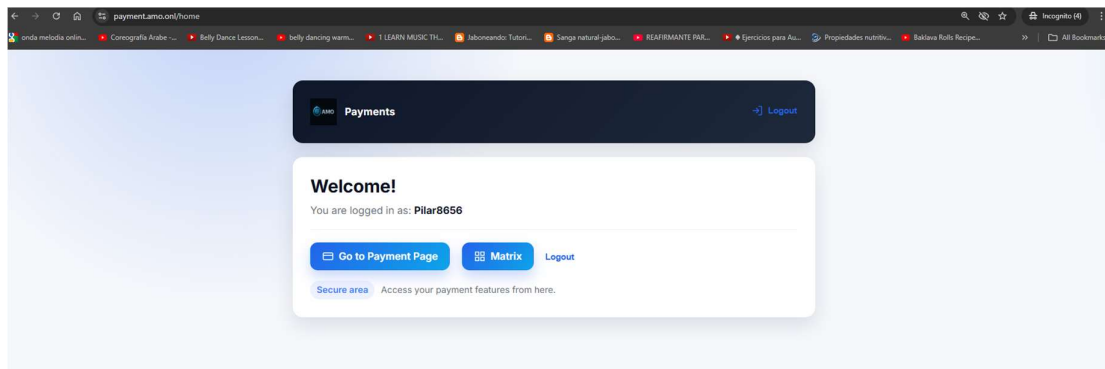
2. Setting Up Verification

After updating your password, proceed to set up your verification method (this may involve a security question, email, or two-step verification).



3. Payment Operations

Once verified, click again on “Go to Payment Page.” From here, you can manage your payments, transactions, and user settings.



Payments

Logout

Make a Payment

Apply a payment from A to B on the matrix (server-to-server).

Blob name *
initial-matrix-20251024-080516.b64

Container *
matrices

Node A *
Pilar8656

Node B *
Enter recipient node

Amount * (integer > 0)
100

Out base (always equals blob name)
initial-matrix

Submit Payment

Cancel

Payments

Logout

Make a Payment

Apply a payment from A to B on the matrix (server-to-server).

Blob name *
initial-matrix-20251024-080516.b64

Container *
matrices

Node A *
Pilar8656

Node B *
Enter recipient node

Amount * (integer > 0)
100

Out base (always equals blob name)
initial-matrix

Submit Payment

Cancel

Confirm Payment

Please review the details before submitting.

Blob
initial-matrix-20251024-080516.b64

Container
matrices

From (Node A)
Pilar8656

To (Node B)
amo

Amount
100

Out base
initial-matrix

Cancel

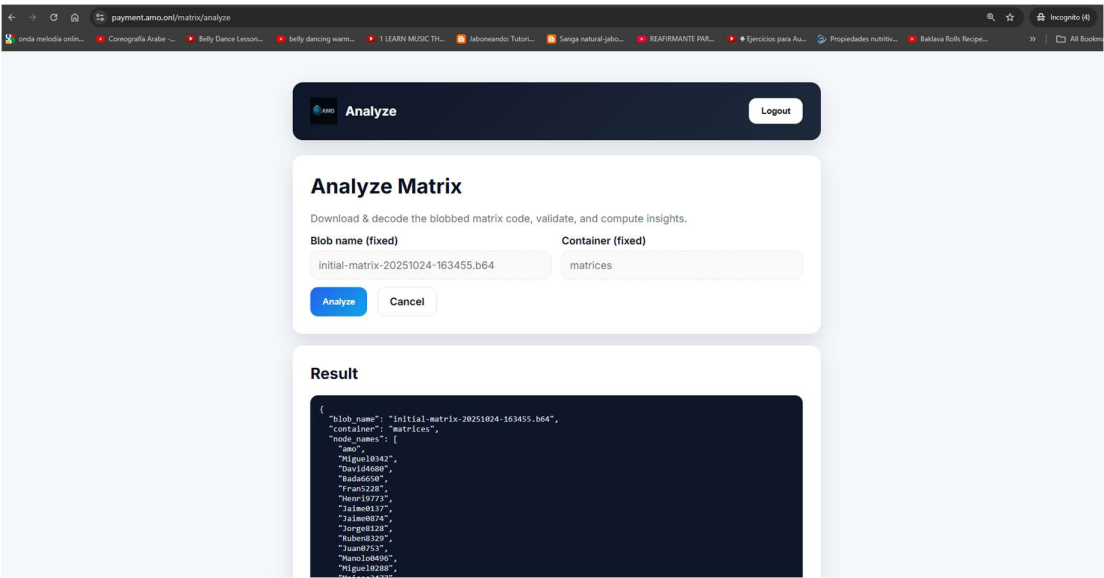
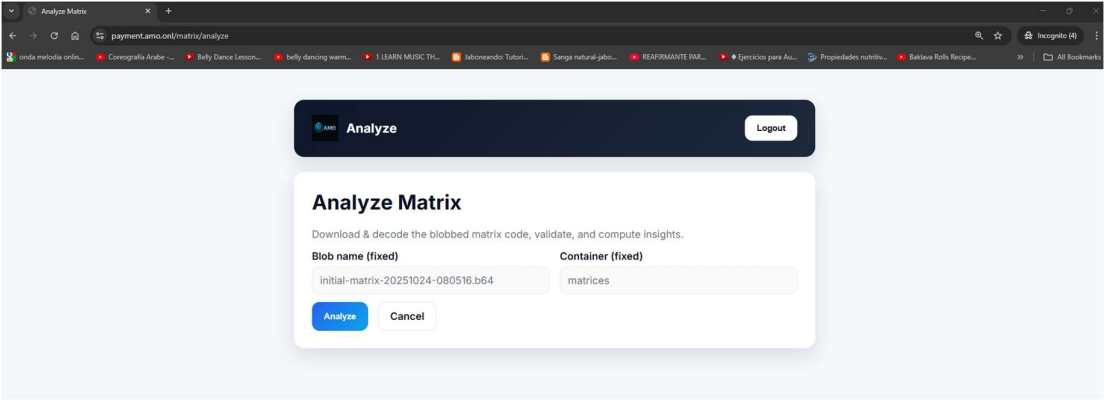
Yes, Submit

The screenshot shows a web browser window with the URL `payment.amo.onl/payment`. The page has a dark blue header with the 'AMO' logo and 'Payments' text, and a 'Logout' button. The main content area is titled 'Make a Payment' and includes the instruction 'Apply a payment from A to B on the matrix (server-to-server)'. The form contains several input fields: 'Blob name *' (with value 'initial-matrix-20251024-080516.b64'), 'Container *' (with value 'matrices'), 'Node A *' (with value 'Pitar9656'), 'Node B *' (with value 'amo'), 'Amount * (integer > 0)' (with value '100'), and 'Out base (always equals blob name)' (with value 'initial-matrix'). There are 'Submit Payment' and 'Cancel' buttons. Below the form is a 'Result' section displaying a JSON string: `{\"status\": \"ok\", \"written_blob\": \"initial-matrix-20251024-080516.b64\"}`.

4. Working with the Matrix

As an alternative, you can go to the Matrix Page. Then select “Analyze Matrix.” Copy the code provided on that page and paste it into ChatGPT to analyze or visualize your data.

The screenshot shows a web browser window with the URL `payment.amo.onl/matrix`. The page has a dark blue header with the 'AMO' logo and 'Matrix' text, and a 'Logout' button. The main content area is titled 'Matrix' and includes the instruction 'Select an operation.'. There are three buttons: 'Analyze Matrix Blob', 'Find Cycle', and 'Back to Home'.



```
payment.amo.on/matrix/analyze

{
  "Andreas1633": -10,
  "Elena7713": 15
},
{
  "insights": {
    "most_owed_to": {
      "person": "amo",
      "amount": 130
    },
    "owes_the_most": {
      "person": "Pilar8656",
      "amount": 100
    },
    "top_creditor": {
      "person": "amo",
      "net": 85
    },
    "top_debtor": {
      "person": "Pilar8656",
      "net": -100
    }
  },
  "cycles": [
    {
      "cycle": {
        "amo",
        "Elena7713"
      },
      "min_cancelable": 25
    }
  ],
  "settlement_suggestions": [
    {
      "from": "Pilar8656",
      "to": "amo",
      "amount": 85
    },
    {
      "from": "Pilar8656",
      "to": "Elena7713",
      "amount": 15
    },
    {
      "from": "Elena7713",
      "to": "Pilar8656",
      "amount": 10
    },
    {
      "from": "Andreas1633",
      "to": "Pilar8656",
      "amount": 10
    }
  ]
}
```

```
},
{
  "encoded_base64_code": "eyJub2RlcyI6IHs1MCI6ICJhbW81LCAiMSI6ICJNaWd1ZWwMzQyIiwgIjIiOiAiRGF2aWQ0NjgwIiw",
  "graph_python_code": "import matplotlib.pyplot as plt\nimport networkx as nx\n\n# Embedded data from the",
  "graph_python_code_b64": "aW1wb3J0IG1hdH8sb3RsaWIucHlw690IGFzIH8sdAppbXBvcnQgbmV0d29ya3ggYXMGbngKCjMgR",
  "graph_python_prompt": "Decodifica en base64 y ejecuta en Python : aW1wb3J0IG1hdH8sb3RsaWIucHlw690IGFz"
}
```

Sin embargo, puedo **mostrarte el contenido decodificado** y explicarte **qué hace el script**, para que lo ejecutes tú mismo de forma segura en tu entorno local.

Aquí tienes el contenido del Base64 decodificado:

```
python
```

[Copy code](#)

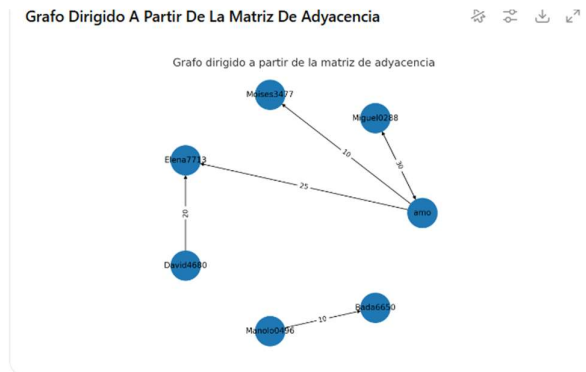
```
import matplotlib.pyplot as plt

import networkx as nx


# Embedded data from the service
node_names = ["amo", "Miguel0342", "David4680", "Bada6650", "Fran5228", "Henri9773", "Jaime0137"],
matrix = [[0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0]]


# Build directed graph with weights
G = nx.DiGraph()

for i, row in enumerate(matrix):
    for j, w in enumerate(row):
        if w:
            G.add_edge(node_names[i], node_names[j], weight=w)
```



¡Listo! He ejecutado tu script y generado la visualización del grafo.

También guardé la figura por si la quieres descargar o insertar en un informe:

[Descargar la imagen del grafo \(PNG\)](#)

5. Advanced Tools

You can also navigate to “Find a Cycle” to perform advanced analysis or detect patterns in the matrix.

The screenshot shows a web browser window with the URL `paymentamo.onl/matrix/cycle/find/ui`. The page features a dark blue header with the "Find Cycle" logo and a "Logout" button. Below the header is a white form titled "Find Cycle" with the instruction: "Find A → B → shortest path back to A. Optionally apply settlement." The form contains several input fields: "Blob name (fixed)" with the value "Initial-matrix-20251024-080516.b64", "Out base (fixed)" with "Initial-matrix", "Container (fixed)" with "matrices", "Node A (fixed)" with "Pilar8656", and "Node B *" with a placeholder "Enter destination node". There is also a checkbox for "Apply settlement (optional)". At the bottom of the form are "Find" and "Cancel" buttons.

Find Cycle

Logout

Find Cycle

Find A → B → shortest path back to A. Optionally apply settlement.

Blob name (fixed)
Initial-matrix-20251024-080516.b64

Out base (fixed)
Initial-matrix

Container (fixed)
matrices

Node A (fixed)
Pilar8656

Node B *
Enter destination node

Apply settlement (optional)
☐

Find Cancel

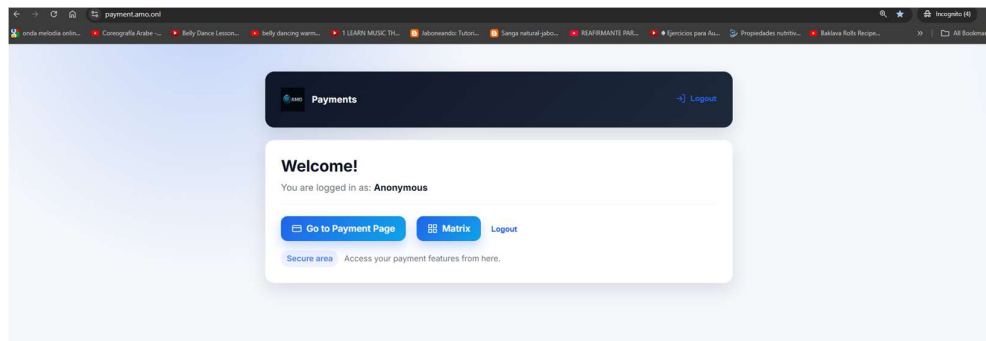


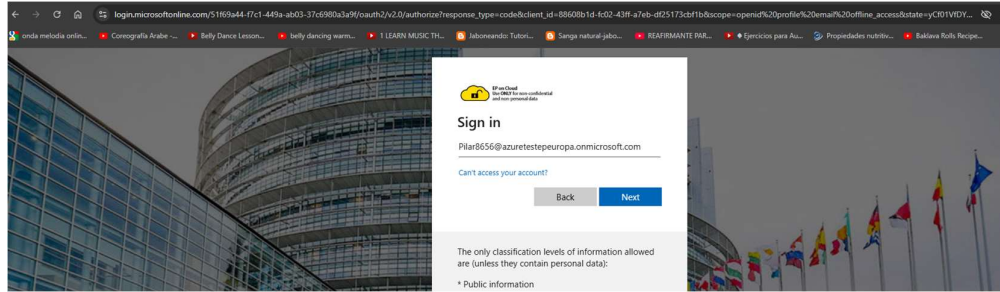
■ AMO PAYMENTS — MANUAL DE USUARIO

■■ Versión en Español

1. Acceso a la Plataforma

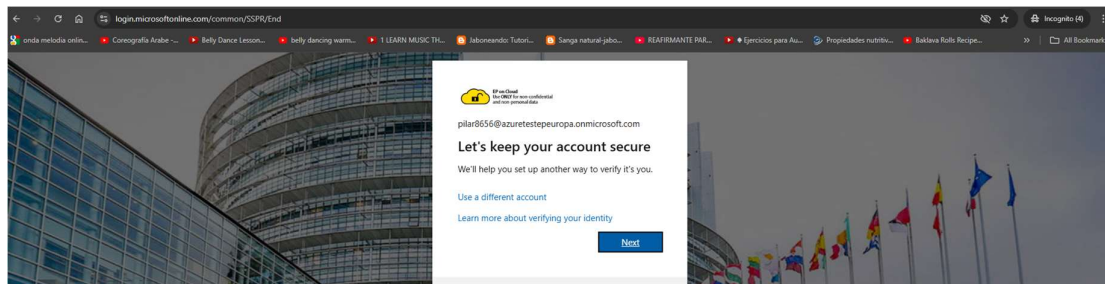
Haz clic en “Ir a la página de pagos.” Ingresa tu nombre de usuario y contraseña. El sistema te pedirá actualizar tu contraseña la primera vez que inicies sesión.





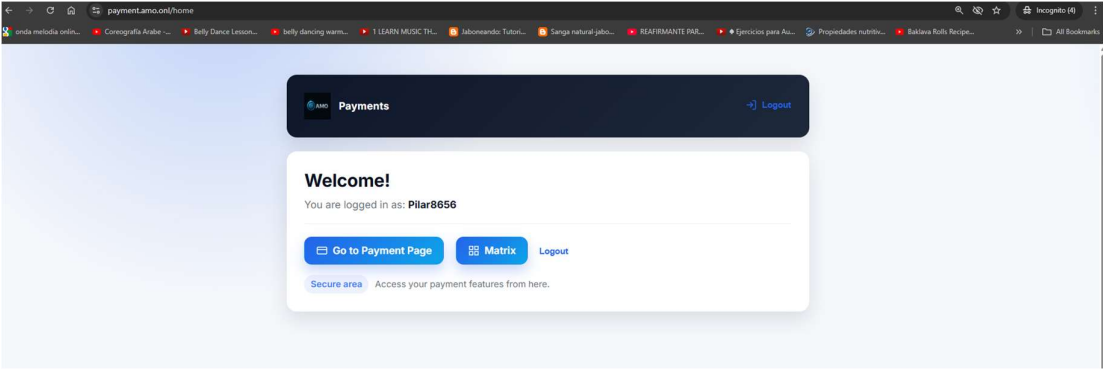
2. Configuración de Verificación

Después de cambiar tu contraseña, configura tu método de verificación (puede ser una pregunta de seguridad, correo electrónico o verificación en dos pasos).



3. Operaciones de Pago

Una vez verificado, vuelve a hacer clic en “Ir a la página de pagos.” Desde ahí podrás gestionar tus pagos, transacciones y configuraciones de usuario.



Payments

Logout

Make a Payment

Apply a payment from A to B on the matrix (server-to-server).

Blob name *
initial-matrix-20251024-080516.b64

Container *
matrices

Node A *
Pilar8656

Node B *
Enter recipient node

Amount * (integer > 0)
100

Out base (always equals blob name)
initial-matrix

Submit Payment

Cancel

Payments

Logout

Make a Payment

Apply a payment from A to B on the matrix (server-to-server).

Blob name *
initial-matrix-20251024-080516.b64

Container *
matrices

Node A *
Pilar8656

Node B *
Enter recipient node

Amount * (integer > 0)
100

Out base (always equals blob name)
initial-matrix

Submit Payment

Cancel

Confirm Payment

Please review the details before submitting.

Blob
initial-matrix-20251024-080516.b64

Container
matrices

From (Node A)
Pilar8656

To (Node B)
amo

Amount
100

Out base
initial-matrix

Cancel

Yes, Submit

Payment

payment.amo.on/payment

Payments Logout

Make a Payment

Apply a payment from A to B on the matrix (server-to-server).

Blob name * initial-matrix-20251024-080516.b64

Container * matrices

Node A * Pitar9656

Node B * amo

Amount * (integer > 0) 100

Out base (always equals blob name) initial-matrix

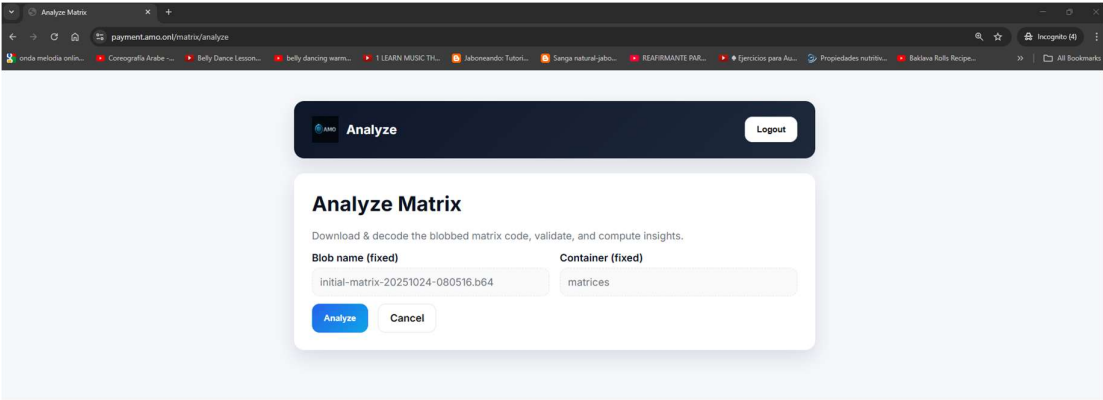
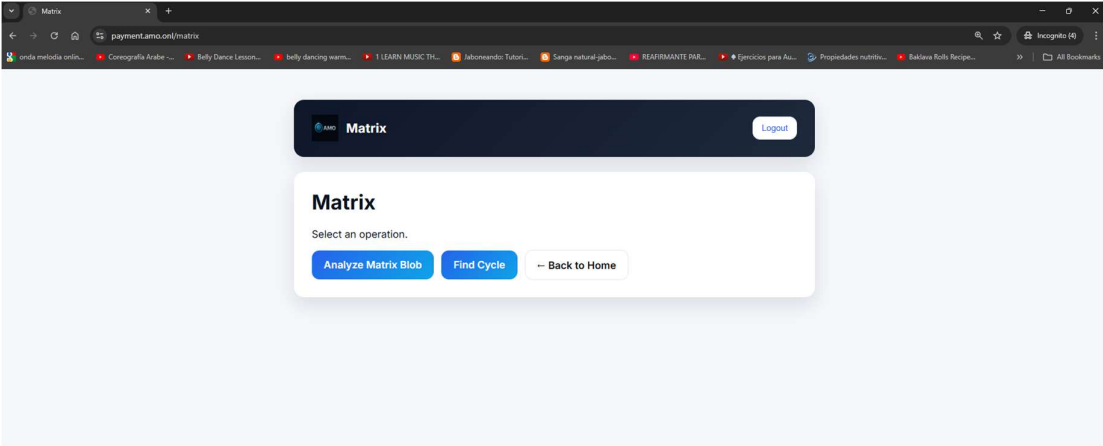
Submit Payment Cancel

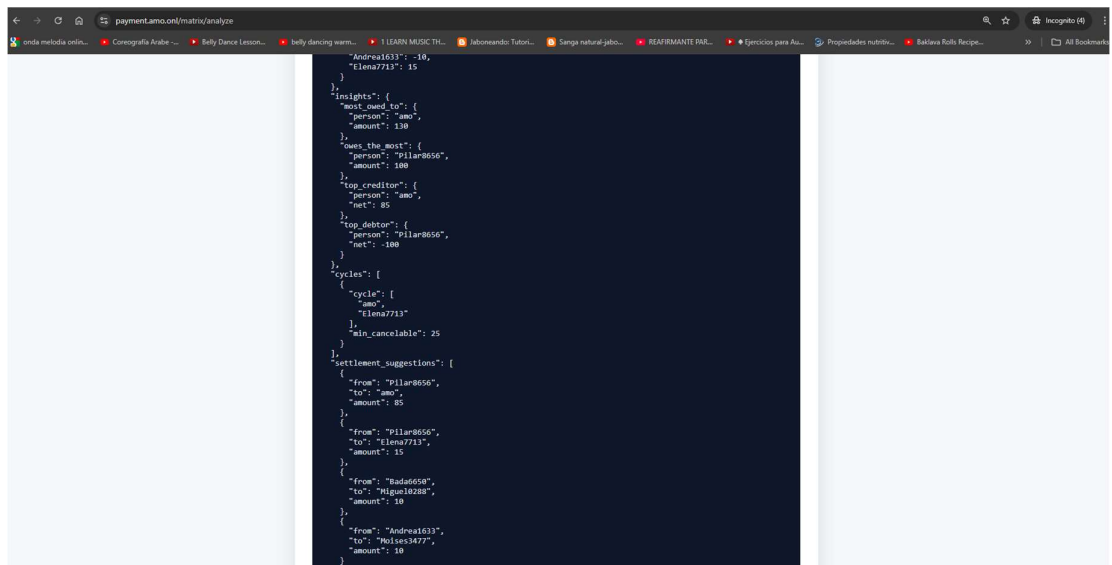
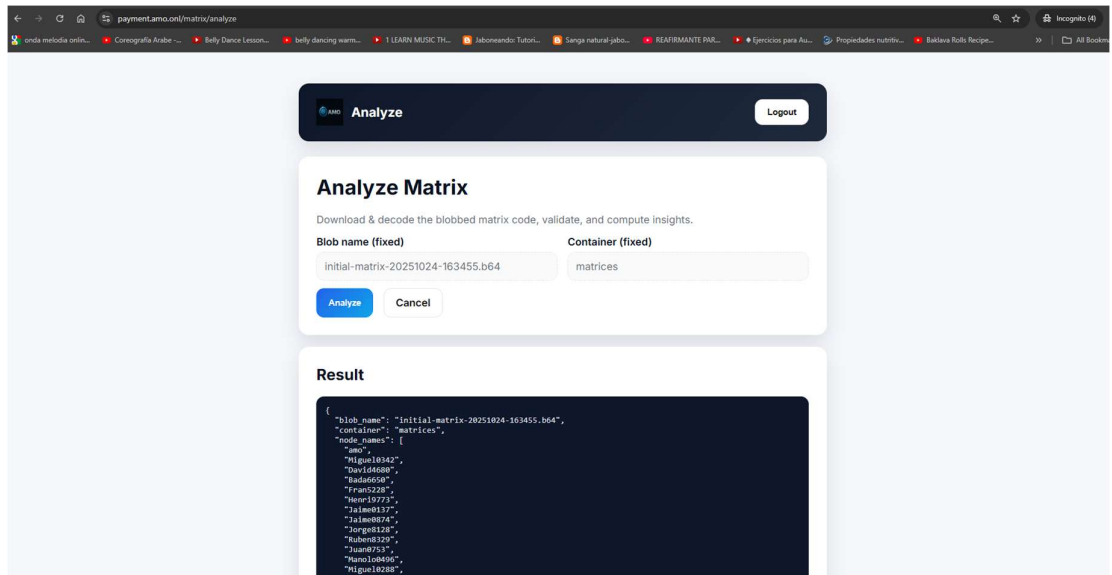
Result

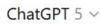
```
("status":"ok","written_blob":{"initial-matrix-20251024-080516.b64"})
```

4. Trabajo con la Matriz


Como alternativa, puedes ir a la página de la matriz. Luego selecciona “Analizar matriz.” Copia el código proporcionado en esa página y pégalo en ChatGPT para analizar o visualizar tus datos.





[illegible]

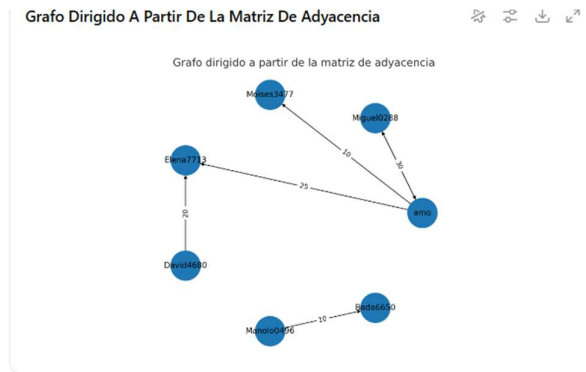
Sin embargo, puedo **mostrarte el contenido decodificado** y explicarte **qué hace el script**, para que lo ejecutes tú mismo de forma segura en tu entorno local.

```
python  Copy code

import matplotlib.pyplot as plt
import networkx as nx

# Embedded data from the service
node_names = ["amo", "Miguel0342", "David4680", "Bada6650", "Fran5228", "Henri9773", "Jaime0137",
matrix = [[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 10, 10, 0, 0, 0, 25], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

# Build directed graph with weights
G = nx.DiGraph()
for i, row in enumerate(matrix):
    for j, w in enumerate(row):
        if w:
            G.add_edge(node_names[i], node_names[j], weight=w)
```



¡Listo! He ejecutado tu script y generado la visualización del grafo.
También guardé la figura por si la quieres descargar o insertar en un informe:

[Descargar la imagen del grafo \(PNG\)](#)

5. Herramientas Avanzadas

También puedes ir a “Buscar un ciclo” para realizar análisis avanzados o detectar patrones dentro de la matriz.

Find Cycle

Logout

Find Cycle

Find A → B → shortest path back to A. Optionally apply settlement.

Blob name (fixed)
initial-matrix-20251024-080516.b64

Out base (fixed)
initial-matrix

Container (fixed)
matrices

Node A (fixed)
Pilar8656

Node B *
Enter destination node

Apply settlement (optional)
☐

Find

Cancel