



# DESPLIEGUE DE SOLUCIONES ANALITICAS – **MANUAL DE INSTALACIÓN**

## INTEGRANTES

- Daniel Arturo Martínez Morales
- Daniel Steven Vargas Guzmán
- José Fernando Duarte Alvarado
- Luis Miguel Ortega Cañate



## Contenido

Manual de Instalación .....	3
Tablero de Preevaluación de Campañas.....	3
Descarga de archivos necesarios para la instalación.....	3
Configuración de repositorios base en GitHub .....	3
Configuración de proyectos en Railway .....	7
Solución de Problemas .....	12
Glosario de Términos.....	12
Contacto y Soporte .....	13

## Lista de ilustraciones

Ilustración 1, Carpeta raíz .....	3
Ilustración 2, Menú para creación de un nuevo repositorio .....	4
Ilustración 3, Configuración del nuevo repositorio .....	5
Ilustración 4, URL del repositorio .....	6
Ilustración 5, Carpeta raíz con las carpetas de los repositorios.....	7
Ilustración 6, Creación de nuevo proyecto en Railway .....	7
Ilustración 7, Despliegue de proyecto desde repositorio .....	8
Ilustración 8, Building del API .....	8
Ilustración 9, URL de la API desplegada.....	9
Ilustración 10, API desplegada .....	9
Ilustración 11, Edición del archivo app.py en el repositorio del tablero.....	10
Ilustración 12, Despliegue del tablero en Railway .....	11
Ilustración 13, Generación de URL publica para el tablero en Railway .....	11
Ilustración 14, Tablero desplegado .....	12



## Manual de Instalación

### Tablero de Preevaluación de Campañas

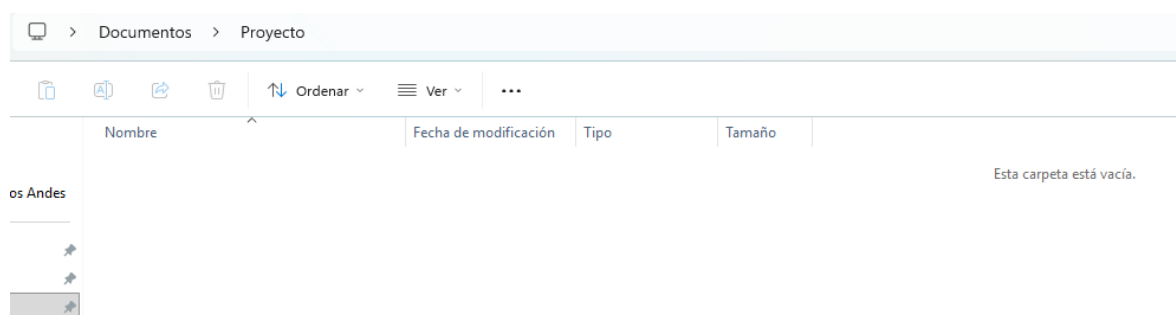
## Descarga de archivos necesarios para la instalación

Para la instalación de este tablero en su sistema deberá contar con los archivos base, estos deben ser descargados desde el siguiente link:

<https://github.com/darturo12/Proyecto-Despliegue/blob/master/Despliegue-API-Tablero.zip>

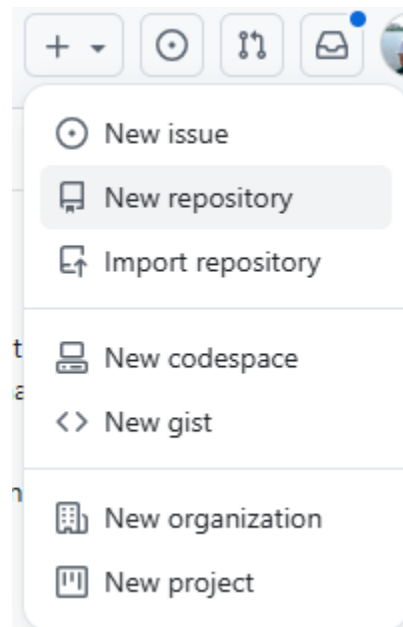
## Configuración de repositorios base en GitHub

1. Cree una carpeta raíz vacía, esta servirá de base del proyecto



*Ilustración 1, Carpeta raíz*

2. Cree dos repositorios vacíos en su cuenta de GitHub (<https://github.com/>) uno para los archivos de la API y otro para los archivos del tablero. A continuación se muestra como crear un repositorio vacío en GitHub:
  - a. Ingrese en su cuenta de GitHub y en el menú superior derecho seleccione el botón con símbolo “+” y luego de clic en “**New Repository**”



*Ilustración 2, Menú para creación de un nuevo repositorio*

- b. Asigne un nombre al repositorio, deje las opciones por defecto tal como aparecen y seleccione “**Create repository**”. Recuerde que se deben crear 2 repositorios, uno referido ala API y otro referido al tablero por tanto se recomienda crear nombres que hagan fácil distinguirlos.



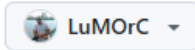
## Create a new repository

Repositories contain a project's files and version history. Have a project elsewhere? [Import a repository](#).

Required fields are marked with an asterisk (\*).

### 1 General

Owner \*



Repository name \*

Proyecto-API

✓ Proyecto-API is available.

Great repository names are short and memorable. How about **legendary-dollop**?

Description

0 / 350 characters

### 2 Configuration

Choose visibility \*

Choose who can see and commit to this repository

 Public ▾

Add README

READMEs can be used as longer descriptions. [About READMEs](#)

Off ☐

Add .gitignore

.gitignore tells git which files not to track. [About ignoring files](#)

No .gitignore ▾

Add license

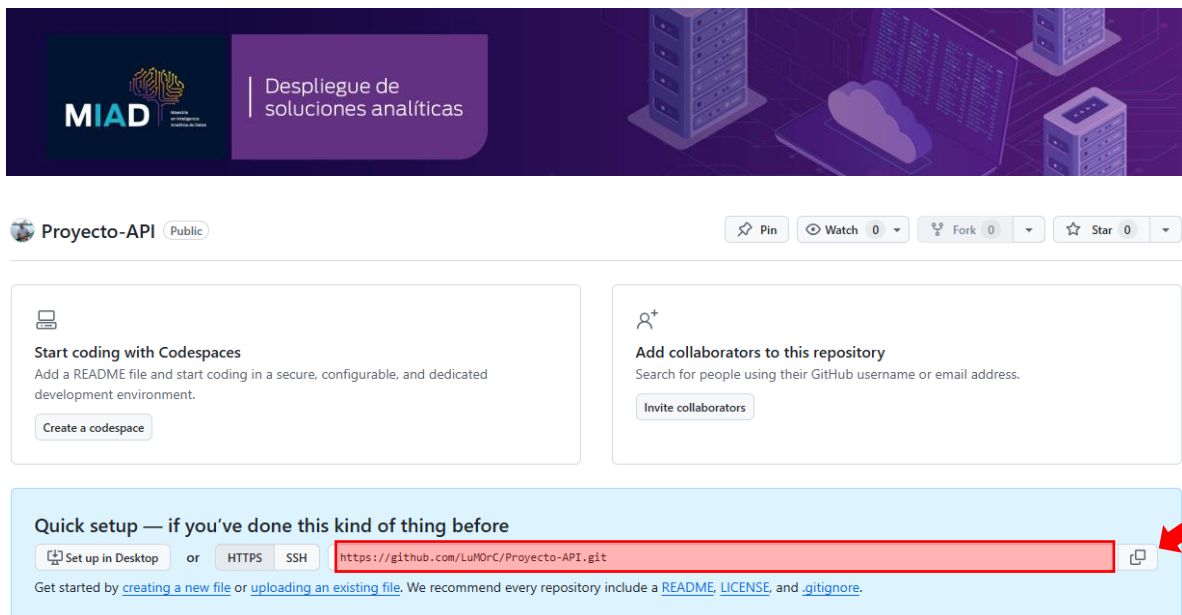
Licenses explain how others can use your code. [About licenses](#)

No license ▾

Create repository

Ilustración 3, Configuración del nuevo repositorio

- c. Una vez creado el repositorio copie la dirección HTTPS



*Ilustración 4, URL del repositorio*

3. Abra una terminal desde la carpeta raíz del proyecto y clone el repositorio vacío que acabamos de crear:

```
git clone https://github.com/usuario/nombre-repo.git
```

4. El comando anterior crea una carpeta en la carpeta raíz del proyecto con el nombre del repositorio. Entre a la carpeta y copie los archivos dentro de esa carpeta.

**NOTA:** Para este proyecto se proveen 2 carpetas de trabajo, una para la API y otra para el tablero. Se deben copiar cada conjunto de archivos en sus correspondientes repositorios.

5. Vuelva a la terminal ingrese a la carpeta del repositorio y agregue los archivos al stage y luego haga un commit:

```
cd nombre-repo
git add .
git commit -m "Initial commit"
```

6. Enviar a GitHub:

```
git push -u origin main
```

7. Repita estos pasos para el repositorio del tablero

Al finalizar estos pasos su carpeta raíz deberá lucir así:

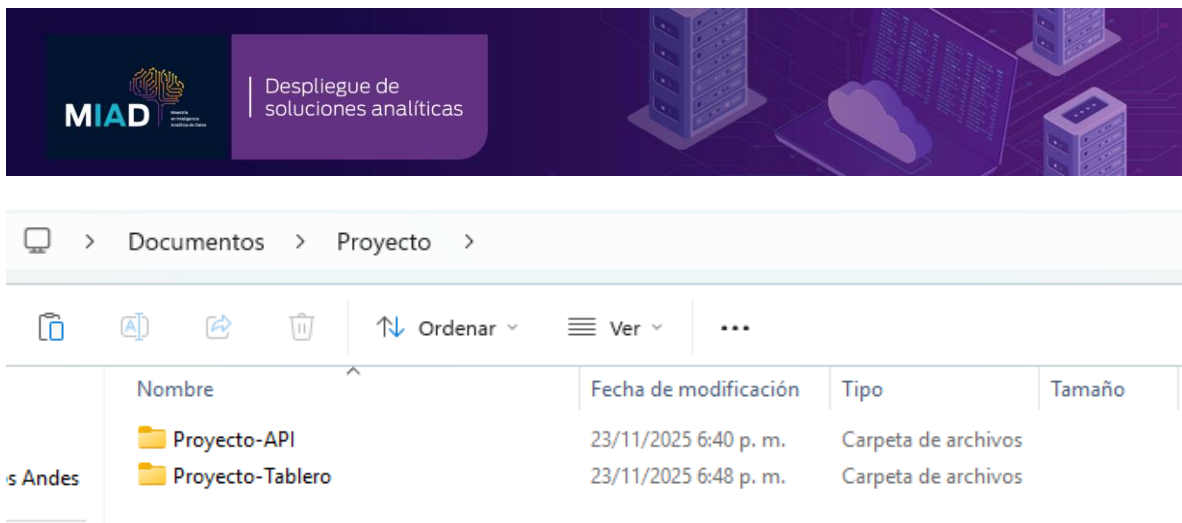


Ilustración 5, Carpeta raíz con las carpetas de los repositorios.

## Configuración de proyectos en Railway

1. Ingrese a su cuenta en Railway (<https://railway.com/dashboard>) si no tiene una cuenta créela.
2. En su navegador asegúrese de tener abiertas sus cuentas de GitHub y Railway.
3. En el navegador vaya a su cuenta de Railway y enlace su cuenta con Github: click en “**New project**”, seleccione “**Deploy from GitHub repo**”, click en “**Configure GitHub App**”.
4. Seleccione la cuenta de Github donde instalará la aplicación Railway. Permita acceso a todos los repositorios. Confirme usando su método preferido de verificación.
5. Con su cuenta ya configurada y enlazada con GitHub, cree un nuevo proyecto. Click en “**New Project**”, seleccione “**GitHub Repository**”.

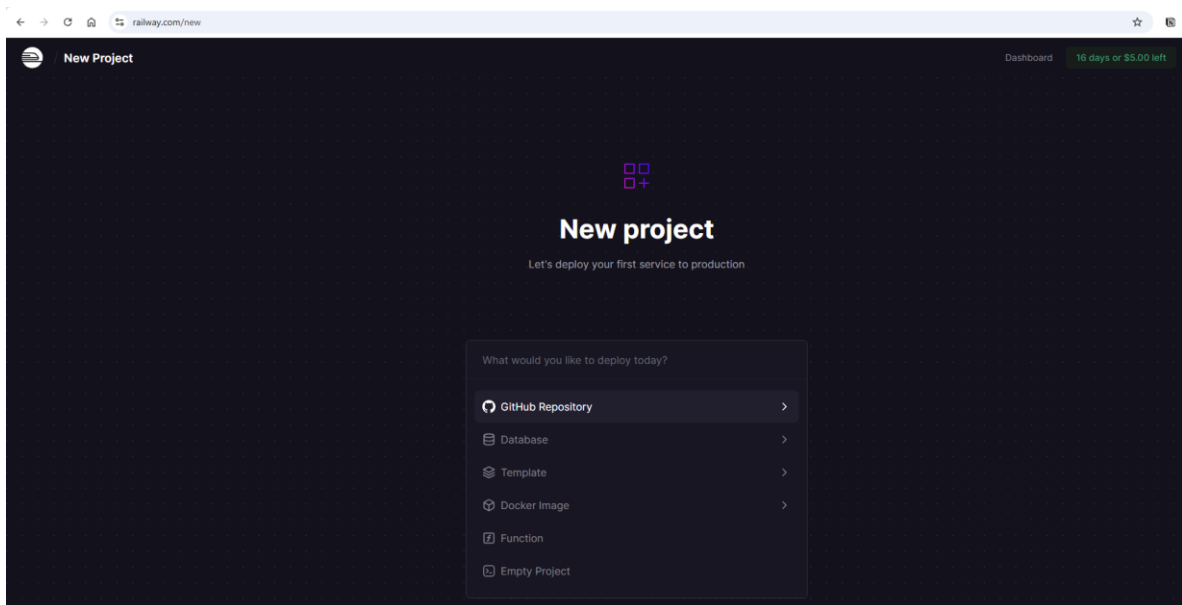
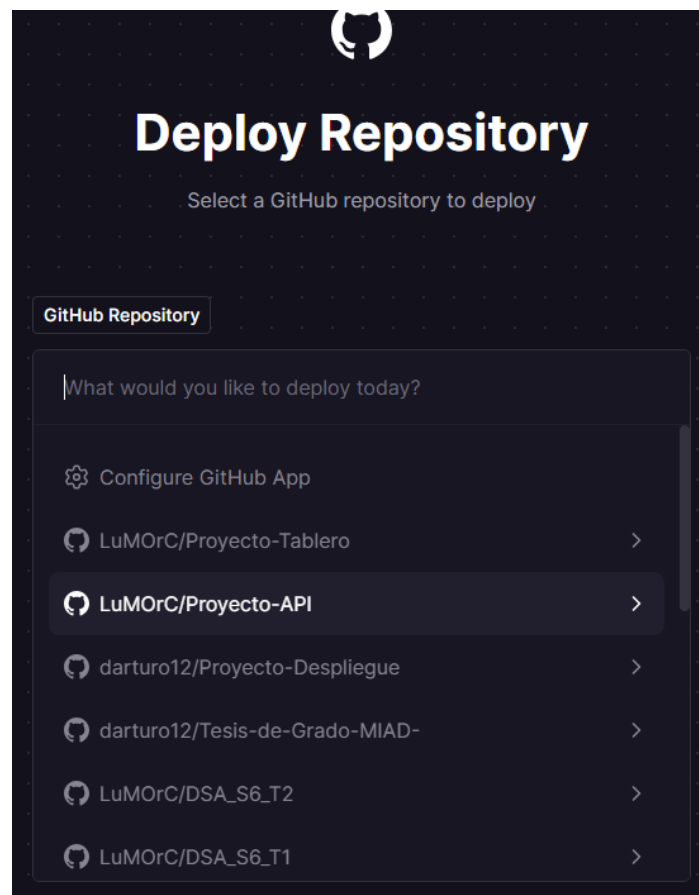


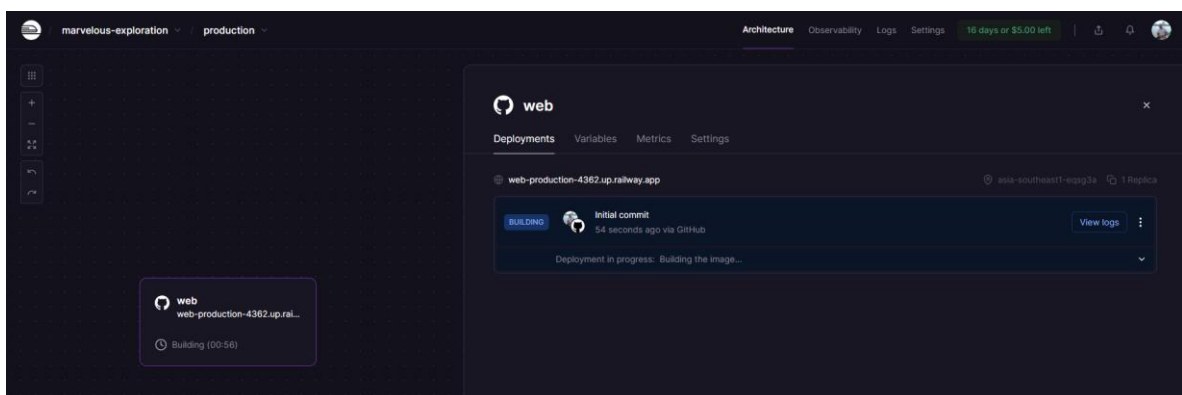
Ilustración 6, Creación de nuevo proyecto en Railway

6. Seleccione el repositorio que creo para la API



*Ilustración 7, Despliegue de proyecto desde repositorio*

Al ejecutar los pasos anteriores Railway automáticamente desplegará la API, esto puede tomar unos minutos.



*Ilustración 8, Building del API*

Una vez el estatus del despliegue pase de **BUILDING** a **ACTIVE** diríjase a la pestaña “**Settings**” y nave hasta la opción “**Networking**” y luego “**Public Networking**”, allí podrá obtener la URL de la API desplegada.



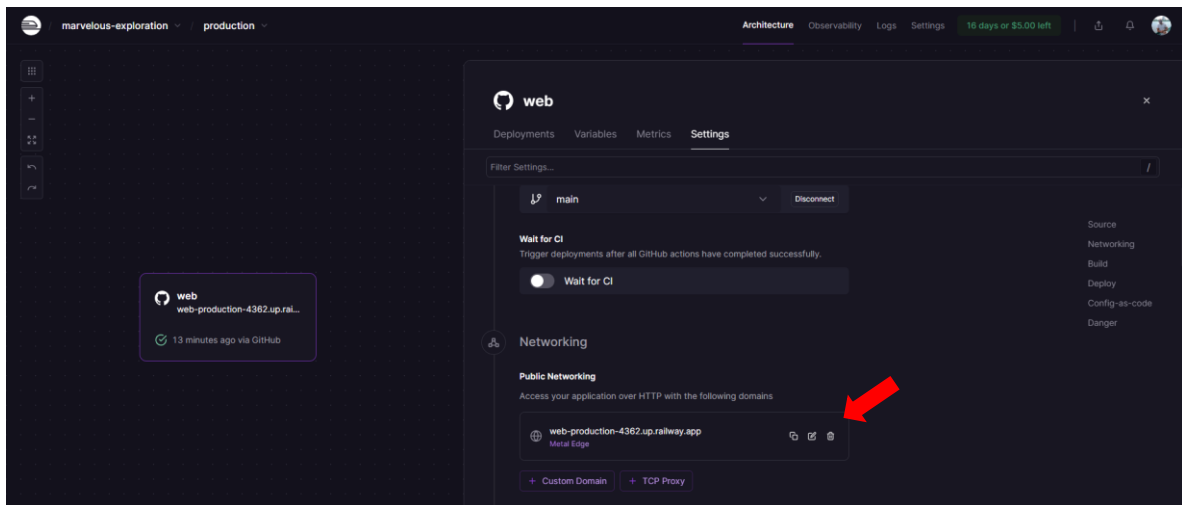


Ilustración 9, URL de la API desplegada

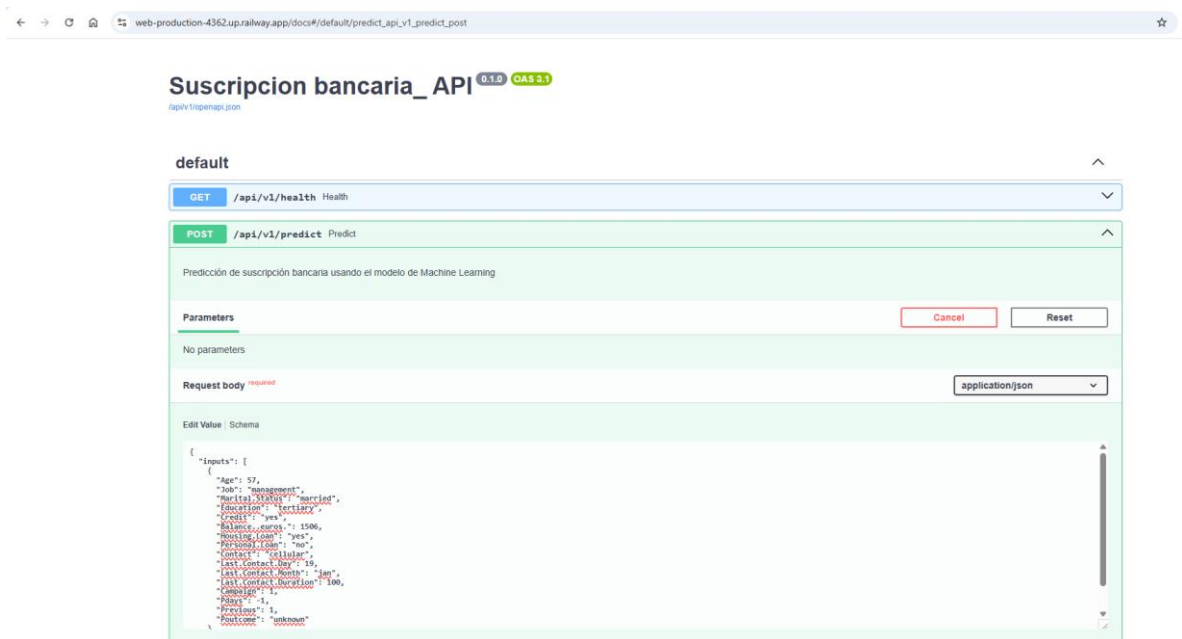


Ilustración 10, API desplegada

7. Copie esta URL y diríjase al repositorio local del tablero
8. Abra y edite el archivo app.py



```
app.py 4 ●
C: > Users > lumor > Documents > Proyecto > Proyecto-Tablero > app.py > ...
1  from dash import Dash, html, dcc, Input, Output, State, ctx
2  import dash_bootstrap_components as dbc
3  import plotly.graph_objects as go
4  import requests
5  import json
6
7  app = Dash(__name__, external_stylesheets=[dbc.themes.BOOTSTRAP])
8
9  # URL de tu API en Railway (REEMPLAZA CON TU URL REAL)
10 API_URL = "web-production-4362.up.railway.app"
11
12 COLORS = {
13     'primary': '#1e3a8a',
14     'secondary': '#3b82f6',
15     'accent': '#10b981',
16     'background': '#f8fafc',
17     'card': '#ffffff',
18     'text': '#1e293b',
19     'text_light': '#64748b',
20     'border': '#e2e8f0'
21 }
22
```

Ilustración 11, Edición del archivo app.py en el repositorio del tablero

9. Edite la línea 10 del archivo reemplazando la URL del archivo con la que acaba de copiar, añada a la parte final de la URL el siguiente texto: `"/api/v1/predict"`
10. Guarde el archivo.
11. De regreso en la terminal ubíquese en la carpeta del repositorio del tablero y registre el cambio en su repositorio remoto.

```
git add .
git commit -m "Actualización de la URL de la API"
git push
```

12. Cree un nuevo proyecto en Railway pero en esta ocasión escoja el repositorio del tablero.
13. Lo anterior dará inicio al despliegue del tablero. Espere a que el estado pase de BUILDING a ACTIVE
14. Una vez el proyecto este activo diríjase a la pestaña de **"Settings"** luego a **"Network"** y finalmente a **"Private Networking"**

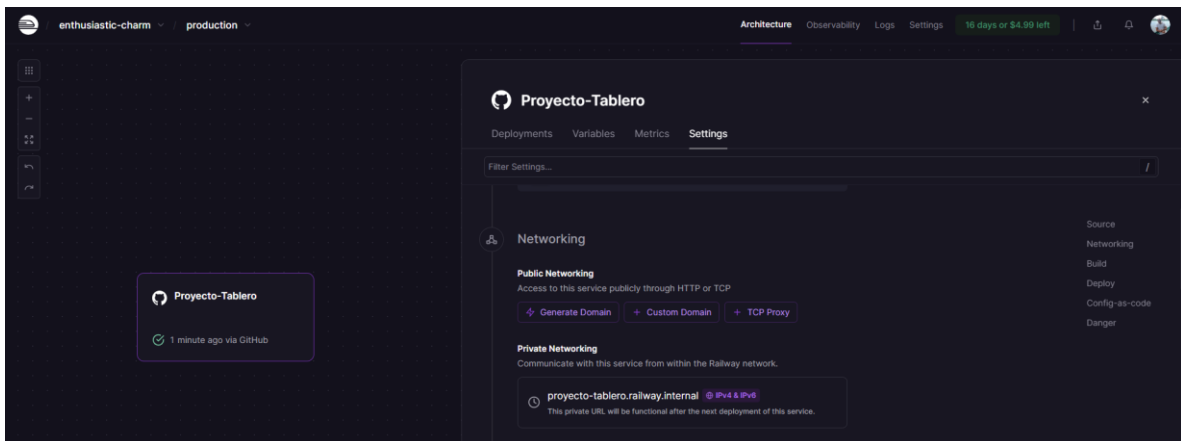


Ilustración 12, Despliegue del tablero en Railway

Seleccione la opción “**Generate Domain**”

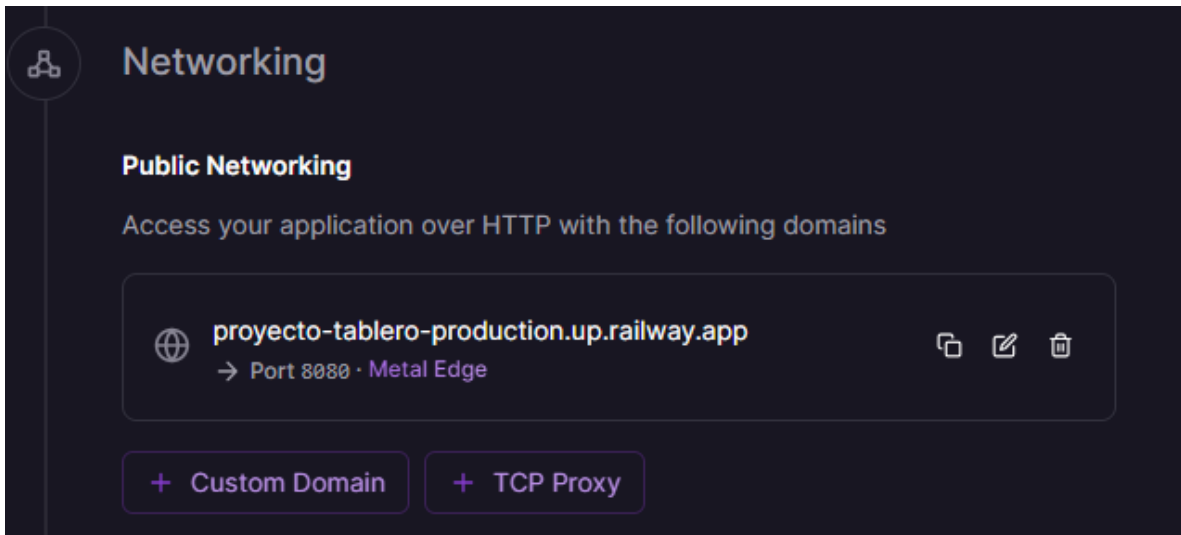


Ilustración 13, Generación de URL publica para el tablero en Railway

15. Ahora tiene el tablero desplegado en la URL generada.

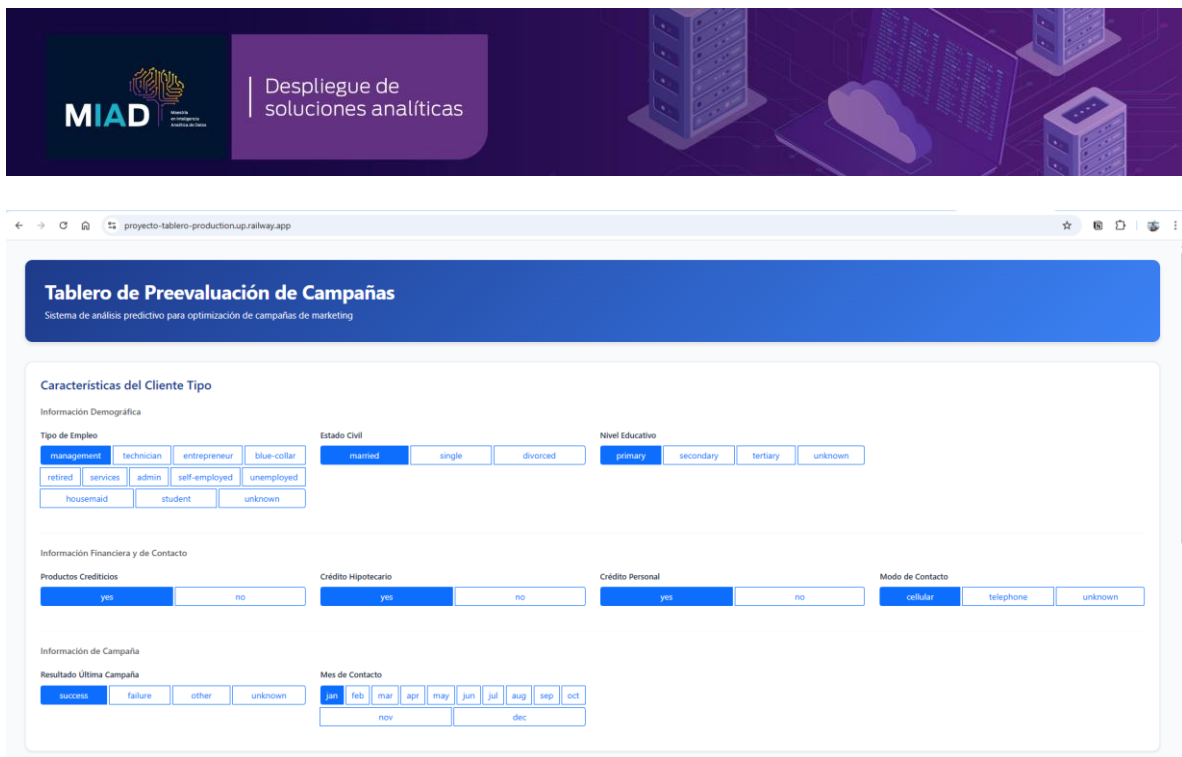


Ilustración 14, Tablero desplegado

## Solución de Problemas

### El botón no cambia de color al hacer clic:

- Verifique que está haciendo clic directamente sobre el botón
- Refresque la página si el problema persiste

### Los sliders no se mueven:

- Asegúrese de arrastrar el círculo deslizante
- Si está usando dispositivo táctil, presione y arrastre con firmeza

### No se muestran los gráficos:

- Verifique su conexión a internet
- Refresque la página
- Asegúrese de que su navegador está actualizado

### La evaluación no responde:

- Verifique que ha seleccionado al menos una opción en cada categoría de botones
- Revise que todos los sliders tienen valores válidos
- Espere unos segundos y vuelva a hacer clic en "Evaluar"

## Glosario de Términos

**Balance:** Saldo promedio en cuenta del cliente

**Contactos Previos:** Número total de veces que se ha contactado al cliente en campañas anteriores

**Duración:** Tiempo en segundos de la última llamada o contacto



**Probabilidad de Suscripción:** Porcentaje que indica qué tan probable es que el cliente acepte el producto

**Segmentación:** Proceso de dividir a los clientes en grupos con características similares

**Sensibilidad:** Análisis de cómo un cambio en una variable afecta el resultado

## Contacto y Soporte

Para asistencia técnica o consultas sobre el uso del tablero, contacte al equipo de soporte de su organización o al administrador del sistema.

**Recuerde:** Este tablero es una herramienta de apoyo para la toma de decisiones. Los resultados deben complementarse con el análisis profesional y el conocimiento del negocio.