



# FinancIA

Jorge Miguel Valero  
Roberto Negrete  
Tania Piña  
Alan Chavez

13 de Octubre del 2024



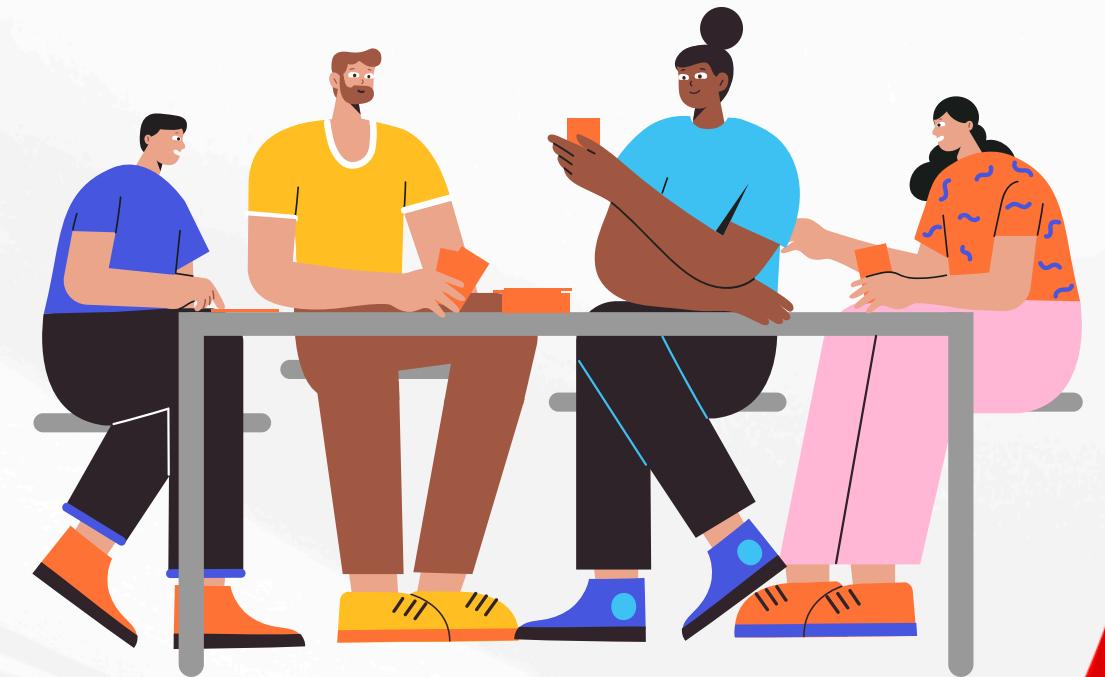
# Índice

- 01** Justificación
- 02** Prototipo Funcional
- 03** User Persona
- 04** Canales de Interacción
- 05** Código Desarrollado
- 06** Documentación
- 07** Conclusiones



# Justificación

El chatbot financiero brinda a los jóvenes una oportunidad de aprendizaje al permitirles formular preguntas específicas y recibir consejos personalizados en un entorno seguro y anónimo. Esto fomenta el desarrollo de habilidades financieras y empodera a los jóvenes para tomar decisiones informadas.



# Prototipo Funcional

## Backend

```
from vertexai.generative_models import GenerativeModel

PROJECT_ID = "gcp-banorte-hackaton-team-3"
vertexai.init(project=PROJECT_ID, location="us-central1")

model = GenerativeModel("gemini-1.5-flash-002")

app = Flask(__name__)
CORS(app) # Habilitamos CORS para permitir solicitudes desde cualquier origen

# ... (Aquí va el código de tu modelo, por ejemplo, el de Vertex AI) ...

# Ruta POST para recibir datos desde FlutterFlow
@app.route('/api/summarize', methods=['POST'])
def summarize():
    try:
        query = request.get_json()

        resp_banorte = search(data_store_id='banorte-datas-id', query=query)
        resp_finanzas = search(data_store_id = 'finanzas-dataset-id', query = query)

        prompt = f"""
Eres un experto en finanzas y buscas proporcionar información al usuario para
mejorar su salud financiera. Evita dar respuestas inventadas si no tienes
información suficiente, y consulta adicionalmente {resp_banorte} y {resp_finanzas}
para obtener información relevante. Usa como país de origen a México.
Usa como idioma de respuesta español.
"""

        #data = request.get_json() # Usamos request.get_json() para obtener los datos JSON
        input_text = query.get('text', '')

        # Lógica de procesamiento en Python (usa tu modelo aquí)
        # ... (reemplaza esto con tu lógica de resumen) ...
        summary = model.generate_content(contents=[prompt])

        return jsonify({"summary": summary})

    except Exception as e:
        return jsonify({'error': str(e)}), 500 # Devolvemos un error 500 si hay algún problema

if __name__ == '__main__':
    app.run(debug=True, host="0.0.0.0", port=5000)
```

información suficiente, y consulta adicionalmente {resp\_banorte} y {resp\_finanzas} para obtener información relevante. Usa como país de origen a México.  
Usa como idioma de respuesta español.  
....

```
#data = request.get_json() # Usamos request.get_json() para obtener los datos JSON
input_text = query.get('text', '')

# Lógica de procesamiento en Python (usa tu modelo aquí)
# ... (reemplaza esto con tu lógica de resumen) ...
summary = model.generate_content(contents=[prompt])

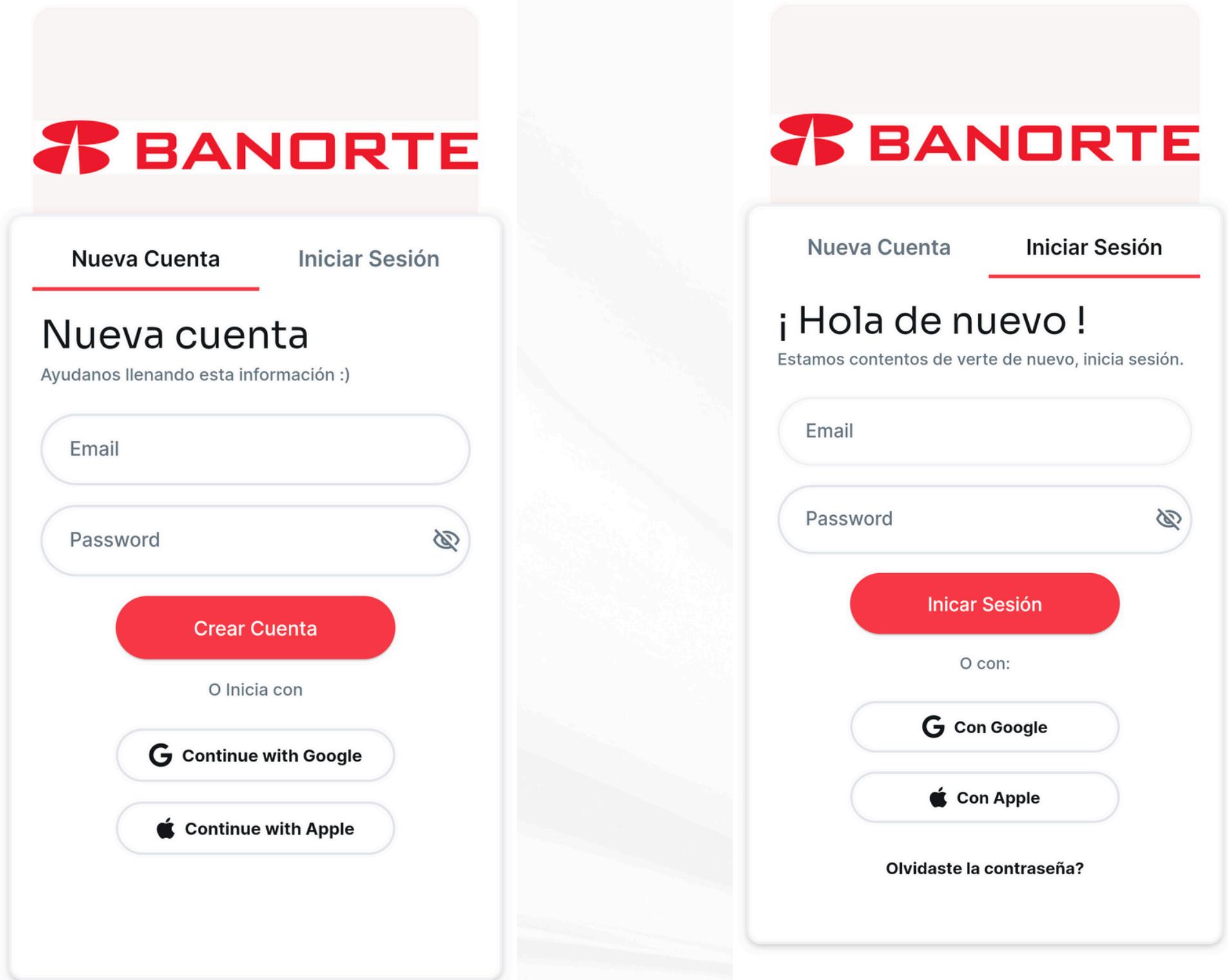
return jsonify({"summary": summary})

except Exception as e:
    return jsonify({'error': str(e)}), 500 # Devolvemos un error 500 si hay algún problema

if __name__ == '__main__':
    app.run(debug=True, host="0.0.0.0", port=5000)
```

# Frontend

## Log In and New Account



The image displays two mobile application screens for Banorte, showing the "Nueva Cuenta" (New Account) and "Iniciar Sesión" (Log In) flows.

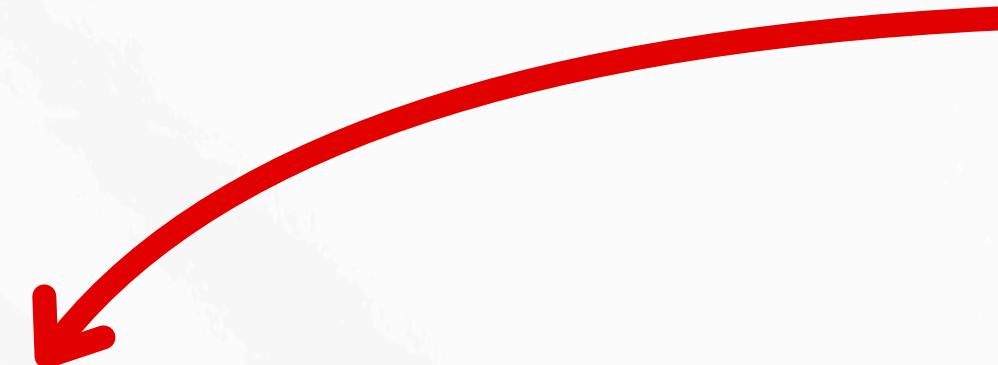
**Nueva Cuenta Screen:**

- Banorte Logo:** At the top center.
- Buttons:** "Nueva Cuenta" (highlighted with a red underline) and "Iniciar Sesión".
- Title:** "Nueva cuenta".
- Text:** "Ayudanos llenando esta información :)"
- Input Fields:** "Email" and "Password" with a visibility toggle icon.
- Primary Action:** "Crear Cuenta" button.
- Secondary Actions:** "O Inicia con" section with "Continue with Google" and "Continue with Apple" buttons.

**Iniciar Sesión Screen:**

- Banorte Logo:** At the top center.
- Buttons:** "Nueva Cuenta" and "Iniciar Sesión" (highlighted with a red underline).
- Title:** "¡Hola de nuevo!".
- Text:** "Estamos contentos de verte de nuevo, inicia sesión."
- Input Fields:** "Email" and "Password" with a visibility toggle icon.
- Primary Action:** "Iniciar Sesión" button.
- Secondary Actions:** "O con:" section with "Con Google" and "Con Apple" buttons.
- Text:** "Olvidaste la contraseña?" at the bottom.

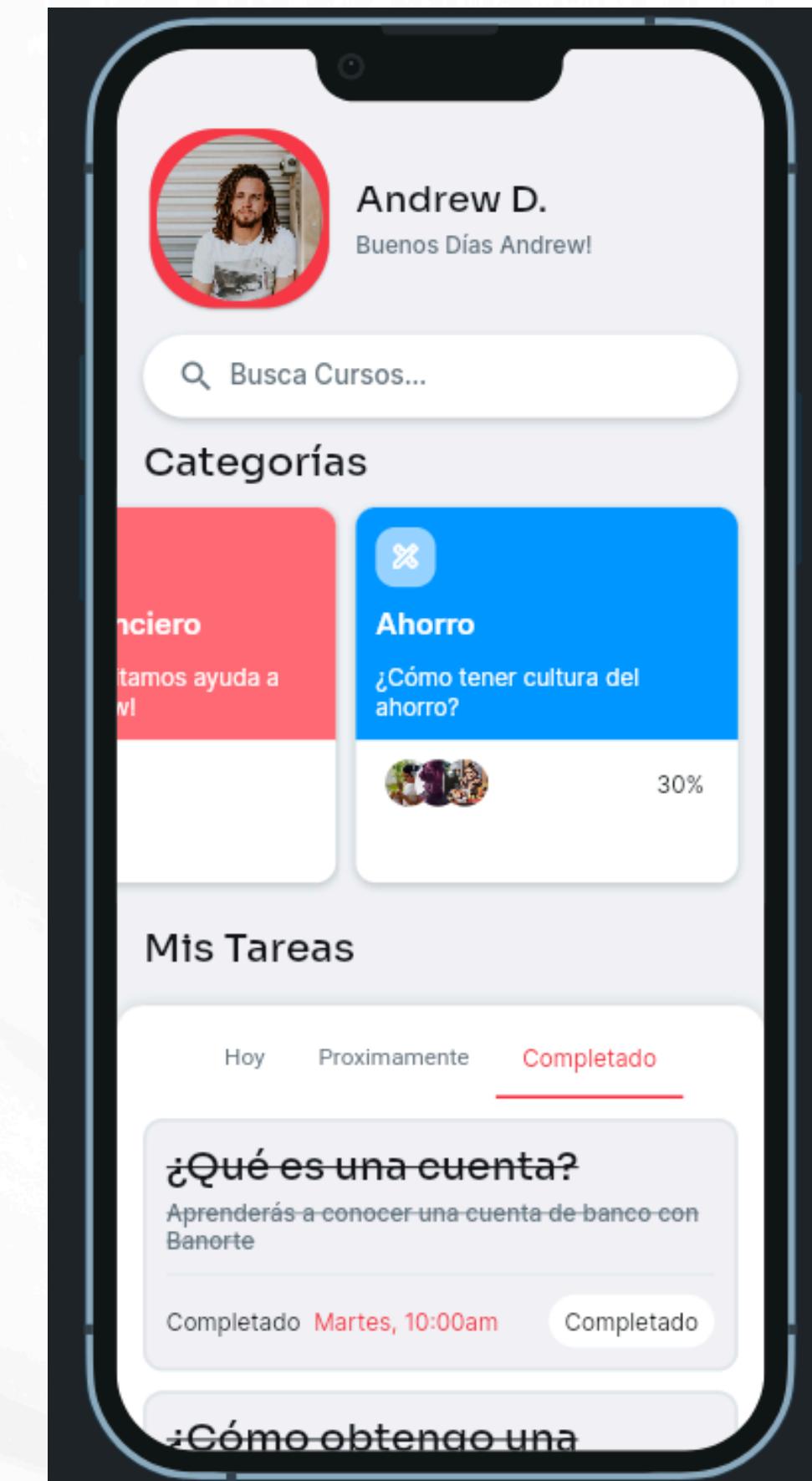
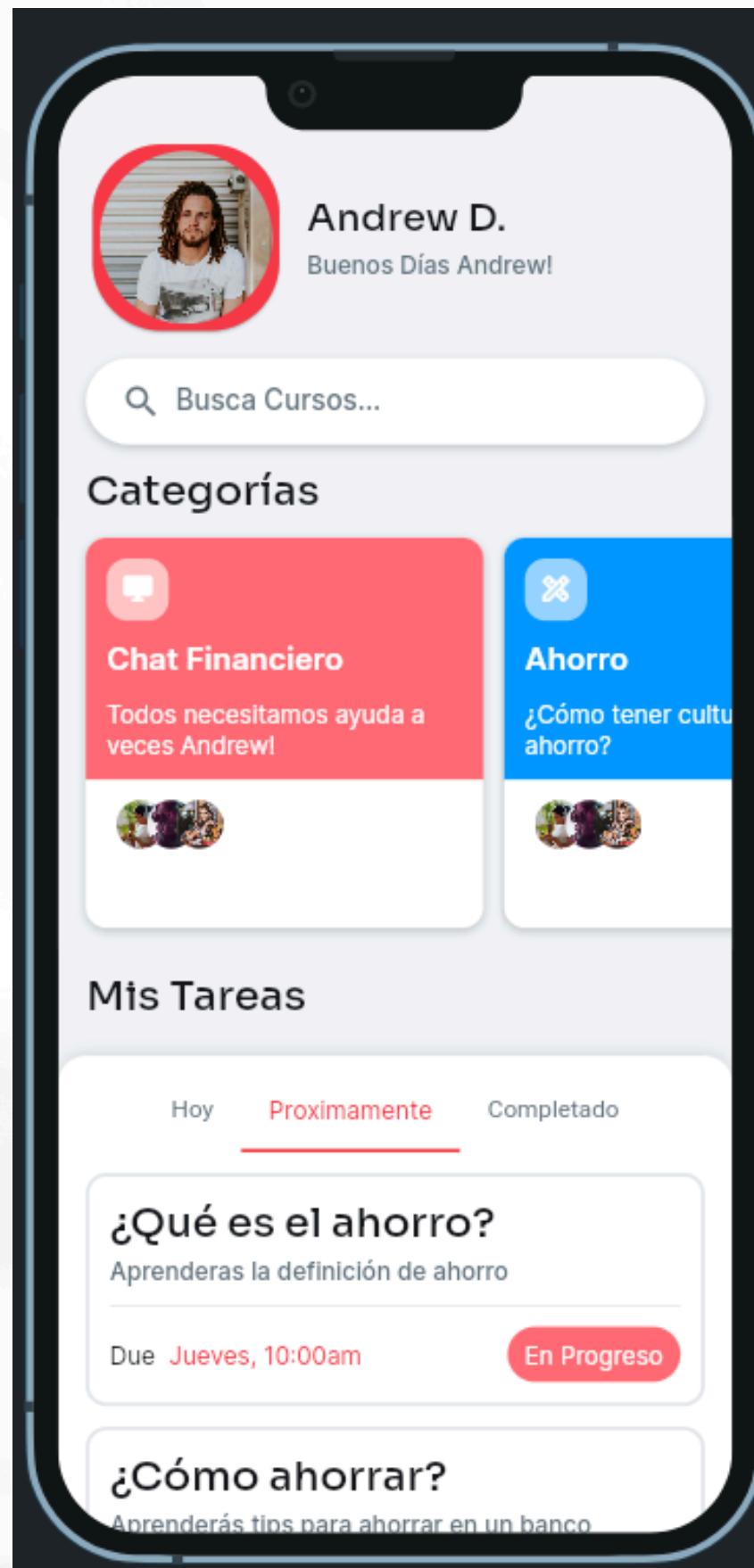
# Cuestionario de Autoconcepción



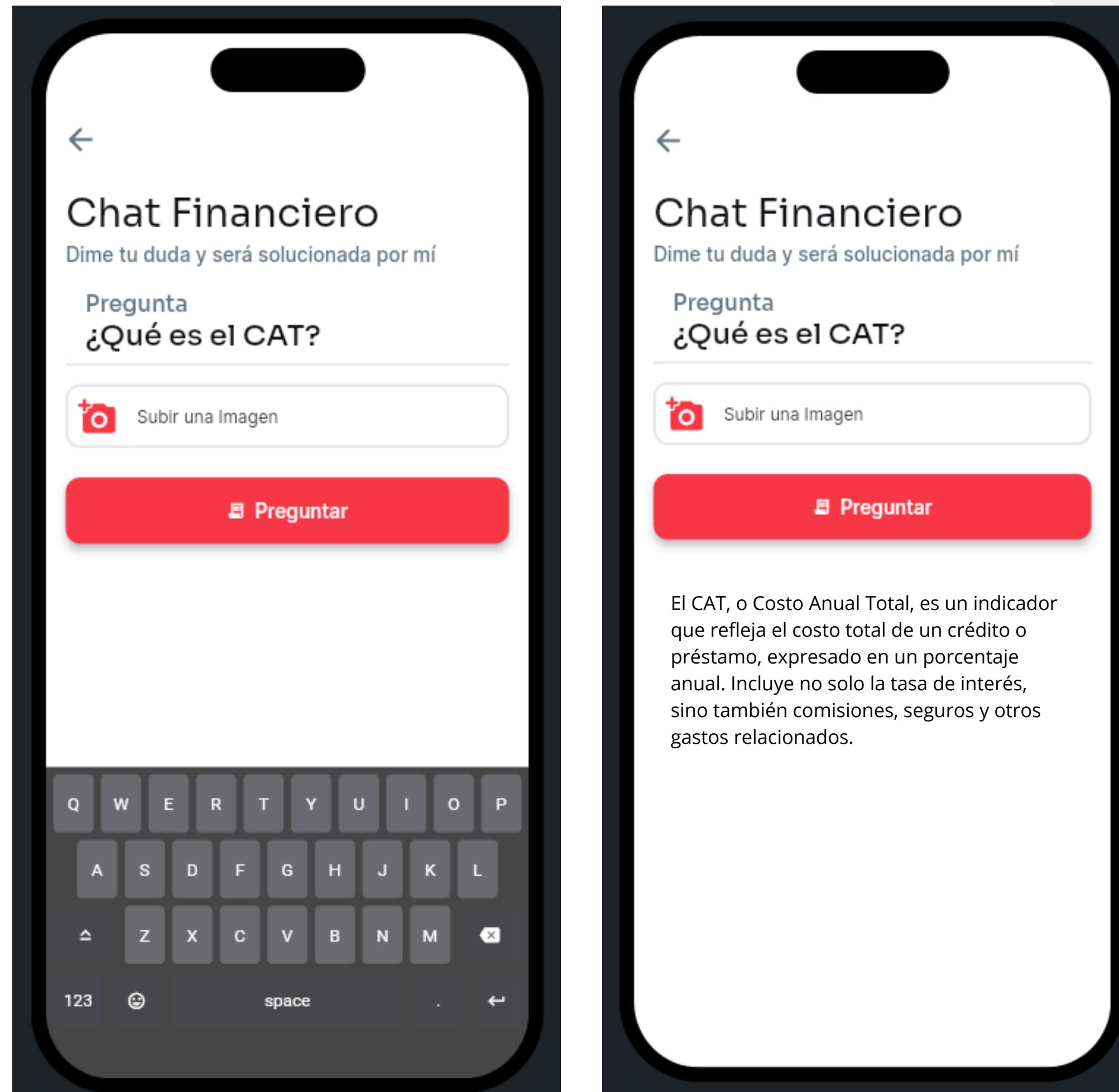
Análisis de Respuestas



# Inicio



# Chat Financiero 24hrs



# Configuración



# User Persona

**INGRESOS** \$290,000 -  
**ANUALES** \$482,000 USD

**OCCUPACIÓN** Diseñadora  
gráfica  
freelance

## Motivación para usar la app

Quiere aprender a ahorrar sin sentirse limitada y pagar sus deudas estudiantiles más rápido. Busca una forma interactiva de aprender sobre gestión de ingresos y manejo de deudas.

## Clara Gómez



**EDAD** 25 años  
**SEXO** Mujer

## Metas financieras

Ahorrar para un viaje al extranjero y pagar deudas de la universidad.

## Desafíos

Clara tiene ingresos irregulares y le cuesta mantener un presupuesto estable. No tiene mucha experiencia manejando sus finanzas, y suele gastar de más en cosas no planificadas.

Aplicaciones  
Móviles

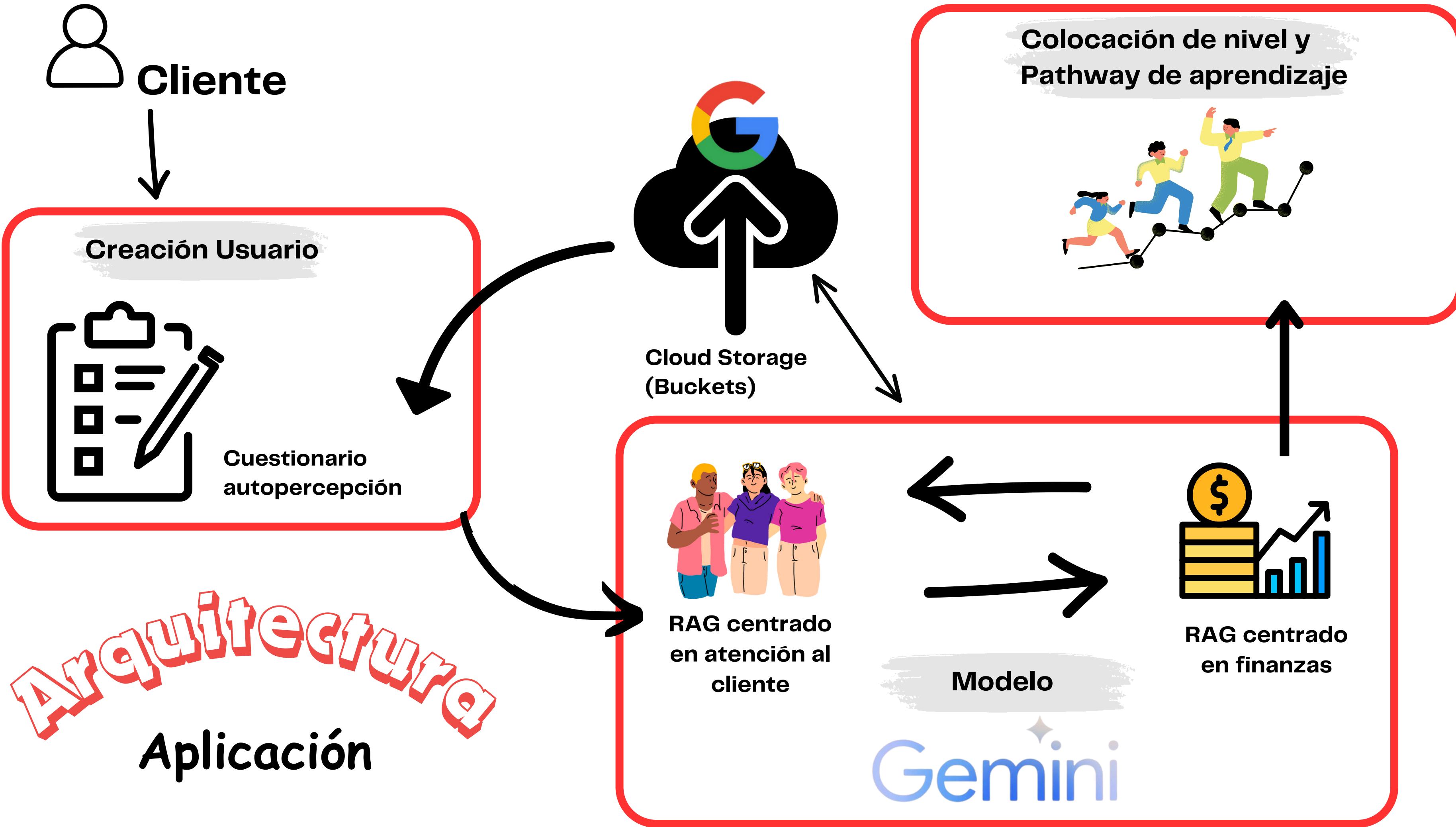


Google Play



## Canales de Interacción

# **Documentación del Proyecto**



# **Tipos de Tecnología Usada Funcionalidades**



# Conclusiones

