

# Software Requirements Specification

for

**PYMER**

Version 1.0.0

2026

## Índice General

<b>1</b>	<b>INTRODUCCIÓN.....</b>	<b>3</b>
<b>2</b>	<b>DETALLES DEL PROYECTO.....</b>	<b>3</b>
2.1	DETALLES GENERALES DEL PROYECTO.....	3
<b>3</b>	<b>DEFINICIÓN PROYECTO.....</b>	<b>3</b>
3.1.1	SISTEMA DE CONTROL DE VERSIONES.....	3
3.1.2	TECH STACK.....	3
<b>4</b>	<b>ESPECIFICACIÓN DE REQUERIMIENTOS DE SOFTWARE.....</b>	<b>4</b>
4.1	ALCANCES.....	4
4.1.1	LIMITES.....	5
4.2	DESCRIPCIÓN GLOBAL DEL PRODUCTO.....	5
4.2.1	INTERFAZ DE USUARIO.....	5
<b>5</b>	<b>REQUERIMIENTOS ESPECÍFICOS.....</b>	<b>5</b>
5.1	REQUERIMIENTOS FUNCIONALES AGENTE.....	5
5.1.1	ACTORES DEL SISTEMA.....	5
5.1.2	ESPECIFICACIÓN DE HISTORIAS DE USUARIO.....	9
5.1.3	ESTRUCTURA AGENTE PYMER.....	23
5.1.4	TOOLS DEL AGENTE.....	25
5.1.5	ESTRUCTURA DEL PITCH GENERADO (7 PASOS).....	27
5.1.6	GESTIÓN DE SESIONES.....	29
5.1.7	API ENDPOINTS.....	29
5.2	REQUEMIENTOS FUNCIONALES LOGICA DE NEGOCIO.....	30
5.2.1	ACTORES DEL SISTEMA.....	30
5.2.2	DESCRIPCIÓN DE ACTORES.....	31
5.2.3	MATRIZ DE TRAZABILIDAD DE REQUERIMIENTOS FUNCIONALES.....	31
5.2.4	ESPECIFICACIÓN HISTORIAS DE USUARIO.....	34
5.3	ANÁLISIS.....	69
5.3.1	CASOS DE USO Y DESCRIPCIÓN.....	69
5.4	MODELAMIENTO DE DATOS.....	71
<b>6</b>	<b>DISEÑO.....</b>	<b>71</b>
6.1	DISEÑO DE ARQUITECTURA FUNCIONAL.....	71
6.2	DISEÑO INTERFAZ Y NAVEGACIÓN.....	77
<b>7</b>	<b>PLAN DE CAPACITACIÓN Y ENTRENAMIENTO.....</b>	<b>78</b>
<b>8</b>	<b>PLAN DE IMPLANTACIÓN Y PUESTA EN MARCHA.....</b>	<b>78</b>
<b>9</b>	<b>ANEXO: PLANIFICACION INICIAL DEL PROYECTO.....</b>	<b>78</b>
9.1	ESQUEMA ESPECIFICACIÓN DE INTERFAZ.....	78
9.2	DIAGRAMA FLUJO MENÚ.....	79

---

# 1 INTRODUCCIÓN

---

Este documento tiene como objetivo detallar los aspectos técnicos del desarrollo frontend para el software Money4Cent (M4C). El objetivo principal es proporcionar una comprensión clara y concisa del funcionamiento del desarrollo frontend de este proyecto, este documento está dirigido principalmente a los desarrolladores involucrados, así como a los equipos de soporte técnico que necesitarán mantener y solucionar problemas en el sistema.

---

## 2 DETALLES DEL PROYECTO

---

### 2.1 Detalles generales del proyecto

- **Fecha tentativa de inicio de proyecto:** Enero 2026
  - **Fecha tentativa de entrega de proyecto:** Salida MVP Mayo 2026
  
  - **Version:** V.1.0.0
- 

## 3 DEFINICIÓN PROYECTO

---

### 3.1.1 Sistema de Control de Versiones

Para asegurar la integridad del ciclo de vida, pyper establece las siguientes directrices operativas:

- **Aislamiento de Entornos:** Cada desarrollador opera desde su IDE workstation mediante ramas /features, evitando cualquier impacto directo sobre la rama productiva.
- **Sincronización Técnica:** Integración Continua y Despliegue Continuo (CI/CD) mediante Google Cloud Build, orquestando flujos de trabajo independientes pero sincronizados tanto para el ecosistema de React.js como para el de Java Spring Boot. Bajo esta directriz, cada vez que se realiza push a la rama productiva se realiza el despliegue automatico de los cambios.

### 3.1.2 Tech Stack

Capa	Tecnología Principal	Herramientas Complementarias
Frontend	React.js + TypeScript	Tailwind CSS (Estilos), TanStack Query (Gestión de estado/API), Vite (Bundler).
Backend	Java + Spring Boot	Spring Security (OAuth2/JWT), Hibernate/JPA, Maven (Gestión de dependencias).
Agente IA	Google SDK /	Modelos Gemini, Vertex AI

---

	ADK	
Infraestructura	Google Cloud Platform	Cloud Build (CI/CD), Cloud Run (Compute), Artifact Registry (Imágenes).
Persistencia	Firestore	Para la gestión de datos de todos los módulos + agente

### 3.1.2.1 Librerías/Dependencias

Categoría	Dependencias	Propósito Arquitectónico
Core & Routing	react, react-dom, react-router-dom	Núcleo de la SPA y gestión de navegación entre los módulos de ideación, costos y proyecciones.
Diseño y UI (MUI)	@mui/material, @mui/icons-material, @emotion/react, @emotion/styled, @toolpad/core	Implementación de la interfaz visual bajo estándares profesionales y dashboards de analítica.
Diseño (Utilidades)	tailwindcss, lucide-react, autoprefixer, postcss	Estilizado ágil mediante clases de utilidad e iconografía minimalista.
Seguridad e Identidad	@react-oauth/google, jwt-decode, firebase, gapi-script	Gestión de autenticación con Google, manejo de tokens JWT y servicios en la nube.
Feedback de Usuario	sweetalert2	Notificaciones y alertas dinámicas para procesos críticos del emprendedor.
Build System	vite, @vitejs/plugin-react, @vitejs/plugin-react-swc	Orquestador de construcción rápida (HMR) y optimización de artefactos para producción.
Calidad de Código	typescript, eslint, typescript-eslint, eslint-plugin-react-hooks	Tipado estático y reglas de análisis para asegurar un código libre de errores comunes.
Entorno de Trabajo	@types/node, @types/react, @types/react-dom, path, globals	Definiciones de tipos y utilidades del sistema para las IDE workstations.

## 4 ESPECIFICACIÓN DE REQUERIMIENTOS DE SOFTWARE

Adaptación basada en *IEEE Software requirements Specifications Std 830-1998*.

### 4.1 Alcances

La plataforma proporcionará un entorno digital centralizado diseñado para guiar a los emprendedores a través de las fases críticas de maduración de negocio, integrando un módulo de ideación para la validación de propuestas de valor, un sistema de estructuración de costos parametrizables y un motor de proyecciones financieras dinámicas. Técnicamente, el sistema se consolidará mediante una arquitectura desacoplada con un frontend robusto en React.js + TypeScript enfocado en la experiencia de usuario (UX), y un backend de grado empresarial en Java Spring Boot encargado de la persistencia, seguridad y lógica computacional.

#### 4.1.1 Límites

El alcance de la fase de Producto Mínimo Viable (MVP) se restringirá exclusivamente a los tres módulos core mencionados ideación, estructura de costos y proyecciones, excluyendo cualquier funcionalidad externa o integración de terceros no especificada en el stack inicial. Asimismo, pymer determina que el motor de cálculo y la lógica de negocio estarán optimizados estrictamente para emprendimientos de manufactura o creación de productos propios; por lo tanto, el sistema no contemplará en esta etapa las complejidades específicas del modelo de reventa, tales como la gestión avanzada de márgenes de intermediación o logística de distribución externa

## 4.2 Descripción Global del Producto

### 4.2.1 Interfaz de usuario

Se relaciona el flujo de interfaz de usuario en el (Anexo 9.1) allí se detallan todas las interfaces, su flujo de navegación y lineamientos gráficos, estas interfaces son mockups desarrollados en la herramienta figma

---

## 5 REQUERIMIENTOS ESPECÍFICOS

---

### 5.1 Requerimientos Funcionales AGENTE

#### 5.1.1 ACTORES DEL SISTEMA

##### 5.1.1.1 Diagrama de Actores

El sistema PYMER cuenta con dos actores principales que interactúan con la plataforma y con el agente financiero conversacional: el Estudiante (emprendedor) y el Profesor (docente). El siguiente diagrama muestra sus relaciones con los módulos del sistema.



##### 5.1.1.2 Descripción de Actores

Nombre del Actor	Descripción del Actor
------------------	-----------------------

Estudiante	Usuario autenticado con el rol de STUDENT. Es responsable de completar los módulos asignados a su aula virtual, ej.: (Emprendimiento, DOFA, Lean Canvas, Golden Circle, Buyer Persona, costos directos e indirectos, configuración y proyección financiera). Puede crear, consultar y actualizar su información en cada módulo. Adicionalmente, puede enviar la completitud de módulos para revisión del docente, consultar las aulas en la que está inscrito, ver su progreso y gestionar su perfil de usuario.
Profesor	Usuario autenticado con el rol de TEACHER. Es responsable de crear y administrar las aulas virtuales, agregar y remover estudiantes, configurar los módulos que se harán disponibles en su aula y revisar las completitudes de módulos enviadas por los estudiantes, asignando una calificación y un comentario. Puede consultar el progreso de sus estudiantes en cada uno de sus módulos.

### 5.1.1.3 MATRIZ DE TRAZABILIDAD DE REQUERIMIENTOS FUNCIONALES

La siguiente tabla relaciona cada requerimiento funcional con las historias de usuario que dan solución al mismo, abarcando las funcionalidades del agente financiero PYMER y los módulos de la plataforma educativa.

REQUERIMIENTO FUNCIONAL	ID_HU	TÍTULO_HU
<b>RF-01 Autenticación y Registro de Usuarios</b>	HU001	COMO usuario no autenticado, QUIERO registrarme e iniciar sesión en la plataforma, PARA acceder a las herramientas de negocio.
	HU002	COMO usuario autenticado, QUIERO cambiar mi contraseña, PARA mantener la seguridad de mi cuenta.
	HU003	COMO usuario no autenticado, QUIERO restablecer mi contraseña mediante correo electrónico, PARA recuperar el acceso a mi cuenta.
<b>RF-02 Gestión de Perfil de Usuario</b>	HU004	COMO usuario autenticado, QUIERO actualizar mi información personal, PARA mantener mis datos de perfil al día.

	HU005	COMO usuario autenticado, QUIERO subir o actualizar mi foto de perfil, PARA personalizar mi cuenta.
	HU006	COMO usuario autenticado, QUIERO consultar mi perfil y ver mi rol asignado, PARA conocer mis permisos en la plataforma.
<b>RF-03 Gestión del Emprendimiento (Módulo 1)</b>	HU007	COMO estudiante, QUIERO registrar la información de mi emprendimiento (nombre, descripción, sector, ciudad, propuesta de valor, diferenciador, etc.), PARA que el agente pueda analizar mi negocio.
	HU008	COMO estudiante, QUIERO completar mi DOFA (fortalezas, oportunidades, debilidades, amenazas), PARA obtener análisis contextualizados de mi negocio.
	HU009	COMO estudiante, QUIERO definir mi Buyer Persona, PARA que el agente calcule mercados y escenarios con base en mi cliente ideal.
	HU010	COMO estudiante, QUIERO completar mi Golden Circle (Qué, Cómo, Por qué), PARA enriquecer el elevator pitch generado por el agente.
<b>RF-04 Gestión Financiera (Módulos 2 y 3)</b>	HU011	COMO estudiante, QUIERO registrar mis costos fijos y variables indirectos, PARA que el agente calcule mi punto de equilibrio y rentabilidad.
	HU012	COMO estudiante, QUIERO configurar mi precio de venta unitario y unidades mensuales planeadas, PARA obtener proyecciones de ventas y análisis de mercado (TAM, SAM, SOM).
	HU013	COMO estudiante, QUIERO consultar mi punto de equilibrio, PARA saber cuántas unidades debo vender para cubrir mis costos.
	HU014	COMO estudiante, QUIERO analizar mi rentabilidad mensual, PARA conocer mi utilidad operativa actual.

	HU015	COMO estudiante, QUIERO proyectar mis ventas con una tasa de crecimiento, PARA planear el crecimiento de mi negocio.
<b>RF-05 Agente Financiero Conversacional</b>	HU016	COMO estudiante, QUIERO enviar mensajes al agente PYMER y recibir respuestas contextualizadas, PARA obtener asesoría financiera sobre mi emprendimiento.
	HU017	COMO estudiante, QUIERO que el agente recuerde el historial de mi conversación, PARA no repetir el contexto en cada interacción.
	HU018	COMO estudiante, QUIERO consultar mi TAM, SAM y SOM, PARA entender el potencial de mi mercado.
	HU019	COMO estudiante, QUIERO consultar escenarios financieros (conservador, moderado, arriesgado), PARA tomar decisiones con base en diferentes proyecciones.
	HU020	COMO estudiante, QUIERO que el agente genere mi elevator pitch en 7 pasos, PARA presentar mi idea de forma persuasiva.
	HU021	COMO estudiante, QUIERO consultar los competidores de mi sector en mi ciudad, PARA conocer el panorama competitivo de mi mercado.
	HU022	COMO estudiante, QUIERO recibir recomendaciones de ubicación óptima para mi punto físico, PARA decidir en qué zona abrir mi negocio.
<b>RF-06 Gestión de Aulas y Módulos</b>	HU023	COMO profesor, QUIERO crear y configurar aulas virtuales con módulos seleccionados, PARA asignarlas a mis estudiantes.
	HU024	COMO profesor, QUIERO agregar y remover estudiantes de mis aulas, PARA gestionar la inscripción.
	HU025	COMO profesor, QUIERO revisar y calificar las completitudes de módulos enviadas por los estudiantes, PARA dar retroalimentación académica.

## 5.1.2 ESPECIFICACIÓN DE HISTORIAS DE USUARIO

A continuación se detallan las historias de usuario identificadas para el agente financiero PYMER y la plataforma educativa. Cada historia incluye descripción, requerimiento funcional asociado, prioridad, precondiciones, reglas de negocio y criterios de aceptación (formato Dado que / Cuando / Entonces).

### 5.1.2.1 HU001 – Registro e inicio de sesión

HU001 Registro e inicio de sesión	
<b>Descripción</b>	El usuario no autenticado puede crear una cuenta con correo y contraseña, o iniciar sesión con credenciales existentes. El sistema valida la identidad y asigna el rol correspondiente (STUDENT o TEACHER).
<b>Requerimiento Funcional Asociado</b>	RF-01 – Autenticación y Registro de Usuarios
<b>Historias de Usuario Asociadas</b>	HU002, HU003
<b>Prioridad</b>	Alta
<b>Precondiciones</b>	El usuario no tiene sesión activa. La plataforma está disponible.
<b>Reglas de Negocio</b>	El correo debe ser único. La contraseña requiere mínimo 8 caracteres. El rol se asigna en el momento del registro o por invitación del profesor.
<b>Criterios de Aceptación</b>	
<b>Escenario 1 – Registro exitoso:</b> Dado que el usuario no tiene cuenta, cuando completa el formulario con correo único y contraseña válida, entonces se crea la cuenta y se redirige al dashboard.	
<b>Escenario 2 – Correo ya registrado:</b> Dado que el correo ya existe en el sistema, cuando el usuario intenta registrarse con ese correo, entonces se muestra el mensaje 'Este correo ya está registrado'.	
<b>Escenario 3 – Inicio de sesión exitoso:</b> Dado que el usuario tiene cuenta activa, cuando ingresa sus credenciales correctas, entonces accede a su dashboard con sus módulos disponibles.	
<b>Escenario 4 – Credenciales incorrectas:</b> Dado que el usuario ingresa contraseña incorrecta, cuando envía el formulario, entonces se muestra 'Correo o contraseña incorrectos' sin revelar cuál campo falló.	

<b>Notas</b>	Integra con Firebase Authentication.
--------------	--------------------------------------

#### 5.1.2.2 HU002 – Cambio de contraseña

<b>HU002 Cambio de contraseña</b>	
<b>Descripción</b>	El usuario autenticado puede cambiar su contraseña actual ingresando la contraseña vigente y la nueva.
<b>Requerimiento Funcional Asociado</b>	RF-01 – Autenticación y Registro de Usuarios
<b>Historias de Usuario Asociadas</b>	HU001
<b>Prioridad</b>	Media
<b>Precondiciones</b>	El usuario tiene sesión activa.
<b>Reglas de Negocio</b>	La contraseña actual debe coincidir con la registrada. La nueva contraseña debe tener mínimo 8 caracteres y ser diferente a la actual.
<b>Criterios de Aceptación</b>	
<b>Escenario 1 – Cambio exitoso:</b> Dado que el usuario está autenticado, cuando ingresa su contraseña actual correcta y una nueva válida, entonces la contraseña se actualiza y se cierra la sesión activa.	
<b>Escenario 2 – Contraseña actual incorrecta:</b> Dado que el usuario ingresa una contraseña actual incorrecta, cuando envía el formulario, entonces se muestra 'Contraseña actual incorrecta'.	
<b>Notas</b>	No aplica

#### 5.1.2.3 HU003 – Restablecimiento de contraseña por correo

<b>HU003 Restablecimiento de contraseña por correo</b>	
<b>Descripción</b>	El usuario no autenticado puede solicitar un enlace de restablecimiento de contraseña a su correo registrado.
<b>Requerimiento Funcional Asociado</b>	RF-01 – Autenticación y Registro de Usuarios

<b>Historias de Usuario Asociadas</b>	HU001
<b>Prioridad</b>	Media
<b>Precondiciones</b>	El usuario tiene correo registrado en la plataforma.
<b>Reglas de Negocio</b>	El enlace de restablecimiento expira en 60 minutos. Solo se envía si el correo existe en el sistema (sin confirmar si existe o no, por seguridad).
<b>Criterios de Aceptación</b>	
<b>Escenario 1 – Solicitud enviada:</b> Dado que el usuario ingresa su correo registrado, cuando solicita el restablecimiento, entonces recibe un correo con el enlace y se muestra 'Si el correo existe, recibirás las instrucciones'.	
<b>Escenario 2 – Enlace expirado:</b> Dado que el usuario accede a un enlace expirado, cuando intenta restablecer la contraseña, entonces se muestra 'El enlace ha expirado. Solicita uno nuevo'.	
<b>Notas</b>	Integra con Firebase Authentication (sendPasswordResetEmail).

#### 5.1.2.4 HU007 – Registro del perfil de emprendimiento

<b>HU007 Registro del perfil de emprendimiento</b>	
<b>Descripción</b>	El estudiante completa el perfil de su emprendimiento en el Módulo 1: nombre, descripción, sector (country), ciudad, valores, objetivos, diferenciador, problema que resuelve, audiencia objetivo, propuesta de valor, fundadores y equipo. Esta información se almacena en la colección 'entrepreneurship' de Firestore y es consumida por todas las tools del agente.
<b>Requerimiento Funcional Asociado</b>	RF-03 – Gestión del Emprendimiento (Módulo 1)
<b>Historias de Usuario Asociadas</b>	HU008, HU009, HU010, HU016, HU018, HU019, HU020, HU021, HU022
<b>Prioridad</b>	Alta
<b>Precondiciones</b>	El estudiante tiene sesión activa. El módulo de emprendimiento está habilitado en su aula.

<b>Reglas de Negocio</b>	Los campos 'name', 'city' y 'country' (sector) son obligatorios para que las tools funcionen. El campo userId se asigna automáticamente. Si el registro ya existe, se actualiza (no se crea uno nuevo).
<b>Criterios de Aceptación</b>	
<b>Escenario 1 – Registro exitoso:</b> Dado que el estudiante completa los campos obligatorios del módulo, cuando guarda el formulario, entonces los datos se persisten en Firestore (colección 'entrepreneurship') y el agente puede consultarlos.	
<b>Escenario 2 – Campos incompletos:</b> Dado que el estudiante omite campos obligatorios (name, city o sector), cuando intenta guardar, entonces se muestra validación indicando los campos faltantes.	
<b>Escenario 3 – Actualización:</b> Dado que el estudiante ya registró su emprendimiento, cuando modifica algún campo y guarda, entonces el documento en Firestore se actualiza y el agente refleja la información nueva en su próxima consulta.	
<b>Notas</b>	Colección Firestore: entrepreneurship. Campos clave: name, city, country (sector), description, differentiator, problem, targetAudience, valueProposition, founders, team.

#### 5.1.2.5 HU008 – Registro del análisis DOFA

<b>HU008 Registro del análisis DOFA</b>	
<b>Descripción</b>	El estudiante registra su análisis DOFA (fortalezas, oportunidades, debilidades, amenazas) en la colección 'dofa' de Firestore. Esta información es consumida por la tool generar_elevator_pitch_tool para enriquecer el pitch del negocio.
<b>Requerimiento Funcional Asociado</b>	RF-03 – Gestión del Emprendimiento (Módulo 1)
<b>Historias de Usuario Asociadas</b>	HU007, HU020
<b>Prioridad</b>	Alta
<b>Precondiciones</b>	El estudiante tiene sesión activa y ha completado el perfil de emprendimiento (HU007).

<b>Reglas de Negocio</b>	Se aceptan campos en inglés (strengths, opportunities, weaknesses, threats) o en español (fortalezas, oportunidades, debilidades, amenazas) por compatibilidad. El userId se asigna automáticamente.
<b>Criterios de Aceptación</b>	
<b>Escenario 1 – Registro exitoso:</b> Dado que el estudiante completa los 4 cuadrantes del DOFA, cuando guarda, entonces los datos se almacenan en la colección 'dofa' y el agente los usa al generar el pitch.	
<b>Escenario 2 – DOFA incompleto:</b> Dado que el estudiante guarda con algún cuadrante vacío, cuando el agente genera el pitch, entonces utiliza solo los campos disponibles e indica en 1 línea cuál completar para enriquecer el resultado.	
<b>Notas</b>	Colección Firestore: dofa. Campos: strengths/fortalezas, opportunities/oportunidades, weaknesses/debilidades, threats/amenazas.

#### 5.1.2.6 HU009 – Registro del Buyer Persona

<b>HU009 Registro del Buyer Persona</b>	
<b>Descripción</b>	El estudiante define el perfil de su cliente ideal en la colección 'buyer_persona' de Firestore: nombre de perfil, edad, profesión, ingresos, ciudad, intereses, retos, objetivos y frase representativa. Esta información es consumida por las tools de mercado (TAM/SAM/SOM), pitch y ubicación.
<b>Requerimiento Funcional Asociado</b>	RF-03 – Gestión del Emprendimiento (Módulo 1)
<b>Historias de Usuario Asociadas</b>	HU007, HU018, HU020, HU022
<b>Prioridad</b>	Alta
<b>Precondiciones</b>	El estudiante tiene sesión activa.
<b>Reglas de Negocio</b>	La ciudad del buyer persona puede diferir de la ciudad del negocio. El campo 'income' (ingresos) afecta el factor_cliente en el cálculo del SAM. Si no se registra buyer persona, el factor de ajuste del SAM es 0.50 (más conservador).
<b>Criterios de Aceptación</b>	

<p><b>Escenario 1 – Registro exitoso:</b> Dado que el estudiante completa el perfil del Buyer Persona, cuando guarda, entonces los datos se almacenan en 'buyer_persona' y el agente los incorpora en sus cálculos de mercado y pitch.</p>	
<p><b>Escenario 2 – Sin Buyer Persona registrado:</b> Dado que el estudiante no ha completado este módulo, cuando el agente calcula el SAM, entonces aplica un factor de cliente conservador (0.50) e indica que completar el Buyer Persona mejorará la precisión del análisis.</p>	
<b>Notas</b>	<p>Colección Firestore: buyer_persona. Campos: name, age, profession, income, city, interests, challenges, objectives, phrase.</p>

#### 5.1.2.7 HU011 – Registro de costos fijos y variables indirectos

HU011 Registro de costos fijos y variables indirectos	
<b>Descripción</b>	El estudiante registra sus costos indirectos fijos (alquiler, seguros, gerente, depreciación) y costos indirectos variables (agua, limpieza, mantenimiento) en las colecciones correspondientes de Firestore. Cada ítem tiene un campo 'costValue'. Esta información es consumida por las tools financieras del agente para calcular punto de equilibrio, rentabilidad y escenarios.
<b>Requerimiento Funcional Asociado</b>	RF-04 – Gestión Financiera (Módulos 2 y 3)
<b>Historias de Usuario Asociadas</b>	HU012, HU013, HU014, HU019
<b>Prioridad</b>	Alta
<b>Precondiciones</b>	El estudiante tiene sesión activa. Los módulos de costos están habilitados en su aula.
<b>Reglas de Negocio</b>	Cada costo es un documento independiente en su colección. El campo 'costValue' debe ser numérico. Si no hay costos registrados, el agente asume costos fijos = 0 e indica el dato faltante. El userId es clave de filtro en cada colección.
<b>Criterios de Aceptación</b>	
<p><b>Escenario 1 – Registro de costo fijo:</b> Dado que el estudiante agrega un costo fijo (ej. alquiler: \$500.000), cuando guarda, entonces se crea un documento en 'indirect_fixed_costs' con costValue=500000 y userId del estudiante.</p>	

<b>Escenario 2 – Eliminación de costo:</b> Dado que el estudiante elimina un costo registrado, cuando confirma la eliminación, entonces el documento se borra de Firestore y los cálculos del agente lo excluyen.	
<b>Escenario 3 – Sin costos registrados:</b> Dado que el estudiante no ha registrado costos, cuando consulta el punto de equilibrio, entonces el agente muestra el cálculo con costos fijos = 0 e indica en 1 línea que debe registrar sus costos para mayor precisión.	
<b>Notas</b>	Colecciones                      Firestore:                      indirect_fixed_costs, indirect_variable_costs. Campo clave: costValue.

#### 5.1.2.8 HU012 – Configuración financiera (precio, unidades, margen)

<b>HU012 Configuración financiera (precio, unidades, margen)</b>	
<b>Descripción</b>	El estudiante registra su configuración financiera base en la colección 'financial_settings': precio por unidad (pricePerUnit), unidades mensuales planeadas (monthlyUnitsPlanned) y margen de ganancia deseado (desiredProfitMargin). El agente usa estos datos para calcular SOM, escenarios y proyecciones.
<b>Requerimiento Funcional Asociado</b>	RF-04 – Gestión Financiera (Módulos 2 y 3)
<b>Historias de Usuario Asociadas</b>	HU011, HU013, HU014, HU015, HU018, HU019
<b>Prioridad</b>	Alta
<b>Precondiciones</b>	El estudiante tiene sesión activa. El módulo de configuración financiera está habilitado.
<b>Reglas de Negocio</b>	Si pricePerUnit no está registrado, el agente lo calcula desde costos + margen. El margen deseado por defecto es 0.15 (15%). Los campos son opcionales en el formulario pero críticos para los cálculos; el agente pedirá el mínimo faltante si es necesario.
<b>Criterios de Aceptación</b>	
<b>Escenario 1 – Configuración completa:</b> Dado que el estudiante registra precio, unidades y margen, cuando el agente calcula el SOM, entonces usa pricePerUnit de Firestore sin pedirle el dato al usuario.	

<b>Escenario 2 – Sin precio registrado:</b> Dado que pricePerUnit no está registrado, cuando el agente calcula el SOM, entonces lo calcula automáticamente desde $(CF/ unidades + CV\_unitario) / (1 - margen)$ e indica la fuente del precio en la respuesta.	
<b>Escenario 3 – Actualización de configuración:</b> Dado que el estudiante actualiza su precio de venta, cuando el agente consulta TAM/SAM/SOM, entonces usa el nuevo precio de Firestore.	
<b>Notas</b>	Colección Firestore: financial_settings. Campos: pricePerUnit, monthlyUnitsPlanned, desiredProfitMargin.

### 5.1.2.9 HU016 – Interacción conversacional con el agente PYMER

HU016 Interacción conversacional con el agente PYMER	
<b>Descripción</b>	El estudiante envía mensajes en lenguaje natural al agente financiero PYMER a través del endpoint POST /chat. El agente interpreta la intención, llama las tools necesarias consultando los datos de Firestore del usuario, y retorna una respuesta estructurada, breve y con una conclusión práctica.
<b>Requerimiento Funcional Asociado</b>	RF-05 – Agente Financiero Conversacional
<b>Historias de Usuario Asociadas</b>	HU017, HU018, HU019, HU020, HU021, HU022
<b>Prioridad</b>	Alta
<b>Precondiciones</b>	El estudiante tiene sesión activa. Su user_id está registrado en Firebase Auth. El agente está desplegado en Cloud Run.
<b>Reglas de Negocio</b>	El agente NUNCA responde 'No puedo ayudarte' sin ofrecer una acción alternativa.   Si una tool devuelve NO_SESSION, el agente detiene toda acción y pide iniciar sesión.   Si el mensaje es ambiguo (ej. 'hola', 'ayuda'), el agente ofrece las 3 opciones principales.   Las respuestas son máx. 6-8 líneas (excepto pitch y análisis de mercado).   El agente no inventa datos: siempre consulta Firestore antes de pedir datos al usuario.
<b>Criterios de Aceptación</b>	
<b>Escenario 1 – Consulta financiera exitosa:</b> Dado que el estudiante tiene costos registrados y envía '¿cuánto necesito vender para no perder?', cuando el agente recibe el mensaje, entonces llama calcular_punto_equilibrio_tool, obtiene los datos de Firestore y responde con las unidades y el valor en dinero del punto de equilibrio.	

<b>Escenario 2 – Mensaje ambiguo:</b> Dado que el estudiante envía 'hola', cuando el agente recibe el mensaje, entonces responde con el saludo estándar y lista las 3 acciones principales disponibles.	
<b>Escenario 3 – Sin sesión activa:</b> Dado que el user_id no tiene sesión en Firebase, cuando cualquier tool devuelve NO_SESSION, entonces el agente responde únicamente: 'Para acceder a tu información debes iniciar sesión en la aplicación PYMER' y no hace más llamadas a tools en ese turno.	
<b>Escenario 4 – Error interno manejado:</b> Dado que ocurre una excepción inesperada en el servidor, cuando se produce el error, entonces el endpoint retorna HTTP 200 con {reply: 'Tuve un problema...', error: true} para que el frontend siempre pueda leer el body.	
<b>Notas</b>	Endpoint: POST /chat. Body: {user_id, message}. Respuesta: {reply, session_id}. Modelo: Gemini 2.5 Pro (Google ADK). Infraestructura: Cloud Run us-central1.

#### 5.1.2.10 HU017 – Persistencia del historial de conversación

HU017 Persistencia del historial de conversación	
<b>Descripción</b>	El agente mantiene el historial de conversación del estudiante en Firestore (colección 'chat_sessions'). Al reiniciar el servidor, el contexto se recupera e inyecta al agente antes del nuevo mensaje, asegurando continuidad de la conversación.
<b>Requerimiento Funcional Asociado</b>	RF-05 – Agente Financiero Conversacional
<b>Historias de Usuario Asociadas</b>	HU016
<b>Prioridad</b>	Alta
<b>Precondiciones</b>	El estudiante tiene sesión activa. El session_id es igual al user_id.
<b>Reglas de Negocio</b>	Máximo 20 mensajes en Firestore (MAX_DB_HISTORY). El agente recibe máximo 10 mensajes de contexto (MAX_LLM_CONTEXT = 5 pares). Al reiniciar el servidor, el historial se inyecta una sola vez como bloque de contexto recuperado. Los mensajes nuevos (user + model) se persisten al final de cada turno.
<b>Criterios de Aceptación</b>	

<b>Escenario 1 – Continuidad en nueva sesión:</b> Dado que el estudiante tuvo una conversación anterior, cuando inicia una nueva sesión en el mismo dispositivo, entonces el agente recuerda el contexto de los últimos 10 mensajes y no pide al usuario repetir información ya proporcionada.	
<b>Escenario 2 – Reinicio del servidor:</b> Dado que el servidor se reinicia, cuando el estudiante envía un nuevo mensaje, entonces el sistema inyecta el historial de Firestore al agente antes de procesar el mensaje, recuperando el contexto de conversación.	
<b>Escenario 3 – Límite de historial:</b> Dado que la sesión acumula más de 20 mensajes, cuando se guarda un mensaje nuevo, entonces el historial se recorta a los últimos 20, descartando los más antiguos.	
<b>Notas</b>	Colección Firestore: chat_sessions. Parámetros: MAX_DB_HISTORY=20, MAX_LLM_CONTEXT=10.

#### 5.1.2.11 HU018 – Consulta de TAM, SAM y SOM

HU018 Consulta de TAM, SAM y SOM	
<b>Descripción</b>	El estudiante solicita al agente el análisis de mercado de su negocio. El agente llama calcular_tam_sam_som_tool, que consulta Firestore para obtener el perfil del negocio y el buyer persona, estima el TAM con Gemini 2.0 Flash, calcula el SAM según ciudad y perfil de cliente, y calcula el SOM con base en unidades planeadas y precio de venta.
<b>Requerimiento Funcional Asociado</b>	RF-05 – Agente Financiero Conversacional
<b>Historias de Usuario Asociadas</b>	HU007, HU009, HU012
<b>Prioridad</b>	Alta
<b>Precondiciones</b>	El estudiante tiene sesión activa y ha registrado al menos el perfil de emprendimiento (nombre, ciudad, sector).
<b>Reglas de Negocio</b>	TAM: estimado con Gemini 2.0 Flash (sector + país). SAM: TAM × factor_ciudad × factor_cliente. Ciudad principal (Bogotá, Medellín, Cali, Barranquilla): factor 20%; otra ciudad: 8%. Con buyer persona: factor 35%; sin buyer persona: 50%. SOM: unidades_mes × 12 × precio_venta. El precio sigue la jerarquía: parámetro > Firestore > cálculo automático. Si TAM no está disponible, el agente lo estima cualitativamente y lo indica como '~\$X (estimado)'. El precio sigue la jerarquía: parámetro > Firestore > cálculo automático. Si TAM no está disponible, el agente lo estima cualitativamente y lo indica como '~\$X (estimado)'.

<b>Criterios de Aceptación</b>	
<b>Escenario 1 – Cálculo completo:</b> Dado que el estudiante tiene registrados sector, ciudad, unidades y precio, cuando consulta su mercado, entonces el agente presenta TAM, SAM y SOM con valores numéricos, explicación por métrica y conclusión en 2 líneas.	
<b>Escenario 2 – Sin precio registrado:</b> Dado que el precio no está en Firestore, cuando el agente calcula el SOM, entonces lo calcula desde costos + margen, indica la fuente y muestra el valor calculado sin pedir el dato al usuario.	
<b>Escenario 3 – SOM pendiente por falta de unidades:</b> Dado que monthlyUnitsPlanned es 0 o nulo, cuando el agente presenta el análisis, entonces muestra TAM y SAM con valores y SOM como 'Pendiente', y en 1 línea al final solicita cuántas unidades planea vender al mes.	
<b>Notas</b>	Tool: <code>calcular_tam_sam_som_tool</code> ( <code>market_tools.py</code> ). API externa: Gemini 2.0 Flash para estimación de TAM.

#### 5.1.2.12 HU019 – Análisis de escenarios financieros

<b>HU019 Análisis de escenarios financieros</b>	
<b>Descripción</b>	El estudiante solicita escenarios financieros (conservador, moderado, arriesgado) a través del agente. La tool <code>calcular_escenarios_tool</code> lee los datos de Firestore del usuario y genera 3 proyecciones con: unidades a vender, ventas mensuales, utilidad mensual y punto de equilibrio, justificando cada escenario con el contexto del sector.
<b>Requerimiento Funcional Asociado</b>	RF-05 – Agente Financiero Conversacional
<b>Historias de Usuario Asociadas</b>	HU011, HU012, HU007
<b>Prioridad</b>	Alta
<b>Precondiciones</b>	El estudiante tiene sesión activa. Idealmente ha registrado costos y configuración financiera.
<b>Reglas de Negocio</b>	Escenario conservador: 60% de las unidades planeadas. Moderado: 100%. Arriesgado: 150%. Si faltan datos ( <code>code: DATOS_INCOMPLETOS</code> ), el agente muestra los escenarios calculables y pide el mínimo faltante en 1 línea. El agente nunca dice que no puede calcular nada; usa los datos disponibles.

<b>Criterios de Aceptación</b>	
<b>Escenario 1 – Análisis completo:</b> Dado que el estudiante tiene costos y configuración financiera registrados, cuando solicita escenarios, entonces el agente presenta los 3 escenarios con sus métricas y una conclusión indicando a cuál apuntar y la acción clave.	
<b>Escenario 2 – Datos parciales:</b> Dado que el estudiante no tiene precio registrado pero sí costos, cuando solicita escenarios, entonces el agente calcula el precio desde costos + margen, presenta los escenarios e indica en 1 línea el supuesto de precio utilizado.	
<b>Escenario 3 – Sin datos financieros:</b> Dado que el estudiante no tiene costos ni configuración registrada, cuando solicita escenarios, entonces el agente indica que necesita al menos el precio de venta y unidades planeadas, y pide estos datos en 1 línea.	
<b>Notas</b>	Tool: calcular_escenarios_tool (scenario_tools.py). Factores: conservador=0.6, moderado=1.0, arriesgado=1.5.

#### 5.1.2.13 HU020 – Generación del elevator pitch en 7 pasos

<b>HU020 Generación del elevator pitch en 7 pasos</b>	
<b>Descripción</b>	El estudiante solicita su elevator pitch al agente. La tool generar_elevator_pitch_tool lee las 4 colecciones del Módulo 1 (entrepreneurship, golden_circle, dofa, buyer_personas) y retorna el contexto estructurado. El agente construye el pitch en 7 pasos: 1) Gancho, 2) Quiénes somos, 3) El Problema, 4) La Solución, 5) Beneficios y Mercado, 6) Nuestra Ventaja, 7) Llamado a la Acción. Al final incluye la versión fluida completa.
<b>Requerimiento Funcional Asociado</b>	RF-05 – Agente Financiero Conversacional
<b>Historias de Usuario Asociadas</b>	HU007, HU008, HU009, HU010
<b>Prioridad</b>	Alta
<b>Precondiciones</b>	El estudiante tiene sesión activa y ha completado al menos el perfil de emprendimiento (HU007).

<b>Reglas de Negocio</b>	La duración por defecto es 60 segundos (~130 palabras). El agente NUNCA usa tablas en el pitch. Si faltan datos clave, genera el pitch con los datos disponibles e indica en 1 línea qué campo completar para mejorar un paso específico. Si la tool devuelve NO_DATA, el agente detiene el proceso e indica que el Módulo 1 no está completo.
<b>Criterios de Aceptación</b>	
<b>Escenario 1 – Pitch completo:</b> Dado que el estudiante tiene los 4 módulos completos, cuando solicita el pitch, entonces el agente presenta los 7 pasos con sus secciones y la versión fluida de ~130 palabras al final.	
<b>Escenario 2 – Datos parciales:</b> Dado que faltan algunos campos (ej. diferenciador o DOFA), cuando el agente genera el pitch, entonces construye los pasos disponibles, completa los faltantes con la información existente, y al final indica en 1 línea qué campo completar para fortalecer un paso específico.	
<b>Escenario 3 – Módulo 1 no iniciado:</b> Dado que la tool retorna NO_DATA, cuando el agente recibe el resultado, entonces responde únicamente: 'Para generar tu pitch debes completar el Módulo 1 (Emprendimiento) en la aplicación PYMER' y se detiene.	
<b>Notas</b>	Tool: generar_elevator_pitch_tool (pitch_tools.py). Colecciones: entrepreneurship, golden_circle, dofa, buyer_persona. Duración default: 60 segundos.

#### 5.1.2.14 HU021 – Consulta de competidores del sector

HU021 Consulta de competidores del sector	
<b>Descripción</b>	El estudiante solicita información sobre sus competidores al agente. La tool buscar_competidores_tool lee ciudad y sector del negocio desde Firestore (colección 'entrepreneurship') y realiza una búsqueda en Google Places Text Search, retornando hasta 10 competidores con nombre, dirección, calificación y número de reseñas.
<b>Requerimiento Funcional Asociado</b>	RF-05 – Agente Financiero Conversacional
<b>Historias de Usuario Asociadas</b>	HU007
<b>Prioridad</b>	Media

<b>Precondiciones</b>	El estudiante tiene sesión activa. Ha registrado ciudad y sector en el perfil de emprendimiento. La Google Places API Key está configurada en el entorno.
<b>Reglas de Negocio</b>	Los parámetros 'industria' y 'ciudad' son opcionales: si no se pasan, se leen de Firestore. La búsqueda usa Text Search con query: '{sector} en {ciudad}'. Se retornan máximo 10 resultados. Si la API Key no está configurada, el agente indica el error. Si no hay resultados, el agente informa y pide verificar el sector o ciudad.
<b>Criterios de Aceptación</b>	
<b>Escenario 1 – Búsqueda exitosa:</b> Dado que el estudiante tiene ciudad y sector registrados y la API está activa, cuando solicita ver sus competidores, entonces el agente presenta de 3 a 5 competidores con nombre, zona, calificación y concluye con una observación práctica en 1 línea.	
<b>Escenario 2 – Sin datos en Firestore:</b> Dado que ciudad o sector no están registrados, cuando el agente llama la tool, entonces retorna status: 'info' y el agente pregunta en 1 línea ciudad y sector para continuar.	
<b>Escenario 3 – API no configurada:</b> Dado que la GOOGLE_PLACES_API_KEY no está en el entorno, cuando se intenta la búsqueda, entonces el agente informa que el servicio de competencia no está disponible y sugiere verificar la configuración.	
<b>Notas</b>	Tool: buscar_competidores_tool (competitor_tools.py). API externa: Google Places Text Search.

#### 5.1.2.15 HU022 – Recomendación de ubicación óptima para punto físico

<b>HU022 Recomendación de ubicación óptima para punto físico</b>	
<b>Descripción</b>	El estudiante solicita recomendación de zonas para abrir su punto físico. La tool predecir_ubicacion_optima_tool combina datos de Firestore (perfil negocio + buyer persona) con Google Places (Text Search + Nearby Search) para analizar hasta 5 zonas, calculando un score por zona basado en competidores cercanos y flujo peatonal (centros comerciales, universidades, oficinas).
<b>Requerimiento Funcional Asociado</b>	RF-05 – Agente Financiero Conversacional
<b>Historias de Usuario Asociadas</b>	HU007, HU009

<b>Prioridad</b>	Media
<b>Precondiciones</b>	El estudiante tiene sesión activa. Ha registrado ciudad y tipo de negocio. La Google Places API Key está configurada.
<b>Reglas de Negocio</b>	Score base por zona: 50 pts. +5 si el rating promedio de competidores cercanos $\geq 3.8$ . +2 pts por lugar de alto flujo peatonal (máx. +20). Radio de análisis: 600 m. Niveles: $\geq 70$ = Alta (⊙), 45-69 = Media (◐), $< 45$ = Baja (◑). El agente presenta máximo 3 zonas top con el formato exacto del system prompt.
<b>Criterios de Aceptación</b>	
<b>Escenario 1 – Análisis exitoso:</b> Dado que el estudiante tiene ciudad y sector registrados y la API está activa, cuando solicita recomendación de ubicación, entonces el agente presenta las 3 zonas top con score, dirección de referencia, factores positivos, competidores cercanos y conclusión en 2 líneas.	
<b>Escenario 2 – Ciudad no registrada:</b> Dado que la ciudad no está en Firestore, cuando el agente llama la tool, entonces retorna CIUDAD_REQUERIDA y el agente pregunta en 1 línea en qué ciudad planea abrir el punto físico.	
<b>Escenario 3 – Sin datos de Google Places:</b> Dado que no se encuentran resultados en la ciudad, cuando la tool retorna SIN_DATOS, entonces el agente informa que no hay información disponible para esa ciudad y sugiere intentar con una ciudad diferente o más grande.	
<b>Notas</b>	Tool: predecir_ubicacion_optima_tool (location_tools.py). Score inicial: 50. API: Google Places Nearby Search (radio 600m).

### 5.1.3 Estructura AGENTE PYMER

#### 5.1.3.1 Estructura del Proyecto

Ruta	Descripción
src/agent.py	Inicialización del agente principal con Google ADK.
src/main.py	Servidor FastAPI con endpoints /health y /chat.
src/session_store.py	Gestión de historial de sesiones en Firestore.
src/prompts/system_prompt.txt	Instrucciones de comportamiento del agente.
src/tools/_init_.py	Registro central de todas las tools disponibles.

Ruta	Descripción
src/tools/financiar_tools.py	Herramientas de análisis financiero.
src/tools/competitor_tools.py	Búsqueda de competidores con Google Places.
src/tools/market_tools.py	Cálculo de TAM, SAM y SOM.
src/tools/pitch_tools.py	Generación de elevator pitch en 7 pasos.
src/tools/location_tools.py	Predicción de ubicación óptima para punto físico.
src/tools/scenario_tools.py	Análisis de escenarios financieros.

### 5.1.3.2 Infraestructura

Componente	Tecnología	Detalle
Modelo de lenguaje	Gemini 2.5 Pro (Google ADK)	Agente principal (asesor_pymer).
Servidor web	FastAPI (Python)	Puerto 8080, ejecutado en Cloud Run.
Persistencia de sesiones	Firestore (Firebase)	Historial por user_id, máx. 20 mensajes.
Contenedor	Docker + Google Cloud Run	Región us-central1, 512 MB RAM.
Build	Google Cloud Build	Imagen publicada en <a href="https://gcr.io/pymer-470217/pymer-agent">gcr.io/pymer-470217/pymer-agent</a> .
API externa (competencia / ubicación)	Google Places API	Text Search y Nearby Search.
API externa (TAM)	Gemini 2.0 Flash	Estimación del mercado total por sector.

### 5.1.3.3 COLECCIONES FIRESTORE UTILIZADAS

Colección	Descripción	Campo clave de filtro
entrepreneurship	Perfil del negocio: nombre, ciudad, sector, descripción, diferenciador, propuesta de valor, equipo.	userId
financial_settings	Configuración financiera: precio por unidad, unidades mensuales planeadas, margen deseado.	userId
indirect_fixed_costs	Costos fijos indirectos (alquiler, gerente, seguros). Cada documento tiene costValue.	userId
indirect_variable_costs	Costos variables indirectos (agua, limpieza, mantenimiento). Cada documento tiene costValue.	userId

<b>Colección</b>	<b>Descripción</b>	<b>Campo clave de filtro</b>
dofa	Análisis DOFA: fortalezas, oportunidades, debilidades, amenazas.	userId
buyer_persona	Perfil del cliente ideal: edad, profesión, ingresos, ciudad, intereses, retos.	userId
golden_circle	Círculo Dorado (Simon Sinek): why, how, what.	userId
chat_sessions	Historial de conversación por sesión. Máx. 20 mensajes en DB, 10 al LLM.	session_id

## 5.1.4 TOOLS DEL AGENTE

### 5.1.4.1 Resumen de Tools Registradas

<b>Tool (FunctionTool)</b>	<b>Archivo</b>	<b>Propósito</b>
obtener_datos_financieros_tool	financial_tools.py	Lee las 3 colecciones financieras del usuario desde Firestore.
calcular_punto_equilibrio_tool	financial_tools.py	Calcula unidades y dinero necesario para cubrir costos fijos.
analizar_rentabilidad_tool	financial_tools.py	Calcula la utilidad operativa del negocio.
proyectar_ventas_tool	financial_tools.py	Proyecta ventas mensuales con tasa de crecimiento compuesta.
buscar_competidores_tool	competitor_tools.py	Busca negocios del mismo sector en la ciudad del usuario.
generar_elevator_pitch_tool	pitch_tools.py	Lee 4 colecciones y entrega contexto para generar el pitch en 7 pasos.
calcular_tam_sam_som_tool	market_tools.py	Calcula TAM (vía Gemini), SAM y SOM del negocio.
calcular_escenarios_tool	scenario_tools.py	Genera 3 escenarios financieros: conservador, moderado, arriesgado.
predecir_ubicacion_optima_tool	location_tools.py	Recomienda zonas físicas óptimas para el negocio usando Google Places.

### 5.1.4.2 Financial Tools — financial\_tools.py

<b>Función</b>	<b>Parámetros</b>	<b>Colecciones leídas</b>	<b>Retorna</b>
obtener_datos_financieros	tool_context	financial_settings, indirect_fixed_costs, indirect_variable_costs	Dict con los 3 conjuntos de datos del usuario.
calcular_punto_equilibrio	precio_venta: float, costo_variable_unitario: float, tool_context	indirect_fixed_costs (vía obtener_datos_financieros)	Unidades en equilibrio y valor en

<b>Función</b>	<b>Parámetros</b>	<b>Colecciones leídas</b>	<b>Retorna</b>
			dinero.
analizar_rentabilidad	ventas_mensuales: float, costo_ventas: float, tool_context	indirect_fixed_costs, indirect_variable_costs	Utilidad operativa mensual.
proyectar_ventas	ventas_actuales: float, crecimiento_porcentaje: float, meses: int	(no requiere Firestore ni sesión)	Lista de ventas proyectadas mes a mes.

#### 5.1.4.2.1 Fórmulas clave

<b>Métrica</b>	<b>Fórmula</b>
Margen de contribución	precio_venta - costo_variable_unitario
Punto de equilibrio (und.)	costos_fijos_totales / margen_contribución
Punto de equilibrio (\$)	pe_unidades × precio_venta
Utilidad operativa	ventas - costo_ventas - costos_fijos - costos_variables
Proyección de ventas	venta × (1 + crecimiento% / 100) por cada mes

#### 5.1.4.3 Competitor Tools — competitor\_tools.py

<b>Función</b>	<b>Parámetros</b>	<b>Fuente de datos</b>	<b>Retorna</b>
buscar_competidores	tool_context, industria?: str, ciudad?: str	Firestore (entrepreneurship) + Google Places Text Search	Lista de hasta 10 competidores con nombre, dirección, rating y reseñas.

#### 5.1.4.3.1 Lógica de prioridad de datos

<b>Dato</b>	<b>Fuente preferida</b>	<b>Fallback</b>
industria	Parámetro explícito del agente	Campo country en entrepreneurship de Firestore
ciudad	Parámetro explícito del agente	Campo city en entrepreneurship de Firestore

#### 5.1.4.4 Market Tools — market\_tools.py

<b>Función</b>	<b>Parámetros</b>	<b>Fuente de datos</b>	<b>Retorna</b>
calcular_tam_sam_som	tool_context, precio_venta?: float, industria?: str, ciudad?: str	Firestore (5 colecciones) + Gemini 2.0 Flash (TAM)	TAM, SAM, SOM con valores, notas y fuente.

#### 5.1.4.4.1 Desglose del cálculo

<b>Métrica</b>	<b>Método de cálculo</b>	<b>Factor aplicado</b>
----------------	--------------------------	------------------------

Métrica	Método de cálculo	Factor aplicado
TAM	Consulta a Gemini 2.0 Flash con el sector y país del negocio.	Global / Latinoamérica
SAM	$TAM \times \text{factor\_ciudad} \times \text{factor\_cliente}$	Ciudad principal: 20% TAM; otra ciudad: 8% TAM
SOM	$\text{unidades\_mes} \times 12 \times \text{precio\_venta}$	Precio desde parámetro > Firestore > costos + margen

#### 5.1.4.4.2 Prioridad del precio de venta para SOM

Prioridad	Fuente
1	Parámetro precio_venta enviado por el agente
2	Campo pricePerUnit en financial_settings de Firestore
3	Cálculo automático: $(CF/und + CV\_unitario) / (1 - \text{margen\_deseado})$





#### 5.1.4.5 6.5. Pitch Tools — pitch\_tools.py



Función	Parámetros	Colecciones leídas	Retorna
generar_elevator_pitch	tool_context, duracion_segundos?: int (default: 60)	entrepreneurship, golden_circle, dofa, buyer_persona	Contexto estructurado con datos del negocio e instrucción para generar el pitch.

#### 5.1.4.5.1 Campos extraídos por colección

Colección	Campos utilizados
entrepreneurship	name, description, city, values, objectives, differentiator, problem, targetAudience, valueProposition, founders, team
golden_circle	what (qué), how (cómo), why (por qué)
dofa	strengths / fortalezas, opportunities / oportunidades, weaknesses / debilidades, threats / amenazas
buyer_persona	name, age, profession, income, city, interests, challenges, objectives, phrase

#### 5.1.5 Estructura del pitch generado (7 pasos)

Paso	Nombre	Datos usados
1	 Gancho	problem, tendencia del sector
2	 Quiénes somos	founders, team, name
3	 El Problema	problem, retos del buyer persona
4	 La Solución	description, golden_circle.what, golden_circle.how
5	Beneficios y Mercado	valueProposition, targetAudience, perfil buyer persona

Paso	Nombre	Datos usados
6	 Nuestra Ventaja	differentiator, fortalezas del DOFA
7	 Llamado a la Acción	Objetivo concreto (reunión, inversión, alianza)




#### 5.1.5.1 Location Tools — location\_tools.py

Función	Parámetros	Fuente de datos	Retorna
predecir_ubicacion_optima	tool_context, ciudad?: str, tipo_negocio?: str	Firestore (entrepreneurship, buyer_persona) + Google Places (Text Search + Nearby Search)	Hasta 5 zonas ordenadas por score con análisis de competencia y flujo peatonal.

#### 5.1.5.1.1 Análisis de zona (por coordenada)



Factor analizado	Fuente	Impacto en score
Competidores del mismo sector (radio 600 m)	Google Places Nearby Search	+5 si rating promedio $\geq$ 3.8 (zona validada por mercado)
Centros comerciales cercanos	Google Places Nearby Search	+2 por lugar (máx. +20)
Universidades cercanas	Google Places Nearby Search	+2 por lugar (máx. +20)
Oficinas cercanas	Google Places Nearby Search	+2 por lugar (máx. +20)

#### 5.1.5.1.2 Niveles de score de zona

Score	Nivel
$\geq$ 70	 Alta
45 - 69	 Media
< 45	 Baja

#### 5.1.5.2 6.7. Scenario Tools — scenario\_tools.py

Genera 3 escenarios financieros para el negocio: conservador, moderado y arriesgado.

Escenario	Descripción	Propósito
 Conservador	Ventas bajas, condiciones difíciles, alta precaución.	Identificar el piso financiero del negocio.
 Moderado	Flujo de clientes normal para el	Proyección más realista

Escenario	Descripción	Propósito
(Base)	sector.	para planeación.
<input checked="" type="checkbox"/> Arriesgado / Optimista	Todas las variables a favor (viralidad, contratos B2B).	Identificar el techo posible del negocio.

#### 5.1.5.2.1 Datos calculados por escenario

Dato	Descripción
Unidades a vender por mes	Volumen de ventas proyectado en el escenario.
Ventas mensuales proyectadas	unidades × precio_venta.
Utilidad mensual	ventas – costos totales.
Punto de equilibrio	Unidades mínimas para cubrir costos fijos.

### 5.1.6 GESTIÓN DE SESIONES

#### 5.1.6.1 Flujo de sesión por petición

Paso	Descripción
1	Se recibe user_id y message en POST /chat.
2	Se busca sesión en Firestore por session_id = user_id. Si no existe, se crea.
3	Se reutiliza (o crea) la sesión ADK en memoria con InMemorySessionService.
4	Si el servidor se reinició y hay historial en Firestore, se inyecta el contexto recuperado al agente antes del mensaje nuevo.
5	Se envía el mensaje al agente y se recoge la respuesta del último evento.
6	Se persisten los dos mensajes nuevos (user + model) en Firestore.

#### 5.1.6.2 Límites de historial

Parámetro	Valor	Descripción
MAX_DB_HISTORY	20 mensajes	Máximo de mensajes guardados en Firestore por sesión.
MAX_LLM_CONTEXT	10 mensajes	Últimos mensajes enviados al LLM en cada llamada (5 pares).

### 5.1.7 API ENDPOINTS

#### 5.1.7.1 Endpoints disponibles

Método	Ruta	Descripción
GET	/health	Verifica que el servidor esté activo. Retorna {"status": "ok"}.
POST	/chat	Envía un mensaje al agente y retorna la respuesta.

#### 5.1.7.2 Esquema del endpoint /chat

Campo	Tipo	Requerido	Descripción
-------	------	-----------	-------------

<b>Campo</b>	<b>Tipo</b>	<b>Requerido</b>	<b>Descripción</b>
user_id	string	Sí	Identificador único del usuario. Actúa como session_id.
message	string	Sí	Mensaje del usuario para el agente.

### 5.1.7.3 Respuesta exitosa

<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
reply	string	Respuesta del agente financiero.
session_id	string	ID de sesión utilizado (igual al user_id).
error	boolean	Solo presente si hubo un error interno manejado.

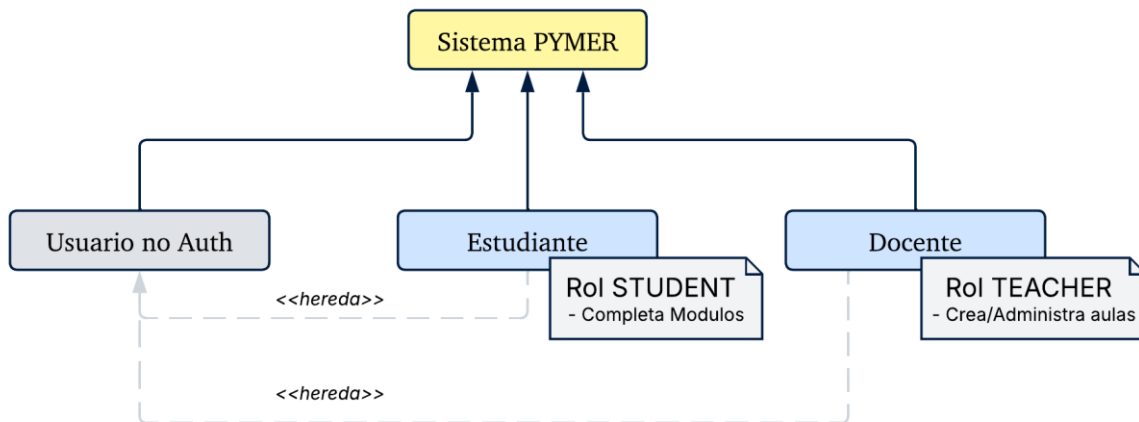
### 5.1.7.4 MANEJO DE ERRORES DEL AGENTE

<b>Código de error</b>	<b>Origen</b>	<b>Acción del agente</b>
NO_SESSION	Cualquier tool cuando user_id es nulo	El agente detiene toda acción y responde: <i>"Para acceder a tu información debes iniciar sesión en la aplicación PYMER."</i>
NO_DATA	pitch_tools / market_tools cuando Firestore no tiene datos del módulo 1	El agente indica que el módulo correspondiente no está completo y se detiene.
DATOS_INCOMPLETOS	scenario_tools cuando faltan datos financieros	El agente muestra los escenarios calculables y pide el dato faltante en 1 línea.
Sin respuesta del agente	El runner no generó texto en el evento final	Se retorna el mensaje de fallback definido en _RESPUESTAS_VACIAS.
Excepción capturada	no Cualquier error inesperado en /chat	HTTP 200 con {"reply": "Tuve un problema...", "error": true} para que el frontend siempre pueda leer el body.

## 5.2 Requerimientos funcionales Logica de negocio

### 5.2.1 ACTORES DEL SISTEMA

### 5.2.1.1 Diagrama de Actores



### 5.2.2 Descripción de Actores

Nombre del Actor	Descripción del Actor
Usuario no Autenticado	Persona que accede al sistema sin haber iniciado sesión. Sus responsabilidades se limitan a registrarse en la plataforma mediante Firebase Authentication, iniciar sesión con su token de identidad, y solicitar o confirmar el restablecimiento de su contraseña. No tiene acceso a ningún recurso protegido del sistema.
Estudiante	Usuario autenticado con el rol de STUDENT. Es responsable de completar los módulos asignados a su aula virtual, ej.: (Emprendimiento, DOFA, Lean Canvas, Golden Circle, Buyer Persona, costos directos e indirectos, configuración y proyección financiera). Puede crear, consultar y actualizar su información en cada módulo. Adicionalmente, puede enviar la completitud de módulos para revisión del docente, consultar las aulas en la que esta inscrito, ver su progreso y gestionar su perfil de usuario.
Profesor	Usuario autenticado con el rol de TEACHER. Es responsable de crear y administrar las aulas virtuales, agregar y remover estudiantes, configurar los módulos que se harán disponibles en su aula y revisar las completitudes de módulos enviadas por los estudiantes, asignando una calificación y un comentario. Puede consultar el progreso de sus estudiantes en cada uno de sus módulos.

### 5.2.3 MATRIZ DE TRAZABILIDAD DE REQUERIMIENTOS FUNCIONALES

REQUERIMIENTO FUNCIONAL	ID_HU	TÍTULO_HU
RF-01 Autenticación y Registro de Usuarios	HU001	<b>COMO</b> usuario no autenticado, <b>QUIERO</b> registrarse e iniciar sesión en la plataforma, <b>PARA</b> acceder a las herramientas de negocio.

	HU002	<b>COMO</b> usuario autenticado, <b>QUIERO</b> cambiar mi contraseña, <b>PARA</b> mantener la seguridad
	HU003	<b>COMO</b> usuario no autenticado, <b>QUIERO</b> restablecer mi contraseña mediante correo electrónico, <b>PARA</b> recuperar el acceso a mi cuenta.
RF-02 Gestión de Perfil de Usuario	HU004	<b>COMO</b> usuario autenticado, <b>QUIERO</b> actualizar mi información personal, <b>PARA</b> mantener mis datos de perfil al día.
	HU005	<b>COMO</b> usuario autenticado, <b>QUIERO</b> consultar mi estado, <b>PARA</b> saber que herramientas he completado.
	HU006	<b>COMO</b> docente, <b>QUIERO</b> consultar la información y estado de un estudiante, <b>PARA</b> hacer seguimiento a su progreso.
RF-03 Gestión de Aulas Virtuales	HU007	<b>COMO</b> docente, <b>QUIERO</b> crear un aula virtual con módulos configurables, <b>PARA</b> organizar a mis estudiantes y definir que módulos deben completar.
	HU008	<b>COMO</b> docente, <b>QUIERO</b> agregar estudiantes a un aula mediante su correo, <b>PARA</b> inscribirlos de forma sencilla.
	HU009	<b>COMO</b> docente, <b>QUIERO</b> remover a un estudiante de un aula, <b>PARA</b> gestionar el grupo.
	HU010	<b>COMO</b> docente, <b>QUIERO</b> actualizar el nombre y módulos habilitados de un aula, <b>PARA</b> adaptar la configuración según las necesidades del curso.
	HU011	<b>COMO</b> docente, <b>QUIERO</b> listar las aulas que he creado, <b>PARA</b> tener una vista general de mis grupos.
	HU012	<b>COMO</b> estudiante, <b>QUIERO</b> listar las aulas en las que estoy inscrito, <b>PARA</b> acceder a mis cursos asignados.
	HU013	<b>COMO</b> miembro del aula, <b>QUIERO</b> consultar el detalle de un aula, <b>PARA</b> conocer la información del grupo.
RF-04 Completitud de Módulos y Retroalimentación	HU014	<b>COMO</b> estudiante, <b>QUIERO</b> enviar la completitud de un módulo en mi aula, <b>PARA</b> que el docente la revise.
	HU015	<b>COMO</b> docente, <b>QUIERO</b> revisar la completitud de un módulo asignando una calificación y un comentario, <b>PARA</b> retroalimentar al estudiante.

	HU016	<b>COMO</b> miembro del aula, <b>QUIERO</b> listar las completitudes del aula, <b>PARA</b> ver el avance general.
	HU017	<b>COMO</b> docente, <b>QUIERO</b> consultar las completitudes de un estudiante en específico, <b>PARA</b> revisar su participación.
	HU018	<b>COMO</b> miembro del aula, <b>QUIERO</b> ver el progreso porcentual de un estudiante, <b>PARA</b> conocer cuántos módulos ha completado.
RF-05 Módulo de Emprendimiento	HU0019	<b>COMO</b> estudiante, <b>QUIERO</b> crear y gestionar mi información de emprendimiento, <b>PARA</b> definir la base de mi idea de negocio.
RF-06 Análisis DOFA	HU020	<b>COMO</b> estudiante, <b>QUIERO</b> crear y gestionar mi análisis DOFA, <b>PARA</b> identificar debilidades, oportunidades, fortalezas y amenazas de mi emprendimiento.
RF-07 Lean Canvas	HU021	<b>COMO</b> estudiante, <b>QUIERO</b> crear y gestionar mi Lean Canvas, <b>PARA</b> modelar mi propuesta de valor y modelo de negocio.
RF-08 Golden Circle	HU022	<b>COMO</b> estudiante, <b>QUIERO</b> crear y gestionar mi Golden Circle, <b>PARA</b> definir el porqué, cómo y qué de mi emprendimiento.
RF-09 Buyer Persona	HU023	<b>COMO</b> estudiante, <b>QUIERO</b> crear y gestionar mi Buyer Persona, <b>PARA</b> definir el perfil de mi cliente ideal.
RF-10 Costos Directos	HU024	<b>COMO</b> estudiante, <b>QUIERO</b> registrar y gestionar materia prima, mano de obra y empaquetación, <b>PARA</b> calcular el costo de fabricación.
	HU025	<b>COMO</b> estudiante, <b>QUIERO</b> consultar el costo de fabricación calculado, <b>PARA</b> conocer el costo de mi producto.
RF-11 Costos Indirectos Fijos	HU026	<b>COMO</b> estudiante, <b>QUIERO</b> registrar y gestionar mis costos indirectos fijos, <b>PARA</b> incluirlos en la estructura de costos.
	HU027	<b>COMO</b> estudiante, <b>QUIERO</b> consultar el cálculo de costos indirectos fijos, <b>PARA</b> conocer el impacto en la unidad producida.

RF-12 Costos Indirectos Variables	HU028	<b>COMO</b> estudiante, <b>QUIERO</b> registrar y gestionar mis costos indirectos variables, <b>PARA</b> incluirlos en la estructura de costos.
	HU029	<b>COMO</b> estudiante, <b>QUIERO</b> consultar el calculo de costos indirectos variables, <b>PARA</b> conocer el impacto en la unidad producida.
RF-13 Configuración Financiera y Precio sugerido	HU030	<b>COMO</b> estudiante, <b>QUIERO</b> configurar los parámetros financieros ( margen de ganancia e impuestos), <b>PARA</b> calcular el precio sugerido de venta.
	HU031	<b>COMO</b> estudiante, <b>QUIERO</b> consultar el precio sugerido de venta calculado, <b>PARA</b> definir mi estrategia de precios.
RF-14 Proyeccion Financiera	HU032	<b>COMO</b> estudiante, <b>QUIERO</b> registrar y gestionar mis activos, <b>PARA</b> proyectar la inversión necesaria de mi negocio.
	HU033	<b>COMO</b> estudiante, <b>QUIERO</b> registrar y gestionar mis deudas, <b>PARA</b> proyectar las obligaciones financieras de mi negocio.
	HU034	<b>COMO</b> estudiante, <b>QUIERO</b> consultar la estructura capital calculada, <b>PARA</b> conocer la composición de financiamiento de mi negocio.

## 5.2.4 ESPECIFICACIÓN HISTORIAS DE USUARIO

### 5.2.4.1 HU001. Registro e Inicio de Sesion

HU001. Registro e Inicio de Sesion	
<b>Descripción</b>	Permite a un usuario registrarse en la plataforma proporcionando sus datos personales (nombre, correo, contraseña). El sistema crea la cuenta en Firebase Authentication y en Firestore, y retorna un token JWT (HS256). Para iniciar sesión, el usuario envia su token de identidad de Firebase y recibe un JWT del backend.
<b>Requerimiento Funcional Asociado</b>	RF-01 Autenticacion y Registro de Usuarios
<b>Historias de Usuario Asociadas</b>	HU002, HU003

<b>Prioridad</b>	Alta
<b>Precondiciones</b>	El usuario debe tener conectividad a Internet. Para el login, debe haber completado autenticación con Firebase en el frontend.
<b>Reglas de Negocio</b>	- El correo electrónico debe ser único en el sistema. - El token de Firebase debe ser válido y no estar expirado. - El JWT generado por el backend tiene una duración configurable vía variable de entorno JWT_EXPIRATION.
<b>Criterios de Aceptación</b>	<p><b>Escenario 1 - Registro exitoso:</b> Dado que el usuario proporciona nombre, correo y contraseña válidos, <b>cuando</b> envía la solicitud de registro a POST /api/auth/register, <b>entonces</b> el sistema crea la cuenta y retorna un token JWT con HTTP 201.</p> <p>Dado que el usuario proporciona nombre, correo y contraseña válidos, cuando envía la solicitud de registro a POST /api/auth/register, entonces el sistema crea la cuenta y retorna un token JWT con HTTP 201.</p> <p><b>Escenario 2 - Registro con correo duplicado:</b> Dado que ya existe un usuario con el mismo correo, <b>cuando</b> envía la solicitud de registro, <b>entonces</b> el sistema retorna un error indicando que el correo ya está registrado.</p> <p>Dado que ya existe un usuario con el mismo correo, cuando envía la solicitud de registro, entonces el sistema retorna un error indicando que el correo ya está registrado.</p> <p><b>Escenario 3 - Login exitoso:</b> Dado que el usuario tiene un token de Firebase válido, <b>cuando</b> envía POST /api/auth/login con el idToken, <b>entonces</b> el sistema verifica el token con Firebase, genera un JWT HS256 y lo retorna con HTTP 200.</p> <p>Dado que el usuario tiene un token de Firebase válido, cuando envía POST /api/auth/login con el idToken, entonces el sistema verifica el token con Firebase, genera un JWT HS256 y lo retorna con HTTP 200.</p> <p><b>Escenario 4 - Login con token inválido:</b> Dado que el usuario envía un token de Firebase expirado o malformado, <b>cuando</b> envía la solicitud de login, <b>entonces</b> el sistema retorna un error de autenticación.</p> <p>Dado que el usuario envía un token de Firebase expirado o malformado, cuando envía la solicitud de login, entonces el sistema retorna un error de autenticación.</p>
<b>Documentos Anexos</b>	No aplica
<b>PROTOTIPO</b>	
No aplica (endpoint API REST)	

### 5.2.4.2 HU002. Cambio de Contraseña

HU002. Cambio de Contraseña	
<b>Descripción</b>	Permite a un usuario autenticado cambiar su contraseña actual proporcionando la contraseña anterior y la nueva contraseña.
<b>Requerimiento Funcional Asociado</b>	RF-01 Autenticacion y Registro de Usuarios
<b>Historias de Usuario Asociadas</b>	HU001, HU003
<b>Prioridad</b>	Media
<b>Precondiciones</b>	El usuario debe estar autenticado con un JWT valido.
<b>Reglas de Negocio</b>	- La contraseña actual debe coincidir con la almacenada. - La nueva contraseña debe cumplir los requisitos minimos de Firebase.
<b>Criterios de Aceptación</b>	<p><b>Escenario 1 - Cambio exitoso:</b> Dado que el usuario esta autenticado y proporciona su contraseña actual correcta y una nueva contraseña valida, <b>cuando</b> envia PUT /api/auth/change-password, <b>entonces</b> el sistema actualiza la contraseña y retorna HTTP 200.</p> <p>Dado que el usuario esta autenticado y proporciona su contraseña actual correcta y una nueva contraseña valida, cuando envia PUT /api/auth/change-password, entonces el sistema actualiza la contraseña y retorna HTTP 200.</p> <p><b>Escenario 2 - Contraseña actual incorrecta:</b> Dado que el usuario proporciona una contraseña actual que no coincide, <b>cuando</b> envia la solicitud, <b>entonces</b> el sistema retorna un error de validacion.</p> <p>Dado que el usuario proporciona una contraseña actual que no coincide, cuando envia la solicitud, entonces el sistema retorna un error de validacion.</p>
<b>Documentos Anexos</b>	No aplica
<b>PROTOTIPO</b>	
No aplica (endpoint API REST)	

### 5.2.4.3 HU003. Restablecimiento de Contraseña

HU003. Restablecimiento de Contraseña	
<b>Descripción</b>	Permite a un usuario no autenticado solicitar el restablecimiento de contraseña mediante su correo electrónico. El sistema envía un enlace de restablecimiento por correo (SMTP Gmail). El usuario confirma con el código OOB y la nueva contraseña.
<b>Requerimiento Funcional Asociado</b>	RF-01 Autenticación y Registro de Usuarios
<b>Historias de Usuario Asociadas</b>	HU001, HU002
<b>Prioridad</b>	Media
<b>Precondiciones</b>	El usuario debe tener una cuenta registrada con un correo electrónico válido.
<b>Reglas de Negocio</b>	<ul style="list-style-type: none"> <li>- El correo debe corresponder a un usuario existente.</li> <li>- El enlace de restablecimiento se genera mediante Firebase Identity Toolkit.</li> <li>- La confirmación requiere un código OOB válido y no expirado.</li> </ul>
<b>Criterios de Aceptación</b>	<p><b>Escenario 1 - Solicitud de restablecimiento exitosa:</b> Dado que el usuario proporciona un correo registrado, cuando envía POST /api/auth/request-password-reset, entonces el sistema envía un correo con el enlace de restablecimiento y retorna HTTP 200.</p> <p>Dado que el usuario proporciona un correo registrado, cuando envía POST /api/auth/request-password-reset, entonces el sistema envía un correo con el enlace de restablecimiento y retorna HTTP 200.</p> <p><b>Escenario 2 - Confirmación de restablecimiento exitosa:</b> Dado que el usuario tiene un código OOB válido y una nueva contraseña, cuando envía POST /api/auth/reset-password, entonces el sistema actualiza la contraseña y retorna HTTP 200.</p> <p>Dado que el usuario tiene un código OOB válido y una nueva contraseña, cuando envía POST /api/auth/reset-password, entonces el sistema actualiza la contraseña y retorna HTTP 200.</p> <p><b>Escenario 3 - Código OOB expirado:</b> Dado que el código OOB ha expirado, cuando el usuario intenta confirmar el restablecimiento, entonces el sistema retorna un error indicando que el código es inválido o ha expirado.</p> <p>Dado que el código OOB ha expirado, cuando el usuario intenta confirmar el restablecimiento, entonces el sistema retorna un error indicando que el código es inválido o ha expirado.</p>
<b>Documentos</b>	No aplica

Anexos	
<b>PROTOTIPO</b>	
No aplica (endpoint API REST)	

#### 5.2.4.4 HU004. Actualizacion de Perfil de Usuario

HU004. Actualizacion de Perfil de Usuario	
Descripción	Permite a un usuario autenticado actualizar parcialmente su informacion personal (nombre, datos adicionales) mediante una solicitud PATCH.
Requerimiento Funcional Asociado	RF-02 Gestion de Perfil de Usuario
Historias de Usuario Asociadas	HU005
Prioridad	Media
Precondiciones	El usuario debe estar autenticado con un JWT valido.
Reglas de Negocio	- Solo se actualizan los campos proporcionados en la solicitud (actualizacion parcial). - El usuario solo puede actualizar su propio perfil.
Criterios de Aceptación	<p><b>Escenario 1 - Actualizacion exitosa:</b> <b>Dado que</b> el usuario esta autenticado y envia campos validos, <b>cuando</b> envia PATCH /api/user/update, <b>entonces</b> el sistema actualiza los campos proporcionados y retorna el perfil actualizado con HTTP 200.</p> <p>Dado que el usuario esta autenticado y envia campos validos, cuando envia PATCH /api/user/update, entonces el sistema actualiza los campos proporcionados y retorna el perfil actualizado con HTTP 200.</p>
Documentos Anexos	No aplica
<b>PROTOTIPO</b>	
No aplica (endpoint API REST)	

#### 5.2.4.5 HU005. Consulta de Estado de Onboarding

HU005. Consulta de Estado de Onboarding	
<b>Descripción</b>	Permite a un usuario autenticado consultar el estado de completitud de las herramientas de negocio (cuales ha iniciado y cuales no).
<b>Requerimiento Funcional Asociado</b>	RF-02 Gestion de Perfil de Usuario
<b>Historias de Usuario Asociadas</b>	HU006, HU019 a HU034
<b>Prioridad</b>	Media
<b>Precondiciones</b>	El usuario debe estar autenticado.
<b>Reglas de Negocio</b>	- El estado se calcula verificando la existencia de registros del usuario en cada modulo de Firestore.
<b>Criterios de Aceptación</b>	<p><b>Escenario 1 - Consulta exitosa:</b> Dado que el usuario esta autenticado, cuando envia GET /api/user/status, entonces el sistema retorna un objeto indicando el estado de cada herramienta (completada/pendiente) con HTTP 200.</p> <p>Dado que el usuario esta autenticado, cuando envia GET /api/user/status, entonces el sistema retorna un objeto indicando el estado de cada herramienta (completada/pendiente) con HTTP 200.</p>
<b>Documentos Anexos</b>	No aplica
<b>PROTOTIPO</b>	
No aplica (endpoint API REST)	

#### 5.2.4.6 HU006. Consulta de Informacion de Estudiante por ID (Docente)

HU006. Consulta de Informacion de Estudiante por ID (Docente)
---

<b>Descripción</b>	Permite al docente consultar la información de perfil y el estado de onboarding de cualquier estudiante mediante su ID.
<b>Requerimiento Funcional Asociado</b>	RF-02 Gestion de Perfil de Usuario
<b>Historias de Usuario Asociadas</b>	HU005
<b>Prioridad</b>	Media
<b>Precondiciones</b>	El usuario debe estar autenticado con rol TEACHER.
<b>Reglas de Negocio</b>	- Solo usuarios con rol TEACHER pueden acceder a este endpoint. - Si el estudiante no existe, se retorna error 404.
<b>Criterios de Aceptación</b>	<p><b>Escenario 1 - Consulta exitosa:</b> Dado que el docente esta autenticado y proporciona un ID de usuario valido, <b>cuando</b> envia GET /api/user/{userId} o GET /api/user/{userId}/status, <b>entonces</b> el sistema retorna la información del estudiante con HTTP 200.</p> <p>Dado que el docente esta autenticado y proporciona un ID de usuario valido, cuando envia GET /api/user/{userId} o GET /api/user/{userId}/status, entonces el sistema retorna la información del estudiante con HTTP 200.</p> <p><b>Escenario 2 - Estudiante no encontrado:</b> Dado que el ID proporcionado no corresponde a ningun usuario, <b>cuando</b> el docente envia la solicitud, <b>entonces</b> el sistema retorna HTTP 404 con mensaje 'Usuario no encontrado'.</p> <p>Dado que el ID proporcionado no corresponde a ningun usuario, cuando el docente envia la solicitud, entonces el sistema retorna HTTP 404 con mensaje 'Usuario no encontrado'.</p>
<b>Documentos Anexos</b>	No aplica
<b>PROTOTIPO</b>	
No aplica (endpoint API REST)	

#### 5.2.4.7 HU007. Creacion de Aula Virtual

#### HU007. Creacion de Aula Virtual

<b>Descripción</b>	Permite al docente crear un aula virtual especificando un nombre y la lista de modulos habilitados. Los modulos disponibles son: ENTREPRENEURSHIP, DOFA, LEAN_CANVAS, GOLDEN_CIRCLE, COSTS, FINANCIAL, PROJECTION. El aula se crea sin miembros iniciales.
<b>Requerimiento Funcional Asociado</b>	RF-03 Gestion de Aulas Virtuales
<b>Historias de Usuario Asociadas</b>	HU008, HU009, HU010, HU011, HU013
<b>Prioridad</b>	Alta
<b>Precondiciones</b>	El usuario debe estar autenticado con rol TEACHER.
<b>Reglas de Negocio</b>	- La lista de modulos no puede estar vacia (@NotEmpty). - Solo valores validos del enum ClassroomModule son aceptados; valores invalidos son rechazados automaticamente por Jackson/Spring. - El docente queda asignado como teacherId del aula.
<b>Criterios de Aceptación</b>	<p><b>Escenario 1 - Creacion exitosa:</b> Dado que el docente proporciona un nombre y una lista de modulos validos, <b>cuando</b> envia POST /api/classrooms, <b>entonces</b> el sistema crea el aula, asigna al docente y retorna el aula creada con HTTP 201.</p> <p>Dado que el docente proporciona un nombre y una lista de modulos validos, cuando envia POST /api/classrooms, entonces el sistema crea el aula, asigna al docente y retorna el aula creada con HTTP 201.</p> <p><b>Escenario 2 - Modulo invalido:</b> Dado que el docente incluye un nombre de modulo que no existe en el enum, <b>cuando</b> envia la solicitud, <b>entonces</b> el sistema retorna HTTP 400 con mensaje de error de deserializacion.</p> <p>Dado que el docente incluye un nombre de modulo que no existe en el enum, cuando envia la solicitud, entonces el sistema retorna HTTP 400 con mensaje de error de deserializacion.</p> <p><b>Escenario 3 - Lista de modulos vacia:</b> Dado que el docente envia una lista de modulos vacia, <b>cuando</b> envia la solicitud, <b>entonces</b> el sistema retorna HTTP 400 por violacion de @NotEmpty.</p> <p>Dado que el docente envia una lista de modulos vacia, cuando envia la solicitud, entonces el sistema retorna HTTP 400 por violacion de @NotEmpty.</p>
<b>Documentos Anexos</b>	No aplica

PROTOTIPO
No aplica (endpoint API REST)

#### 5.2.4.8 HU008. Agregar Estudiantes al Aula por Correo Electronico

HU008. Agregar Estudiantes al Aula por Correo Electronico	
Descripción	Permite al docente agregar estudiantes a un aula virtual proporcionando una lista de correos electronicos. El sistema resuelve cada correo a su ID de usuario correspondiente. Si algun correo no se encuentra, se rechaza la operacion completa.
Requerimiento Funcional Asociado	RF-03 Gestion de Aulas Virtuales
Historias de Usuario Asociadas	HU007, HU009
Prioridad	Alta
Precondiciones	El usuario debe estar autenticado con rol TEACHER. El aula debe existir y pertenecer al docente.
Reglas de Negocio	- Solo el docente propietario del aula puede agregar miembros. - Los correos se resuelven contra la coleccion de usuarios en Firestore. - Si un estudiante ya es miembro, no se duplica. - Si algun correo no corresponde a un usuario registrado, se retorna un error con la lista de correos no encontrados.
Criterios de Aceptación	<p><b>Escenario 1 - Adicion exitosa:</b> Dado que todos los correos proporcionados corresponden a usuarios registrados, <b>cuando</b> el docente envia POST /api/classrooms/{classroomId}/members, <b>entonces</b> el sistema agrega los estudiantes al aula y retorna el aula actualizada con HTTP 200.</p> <p>Dado que todos los correos proporcionados corresponden a usuarios registrados, cuando el docente envia POST /api/classrooms/{classroomId}/members, entonces el sistema agrega los estudiantes al aula y retorna el aula actualizada con HTTP 200.</p> <p><b>Escenario 2 - Correos no encontrados:</b> Dado que uno o mas correos no corresponden a usuarios registrados, <b>cuando</b> el docente envia la solicitud, <b>entonces</b> el sistema retorna HTTP 404 indicando 'Usuarios no encontrados con emails: [lista]'. Dado que uno o mas correos no corresponden a usuarios registrados, cuando el docente envia la solicitud, entonces el sistema retorna HTTP 404 indicando 'Usuarios no encontrados con</p>

	<p>emails: [lista]'.  <b>Escenario 3 - Docente no es propietario: Dado que</b> el docente autenticado no es el creador del aula, <b>cuando</b> intenta agregar miembros, <b>entonces</b> el sistema retorna HTTP 403 'Solo el profesor puede agregar miembros al aula'.  Dado que el docente autenticado no es el creador del aula, cuando intenta agregar miembros, entonces el sistema retorna HTTP 403 'Solo el profesor puede agregar miembros al aula'.</p>
<b>Documentos Anexos</b>	No aplica
<b>PROTOTIPO</b>	
No aplica (endpoint API REST)	

#### 5.2.4.9 HU009. Remover Estudiante del Aula

<b>HU009. Remover Estudiante del Aula</b>	
<b>Descripción</b>	Permite al docente remover un estudiante específico de un aula virtual mediante su ID.
<b>Requerimiento Funcional Asociado</b>	RF-03 Gestion de Aulas Virtuales
<b>Historias de Usuario Asociadas</b>	HU007, HU008
<b>Prioridad</b>	Media
<b>Precondiciones</b>	El usuario debe estar autenticado con rol TEACHER. El aula debe existir y pertenecer al docente.
<b>Reglas de Negocio</b>	- Solo el docente propietario del aula puede remover miembros.

<b>Criterios de Aceptación</b>	<p><b>Escenario 1 - Remocion exitosa:</b> Dado que el docente es propietario del aula y el estudiante es miembro, <b>cuando</b> envia DELETE /api/classrooms/{classroomId}/members/{memberId}, <b>entonces</b> el sistema remueve al estudiante y retorna el aula actualizada con HTTP 200.</p> <p>Dado que el docente es propietario del aula y el estudiante es miembro, cuando envia DELETE /api/classrooms/{classroomId}/members/{memberId}, entonces el sistema remueve al estudiante y retorna el aula actualizada con HTTP 200.</p>
<b>Documentos Anexos</b>	No aplica
<b>PROTOTIPO</b>	
No aplica (endpoint API REST)	

#### 5.2.4.10 HU010. Actualizacion de Aula Virtual

<b>HU010. Actualizacion de Aula Virtual</b>	
<b>Descripción</b>	Permite al docente actualizar el nombre y/o los modulos habilitados de un aula virtual existente. Ambos campos son opcionales.
<b>Requerimiento Funcional Asociado</b>	RF-03 Gestion de Aulas Virtuales
<b>Historias de Usuario Asociadas</b>	HU007
<b>Prioridad</b>	Media
<b>Precondiciones</b>	El usuario debe estar autenticado con rol TEACHER. El aula debe existir y pertenecer al docente.
<b>Reglas de Negocio</b>	- Solo se actualizan los campos proporcionados (actualizacion parcial). - Solo el docente propietario puede actualizar el aula. - Los modulos enviados reemplazan la lista anterior completamente.

<b>Criterios de Aceptación</b>	<p><b>Escenario 1 - Actualización exitosa:</b> Dado que el docente proporciona campos validos, cuando envia PUT /api/classrooms/{classroomId}, entonces el sistema actualiza los campos proporcionados y retorna el aula actualizada con HTTP 200.</p> <p>Dado que el docente proporciona campos validos, cuando envia PUT /api/classrooms/{classroomId}, entonces el sistema actualiza los campos proporcionados y retorna el aula actualizada con HTTP 200.</p>
<b>Documentos Anexos</b>	No aplica
<b>PROTOTIPO</b>	
No aplica (endpoint API REST)	

#### 5.2.4.11 HU011. Listar Aulas del Docente

<b>HU011. Listar Aulas del Docente</b>	
<b>Descripción</b>	Permite al docente listar todas las aulas virtuales que ha creado.
<b>Requerimiento Funcional Asociado</b>	RF-03 Gestion de Aulas Virtuales
<b>Historias de Usuario Asociadas</b>	HU007
<b>Prioridad</b>	Alta
<b>Precondiciones</b>	El usuario debe estar autenticado con rol TEACHER.
<b>Reglas de Negocio</b>	- Solo se retornan las aulas donde el teacherId coincide con el usuario autenticado.
<b>Criterios de Aceptación</b>	<p><b>Escenario 1 - Listado exitoso:</b> Dado que el docente esta autenticado, cuando envia GET /api/classrooms/teaching, entonces el sistema retorna la lista de aulas del docente con HTTP 200.</p> <p>Dado que el docente esta autenticado, cuando envia GET /api/classrooms/teaching, entonces el sistema retorna la lista de aulas del docente con HTTP 200.</p> <p><b>Escenario 2 - Sin aulas:</b> Dado que el docente no ha creado</p>

	<p>ninguna aula, <b>cuando</b> envia la solicitud, <b>entonces</b> el sistema retorna una lista vacia con HTTP 200.</p> <p>Dado que el docente no ha creado ninguna aula, cuando envia la solicitud, entonces el sistema retorna una lista vacia con HTTP 200.</p>
<b>Documentos Anexos</b>	No aplica
<b>PROTOTIPO</b>	
No aplica (endpoint API REST)	

#### 5.2.4.12 HU012. Listar Aulas del Estudiante

<b>HU012. Listar Aulas del Estudiante</b>	
<b>Descripción</b>	Permite al estudiante listar todas las aulas virtuales en las que esta inscrito como miembro.
<b>Requerimiento Funcional Asociado</b>	RF-03 Gestion de Aulas Virtuales
<b>Historias de Usuario Asociadas</b>	HU013
<b>Prioridad</b>	Alta
<b>Precondiciones</b>	El usuario debe estar autenticado con rol STUDENT.
<b>Reglas de Negocio</b>	- Solo se retornan las aulas donde el ID del usuario aparece en la lista de memberIds.
<b>Criterios de Aceptación</b>	<p><b>Escenario 1 - Listado exitoso:</b> Dado que el estudiante esta autenticado, <b>cuando</b> envia GET /api/classrooms/enrolled, <b>entonces</b> el sistema retorna la lista de aulas donde es miembro con HTTP 200.</p> <p>Dado que el estudiante esta autenticado, cuando envia GET /api/classrooms/enrolled, entonces el sistema retorna la lista de aulas donde es miembro con HTTP 200.</p>

Documentos Anexos	No aplica
<b>PROTOTIPO</b>	
No aplica (endpoint API REST)	

#### 5.2.4.13 HU013. Consultar Detalle de Aula

<b>HU013. Consultar Detalle de Aula</b>	
<b>Descripción</b>	Permite a cualquier miembro del aula (docente o estudiante inscrito) consultar el detalle completo de un aula, incluyendo nombre del docente, lista de miembros con nombre y correo, y modulos habilitados.
<b>Requerimiento Funcional Asociado</b>	RF-03 Gestion de Aulas Virtuales
<b>Historias de Usuario Asociadas</b>	HU007, HU012
<b>Prioridad</b>	Media
<b>Precondiciones</b>	El usuario debe estar autenticado y ser miembro o docente del aula.
<b>Reglas de Negocio</b>	- Si el usuario no es docente ni miembro del aula, se retorna error 403. - Los nombres de los miembros se resuelven desde la coleccion de usuarios.
<b>Criterios de Aceptación</b>	<p><b>Escenario 1 - Consulta exitosa:</b> Dado que el usuario es miembro o docente del aula, cuando envia GET /api/classrooms/{classroomId}, entonces el sistema retorna el detalle con nombres resueltos y HTTP 200.</p> <p>Dado que el usuario es miembro o docente del aula, cuando envia GET /api/classrooms/{classroomId}, entonces el sistema retorna el detalle con nombres resueltos y HTTP 200.</p> <p><b>Escenario 2 - Sin acceso:</b> Dado que el usuario no es miembro ni docente del aula, cuando envia la solicitud, entonces el sistema retorna HTTP 403 'No tiene acceso a esta aula'.</p> <p>Dado que el usuario no es miembro ni docente del aula, cuando envia la solicitud, entonces el sistema retorna HTTP 403 'No tiene acceso a esta aula'.</p>
<b>Documentos</b>	No aplica

Anexos	
PROTOTIPO	
No aplica (endpoint API REST)	

#### 5.2.4.14 HU014. Enviar Completitud de Modulo

HU014. Enviar Completitud de Modulo	
Descripción	Permite al estudiante enviar la completitud de un modulo especifico dentro de un aula virtual. El sistema valida que el modulo este habilitado en el aula y que no exista ya una completitud previa para ese modulo. El estado inicial es PENDING_REVIEW.
Requerimiento Funcional Asociado	RF-04 Completitud de Modulos y Retroalimentacion
Historias de Usuario Asociadas	HU015, HU018
Prioridad	Alta
Precondiciones	El usuario debe estar autenticado con rol STUDENT. Debe ser miembro del aula. El modulo debe estar habilitado en el aula.
Reglas de Negocio	- El estudiante debe ser miembro del aula. - El modulo enviado debe pertenecer a la lista de modulos configurados del aula. - No se permite enviar dos veces el mismo modulo en la misma aula (duplicado). - Se registra el nombre completo del estudiante en el documento para facilitar consultas.
Criterios de Aceptación	<p><b>Escenario 1 - Envio exitoso:</b> <b>Dado que</b> el estudiante es miembro del aula y el modulo esta habilitado y no ha sido enviado previamente, <b>cuando</b> envia POST /api/classrooms/{classroomId}/completions con { "module": "DOFA" }, <b>entonces</b> el sistema crea la completitud con estado PENDING_REVIEW y retorna HTTP 201.</p> <p>Dado que el estudiante es miembro del aula y el modulo esta habilitado y no ha sido enviado previamente, cuando envia POST /api/classrooms/{classroomId}/completions con { "module": "DOFA" }, entonces el sistema crea la completitud con estado PENDING_REVIEW y retorna HTTP 201.</p> <p><b>Escenario 2 - Modulo no habilitado:</b> <b>Dado que</b> el modulo enviado no esta en la lista de modulos del aula, <b>cuando</b> el estudiante envia la solicitud, <b>entonces</b> el sistema retorna HTTP 400 indicando que el modulo no esta habilitado en el aula.</p>

	<p>Dado que el modulo enviado no esta en la lista de modulos del aula, cuando el estudiante envia la solicitud, entonces el sistema retorna HTTP 400 indicando que el modulo no esta habilitado en el aula.</p> <p><b>Escenario 3 - Modulo ya enviado:</b> Dado que el estudiante ya envio la completitud de ese modulo, <b>cuando</b> intenta enviarlo nuevamente, <b>entonces</b> el sistema retorna HTTP 400 indicando duplicado.</p> <p>Dado que el estudiante ya envio la completitud de ese modulo, cuando intenta enviarlo nuevamente, entonces el sistema retorna HTTP 400 indicando duplicado.</p> <p><b>Escenario 4 - No es miembro del aula:</b> Dado que el estudiante no pertenece al aula, <b>cuando</b> envia la solicitud, <b>entonces</b> el sistema retorna HTTP 403.</p> <p>Dado que el estudiante no pertenece al aula, cuando envia la solicitud, entonces el sistema retorna HTTP 403.</p>
<b>Documentos Anexos</b>	No aplica
<b>PROTOTIPO</b>	
No aplica (endpoint API REST)	

#### 5.2.4.15HU015. Revisar Completitud de Modulo (Docente)

<b>HU015. Revisar Completitud de Modulo (Docente)</b>	
<b>Descripción</b>	Permite al docente revisar la completitud de un modulo enviada por un estudiante, asignando opcionalmente una calificacion numerica y un comentario obligatorio. El estado cambia a REVIEWED.
<b>Requerimiento Funcional Asociado</b>	RF-04 Completitud de Modulos y Retroalimentacion
<b>Historias de Usuario Asociadas</b>	HU014, HU016
<b>Prioridad</b>	Alta
<b>Precondiciones</b>	El usuario debe estar autentificado con rol TEACHER. Debe ser el docente propietario del aula. La completitud debe existir.

<b>Reglas de Negocio</b>	- Solo el docente propietario del aula puede revisar. - La calificación (grade) es opcional (tipo Double). - El comentario es obligatorio (@NotBlank). - El estado cambia de PENDING_REVIEW a REVIEWED.
<b>Criterios de Aceptación</b>	<p><b>Escenario 1 - Revisión exitosa:</b> Dado que el docente es propietario del aula y la completitud existe, <b>cuando</b> envía POST /api/classrooms/{classroomId}/completions/{completionId}/review con calificación y comentario, <b>entonces</b> el sistema actualiza el estado a REVIEWED y retorna HTTP 200.</p> <p>Dado que el docente es propietario del aula y la completitud existe, cuando envía POST /api/classrooms/{classroomId}/completions/{completionId}/review con calificación y comentario, entonces el sistema actualiza el estado a REVIEWED y retorna HTTP 200.</p> <p><b>Escenario 2 - No es docente del aula:</b> Dado que el usuario autenticado no es el docente del aula, <b>cuando</b> intenta revisar, <b>entonces</b> el sistema retorna HTTP 403.</p> <p>Dado que el usuario autenticado no es el docente del aula, cuando intenta revisar, entonces el sistema retorna HTTP 403.</p>
<b>Documentos Anexos</b>	No aplica
<b>PROTOTIPO</b>	
No aplica (endpoint API REST)	

#### 5.2.4.16 HU016. Listar Completitudes del Aula

<b>HU016. Listar Completitudes del Aula</b>	
<b>Descripción</b>	Permite a cualquier miembro del aula listar las completitudes de módulos, con filtro opcional por estado (PENDING_REVIEW, REVIEWED).
<b>Requerimiento Funcional Asociado</b>	RF-04 Completitud de Módulos y Retroalimentación
<b>Historias de Usuario Asociadas</b>	HU014, HU015
<b>Prioridad</b>	Media

<b>Precondiciones</b>	El usuario debe estar autenticado y ser miembro o docente del aula.
<b>Reglas de Negocio</b>	- Se resuelven los nombres de los estudiantes en lote para la respuesta. - El parametro status es opcional; si se omite, retorna todas las completitudes.
<b>Criterios de Aceptación</b>	<p><b>Escenario 1 - Listado completo:</b> Dado que el usuario es miembro del aula, cuando envia GET /api/classrooms/{classroomId}/completions, entonces el sistema retorna todas las completitudes del aula con HTTP 200.</p> <p>Dado que el usuario es miembro del aula, cuando envia GET /api/classrooms/{classroomId}/completions, entonces el sistema retorna todas las completitudes del aula con HTTP 200.</p> <p><b>Escenario 2 - Filtrado por estado:</b> Dado que el usuario filtra por ?status=PENDING_REVIEW, cuando envia la solicitud, entonces el sistema retorna solo las completitudes con estado PENDING_REVIEW.</p> <p>Dado que el usuario filtra por ?status=PENDING_REVIEW, cuando envia la solicitud, entonces el sistema retorna solo las completitudes con estado PENDING_REVIEW.</p>
<b>Documentos Anexos</b>	No aplica
<b>PROTOTIPO</b>	
No aplica (endpoint API REST)	

#### 5.2.4.17 HU017. Consultar Completitudes de un Estudiante

<b>HU017. Consultar Completitudes de un Estudiante</b>	
<b>Descripción</b>	Permite al docente o al propio estudiante consultar todas las completitudes de modulos de un estudiante especifico dentro de un aula.
<b>Requerimiento Funcional Asociado</b>	RF-04 Completitud de Modulos y Retroalimentacion
<b>Historias de Usuario Asociadas</b>	HU014, HU018
<b>Prioridad</b>	Media

<b>Precondiciones</b>	El usuario debe estar autenticado. Debe ser el docente del aula o el propio estudiante.
<b>Reglas de Negocio</b>	- El docente puede ver las completitudes de cualquier estudiante del aula. - Un estudiante solo puede ver sus propias completitudes.
<b>Criterios de Aceptación</b>	<b>Escenario 1 - Consulta exitosa: Dado que</b> el usuario tiene acceso (es docente del aula o el propio estudiante), <b>cuando</b> envía GET /api/classrooms/{classroomId}/completions/students/{studentId}, <b>entonces</b> el sistema retorna las completitudes del estudiante con HTTP 200.  Dado que el usuario tiene acceso (es docente del aula o el propio estudiante), cuando envía GET /api/classrooms/{classroomId}/completions/students/{studentId}, entonces el sistema retorna las completitudes del estudiante con HTTP 200.
<b>Documentos Anexos</b>	No aplica
<b>PROTOTIPO</b>	
No aplica (endpoint API REST)	

#### 5.2.4.18HU018. Consultar Progreso Porcentual del Estudiante

<b>HU018. Consultar Progreso Porcentual del Estudiante</b>	
<b>Descripción</b>	Permite consultar el porcentaje de avance de un estudiante en un aula, calculado como la cantidad de modulos con completitud enviada (cualquier estado) dividida entre el total de modulos habilitados en el aula. Incluye el detalle de cada modulo.
<b>Requerimiento Funcional Asociado</b>	RF-04 Completitud de Modulos y Retroalimentacion
<b>Historias de Usuario Asociadas</b>	HU014, HU017
<b>Prioridad</b>	Alta
<b>Precondicione s</b>	El usuario debe estar autenticado. Debe ser el docente del aula o el propio estudiante.

<b>Reglas de Negocio</b>	<p>- El progreso se calcula contra los modulos configurados del aula (classroom.getModules().size()). - Si el aula no tiene modulos configurados, se usa el total del enum ClassroomModule como fallback. - Tanto modulos en PENDING_REVIEW como REVIEWED cuentan como completados para el porcentaje.</p>
<b>Criterios de Aceptación</b>	<p><b>Escenario 1 - Progreso parcial:</b> Dado que el aula tiene 5 modulos habilitados y el estudiante ha enviado 3, <b>cuando</b> se consulta GET /api/classrooms/{classroomId}/completions/students/{studentId}/progress, <b>entonces</b> el sistema retorna completionPercentage: 60.0 con la lista de modulos y su estado.</p> <p>Dado que el aula tiene 5 modulos habilitados y el estudiante ha enviado 3, cuando se consulta GET /api/classrooms/{classroomId}/completions/students/{studentId}/progress, entonces el sistema retorna completionPercentage: 60.0 con la lista de modulos y su estado.</p> <p><b>Escenario 2 - Sin completitudes:</b> Dado que el estudiante no ha enviado ninguna completitud, <b>cuando</b> se consulta el progreso, <b>entonces</b> el sistema retorna completionPercentage: 0.0 con todos los modulos marcados como no completados.</p> <p>Dado que el estudiante no ha enviado ninguna completitud, cuando se consulta el progreso, entonces el sistema retorna completionPercentage: 0.0 con todos los modulos marcados como no completados.</p>
<b>Documentos Anexos</b>	No aplica
<b>PROTOTIPO</b>	
No aplica (endpoint API REST)	

#### 5.2.4.19 HU019. Gestion del Modulo de Emprendimiento

<b>HU019. Gestion del Modulo de Emprendimiento</b>	
<b>Descripción</b>	Permite al estudiante crear, consultar y actualizar su informacion de emprendimiento (nombre del negocio, descripcion, sector, etc.). Cada usuario tiene un unico registro de emprendimiento.
<b>Requerimiento Funcional Asociado</b>	RF-05 Modulo de Emprendimiento
<b>Historias de Usuario Asociadas</b>	HU005

<b>Prioridad</b>	Alta
<b>Precondiciones</b>	El usuario debe estar autenticado.
<b>Reglas de Negocio</b>	- Cada usuario tiene maximo un registro de emprendimiento. - Los campos deben cumplir las validaciones definidas en CreateEntrepreneurshipRequest.
<b>Criterios de Aceptación</b>	<p><b>Escenario 1 - Creacion exitosa:</b> Dado que el estudiante no tiene registro de emprendimiento, <b>cuando</b> envia POST /api/entrepreneurship con datos validos, <b>entonces</b> el sistema crea el registro y retorna HTTP 201.</p> <p>Dado que el estudiante no tiene registro de emprendimiento, cuando envia POST /api/entrepreneurship con datos validos, entonces el sistema crea el registro y retorna HTTP 201.</p> <p><b>Escenario 2 - Consulta exitosa:</b> Dado que el estudiante tiene un registro de emprendimiento, <b>cuando</b> envia GET /api/entrepreneurship, <b>entonces</b> el sistema retorna la informacion con HTTP 200.</p> <p>Dado que el estudiante tiene un registro de emprendimiento, cuando envia GET /api/entrepreneurship, entonces el sistema retorna la informacion con HTTP 200.</p> <p><b>Escenario 3 - Actualizacion exitosa:</b> Dado que el estudiante tiene un registro existente, <b>cuando</b> envia PUT /api/entrepreneurship con datos actualizados, <b>entonces</b> el sistema actualiza y retorna el registro con HTTP 200.</p> <p>Dado que el estudiante tiene un registro existente, cuando envia PUT /api/entrepreneurship con datos actualizados, entonces el sistema actualiza y retorna el registro con HTTP 200.</p>
<b>Documentos Anexos</b>	No aplica
<b>PROTOTIPO</b>	
No aplica (endpoint API REST)	

#### 5.2.4.20 HU020. Gestion del Analisis DOFA

<b>HU020. Gestion del Analisis DOFA</b>	
<b>Descripción</b>	Permite al estudiante crear, consultar y actualizar su analisis DOFA (Debilidades, Oportunidades, Fortalezas, Amenazas).
<b>Requerimiento Funcional</b>	RF-06 Analisis DOFA

<b>Asociado</b>	
<b>Historias de Usuario Asociadas</b>	HU005
<b>Prioridad</b>	Alta
<b>Precondiciones</b>	El usuario debe estar autenticado.
<b>Reglas de Negocio</b>	- Cada usuario tiene maximo un registro DOFA. - La consulta por ID valida que el registro pertenezca al usuario autenticado.
<b>Criterios de Aceptación</b>	<p><b>Escenario 1 - Creacion exitosa:</b> Dado que el estudiante proporciona datos validos, <b>cuando</b> envia POST /api/dofa, <b>entonces</b> el sistema crea el analisis DOFA y retorna HTTP 201.</p> <p>Dado que el estudiante proporciona datos validos, cuando envia POST /api/dofa, entonces el sistema crea el analisis DOFA y retorna HTTP 201.</p> <p><b>Escenario 2 - Consulta y actualizacion:</b> Dado que el estudiante tiene un registro DOFA, <b>cuando</b> consulta con GET /api/dofa/{id} o actualiza con PUT /api/dofa/{id}, <b>entonces</b> el sistema retorna los datos correspondientes con HTTP 200.</p> <p>Dado que el estudiante tiene un registro DOFA, cuando consulta con GET /api/dofa/{id} o actualiza con PUT /api/dofa/{id}, entonces el sistema retorna los datos correspondientes con HTTP 200.</p>
<b>Documentos Anexos</b>	No aplica
<b>PROTOTIPO</b>	
No aplica (endpoint API REST)	

#### 5.2.4.21 HU021. Gestion del Lean Canvas

<b>HU021. Gestion del Lean Canvas</b>	
<b>Descripción</b>	Permite al estudiante crear, consultar y actualizar su Lean Canvas (problema, solucion, propuesta de valor, segmentos, canales, ingresos, costos, metricas, ventaja competitiva).
<b>Requerimiento Funcional</b>	RF-07 Lean Canvas

<b>Asociado</b>	
<b>Historias de Usuario Asociadas</b>	HU005
<b>Prioridad</b>	Alta
<b>Precondiciones</b>	El usuario debe estar autenticado.
<b>Reglas de Negocio</b>	- Cada usuario tiene maximo un registro de Lean Canvas.
<b>Criterios de Aceptación</b>	<p><b>Escenario 1 - CRUD exitoso: Dado que</b> el estudiante esta autenticado, <b>cuando</b> crea (POST /api/lc), consulta (GET /api/lc/{id}) o actualiza (PUT /api/lc/{id}), <b>entonces</b> el sistema ejecuta la operacion y retorna el Lean Canvas con el codigo HTTP correspondiente.</p> <p>Dado que el estudiante esta autenticado, cuando crea (POST /api/lc), consulta (GET /api/lc/{id}) o actualiza (PUT /api/lc/{id}), entonces el sistema ejecuta la operacion y retorna el Lean Canvas con el codigo HTTP correspondiente.</p>
<b>Documentos Anexos</b>	No aplica
<b>PROTOTIPO</b>	
No aplica (endpoint API REST)	

#### 5.2.4.22 HU022. Gestion del Golden Circle

<b>HU022. Gestion del Golden Circle</b>	
<b>Descripción</b>	Permite al estudiante crear, consultar y actualizar su Golden Circle (Why, How, What).
<b>Requerimiento Funcional Asociado</b>	RF-08 Golden Circle
<b>Historias de Usuario Asociadas</b>	HU005

<b>Prioridad</b>	Alta
<b>Precondiciones</b>	El usuario debe estar autenticado.
<b>Reglas de Negocio</b>	- Cada usuario tiene maximo un registro de Golden Circle.
<b>Criterios de Aceptación</b>	<p><b>Escenario 1 - CRUD exitoso: Dado que</b> el estudiante esta autenticado, <b>cuando</b> crea (POST /api/gc), consulta (GET /api/gc/{id}) o actualiza (PUT /api/gc/{id}), <b>entonces</b> el sistema ejecuta la operacion y retorna el Golden Circle con el codigo HTTP correspondiente.</p> <p>Dado que el estudiante esta autenticado, cuando crea (POST /api/gc), consulta (GET /api/gc/{id}) o actualiza (PUT /api/gc/{id}), entonces el sistema ejecuta la operacion y retorna el Golden Circle con el codigo HTTP correspondiente.</p>
<b>Documentos Anexos</b>	No aplica
<b>PROTOTIPO</b>	
No aplica (endpoint API REST)	

#### 5.2.4.23 HU023. Gestion del Buyer Persona

<b>HU023. Gestion del Buyer Persona</b>	
<b>Descripción</b>	Permite al estudiante crear, consultar y actualizar su Buyer Persona (perfil demografico, comportamientos, motivaciones, frustraciones del cliente ideal).
<b>Requerimiento Funcional Asociado</b>	RF-09 Buyer Persona
<b>Historias de Usuario Asociadas</b>	HU005
<b>Prioridad</b>	Alta
<b>Precondiciones</b>	El usuario debe estar autenticado.

<b>Reglas de Negocio</b>	- Cada usuario tiene maximo un registro de Buyer Persona.
<b>Criterios de Aceptación</b>	<p><b>Escenario 1 - CRUD exitoso: Dado que</b> el estudiante esta autenticado, <b>cuando</b> crea (POST /api/bp), consulta (GET /api/bp/{id}) o actualiza (PUT /api/bp/{id}), <b>entonces</b> el sistema ejecuta la operacion y retorna el Buyer Persona con el codigo HTTP correspondiente.</p> <p>Dado que el estudiante esta autenticado, cuando crea (POST /api/bp), consulta (GET /api/bp/{id}) o actualiza (PUT /api/bp/{id}), entonces el sistema ejecuta la operacion y retorna el Buyer Persona con el codigo HTTP correspondiente.</p>
<b>Documentos Anexos</b>	No aplica
<b>PROTOTIPO</b>	
No aplica (endpoint API REST)	

#### 5.2.4.24 HU024. Gestion de Costos Directos (Materia Prima, Mano de Obra, Empaque)

<b>HU024. Gestion de Costos Directos (Materia Prima, Mano de Obra, Empaque)</b>	
<b>Descripción</b>	Permite al estudiante registrar, consultar, actualizar y eliminar los componentes de costos directos de su producto: materia prima, mano de obra y empaque. Soporta operaciones individuales y en lote (batch).
<b>Requerimiento Funcional Asociado</b>	RF-10 Costos Directos
<b>Historias de Usuario Asociadas</b>	HU025
<b>Prioridad</b>	Alta
<b>Precondiciones</b>	El usuario debe estar autenticado.
<b>Reglas de Negocio</b>	- Cada item pertenece al usuario que lo creo; no se pueden modificar items de otros usuarios. - Las operaciones batch procesan listas de registros en una sola solicitud. - Los calculos monetarios usan BigDecimal con escala 4 y RoundingMode.HALF_UP.

<p><b>Criterios de Aceptación</b></p>	<p><b>Escenario 1 - Creacion individual de materia prima:</b> Dado que el estudiante proporciona datos validos de un material, <b>cuando</b> envia POST /api/costs/direct/material, <b>entonces</b> el sistema crea el registro y retorna HTTP 201.</p> <p>Dado que el estudiante proporciona datos validos de un material, cuando envia POST /api/costs/direct/material, entonces el sistema crea el registro y retorna HTTP 201.</p> <p><b>Escenario 2 - Creacion en lote:</b> Dado que el estudiante proporciona una lista de registros validos, <b>cuando</b> envia POST /api/costs/direct/material/batch, <b>entonces</b> el sistema crea todos los registros y retorna la lista con HTTP 201.</p> <p>Dado que el estudiante proporciona una lista de registros validos, cuando envia POST /api/costs/direct/material/batch, entonces el sistema crea todos los registros y retorna la lista con HTTP 201.</p> <p><b>Escenario 3 - Eliminacion:</b> Dado que el estudiante es propietario del registro, <b>cuando</b> envia DELETE /api/costs/direct/material/{id}, <b>entonces</b> el sistema elimina el registro y retorna HTTP 204.</p> <p>Dado que el estudiante es propietario del registro, cuando envia DELETE /api/costs/direct/material/{id}, entonces el sistema elimina el registro y retorna HTTP 204.</p> <p><b>Escenario 4 - Operaciones de mano de obra y empaque:</b> Las mismas operaciones aplican para /api/costs/direct/labour y /api/costs/direct/packaging con sus respectivos DTOs.</p> <p>Las mismas operaciones aplican para /api/costs/direct/labour y /api/costs/direct/packaging con sus respectivos DTOs.</p>
<p><b>Documentos Anexos</b></p>	<p>No aplica</p>
<p style="text-align: center;"><b>PROTOTIPO</b></p>	
<p style="text-align: center;">No aplica (endpoint API REST)</p>	

#### 5.2.4.25 HU025. Consulta del Costo de Fabricacion

<p style="text-align: center;"><b>HU025. Consulta del Costo de Fabricacion</b></p>	
<p><b>Descripción</b></p>	<p>Permite al estudiante consultar el costo de fabricacion calculado, que agrega los costos de materia prima, mano de obra y empaque por unidad de producto.</p>
<p><b>Requerimiento Funcional Asociado</b></p>	<p>RF-10 Costos Directos</p>

<b>Historias de Usuario Asociadas</b>	HU024
<b>Prioridad</b>	Alta
<b>Precondiciones</b>	El usuario debe estar autenticado y tener registros de costos directos.
<b>Reglas de Negocio</b>	- El calculo agrega costos unitarios de los tres componentes directos. - Los calculos usan BigDecimal con precision de 4 decimales.
<b>Criterios de Aceptación</b>	<b>Escenario 1 - Calculo exitoso: Dado que</b> el estudiante tiene registros de costos directos, <b>cuando</b> envia GET /api/costs/direct/fabrication-cost, <b>entonces</b> el sistema retorna el costo de fabricacion desglosado con HTTP 200.  Dado que el estudiante tiene registros de costos directos, cuando envia GET /api/costs/direct/fabrication-cost, entonces el sistema retorna el costo de fabricacion desglosado con HTTP 200.
<b>Documentos Anexos</b>	No aplica
<b>PROTOTIPO</b>	
No aplica (endpoint API REST)	

#### 5.2.4.26 HU026. Gestion de Costos Indirectos Fijos

<b>HU026. Gestion de Costos Indirectos Fijos</b>	
<b>Descripción</b>	Permite al estudiante registrar, consultar, actualizar y eliminar costos indirectos fijos (arriendo, servicios publicos, seguros, etc.). Soporta operaciones individuales y en lote.
<b>Requerimiento Funcional Asociado</b>	RF-11 Costos Indirectos Fijos
<b>Historias de Usuario Asociadas</b>	HU027
<b>Prioridad</b>	Alta

<b>Precondiciones</b>	El usuario debe estar autenticado.
<b>Reglas de Negocio</b>	- Cada costo fijo tiene un monto mensual y una descripción. - Las operaciones batch permiten crear/actualizar/eliminar múltiples registros.
<b>Criterios de Aceptación</b>	<p><b>Escenario 1 - CRUD individual y batch:</b> Dado que el estudiante está autenticado, <b>cuando</b> realiza operaciones CRUD en /api/costos/indirect/fixed (individual o /batch), <b>entonces</b> el sistema ejecuta la operación y retorna el resultado con el código HTTP correspondiente.</p> <p>Dado que el estudiante está autenticado, cuando realiza operaciones CRUD en /api/costos/indirect/fixed (individual o /batch), entonces el sistema ejecuta la operación y retorna el resultado con el código HTTP correspondiente.</p>
<b>Documentos Anexos</b>	No aplica
<b>PROTOTIPO</b>	
No aplica (endpoint API REST)	

#### 5.2.4.27 HU027. Calculo de Costos Indirectos Fijos Prorrateados

<b>HU027. Calculo de Costos Indirectos Fijos Prorrateados</b>	
<b>Descripción</b>	Permite al estudiante consultar el cálculo de prorratio de sus costos indirectos fijos, distribuyendo el costo mensual total entre las unidades producidas planificadas para obtener el costo fijo por unidad.
<b>Requerimiento Funcional Asociado</b>	RF-11 Costos Indirectos Fijos
<b>Historias de Usuario Asociadas</b>	HU026
<b>Prioridad</b>	Alta
<b>Precondiciones</b>	El usuario debe tener registros de costos indirectos fijos y configuración de unidades producidas.
<b>Reglas de Negocio</b>	- La fórmula de prorratio es: $\text{costoMensualTotal} / \text{unidadesPlanificadasMensuales}$ . - Se usa BigDecimal con escala 4 y RoundingMode.HALF_UP para evitar pérdida de precisión.

<b>Criterios de Aceptación</b>	<p><b>Escenario 1 - Calculo exitoso:</b> Dado que el estudiante tiene costos fijos registrados, cuando envia GET /api/costs/indirect/fixed/calculated, entonces el sistema retorna el costo prorrateado por unidad con HTTP 200.</p> <p>Dado que el estudiante tiene costos fijos registrados, cuando envia GET /api/costs/indirect/fixed/calculated, entonces el sistema retorna el costo prorrateado por unidad con HTTP 200.</p>
<b>Documentos Anexos</b>	No aplica
<b>PROTOTIPO</b>	
No aplica (endpoint API REST)	

#### 5.2.4.28 HU028. Gestion de Costos Indirectos Variables

<b>HU028. Gestion de Costos Indirectos Variables</b>	
<b>Descripción</b>	Permite al estudiante registrar, consultar, actualizar y eliminar costos indirectos variables (comisiones, transporte, insumos variables, etc.). Soporta operaciones individuales y en lote.
<b>Requerimiento Funcional Asociado</b>	RF-12 Costos Indirectos Variables
<b>Historias de Usuario Asociadas</b>	HU029
<b>Prioridad</b>	Alta
<b>Precondiciones</b>	El usuario debe estar autenticado.
<b>Reglas de Negocio</b>	- Mismas reglas de negocio que costos indirectos fijos aplicadas a costos variables.

<b>Criterios de Aceptación</b>	<p><b>Escenario 1 - CRUD individual y batch:</b> Dado que el estudiante esta autenticado, <b>cuando</b> realiza operaciones CRUD en /api/costs/indirect/variable (individual o /batch), <b>entonces</b> el sistema ejecuta la operacion y retorna el resultado con el codigo HTTP correspondiente.</p> <p>Dado que el estudiante esta autenticado, cuando realiza operaciones CRUD en /api/costs/indirect/variable (individual o /batch), entonces el sistema ejecuta la operacion y retorna el resultado con el codigo HTTP correspondiente.</p>
<b>Documentos Anexos</b>	No aplica
<b>PROTOTIPO</b>	
No aplica (endpoint API REST)	

#### 5.2.4.29 HU029. Calculo de Costos Indirectos Variables Prorrateados

<b>HU029. Calculo de Costos Indirectos Variables Prorrateados</b>	
<b>Descripción</b>	Permite al estudiante consultar el calculo de prorrateo de sus costos indirectos variables por unidad producida.
<b>Requerimiento Funcional Asociado</b>	RF-12 Costos Indirectos Variables
<b>Historias de Usuario Asociadas</b>	HU028
<b>Prioridad</b>	Alta
<b>Precondiciones</b>	El usuario debe tener registros de costos indirectos variables.
<b>Reglas de Negocio</b>	- Misma formula de prorrateo que costos fijos: $\text{costoMensualTotal} / \text{unidadesPlanificadasMensuales}$ . - Usa el componente compartido CalculateProratedCost.

<b>Criterios de Aceptación</b>	<p><b>Escenario 1 - Calculo exitoso: Dado que</b> el estudiante tiene costos variables registrados, <b>cuando</b> envia GET /api/costs/indirect/variable/calculated, <b>entonces</b> el sistema retorna el costo variable prorrateado por unidad con HTTP 200.</p> <p>Dado que el estudiante tiene costos variables registrados, cuando envia GET /api/costs/indirect/variable/calculated, entonces el sistema retorna el costo variable prorrateado por unidad con HTTP 200.</p>
<b>Documentos Anexos</b>	No aplica
<b>PROTOTIPO</b>	
No aplica (endpoint API REST)	

#### 5.2.4.30 HU030. Configuración de Parametros Financieros

<b>HU030. Configuración de Parametros Financieros</b>	
<b>Descripción</b>	Permite al estudiante crear, consultar y actualizar su configuración financiera (margen de ganancia, porcentaje de impuestos, y demas parametros necesarios para el calculo del precio sugerido de venta).
<b>Requerimiento Funcional Asociado</b>	RF-13 Configuración Financiera y Precio Sugerido
<b>Historias de Usuario Asociadas</b>	HU031
<b>Prioridad</b>	Alta
<b>Precondiciones</b>	El usuario debe estar autenticado.
<b>Reglas de Negocio</b>	- Cada usuario tiene una unica configuración financiera. - Los parametros se expresan como porcentajes o valores absolutos segun el campo.
<b>Criterios de Aceptación</b>	<p><b>Escenario 1 - Creación exitosa: Dado que</b> el estudiante no tiene configuración financiera, <b>cuando</b> envia POST /api/financiamiento/settings con datos validos, <b>entonces</b> el sistema crea la configuración y retorna HTTP 201.</p> <p>Dado que el estudiante no tiene configuración financiera, cuando envia POST /api/financiamiento/settings con datos validos, entonces el sistema crea la configuración y retorna HTTP 201.</p> <p><b>Escenario 2 - Actualización exitosa: Dado que</b> el estudiante</p>

	<p>tiene configuracion existente, <b>cuando</b> envia PATCH /api/financiamiento/settings, <b>entonces</b> el sistema actualiza parcialmente y retorna HTTP 200.</p> <p>Dado que el estudiante tiene configuracion existente, cuando envia PATCH /api/financiamiento/settings, entonces el sistema actualiza parcialmente y retorna HTTP 200.</p>
<b>Documentos Anexos</b>	No aplica
<b>PROTOTIPO</b>	
No aplica (endpoint API REST)	

#### 5.2.4.31 HU031. Consulta del Precio Sugerido de Venta

<b>HU031. Consulta del Precio Sugerido de Venta</b>	
<b>Descripción</b>	Permite al estudiante consultar el precio sugerido de venta, calculado a partir del costo de fabricacion, costos indirectos prorrateados, margen de ganancia e impuestos configurados.
<b>Requerimiento Funcional Asociado</b>	RF-13 Configuración Financiera y Precio Sugerido
<b>Historias de Usuario Asociadas</b>	HU025, HU027, HU029, HU030
<b>Prioridad</b>	Alta
<b>Precondiciones</b>	El usuario debe tener costos directos, costos indirectos y configuración financiera registrados.
<b>Reglas de Negocio</b>	- El calculo integra: costo de fabricacion + costos indirectos fijos prorrateados + costos indirectos variables prorrateados + margen de ganancia + impuestos. - Todos los calculos usan BigDecimal con escala 4 y RoundingMode.HALF_UP.

<b>Criterios de Aceptación</b>	<p><b>Escenario 1 - Calculo exitoso: Dado que</b> el estudiante tiene todos los componentes de costo y la configuracion financiera, <b>cuando</b> envia GET /api/financiamiento/settings/calculated, <b>entonces</b> el sistema retorna el precio sugerido con desglose completo y HTTP 200.</p> <p>Dado que el estudiante tiene todos los componentes de costo y la configuracion financiera, cuando envia GET /api/financiamiento/settings/calculated, entonces el sistema retorna el precio sugerido con desglose completo y HTTP 200.</p>
<b>Documentos Anexos</b>	No aplica
<b>PROTOTIPO</b>	
No aplica (endpoint API REST)	

#### 5.2.4.32 HU032. Gestion de Activos

<b>HU032. Gestion de Activos</b>	
<b>Descripción</b>	Permite al estudiante registrar, consultar, actualizar y eliminar activos necesarios para su emprendimiento (maquinaria, equipo, mobiliario, etc.). Soporta operaciones individuales y en lote. Tambien permite consultar el total calculado de activos.
<b>Requerimiento Funcional Asociado</b>	RF-14 Proyeccion Financiera
<b>Historias de Usuario Asociadas</b>	HU034
<b>Prioridad</b>	Media
<b>Precondiciones</b>	El usuario debe estar autenticado.
<b>Reglas de Negocio</b>	- Cada activo tiene nombre, valor y categoria. - El calculo totaliza el valor de todos los activos del usuario.

<b>Criterios de Aceptación</b>	<p><b>Escenario 1 - CRUD y calculo:</b> Dado que el estudiante esta autenticado, <b>cuando</b> realiza operaciones CRUD en /api/projection/asset o consulta /api/projection/asset/calculated, <b>entonces</b> el sistema ejecuta la operacion correspondiente con el codigo HTTP adecuado.</p> <p>Dado que el estudiante esta autenticado, cuando realiza operaciones CRUD en /api/projection/asset o consulta /api/projection/asset/calculated, entonces el sistema ejecuta la operacion correspondiente con el codigo HTTP adecuado.</p>
<b>Documentos Anexos</b>	No aplica
<b>PROTOTIPO</b>	
No aplica (endpoint API REST)	

#### 5.2.4.33 HU033. Gestion de Deudas/Creditos

<b>HU033. Gestion de Deudas/Creditos</b>	
<b>Descripción</b>	Permite al estudiante registrar, consultar, actualizar y eliminar deudas o creditos proyectados para su emprendimiento. Soporta operaciones individuales y en lote. Tambien permite consultar el total calculado de deudas.
<b>Requerimiento Funcional Asociado</b>	RF-14 Proyeccion Financiera
<b>Historias de Usuario Asociadas</b>	HU034
<b>Prioridad</b>	Media
<b>Precondiciones</b>	El usuario debe estar autenticado.
<b>Reglas de Negocio</b>	<ul style="list-style-type: none"> <li>- Cada deuda tiene monto, plazo y demas parametros financieros.</li> <li>- El calculo totaliza las obligaciones del usuario.</li> </ul>

<b>Criterios de Aceptación</b>	<p><b>Escenario 1 - CRUD y calculo:</b> Dado que el estudiante esta autenticado, <b>cuando</b> realiza operaciones CRUD en /api/projection/debts o consulta /api/projection/debts/calculated, <b>entonces</b> el sistema ejecuta la operacion correspondiente con el codigo HTTP adecuado.</p> <p>Dado que el estudiante esta autenticado, cuando realiza operaciones CRUD en /api/projection/debts o consulta /api/projection/debts/calculated, entonces el sistema ejecuta la operacion correspondiente con el codigo HTTP adecuado.</p>
<b>Documentos Anexos</b>	No aplica
<b>PROTOTIPO</b>	
No aplica (endpoint API REST)	

#### 5.2.4.34 HU034. Consulta de Estructura de Capital

<b>HU034. Consulta de Estructura de Capital</b>	
<b>Descripción</b>	Permite al estudiante consultar la estructura de capital calculada, que combina activos y deudas para mostrar la composicion del financiamiento de su emprendimiento (capital propio vs. financiado).
<b>Requerimiento Funcional Asociado</b>	RF-14 Proyeccion Financiera
<b>Historias de Usuario Asociadas</b>	HU032, HU033
<b>Prioridad</b>	Media
<b>Precondiciones</b>	El usuario debe tener activos y/o deudas registrados.
<b>Reglas de Negocio</b>	- El calculo integra los totales de activos y deudas para determinar la estructura de capital.

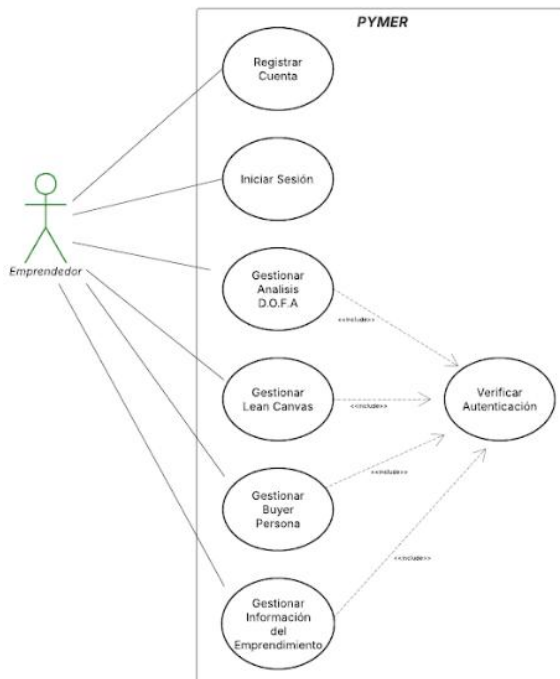
<b>Criterios de Aceptación</b>	<p><b>Escenario 1 - Calculo exitoso:</b> Dado que el estudiante tiene activos y deudas registrados, <b>cuando</b> envía GET /api/projection/calculated, <b>entonces</b> el sistema retorna la estructura de capital con HTTP 200.</p> <p>Dado que el estudiante tiene activos y deudas registrados, cuando envía GET /api/projection/calculated, entonces el sistema retorna la estructura de capital con HTTP 200.</p>
<b>Documentos Anexos</b>	No aplica
<b>PROTOTIPO</b>	
No aplica (endpoint API REST)	

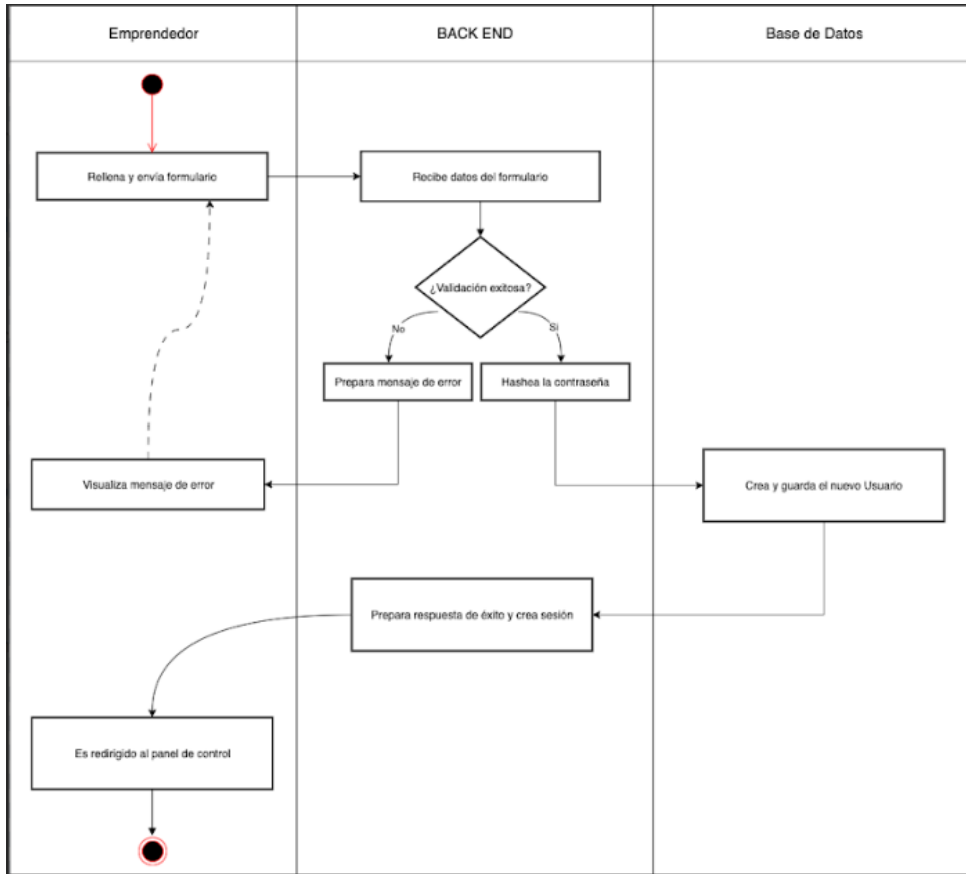
## 5.3 Análisis

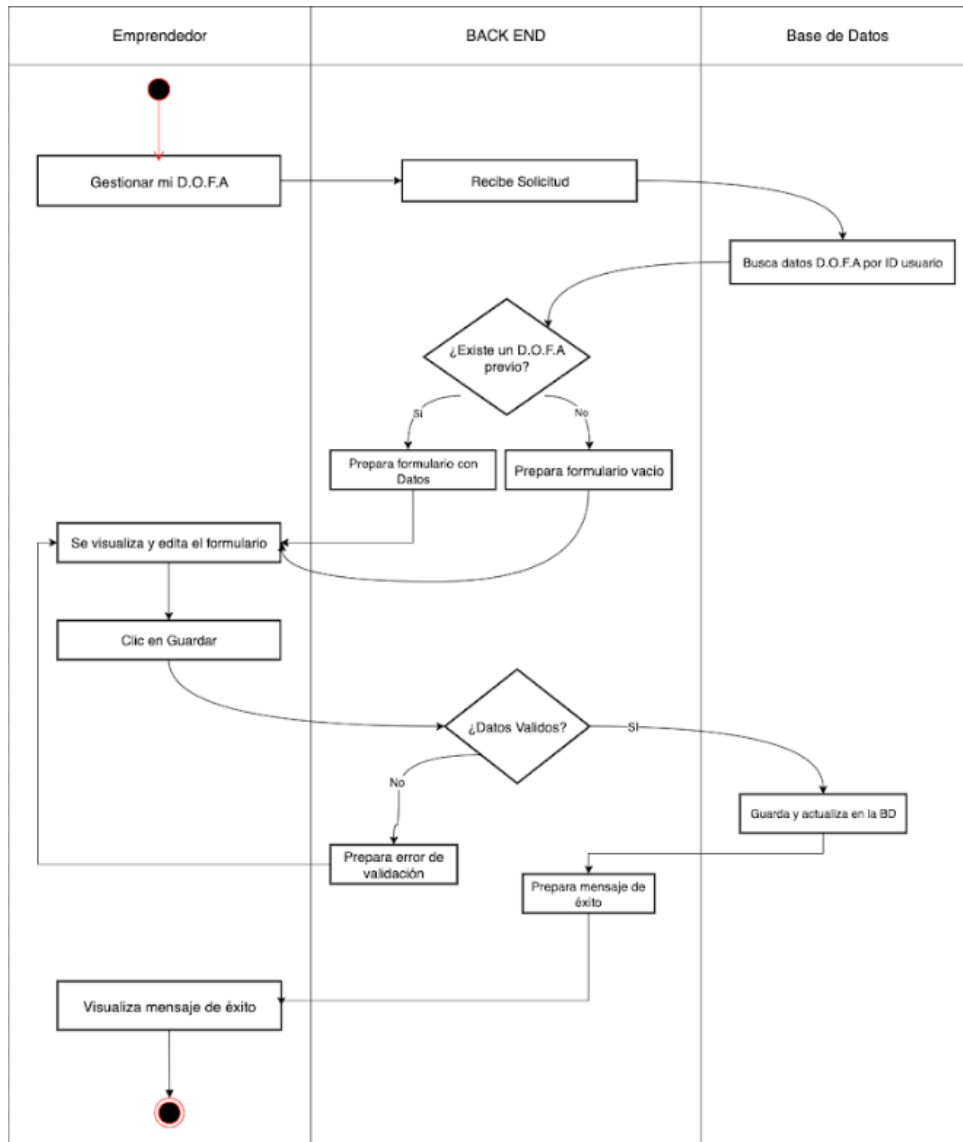
### 5.3.1 Casos de Uso y descripción

Aquí se presenta el diagrama de los casos de uso para mostrar la interacción entre los actores y los casos de uso.

#### 5.3.1.1 Caso de Uso: Autenticación + Modulo de ideación







## 5.4 Modelamiento de datos

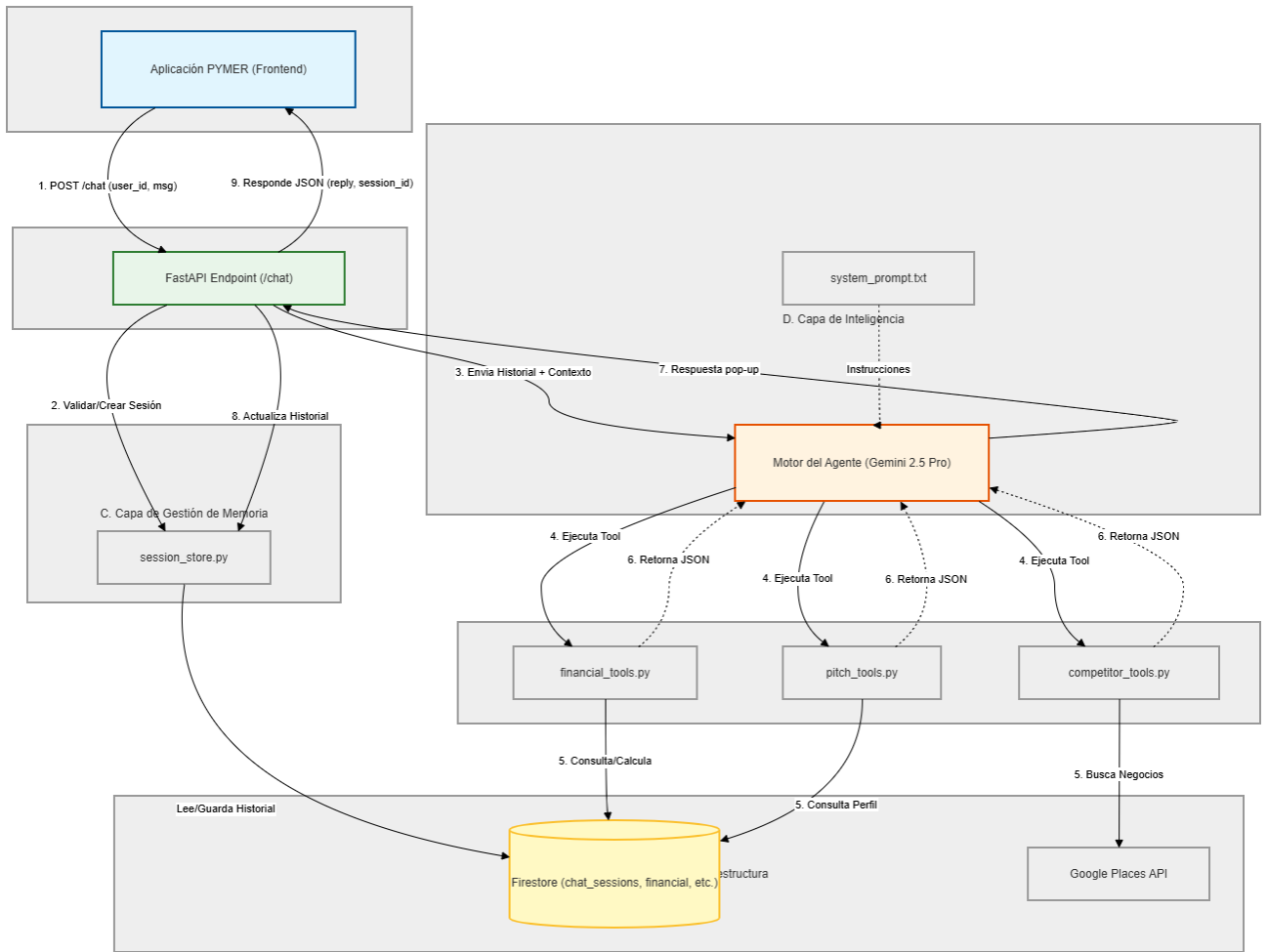
En este ítem se incluye una introducción al modelo y el diagrama. Esta introducción es una explicación, en términos de la empresa, de las entidades o clases y relaciones más representativas del software.

Para el modelamiento se puede utilizar modelos E-R o de clases. Recuerde respetar una codificación para nombrar distintos elementos del modelo.

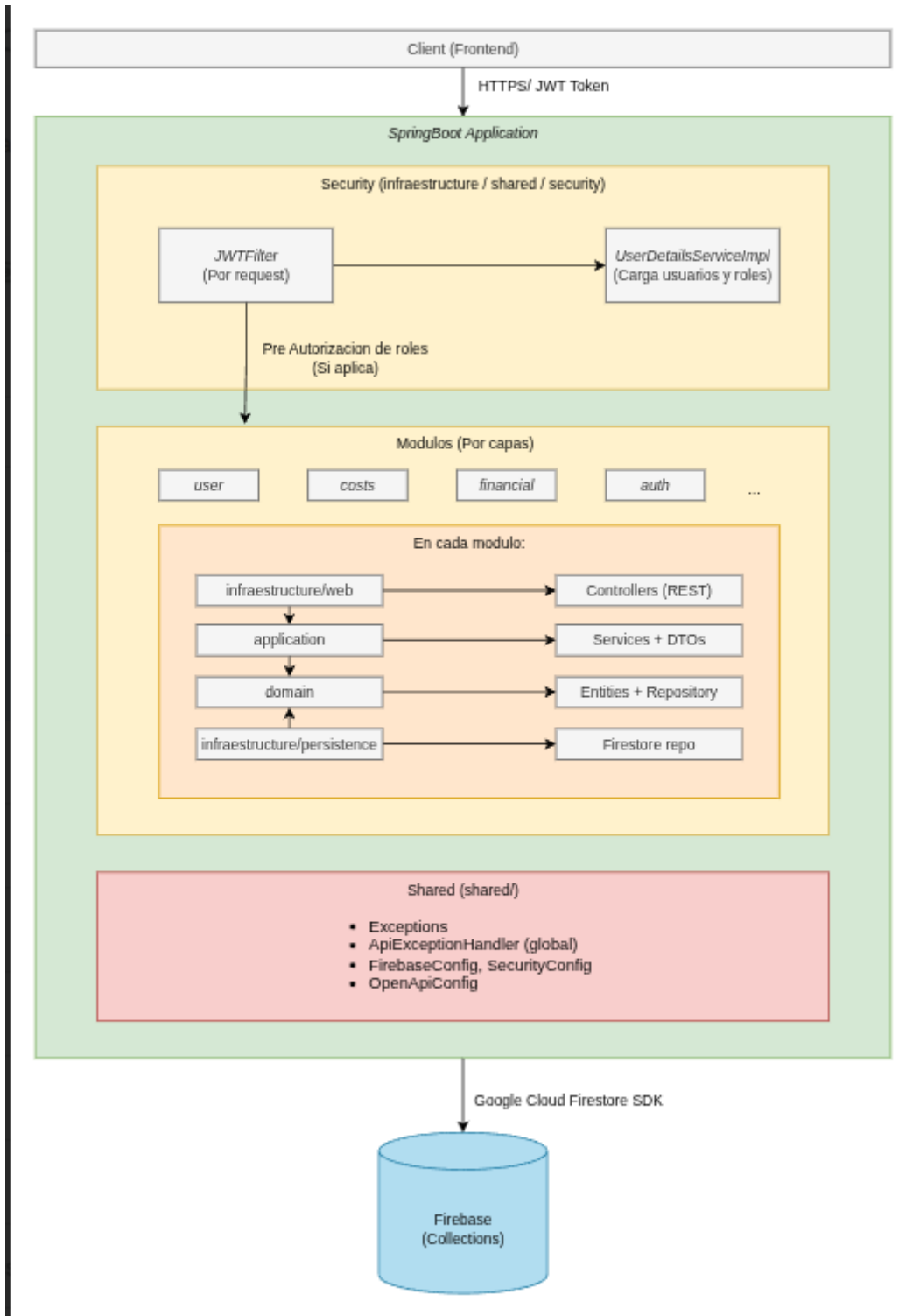
## 6 DISEÑO

### 6.1 Diseño de arquitectura funcional

A continuación, se detalla la arquitectura del Agente que se está desarrollando con ADK y Python



Y también se relaciona la arquitectura del backend

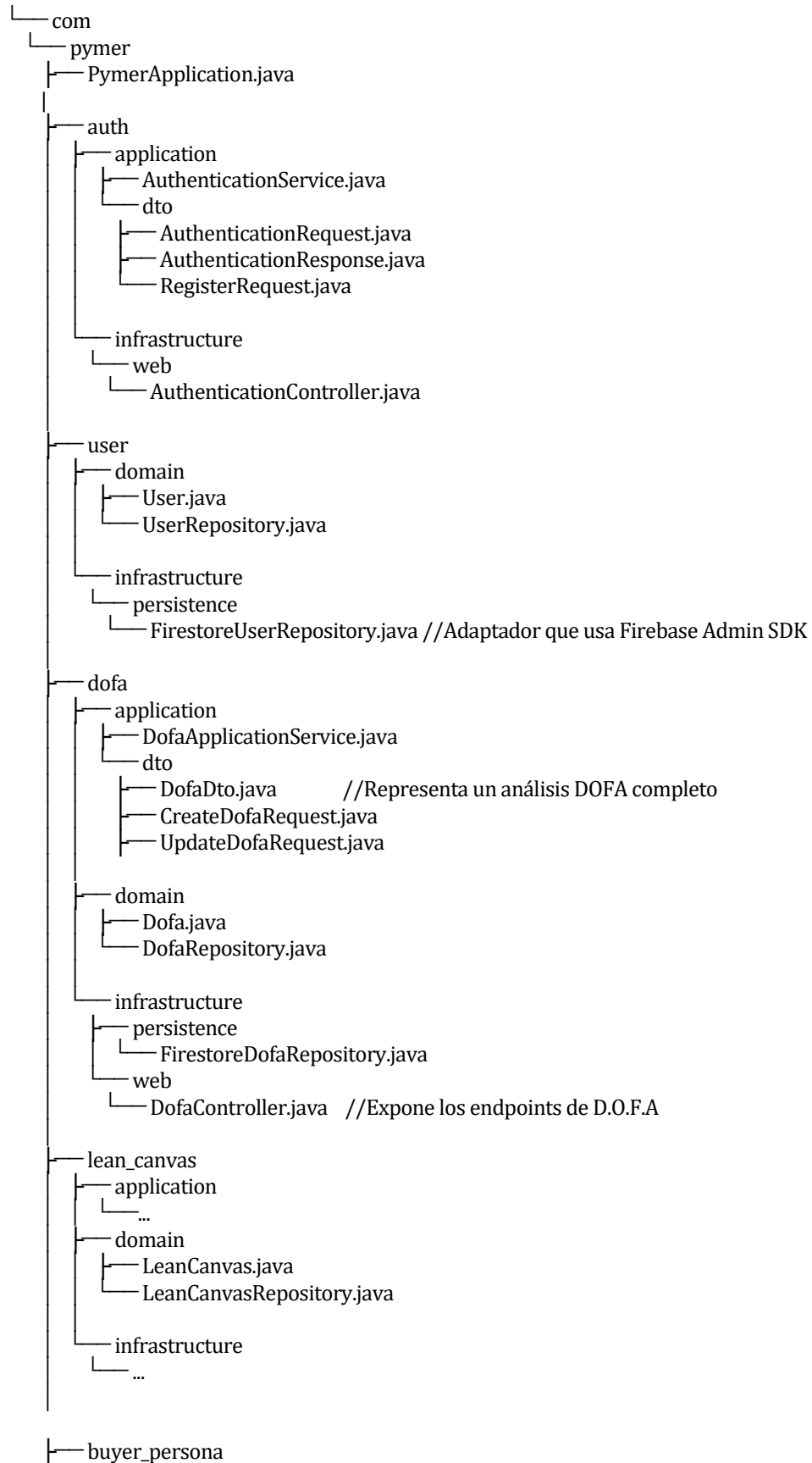


### 6.1.1.1 Estructura de Directorios

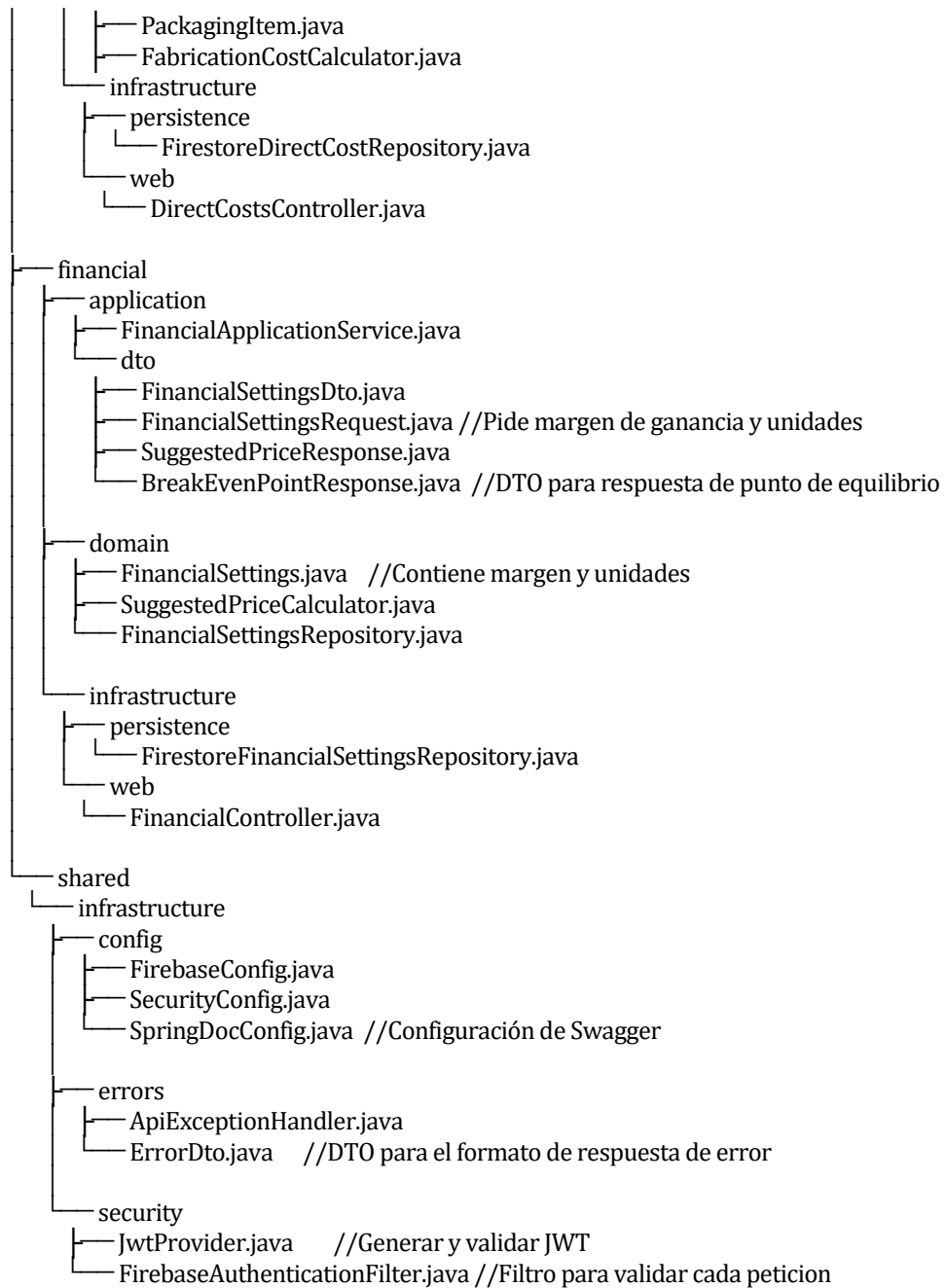
```

src
├── main
└── java

```

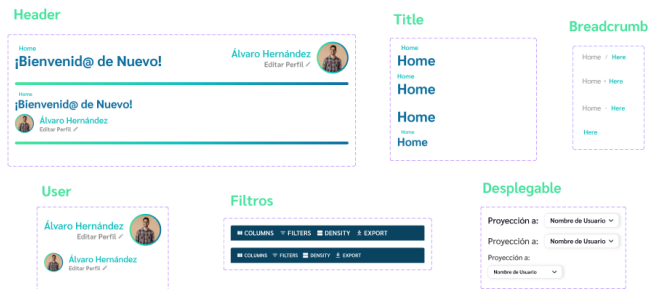






## 6.2 Diseño interfaz y navegación

La integridad de la plataforma no solo reside en su lógica de backend, sino en una arquitectura visual cohesiva y predecible que facilite la curva de aprendizaje del emprendedor. Para ello, se ha definido un estándar de diseño que unifica la estética y la disposición funcional en todas las vistas, asegurando que elementos como la paleta de colores, la iconografía y la jerarquía de navegación respondan a una organización lógica y ergonómica, este diseño estructural garantiza que cada componente, desde los menús hasta las áreas de ingreso de datos, ocupe un lugar estratégico que optimice el flujo de trabajo y la navegación dentro de los módulos de ideación y finanzas.



### Componentes To - Do

#### Usuario

- Usuario + Nombre + Cerrar Sesión

Nombre usuario

#### Dropdown

- Menu dropdown texto

Quitar

- Titulo dropdown

**Movimientos Reales**  
 Ingresos, gastos, pagos y abonos  
 Detalle de gastos  
 Distribución de gastos Top 3 gastos

- Dropdown Fecha

20/07/2024

#### Header

- Header sección

Proveedores

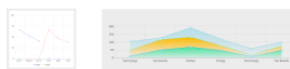
#### Lista

- Lista de categorías

Lista de categorías

#### Grafico

- Grafico de Lineas



#### Tablas

- Gastos y subtotal (same-ish)

Descripción	Cantidad	Moneda
...	...	...
<b>Subtotal</b>	<b>...</b>	<b>...</b>

- Subtotal (same-ish)

...	...	...	...
...	...	...	...
...	...	...	...

- Estatica (same-ish)

...	...	...	...
...	...	...	...
...	...	...	...

...	...	...
...	...	...
...	...	...

...	...	...	...
...	...	...	...
...	...	...	...

#### Formularios

- Crear meta

Establecer objetivo

Crear meta

- Añadir transacción

Transacción N°  Monto transacción:

Añadir

- Añadir crear categoría

Añadir de categoría

Crear categoría

#### Botones

- Continuar

- Guardar

#### Barras

- Estadísticas horizontal



- Estadísticas verticales (diferentes tipos)



<https://www.figma.com/design/C1RQsBcRJPLYlvV4GIUNri/PYME?node-id=57-7036&t=zxyxrT71XLlwZTe9-1> (pagina style-guide)

Adicional se encuentra en la pagina (Branbook) toda la guía de diseño de PYMER

---

## 7 PLAN DE CAPACITACIÓN Y ENTRENAMIENTO

---

En construcción

---

## 8 PLAN DE IMPLANTACIÓN Y PUESTA EN MARCHA

---

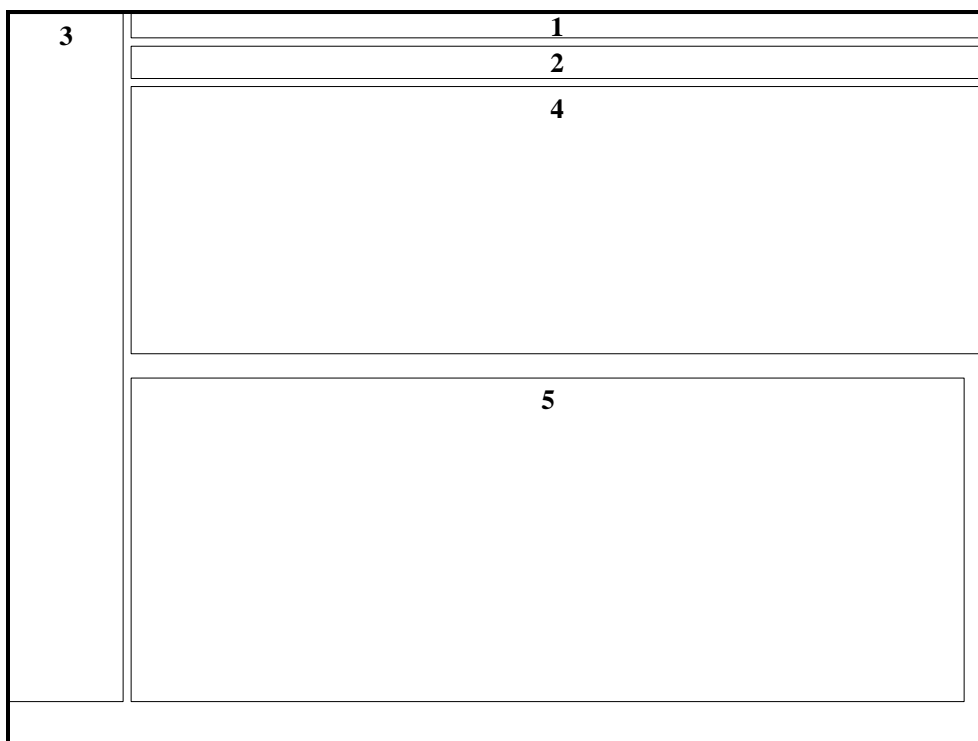
En construcción

---

## 9 ANEXO: PLANIFICACION INICIAL DEL PROYECTO

---

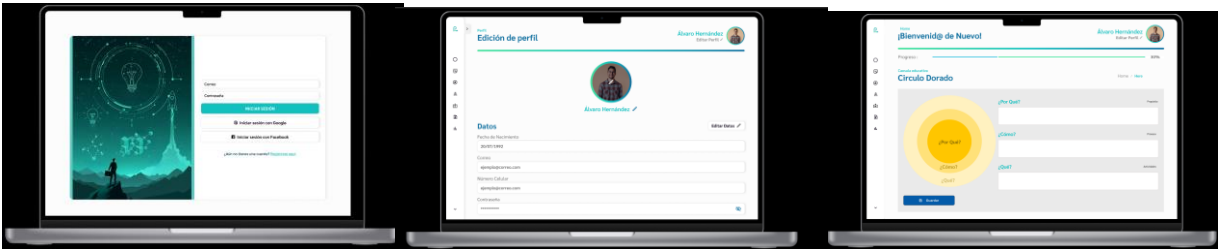
### 9.1 Esquema especificación de Interfaz



- Area 1. Encabezado incluye texto bienvenida + perfil
- Area 2. Barra de progreso
- Area 3. Menu
- Area 4. Capsula educativa
- Area 5. Actividad dinámica del modulo

<https://www.figma.com/design/C1RQSbCRJPLYvV4GIUNri/PYME?node-id=1868-6853&t=zxyxrT71XLlwZTe9-1>

## 9.2 Diagrama flujo menú



<https://www.figma.com/proto/C1RQSbCRJPLYvV4GIUNri/PYME?node-id=229-7077&t=xBcfARo0Tk4g2OM-1&scaling=scale-down&content-scaling=fixed&page-id=160%3A12382&starting-point-node-id=229%3A7077&show-prot-sidebar=1>