

Proyecto V&V

CRM para agencia de viajes



Grupo 5

Chiluiza Anthony

Guachamin Emilia

Ludeña Henry

Nagua Fernando

Bryan Salazar

Ximenez Joel

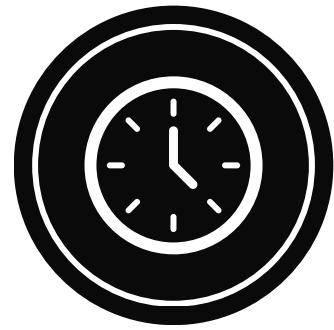
El problema



Gestión
manual de
solicitudes y
documentos.



Falta de
trazabilidad del
estado de los
trámites.



Retrasos en
fechas críticas
y citas.



Alta carga
operativa para
los asesores.

Nuestra solución



Desarrollo de un CRM web interno para asesores y supervisores de la agencia de viajes que permite:

- Centralizar información migratoria.
- Controlar solicitudes, documentos y citas.
- Gestionar tareas y alertas automáticas.
- Generar reportes administrativos.



Nuestra solución



SGPM
Gestión Migratoria

Solicitudes
Gestión de solicitudes

Gestión de Solicitud
Administra y consulta las solicitudes migratorias

Gestión de Solicitud

- Registrar Nueva Solicitud: Crear una nueva solicitud migratoria para un solicitante
- Listado de Solicitud: Ver todas las solicitudes registradas en el sistema
- Detalle de Solicitud: Consultar información completa de una solicitud
- Cambio de Estado: Actualizar el estado de las solicitudes
- Registro de Fechas Clave: Gestionar fechas importantes de las solicitudes
- Documentos de Solicitud: Subir y gestionar documentos PDF asociados a solicitudes

Estadísticas Rápidas

TOTAL SOLICITUDES	APROBADAS	EN PROCESO	PENDIENTES HOY
248 <small>12% más que el mes anterior</small>	186 <small>8% más que el mes anterior</small>	45 <small>— Sin cambios</small>	17 <small>↓ 3 menos que ayer</small>

Activar Windows
Ve a Configuración para activar Windows.

anderson gonzal...
anderson@gmail.com

Gestión del trabajo

* Reunión en "General"

lunes, 12 enero 2026 21:01 - 22:00 ▾ Más ↓ Descarga

Asistencia Participación

6 Asistieron	21:01 - 22:00 Hora de inicio y finalización
59m 31s Duración de la reunión	48m 51s Tiempo medio de asistencia

Participantes

Nombre	Primera reunión	Última salida	Duración de la reunión	Rol	Compromiso
 JOEL DAVID XIMENEZ CALDERON joel.ximenez@epn.edu.ec	21:01	22:00	59m 17s	Organizador	0 - 0 - 0 - 🤝 - ❤️ - 🌟 - 😊 - 😊 - 🙌 -
 HENRY FRANCISCO LUDENA CARRASCO henry.ludena@epn.edu.ec	21:04	22:00	55m 56s	Moderador	0 - 0 - 0 - 🤝 - ❤️ - 🌟 - 😊 - 😊 - 🙌 -
 FERNANDO ALEXANDER NAGUA fernando.nagua@epn.edu.ec	21:05	22:00	55m 16s	Moderador	0 - 0 - 0 - 🤝 - ❤️ - 🌟 - 😊 - 😊 - 🙌 -
 BRYAN DAVID SALAZAR MORO bryan.salazar02@epn.edu.ec	21:10	22:00	50m 2s	Moderador	0 - 0 - 0 - 🤝 - ❤️ - 🌟 - 😊 - 😊 - 🙌 -
 EMILIA MAITE GUACHAMIN GUTIERREZ emilia.guachamin@epn.edu.ec	21:19	22:00	41m 11s	Moderador	0 - 0 - 0 - 🤝 - ❤️ - 🌟 - 😊 - 😊 - 🙌 -
 ANTHONY FRANCISCO CHILUIZA anthony.chiluiza@epn.edu.ec	21:29	22:00	31m 26s	Moderador	0 - 0 - 0 - 🤝 - ❤️ - 🌟 - 😊 - 😊 - 🙌 -

Pair Programming

Controlar Chat Gente Participar Reaccionar Vista Más Cámara Micro Compartir Salir

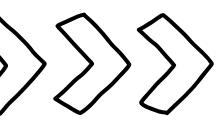
tests/features current file > reservation_citas_solicitantes.feature reservation_citas_solicitantes.py DiagramaDeClases.puml

```
1 import behave.runner
2 from behave import *
3
4 use_step_matcher("re")
5
6
7 @step('que el asesor "Juan Pérez" está autenticado') new
8 def step_impl(context):
9     raise NotImplementedError(u"STEP: Dado que el asesor \"Juan Pérez\" está autenticado")
10
11
12 @step('existe un solicitante "María López" con cédula "ABC123"') new
13 def step_impl(context):
14     raise NotImplementedError(u"STEP: Y existe un solicitante \"María López\" con cédula \"ABC123\"")
15
16
17 @step('existe una solicitud migratoria con código "SOL-001" del solicitante "ABC123" gestionada por el asesor "Juan Pérez"')
18 def step_impl(context):
19     raise NotImplementedError()
20     u"STEP: Y existe una solicitud migratoria con código \"SOL-001\" del solicitante \"ABC123\" gestionada por el asesor
21
22
23
24 @step('el asesor agenda una cita de tipo "ASESORIA" para la solicitud "SOL-001" desde "2026-04-19 10:00" hasta "2026-04-19
25 def step_impl(context):
26     raise NotImplementedError()
27     u"STEP: Cuando el asesor agenda una cita de tipo \"ASESORIA\" para la solicitud \"SOL-001\" desde \"2026-04-19 10:00\""
28
29
30
31 @step('la cita queda en estado "PROGRAMADA"') new
32 def step_impl(context):
```

JOEL DAVID XIMENEZ CALDERON

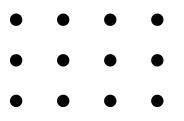
EG

8:22 CRLF UTF-8 4 spaces Python 3.13 (Sistema de Gestión del Proceso Migratorio)



BDD

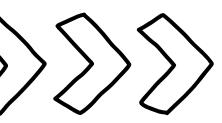




Visión



Ser una plataforma ágil en el flujo administrativo de los procesos de migración.



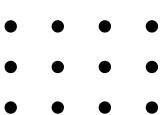
BDD

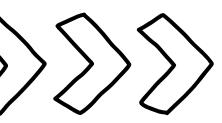


Objetivos comerciales



- Centralizar el 100% de los documentos y el historial de solicitudes en el repositorio digital durante el primer trimestre tras la implementación, eliminando el uso de hojas de cálculo para erradicar errores manuales y agilizar la consulta de expedientes.
- Establecer un proceso automatizado de alertas y administración de tareas que, en los primeros 3 meses, cubra el 100% de las citas, reduzca las inasistencias y disminuya la carga administrativa del equipo, para incrementar el tiempo destinado a la atención personalizada.
-





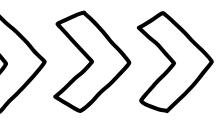
BDD



.....

Capacidades

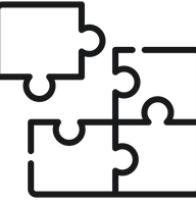
- Tramitación de solicitudes de visa
- Administración de citas solicitantes
- Coordinación de tareas a los asesores



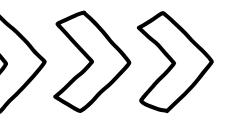
BDD



Características



- *Tramitación de solicitudes de visa*
 - Recepción de documentos.
 - Manejo de los datos de los solicitantes.
 - Control de estado y fechas respectivas de las solicitudes.
- *Administración de citas solicitantes*
 - Reservación de citas con solicitantes.
- *Coordinación de tareas a los asesores*
 - Asignación de tareas.
 - Generación de reporte de estadísticas de tareas.



BDD



Criterios de aceptación y ejemplos



⋮⋮⋮

Asignar una tarea a un asesor

- El supervisor puede asignar una tarea a un asesor.
- Tras la asignación, queda claro quién es el responsable.
- La tarea queda marcada como **pendiente** hasta que el asesor la actualice.

⋮⋮⋮

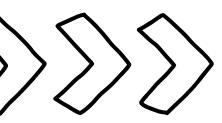
Ejemplo

- El supervisor asigna "Revisar documentación" a un asesor → el asesor queda como **responsable** y **pendiente**.

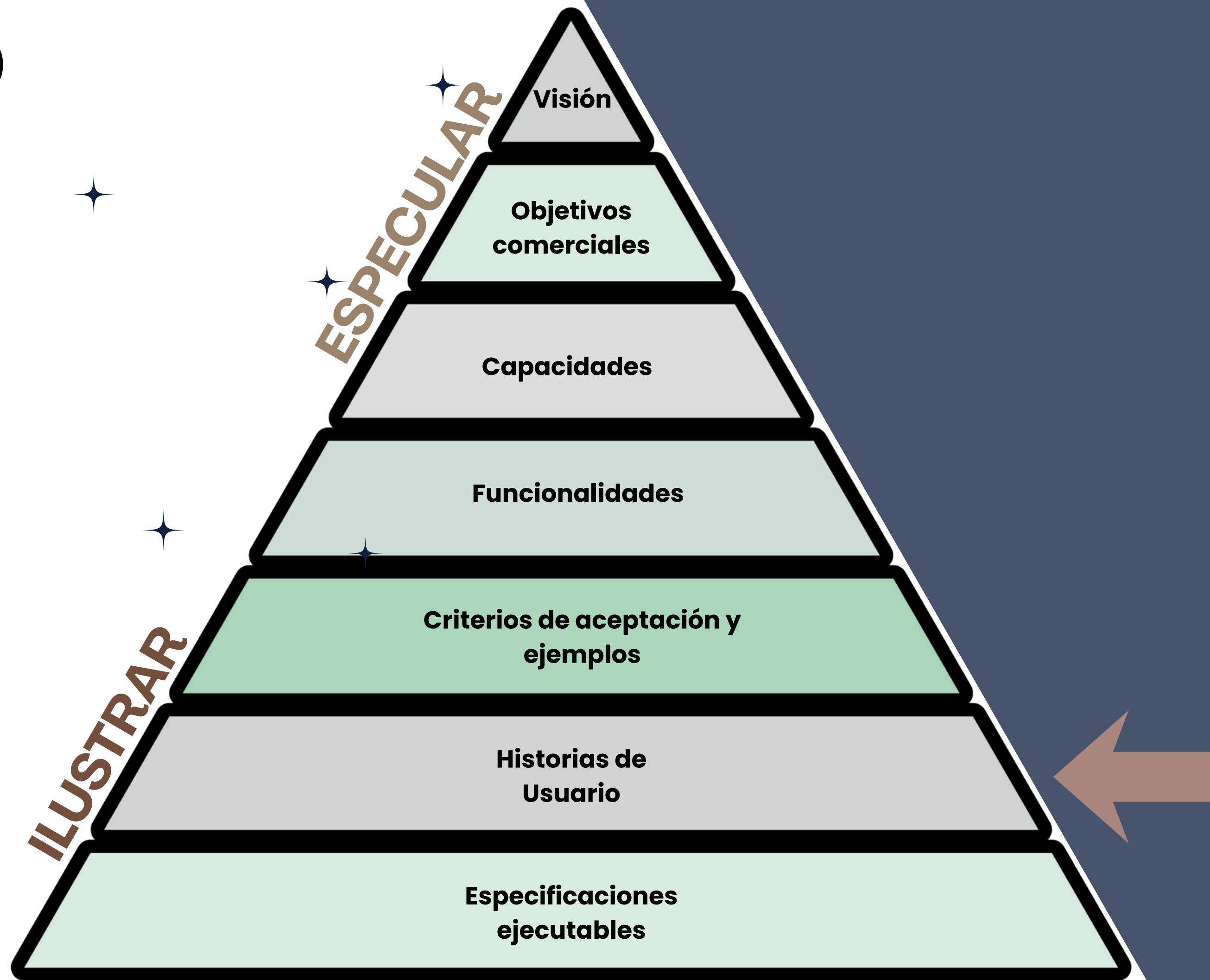
Asignar varias tareas al mismo asesor

- El supervisor puede asignar más de una tarea al mismo asesor.
- Al asignar una nueva tarea, las tareas anteriores del asesor se mantienen.

Ejemplo



BDD



Historias de Usuario



HU — Asignación de tareas a asesores

Como supervisor

Quiero asignar tareas a los asesores

Para asegurar el seguimiento del trabajo y el cumplimiento de plazos

Criterios de aceptación

1. Asignación básica

- **Criterio:** Al asignar una tarea, queda definido un responsable y la tarea queda **pendiente**.
- **Ejemplo:** El supervisor asigna “Revisar expediente” a un asesor → responsable definido; estado **pendiente**.

2. Múltiples tareas por asesor

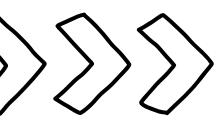
- **Criterio:** Al asignar una nueva tarea a un asesor, no se eliminan sus tareas anteriores.
- **Ejemplo:** Un asesor ya tiene dos tareas. Se asigna una tercera → se conservan las anteriores.

3. Asignación con vencimiento

- **Criterio:** Si se define una fecha/hora límite, esta queda registrada y asociada a la tarea.
- **Ejemplo:** “Entregar informe” con vencimiento “20 de enero, 17:00” → la tarea muestra ese vencimiento.

4. Edición por supervisor

- **Criterio:** El supervisor puede cambiar atributos como la **prioridad** (y/o vencimiento) y el cambio queda reflejado.
- **Ejemplo:** El supervisor cambia la prioridad de “Preparar reporte” a “alta” → se refleja como “alta”.



BDD



Especificaciones ejecutables

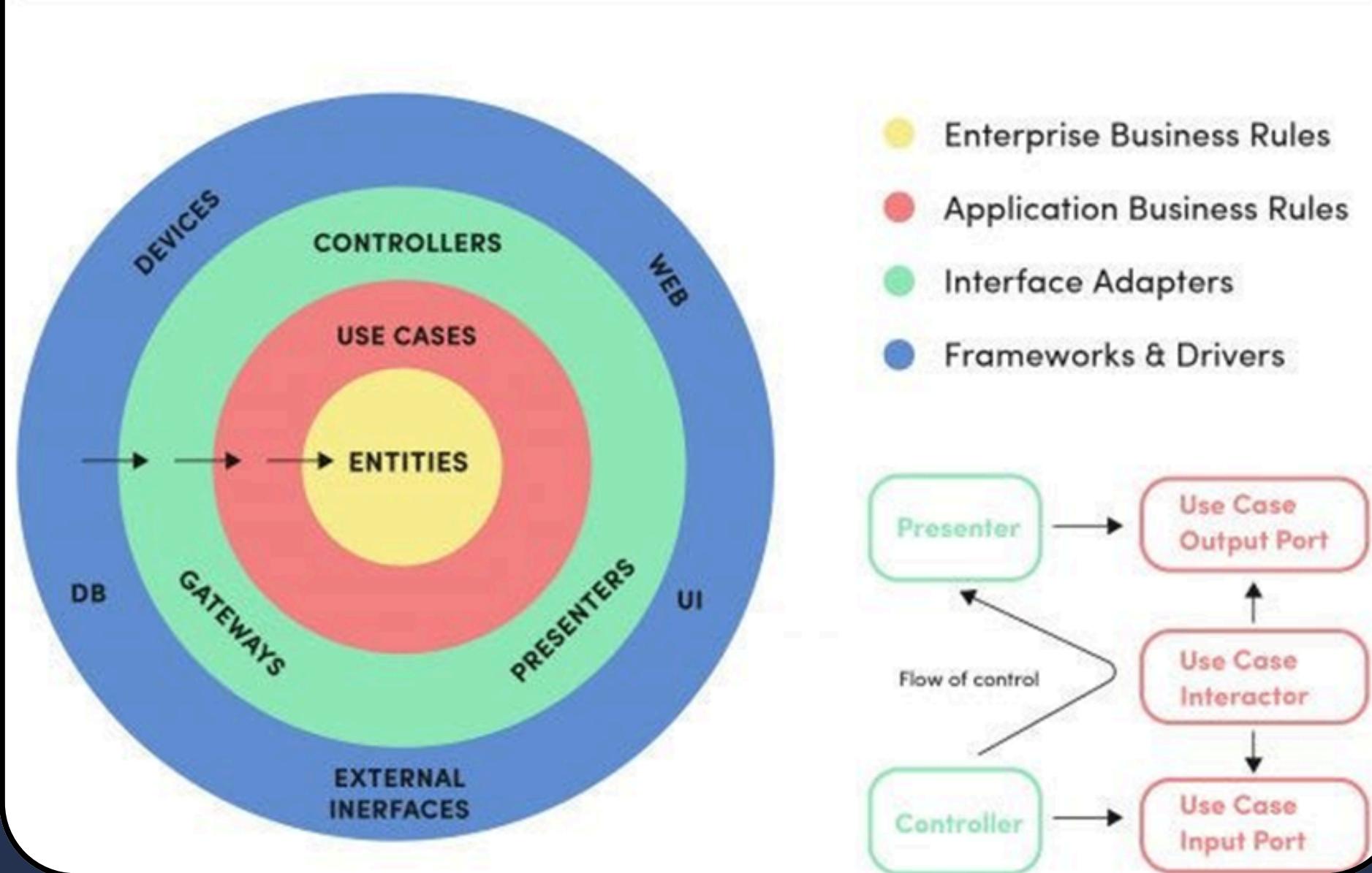


```
asignacion_tareas.feature  manej_datos_solicitantes.feature ×  generacion_reporte_estadisticas_tareas.feature
```

33 Entonces los cambios se guardan correctamente
34 Y la ficha del solicitante muestra los datos actualizados
35
36 @validacion
37 ►► Esquema del escenario: Evitar registrar un solicitante si falta un dato obligatorio
38 Dado que no existe un solicitante con cédula "<cedula>"
39 Cuando intento registrar un solicitante con los siguientes datos:
40 | cedula | nombre | apellido | correo | fecha_nacimiento | telefono
41 | <cedula> | <nombre> | <apellido> | <correo> | <fecha> | <telefono>
42 Entonces se muestra un mensaje de error por datos obligatorios
43 Y el solicitante con cédula "<cedula>" no se registra
44
45 Ejemplos:
46 | cedula | nombre | apellido | correo | fecha | telefono
47 | 0102030406 | | Pérez | sin.nombre@mail.com | 2000-01-10 | 09123456
48
49 @regla_negocio
50 ►► Escenario: Evitar registrar un solicitante con cédula ya registrada
51 Dado que existe un solicitante con los siguientes datos:
52 | cedula | nombre | apellido | correo | fecha_nacimiento | telefono
53 | 0102030405 | Ana | Torres | ana.torres@mail.com | 1998-05-12 | 09123456
54 Cuando intento registrar un solicitante con los siguientes datos:
55 | cedula | nombre | apellido | correo | fecha_nacimiento | telefono
56 | 0102030405 | Carlos | Vera | carlos.vera@mail.com | 1995-08-20 | 09123456
57 Entonces se muestra un mensaje indicando que la cédula ya está registrada
58 Y no se crea un nuevo solicitante

Arquitectura

The Clean Architecture



Project

```
<pre>>> SistemadeGestióndelProcesoMigratorio C:\User
> .venv
> docs
> SGPM
> application
> domain
> infrastructure
> management
> migrations
> presentation
> __init__.py
> admin.py
> apps.py
> models.py
> tests.py
> SistemadeGestióndelProcesoMigratorio
> static
> templates
> tests
> features
> .gitignore
> db.sqlite3
> manage.py
> External Libraries
> Scratches and Consoles</pre>
```

Verificación y Validación

Verificación

- Revisión de requisitos y código.
- Inspecciones y code reviews.

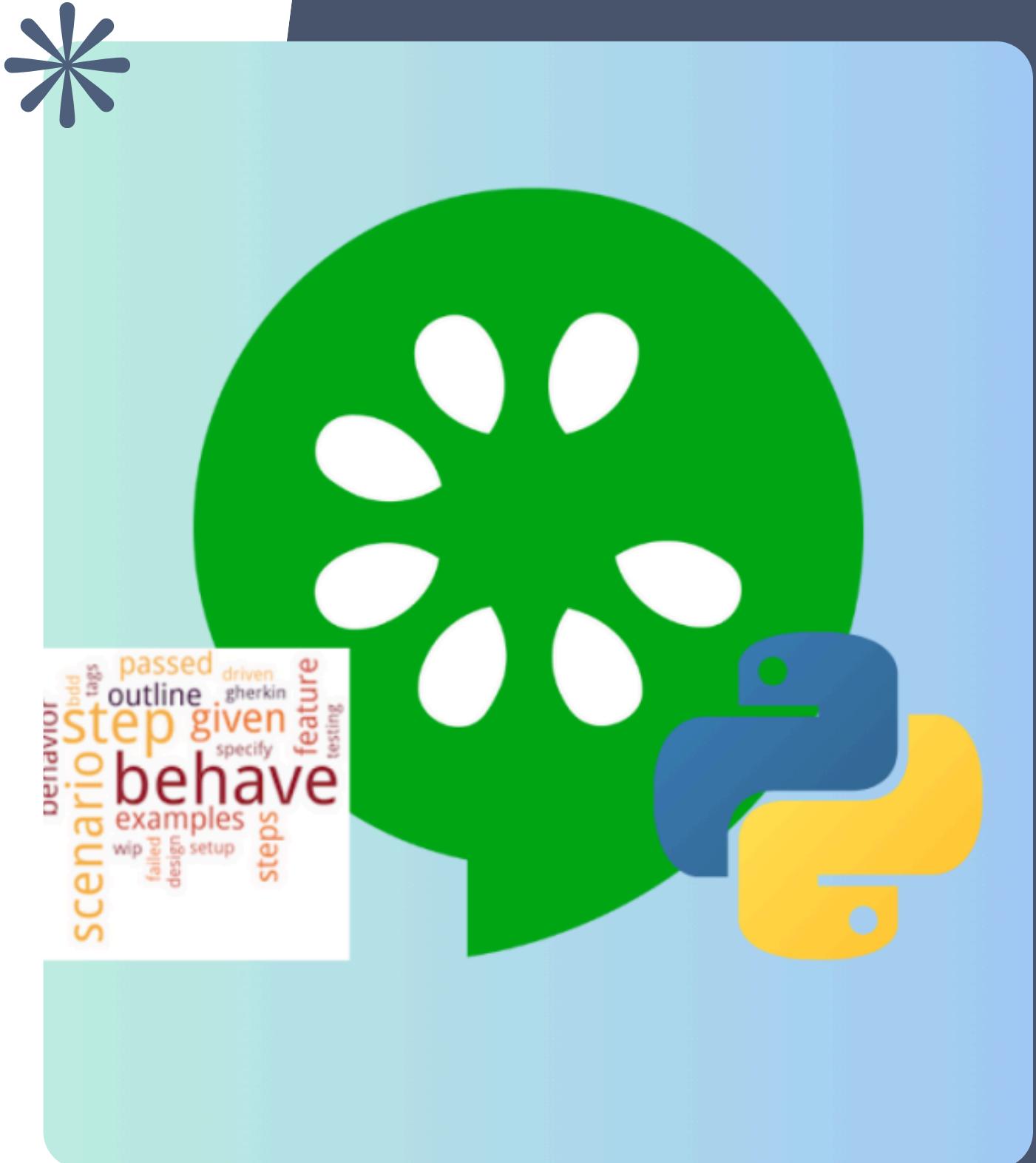
Validación

- Escenarios BDD ejecutables.

⋮⋮⋮

Tecnologías utilizadas

- Python – Lógica de negocio.
- Django – Framework web.
- Gherkin / Cucumber – Especificación BDD.
- Behave – Pruebas funcionales.
- Bootstrap + HTML5 – Interfaz web.





Estimaciones

- **Alcance:** CRM interno (asesores/supervisores): solicitantes, solicitudes, documentos, citas, tareas y reportes.
- **Tiempo:** 26/11/2025–28/01/2026 → 40 días laborables efectivos.
- **Costo:**
 - Licencias: PyCharm Pro 6 usuarios = \$ 1.794/año
 - Esfuerzo humano: 6 personas × 40 días × 2 h/día = 480 h →
 $(480 \times \$35/h) = \16.800
 - Contingencia (15%): \$2.789
 - Total estimado: \$21.383



GRACIAS