



Carrera Analista Programador Computacional

Semana 5

Portafolio de Título (PTY4479)

Nombre estudiante:	
<ul style="list-style-type: none">• Gunther Hemmelmann• Erika Muñoz• Ignacio Manríquez	
Asignatura: Portafolio de Título	Carrera: Analista Programador Computacional
Profesor: Patricio Oliva	Fecha: 14/09/2025

Descripción de la actividad

En esta quinta semana se realizará la actividad formativa 4 de manera grupal, a través de un encargo, donde deberán entregar las épicas, historias de usuario, impact mapping, product backlog y la story mapping. Considerando los aportes de la semana anterior.

Instrucciones específicas

A continuación, se detalla la documentación base de tu proyecto, deberás entregar los formatos de respuesta y los archivos de las imágenes solicitadas.

Sigue las siguientes recomendaciones para cada uno.

1. **Épicas:** Detalla de manera clara las épicas y describir las funcionalidades o características del software con una mirada de alto nivel. Su desarrollo debe ser de cara al usuario, por lo que la sintaxis debe ser basada en el usuario. Las épicas son útiles para organizar y planificar el trabajo a largo plazo por lo que su descripción debe permitir que la épica luego pueda dividirse en historias de usuario.
 - Entrega: Deberás responder en este formato de respuesta.
2. **Historias de usuario:** Elabora las historias de usuario provienen de las épicas (por lo que deben tener relación entre ellas) y hacen referencia a una característica o funcionalidad específica del software, siempre desde la perspectiva del usuario. Debe poseer tres elementos claves: El usuario, la acción que realiza y el objetivo que persigue el usuario con esa acción. Además, a cada historia de usuario se le deben agregar los criterios de aceptación para conocer el cumplimiento de ella una vez creada la historia.
 - Entrega: Deberás responder formato de respuesta de Word.
3. **Impact mapping:** Realiza el esquema impact Mapping visualiza de manera clara los objetivos del proyecto, los actores involucrados, los resultados esperados y el impacto general. Proporciona respuestas a las preguntas clave de "qué", "cómo" y "quién" en el contexto del desarrollo del proyecto TI.

- Entrega: Archivo de imagen, usando herramientas web de desarrollo de esquemas (Canva, Cacoo, entre otros)
4. **Product Backlog:** Elabora una lista priorizada de todas las funcionalidades que se planean implementar en el sistema a desarrollar. El product Backlog sirve para gestionar el trabajo del equipo de desarrollo por lo que su enfoque es hacia ese equipo.
- Entrega: Deberás responder en el formato de respuesta Excel.
5. **Story mapping:** Realiza un esquema que muestra cómo serán distribuidas las historias de usuarios en los Sprint que se trabajarán durante el proyecto. Los Sprint que se ejecutarán, se basan en lo proyectado en este esquema. Se debe revisar que las historias de usuario que se desarrolle en un sprint específico se puedan lograr dentro de una semana de desarrollo.
- Entrega: Archivo de imagen, usando herramientas web de desarrollo de esquemas (Canva, Cacoo, entre otros)
6. **Avance del desarrollo del sistema:** De acuerdo con la planificación de la Fase I, en base a las evidencias y el plan de trabajo, entrega los artefactos relacionados con el desarrollo del sistema.
- Entrega: Un Zip que contenga todos los archivos del desarrollo.

Importante

Tanto los dos formatos de respuesta (Excel y Word), las imágenes y el Zip (avances del desarrollo), deberás guardarlos en un Zip (entregable de la semana) y adjuntarlo en la plataforma, tal como se indicó en las Introducción de la semana.

1. Épicas:

Responde en este espacio y observa el ejemplo que podrás borrar después de leer

Épica 1: Gestión de Usuarios y Autenticación

Descripción: Esta épica abarca el registro e inicio de sesión de los usuarios, ademas de la funcionalidad para editar el perfil personal. También, incluye la gestión de permisos por parte del administrador, el cual podrá asignar los roles: paciente, funcionario y administrador.

Épica 2: Reserva y Gestión de Citas Médicas

Descripción: Esta épica engloba las funcionalidades de consulta, reserva y visualización de horas medicas, junto con la confirmación por correo de las reservas solicitadas.

Épica 3: Módulo Administrativo

Descripción: Esta épica contiene la función que corresponde al modulo administrativo, en donde los funcionarios podrán ver y gestionar la agenda de citas medicas.

Épica 4: Cancelación y Reprogramación de citas

Descripción: En esta épica están las funciones para cancelar una cita, dejando el cupo disponible y, también, esta la función para reagendar la cita en una fecha posterior.

Épica 5: Reportes y Estadísticas

Descripción: Como personal administrativo, quiero generar reportes sobre citas, pacientes y rendimiento para obtener información valiosa que ayude a la toma de decisiones y a la mejora del servicio.

2. Historia de usuario:

Historia de usuario 1: Como paciente, quiero registrarme en el sistema ingresando mis datos personales para crear un perfil.

Épica Asociada: Gestión de usuarios y seguridad

Puntos de Historia: 5

Criterios de aceptación:

- a. El usuario debe poder acceder a un formulario de registro desde la página principal.
- b. El formulario debe solicitar campos obligatorios: nombre, apellido, RUT, correo electrónico y contraseña.
- c. El sistema debe validar en el backend que el correo electrónico no esté registrado previamente y mostrar un error específico si lo está.
- d. La contraseña debe cumplir con requisitos mínimos de seguridad (ej. largo, caracteres especiales), validados tanto en el frontend como en el backend.
- e. Al enviar el formulario con datos válidos, se creará una nueva cuenta en la base de datos y se redirigirá al usuario al login con un mensaje de éxito.

Historia de usuario 2: Como paciente, quiero iniciar sesión con RUT o correo, y contraseña para acceder de forma segura a mi perfil.

Épica Asociada: Gestión de usuarios y seguridad

Puntos de Historia: 5

Criterios de aceptación:

- a. La página de inicio debe contener un formulario de login (correo/RUT y contraseña).
- b. El sistema debe verificar en el backend que las credenciales coincidan con un usuario registrado.
- c. Tras una autenticación exitosa, el sistema debe generar un token de sesión y redirigir al usuario a su panel personal.
- d. Si las credenciales son incorrectas, se debe mostrar un mensaje de error claro desde el backend.
- e. La sesión del usuario debe mantenerse activa mientras navega por la aplicación.

Historia de usuario 3: Como administrador, quiero asignar roles (paciente, funcionario, administrador) a un usuario para gestionar sus permisos.

Épica Asociada: Gestión de usuarios y seguridad

Puntos de Historia: 3

Criterios de aceptación:

- a. El administrador debe tener acceso a una sección "Gestión de Usuarios" en su panel.
- b. La interfaz debe obtener y mostrar una lista de todos los usuarios y sus roles actuales.
- c. El administrador debe poder modificar el rol de un usuario, y el cambio debe enviarse al backend para ser persistido en la base de datos.
- d. El cambio de rol debe ser efectivo inmediatamente y afectar los permisos del usuario.

Historia de usuario 4: Como paciente, quiero ver y editar la información de mi perfil para mantener mis datos actualizados.

Épica Asociada: Gestión de usuarios y seguridad

Puntos de Historia: 3

Criterios de aceptación:

- a. El paciente debe tener una sección "Mi Perfil" que cargue sus datos desde el backend.
- b. La página debe mostrar los datos personales del usuario (nombre, apellido, correo) en un formulario editable.
- c. Al guardar, los nuevos datos deben ser enviados al backend y actualizados en la base de datos.
- d. Se debe mostrar un mensaje de confirmación en la interfaz tras una actualización exitosa.

Historia de usuario 5: Como paciente, quiero visualizar la disponibilidad de horas médicas por especialidad para elegir la que más me convenga.

Épica Asociada: Reserva y gestión de citas médicas

Puntos de Historia: 8

Criterios de aceptación:

- a. La interfaz debe permitir al paciente filtrar por especialidad médica.
- b. El frontend debe solicitar al backend la disponibilidad de horarios según el filtro seleccionado.

- c. El sistema debe mostrar un calendario con los días y horarios disponibles.
- d. Los horarios ya ocupados (obtenidos del backend) no deben ser seleccionables en la interfaz.
- e. El usuario debe poder navegar entre diferentes semanas o meses para consultar disponibilidad futura.

Historia de usuario 6: Como paciente, quiero reservar una cita confirmando día y hora para asegurar mi atención.

Épica Asociada: Reserva y gestión de citas médicas

Puntos de Historia: 5

Criterios de aceptación:

- a. Al seleccionar un horario disponible, la interfaz debe mostrar un resumen de la cita para confirmación.
- b. Al confirmar, el frontend debe enviar una petición al backend para crear la cita en la base de datos.
- c. Una vez que el backend confirma la creación, el horario debe marcarse como "no disponible" para otros pacientes.

Historia de usuario 7: Como sistema, quiero enviar una confirmación visual en pantalla y una notificación por correo al paciente tras una reserva exitosa.

Épica Asociada: Reserva y gestión de citas médicas

Puntos de Historia: 3

Criterios de aceptación:

- a. Inmediatamente después de que el backend confirme la reserva, el frontend debe mostrar un mensaje de éxito.
- b. El proceso de reserva en el backend debe gatillar el envío de un correo electrónico de confirmación al paciente.

Historia de usuario 8: Como paciente, quiero visualizar mis próximas citas y el historial de citas pasadas para llevar un control de mi atención.

Épica Asociada: Reserva y gestión de citas médicas

Puntos de Historia: 3

Criterios de aceptación:

- a. El paciente debe tener una sección "Mis Citas" que solicita los datos al backend.
- b. El frontend debe organizar la información recibida en dos secciones: "Próximas Citas" e "Historial de Citas".
- c. Ambas secciones deben mostrar los detalles relevantes de cada cita (médico, especialidad, fecha, hora).

Historia de usuario 9: Como funcionario, quiero cargar los horarios y especialidades de los médicos para gestionar citas en el sistema.

Épica Asociada: Administración de agenda médica

Puntos de Historia: 8

Criterios de aceptación:

- a. El funcionario debe tener una interfaz para "Gestionar Agendas".
- b. La interfaz debe permitir seleccionar un médico y definir sus horarios de trabajo, guardando esta configuración en la base de datos.
- c. Debe poder asignar especialidades a cada médico.
- d. Debe existir una funcionalidad para bloquear fechas o rangos de horarios específicos, lo que se reflejará en la disponibilidad.
- e. Los cambios realizados deben impactar en tiempo real los datos que el backend sirve a los pacientes.

Historia de usuario 10: Como paciente, quiero cancelar una cita agendada para liberar el espacio si no puedo asistir.

Épica Asociada: Cancelación y reprogramación de citas

Puntos de Historia: 5

Criterios de aceptación:

- a. En la sección "Próximas Citas", cada cita debe tener un botón "Cancelar" que llame al endpoint de cancelación del backend.
- b. La interfaz debe pedir una confirmación antes de enviar la petición de borrado.
- c. Al confirmar, el backend debe eliminar la cita y liberar el horario, lo cual debe reflejarse en la base de datos.
- d. El paciente debe recibir una notificación de la cancelación.

Historia de usuario 11: Como paciente, quiero reprogramar mi cita seleccionando otra fecha disponible para no perder la atención médica.

Épica Asociada: Cancelación y reprogramación de citas

Puntos de Historia: 5

Criterios de aceptación:

- a. En la sección "Próximas Citas", cada cita debe tener un botón para "Reprogramar".
- b. Al hacer clic, el sistema debe mostrar el calendario de disponibilidad del médico para seleccionar un nuevo horario.
- c. Al confirmar un nuevo horario, el frontend debe enviar una petición al backend para actualizar la cita en la base de datos de forma atómica.
- d. El horario anterior debe quedar liberado automáticamente.

Historia de usuario 12: Como administrador, quiero generar reportes de pacientes atendidos por especialidad para evaluar la demanda.

Épica Asociada: Reportes administrativos

Puntos de Historia: 5

Criterios de aceptación:

- a. La interfaz de "Reportes" debe permitir al administrador seleccionar filtros (rango de fechas, especialidad).

- b. El frontend debe enviar estos filtros al backend para obtener los datos del reporte.
- c. La consulta en la base de datos debe estar optimizada para manejar grandes volúmenes de datos.
- d. El sistema debe mostrar un resumen de los datos obtenidos en la interfaz.

Historia de usuario 13: Como administrador, quiero exportar reportes de asistencia y ausencias en formato CSV para mejorar la planificación.

Épica Asociada: Reportes administrativos

Puntos de Historia: 3

Criterios de aceptación:

- a. En la interfaz de reportes, debe existir un botón "Exportar a CSV".
- b. Al hacer clic, el frontend debe solicitar al backend la generación del archivo CSV con los datos del reporte actual.
- c. El backend debe generar el archivo CSV y enviarlo para que el navegador del usuario inicie la descarga.

Historia de usuario 14: Como usuario, quiero poder restablecer mi contraseña si la olvido para recuperar el acceso a mi cuenta.

Épica Asociada: Gestión de usuarios y seguridad

Puntos de Historia: 5

Criterios de aceptación:

- a. La página de login debe tener un enlace de "¿Olvidaste tu contraseña?".
- b. Al hacer clic, se debe pedir el correo electrónico del usuario y enviarlo al backend.
- c. El backend debe generar un token único y enviar un correo con un enlace seguro para restablecer la contraseña.
- d. El enlace debe llevar a una página de frontend donde se pueda ingresar y confirmar una nueva contraseña, que será actualizada en la base de datos.

Historia de usuario 15: Como funcionario, quiero poder agendar, cancelar y reprogramar citas en nombre de un paciente.

Épica Asociada: Administración de agenda médica

Puntos de Historia: 8

Criterios de aceptación:

- a. La interfaz del funcionario debe permitir buscar a un paciente por nombre o RUT a través de una consulta al backend.
- b. Una vez seleccionado el paciente, el funcionario debe poder usar la misma interfaz de agendamiento que vería un paciente.
- c. Debe poder ver las citas del paciente y tener botones para cancelar o reprogramar, que consuman los mismos endpoints que usaría el paciente.
- d. Cualquier acción realizada por el funcionario debe quedar registrada y disparar las notificaciones por correo correspondientes.

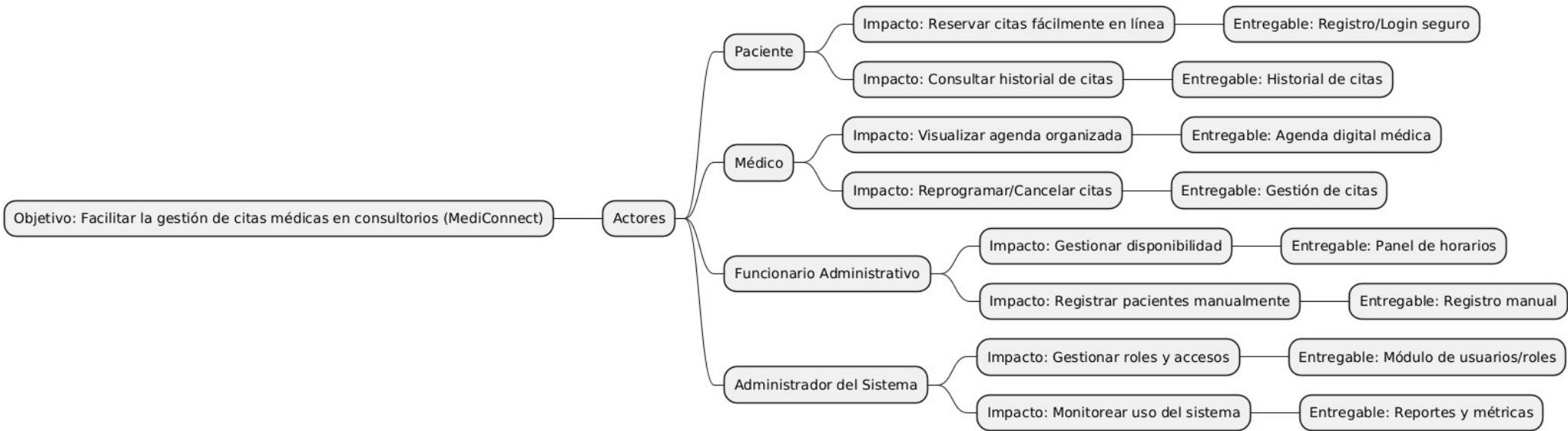
Historia de usuario 16: Como sistema, quiero enviar notificaciones por correo cuando una cita sea cancelada o reprogramada.

Épica Asociada: Reserva y gestión de citas médicas

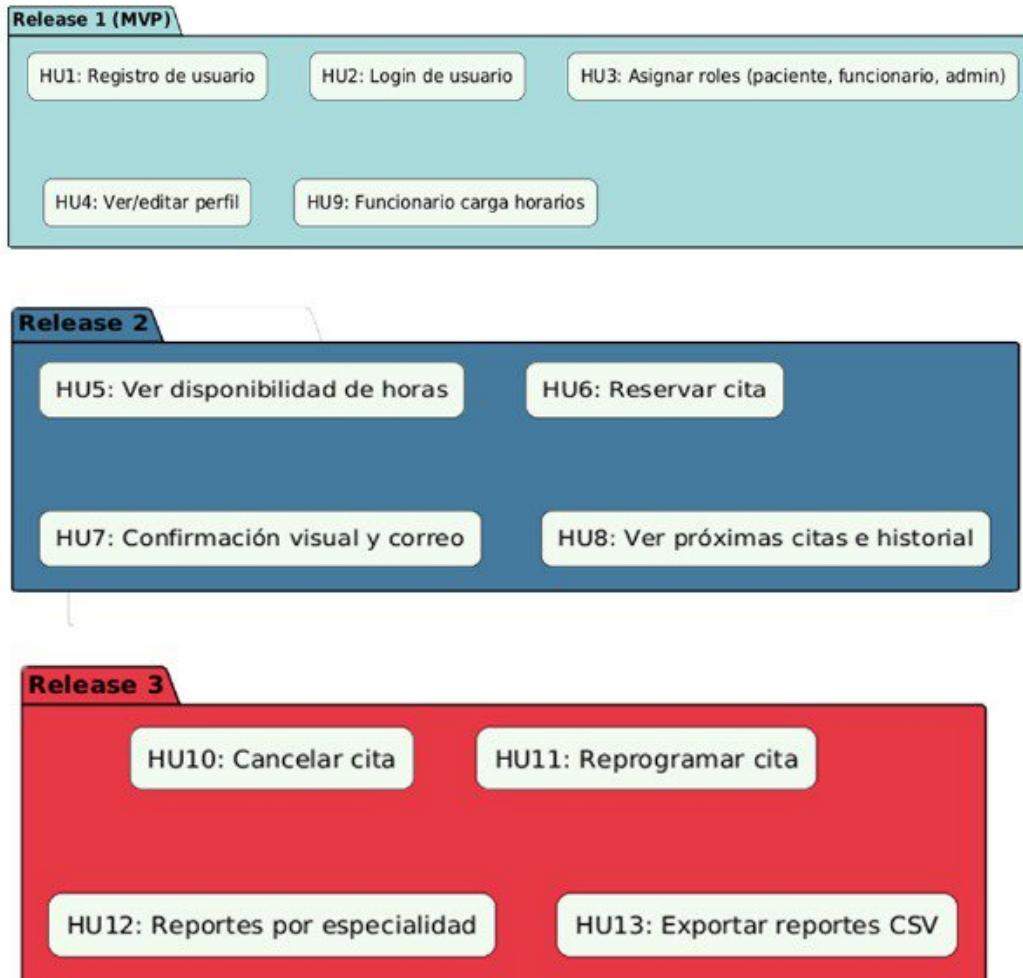
Puntos de Historia: 3

Criterios de aceptación:

- a. La lógica de negocio en el backend para cancelar o reprogramar una cita debe incluir el envío de una notificación por correo.
- b. El correo de cancelación debe informar claramente que la cita ya no está agendada.
- c. El correo de reprogramación debe contener los nuevos detalles de la cita.

3. Impact mapping:

4. Story mapping:



Links a los repositorios y plataformas

Jira: <https://mediconnect-10.atlassian.net/jira/software/projects/MEDI/boards/1/backlog>

GitHub: <https://github.com/Erika10120808/MediConnect-/tree/main>

G-Drive: <https://drive.google.com/drive/u/4/folders/1jCft1gyLMsDkSz78ct52fVN4ipUJfcyn>



Duoc UC