**Documento de**

**Definition of Done para el Proyecto**

**“Mediconecta ”**

***Fecha:xx/xx/xxxx***

Indice

[Datos del documento 3](#__RefHeading___Toc8805_2253226328)

[Introducción 4](#__RefHeading___Toc8807_2253226328)

[Criterios de Completitud 5](#__RefHeading___Toc8809_2253226328)

[Pruebas Automatizadas 6](#__RefHeading___Toc8811_2253226328)

[Documentación 8](#__RefHeading___Toc8813_2253226328)

[Integración 9](#__RefHeading___Toc8815_2253226328)

# Datos del documento

Histórico de Revisiones

|  |  |  |  |
| --- | --- | --- | --- |
| Versión | Fecha | Descripción/cambio | autor |
| 0.1 |  | Inicio Proyecto | Product Owner |
|  |  |  |  |
|  |  |  |  |

Información del Proyecto

|  |  |
| --- | --- |
| Organización | Duoc UC. Escuela de Informática y Telecomunicaciones |
| Sección |  |
| Proyecto (Nombre) | Mediconecta |
| Fecha de Inicio |  |
| Fecha de Término |  |
| Patrocinador principal |  |
| Docente | FABIAN ALEJANDRO ALCANTARA GUAJARDO |

Integrantes

| Rut | Nombre | Correo |
| --- | --- | --- |
| **19916403-1** | **Roberto Valenzuela** |  |
| **20044378-0** | **Efraín Pasarín** |  |
| **20604587-6** | **Sebastián Mora** | **s.mora@duocuc.cl** |

# Introducción

En el contexto del proyecto MediConecta, la Definition of Done (DoD) es esencial para garantizar que cada componente del sistema se desarrolle con la calidad y eficiencia necesarias para satisfacer las necesidades de nuestros usuarios finales: pacientes y doctores. MediConecta busca ofrecer una plataforma integral para la gestión de citas médicas, historial clínico y recomendaciones personalizadas, y para lograr estos objetivos es fundamental establecer criterios claros que definan cuándo una tarea, historia de usuario o funcionalidad está completamente terminada.  
  
La DoD proporciona un marco que asegura que todos los aspectos técnicos, funcionales y de negocio sean considerados antes de marcar un incremento como completado. Esto incluye el desarrollo de características relacionadas con la gestión de pacientes y doctores, la implementación de la base de datos relacional en PostgreSQL, y el correcto funcionamiento de la interfaz de usuario.  
  
Al seguir esta Definition of Done, nuestro equipo garantiza que cada entrega no solo cumpla con los requisitos establecidos, sino que también mantenga altos estándares de calidad y rendimiento, minimizando el riesgo de problemas futuros. La revisión y actualización periódica de la DoD en el contexto de MediConecta también permitirá incorporar lecciones aprendidas y adaptaciones necesarias para mejorar continuamente nuestro proceso de desarrollo y entrega.

# Criterios de Completitud

Desarrollo de Código

* Implementación Completa:  
    
  El código está completamente implementado y cumple con las especificaciones detalladas en la historia de usuario correspondiente. Esto incluye la funcionalidad relacionada con la gestión de pacientes, doctores y citas médicas.
* Convenciones de Codificación:  
    
  Se han seguido rigurosamente las convenciones de codificación establecidas para el proyecto MediConecta. Esto garantiza la coherencia y legibilidad del código, facilitando el mantenimiento y la colaboración entre los miembros del equipo.
* Revisión por Pares:  
    
  El código ha sido revisado por al menos un compañero de equipo antes de ser integrado en la rama principal. Esta revisión busca identificar posibles errores, mejorar la calidad del código y compartir conocimiento entre los miembros del equipo, asegurando que todos estén alineados con las mejores prácticas de desarrollo.

# Pruebas Automatizadas

* Cobertura de Pruebas Unitarias:  
    
  Se han implementado pruebas unitarias que cubren al menos el 80% de la funcionalidad nueva relacionada con la gestión de pacientes, doctores y citas médicas. Esta cobertura asegura que las partes críticas del código están correctamente validadas y funcionan como se espera.
* Ejecución de Pruebas Unitarias:  
    
  Todas las pruebas unitarias han sido ejecutadas y han pasado sin errores. Esto confirma que la implementación cumple con los criterios de calidad y no introduce fallos en las funcionalidades existentes.
* Pruebas de Integración:  
    
  Se han creado pruebas de integración para las funcionalidades que requieren la interacción de múltiples componentes del sistema, como la conexión entre la base de datos y la interfaz de usuario. Estas pruebas garantizan que los distintos módulos del sistema funcionen correctamente en conjunto.
* Pruebas de Regresión:  
    
  Se han ejecutado pruebas de regresión en cada iteración del proyecto, asegurando que las modificaciones o nuevas implementaciones no afecten negativamente a las funcionalidades existentes. Todas las pruebas de regresión han pasado sin errores, validando así la estabilidad del sistema.

# Documentación

* Documentación de la API:  
    
  La API ha sido documentada utilizando Swagger. Esta documentación incluye descripciones detalladas de los endpoints, parámetros, respuestas y ejemplos de uso, lo que facilita la comprensión y utilización de la API por parte de otros desarrolladores y equipos.
* Comentarios en el Código:  
    
  Se han actualizado los comentarios en el código para explicar la lógica compleja y las decisiones clave tomadas durante la implementación. Esto incluye anotaciones sobre algoritmos utilizados, estructuras de datos específicas y justificaciones para ciertas elecciones de diseño, garantizando que otros desarrolladores puedan entender fácilmente la lógica detrás del código.
* Actualización de Documentación del Usuario y del Sistema:  
    
  La documentación del usuario y del sistema ha sido actualizada para reflejar los cambios y nuevas funcionalidades implementadas en el sistema. Esto incluye manuales de usuario, guías de instalación y mantenimiento, así como documentación técnica que describe la arquitectura y el diseño del sistema, asegurando que todos los interesados tengan acceso a la información necesaria para utilizar y mantener MediConecta.

# Integración

* Integración del Código:  
    
  El código ha sido integrado en la rama principal del repositorio de MediConecta. Este proceso asegura que las funcionalidades desarrolladas están disponibles para todo el equipo y para futuras implementaciones.
* Conflictos de Fusión:  
    
  No hay conflictos de fusión en el código tras la integración. Se ha verificado que todos los cambios realizados por diferentes miembros del equipo se han consolidado sin discrepancias, manteniendo la coherencia y estabilidad del código base.
* Construcción en el Entorno de Integración Continua (CI):  
    
  Se ha ejecutado una construcción exitosa en el entorno de integración continua (CI). Esto incluye la ejecución de pruebas automatizadas y la validación de que el código se compila y se integra correctamente, garantizando que las nuevas funcionalidades no rompan la aplicación existente.

**Revisión de Calidad**

* Revisión de Calidad del Código (QA):  
    
  Se ha llevado a cabo una revisión de calidad del código (QA) por parte de miembros del equipo, donde se han evaluado aspectos como la legibilidad, la adherencia a las convenciones de codificación y la optimización de la lógica implementada. Esta revisión busca asegurar que el código cumple con los estándares establecidos para MediConecta.
* Pruebas de Aceptación:  
    
  Se han realizado pruebas de aceptación por parte del Product Owner o un cliente designado. Estas pruebas validan que las funcionalidades desarrolladas cumplen con los requisitos y expectativas del usuario final, asegurando que el producto entregado sea valioso y útil.
* Resolución de Errores Críticos:  
    
  Todos los errores críticos o bloqueos identificados durante la revisión de calidad y las pruebas de aceptación han sido abordados y resueltos. Se han documentado las soluciones implementadas para cada problema, garantizando que el sistema sea robusto y confiable antes de avanzar a la siguiente fase del proyecto.

**Revisión de Calidad**

* Revisión de Calidad del Código (QA):  
    
  Se ha llevado a cabo una revisión de calidad del código (QA) por parte de miembros del equipo, donde se han evaluado aspectos como la legibilidad, la adherencia a las convenciones de codificación y la optimización de la lógica implementada. Esta revisión busca asegurar que el código cumple con los estándares establecidos para MediConecta.
* Pruebas de Aceptación:  
    
  Se han realizado pruebas de aceptación por parte del Product Owner o un cliente designado. Estas pruebas validan que las funcionalidades desarrolladas cumplen con los requisitos y expectativas del usuario final, asegurando que el producto entregado sea valioso y útil.
* Resolución de Errores Críticos:  
    
  Todos los errores críticos o bloqueos identificados durante la revisión de calidad y las pruebas de aceptación han sido abordados y resueltos. Se han documentado las soluciones implementadas para cada problema, garantizando que el sistema sea robusto y confiable antes de avanzar a la siguiente fase del proyecto.

**Despliegue**

* Disponibilidad en Entorno de Desarrollo o Pruebas:  
    
  La funcionalidad desarrollada ha sido desplegada y está disponible en el entorno de desarrollo o pruebas de MediConecta. Esto permite al equipo de desarrollo y a los testers realizar verificaciones exhaustivas y asegurar que la nueva implementación funciona como se espera antes de su lanzamiento en producción.
* Verificación en Entorno de Producción:  
    
  Se ha verificado que la implementación en el entorno de producción no afecta a las funcionalidades existentes. Se han realizado pruebas de regresión y validaciones para confirmar que todas las características previas continúan funcionando correctamente, garantizando así una transición fluida y sin interrupciones para los usuarios finales.

**Feedback y Mejoras**

* Recopilación de Comentarios:  
    
  Se han recogido comentarios del equipo después de cada iteración, incluyendo opiniones sobre la implementación, la colaboración y el proceso en general. Estos comentarios han sido documentados para su análisis posterior y para garantizar que las lecciones aprendidas se incorporen en futuras fases del desarrollo.
* Identificación y Planificación de Acciones:  
    
  Se han identificado áreas de mejora en el proceso de desarrollo y se han planificado acciones concretas para abordar estas cuestiones en futuras iteraciones. Esto incluye optimizaciones en la gestión del backlog, ajustes en la planificación de sprints, y la adopción de nuevas herramientas o prácticas que faciliten la colaboración y la eficiencia del equipo.

**Revisión y Aprobación**

Este documento debe ser revisado y aprobado por el equipo de desarrollo y el Product Owner en intervalos regulares para asegurar su relevancia y eficacia.  
  
Firmas:  
  
Product Owner: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Fecha: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
Líder del Equipo de Desarrollo: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Fecha: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
Scrum Master: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Fecha: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_