**Historias Técnicas**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Fecha** | **Versión** | **Autor** | **Organización** | **Descripción** |
| 23/04/2025 | 1.0 | Cristhian Chimbo | - | Primera versión de historias técnicas |
| 24/04/2025 | 1.1 | Cristhian Chimbo | - | Cambios en las historias técnicas |

|  |  |  |
| --- | --- | --- |
| HT-001 | Configuración del entorno de desarrollo local para Blazor Server | |
| Descripción | | Configurar correctamente el entorno de desarrollo local para trabajar el proyecto Blazor Server, con todas las herramientas necesarias estén instaladas y funcionando. |
| Criterios de Aceptación | | * Visual Studio configurado con soporte para Blazor Server. * SDK de .NET instalado y funcionando. * El proyecto se puede compilar y ejecutar en local sin errores. |
| Tareas | | * Instalar Visual Studio 2022 o Visual Studio Code con extensiones de C#. * Instalar el SDK de .NET (versión específica, por ejemplo, .NET 8). * Compilar y correr el proyecto localmente para verificar que funciona. |

|  |  |  |
| --- | --- | --- |
| HT-002 | Configuración del repositorio para despliegue de Blazor. | |
| Descripción | | Como desarrollador quiero configurar el repositorio GitHub para el proyecto Blazor Server, incluyendo todos los archivos para su funcionamiento. |
| Criterios de Aceptación | | * El proyecto tiene una rama main organizada. * Los commits se realizan frecuentemente. |
| Tareas | | * Crear repositorio nuevo en GitHub * Crear la rama main * Subir el proyecto Blazor Server. |

|  |  |  |
| --- | --- | --- |
| HT-003 | Instalación y configuración de SQL Server Express para entorno local | |
| Descripción | | Instalar SQL Server Express y asegurar que la instancia esté en ejecución, configurada correctamente y lista para ser usada en el proyecto Blazor Server. |
| Criterios de Aceptación | | * SQL Server Express está instalado exitosamente. * Se puede acceder a la instancia usando SQL Server Management Studio (SSMS). * La instancia local permite crear y administrar bases de datos. |
| Tareas | | * Descargar e instalar SQL Server Express Edition. * Descargar e instalar SQL Server Management Studio (SSMS) para administrar la base de datos. * Configurar la instancia para permitir conexiones locales. * Verificar que el servicio de SQL Server esté corriendo. * Crear una base de datos para pruebas iniciales. |

|  |  |  |
| --- | --- | --- |
| HT-004 | Configuración de la API REST y configuración inicial de acceso a base de datos | |
| Descripción | | Configurar el proyecto API REST con ASP.NET Core, para acceder a la base de datos SQL Server usando Entity Framework Core, y preparar la base para exponer servicios que serán consumidos por la aplicación Blazor Server. |
| Criterios de Aceptación | | * Proyecto API creado (ASP.NET Core Web API). * Conexión a SQL Server configurada (DbContext). * Primera entidad y primer controlador creados. |
| Tareas | | * Crear proyecto API en Visual Studio. * Instalar EF Core y configurar DbContext. * Crear modelo de datos |

|  |  |  |
| --- | --- | --- |
| HT-005 | Diseño e implementación inicial de la base de datos | |
| Descripción | | Definir las tablas principales y relaciones de la base de datos SQL Server, crear el esquema inicial y asegurarse que sea funcional para la aplicación. |
| Criterios de Aceptación | | * Las tablas están correctamente creadas en SQL Server. * Se definen columnas, tipos de datos y relaciones. * Se crean los scripts o migraciones necesarias. * La base de datos es accesible sin errores desde la API. |
| Tareas | | * Diseñar el modelo de datos (entidades y relaciones). * Crear scripts SQL o configuraciones de Entity Framework Core. * Ejecutar migraciones o scripts para construir la base de datos. * Verificar que la API puede acceder a la base de datos correctamente. |

|  |  |
| --- | --- |
| HT-006 | Separación de capas en la arquitectura |
| Descripción | Separar el proyecto en capas: presentación (UI), lógica de negocio y acceso a datos, para mejorar mantenimiento y escalabilidad. |
| Criterios de Aceptación | * Cada capa está en un proyecto distinto. * No hay dependencias cruzadas incorrectas. |
| Tareas | * Crear proyectos independientes en la solución. * Configurar inyección de dependencias. |

|  |  |  |
| --- | --- | --- |
| HT-007 | Configuración de consumo de API desde Blazor Server | |
| Descripción | | Configurar el proyecto Blazor Server para consumir datos de una API externa que accede a la base de datos, usando servicios basados en HttpClient. |
| Criterios de Aceptación | | * Se configura un servicio HttpClient para realizar peticiones a la API. * El proyecto puede consumir datos de prueba de la API (GET, POST). * Se implementa un servicio Blazor que maneja la comunicación con la API. |
| Tareas | | * Configurar HttpClient en Program.cs. * Crear un servicio Blazor que realice solicitudes HTTP (IApiService y ApiService). * Probar que Blazor Server obtiene información de la API. |

|  |  |  |
| --- | --- | --- |
| HT-008 | Creación del primer servicio en Blazor Server para consumir la API | |
| Descripción | | Implementar un servicio en Blazor Server que consuma los datos de la API mediante HttpClient, y lo exponga a los componentes de la aplicación. |
| Criterios de Aceptación | | * Servicio IApiService y ApiService creados. * Los componentes de Blazor pueden obtener datos desde la API. * Manejo básico de errores en llamadas HTTP implementado. |
| Tareas | | * Crear una carpeta Services en Blazor Server. * Crear interfaces y clases de servicio para consumir la API. * Inyectar el servicio en componentes Blazor. |

|  |  |
| --- | --- |
| HT-009 | Integración de ASP.NET Identity en Blazor Server |
| Descripción | Implementar autenticación y autorización basada en roles (administrador, laboratorista, recepcionista) usando ASP.NET Identity con base de datos SQL. |
| Criterios de Aceptación | * Flujo de login y logout funcional. * Usuarios y roles gestionables desde la base de datos. |
| Tareas | * Añadir Identity al proyecto. * Crear migraciones para esquema de usuarios. * Proteger páginas según roles. |

|  |  |
| --- | --- |
| HT-010 | Configuración de CI/CD con GitHub Actions |
| Descripción | Implementar pipeline de integración y despliegue continuo para automatizar compilación, pruebas y publicación. |
| Criterios de Aceptación | * Push a rama principal desencadena despliegue exitoso. * Logs disponibles en GitHub. |
| Tareas | * Probar con cambios reales. |

|  |  |
| --- | --- |
| HT-011 | Despliegue de la Aplicación Blazor Server en Azure App Service |
| Descripción | Como desarrollador quiero desplegar la aplicación Blazor Server en Azure App Service, para que esté accesible públicamente en un entorno de producción. |
| Criterios de Aceptación | * La aplicación está publicada y accesible desde una URL pública de Azure. * Se valida que la aplicación corre en modo Production. |
| Tareas | * Crear recurso Azure App Service para .NET Core. * Configurar la publicación desde Visual Studio o GitHub Actions. * Probar acceso a la aplicación desplegada. |

|  |  |
| --- | --- |
| HT-012 | Creación de Azure SQL Database para producción |
| Descripción | Como desarrollador quiero crear una instancia de Azure SQL Database, para alojar los datos de la aplicación en un entorno seguro y accesible en la nube. |
| Criterios de Aceptación | * Azure SQL Database creada y accesible. * Firewall configurado para permitir acceso controlado. |
| Tareas | * Crear recurso Azure SQL Database. * Configurar usuarios y permisos. * Probar la conexión desde la API o desde SQL Server Management Studio. |

|  |  |
| --- | --- |
| HT-013 | Migración de Base de Datos Local a Azure SQL Database |
| Descripción | Como desarrollador quiero migrar la base de datos local de desarrollo hacia Azure SQL Database, para que la aplicación en producción use datos reales en la nube. |
| Criterios de Aceptación | * Migración de tablas, datos y relaciones completa. * Validación de integridad de datos después de la migración. |
| Tareas | * Usar Data Migration Assistant (DMA) o exportar/importar .bacpac. * Subir la base de datos al servidor Azure SQL. * Actualizar cadena de conexión en la aplicación. |

|  |  |
| --- | --- |
| HT-014 | Configuración de dominio personalizado para App Service |
| Descripción | Como desarrollador quiero configurar un dominio personalizado para la aplicación desplegada en Azure, para mejorar la presencia profesional y facilitar el acceso. |
| Criterios de Aceptación | * Dominio correctamente asignado al Azure App Service. * Certificado SSL/TLS habilitado para HTTPS. |
| Tareas | * Configurar registros DNS en el proveedor del dominio. * Asociar dominio personalizado en Azure App Service. * Habilitar redirección automática a HTTPS. |

|  |  |
| --- | --- |
| HT-015 | Configuración de entorno de producción (appsettings y secretos) |
| Descripción | Como desarrollador quiero separar las configuraciones sensibles en el entorno de producción, usando appsettings.Production.json y Application Settings en Azure, para mejorar la seguridad. |
| Criterios de Aceptación | * El proyecto no expone datos sensibles en el repositorio. * Se leen configuraciones de Application Settings en Azure. |
| Tareas | * Crear appsettings.Production.json. * Configurar Application Settings en Azure App Service. * Verificar que la app corre en modo Production usando configuraciones seguras. |