

## **Actividad 1 – Organizando el equipo**

### **Servicios en la nube**

### **Ingeniería en Desarrollo de Software**

#### **Tutor:**

Urbano Francisco Ortega Rivera

#### **Alumno:**

Alejandro Abarca Gerónimo

**Equipo con:** Luis Felipe Jacobo Gallardo

#### **Fecha:**

26 de agosto de 2024

# Indice

<b>Introducción .....</b>	<b>3</b>
<b>Descripción .....</b>	<b>3</b>
<b>Justificación .....</b>	<b>4</b>
<b>Capturas de pantalla.....</b>	<b>5</b>
<b>Explicación breve de cada tarea .....</b>	<b>8</b>
<b>Conclusión .....</b>	<b>11</b>
<b>Referencias .....</b>	<b>12</b>

## **Introducción**

En el contexto del desarrollo de software, la adecuada planificación y ejecución del proyecto es esencial para alcanzar los objetivos establecidos y garantizar la calidad del producto final. Esta actividad tiene como objetivo crear una enciclopedia digital de desarrollo de software que servirá como una fuente integral de información para profesionales y estudiantes de la industria. Esta enciclopedia cubrirá temas clave como métodos de desarrollo, lenguajes de programación, herramientas de desarrollo y mejores prácticas de la industria del software.

El evento sigue un enfoque estructurado basado en fases de desarrollo claramente definidas: análisis, diseño, construcción y transición. Estas fases ayudan a los equipos a organizar y ejecutar proyectos utilizando Azure DevOps Services, una herramienta que facilita la colaboración y la gestión de proyectos.

Durante este proceso se abordarán requisitos funcionales y no funcionales, se desarrollarán esquemas técnicos y se implementarán soluciones para la construcción de sistemas de red eficiente y asequible. El objetivo es proporcionar recursos educativos y útiles para promover el aprendizaje y la referencia en el desarrollo de software.

## **Descripción**

El objetivo de esta actividad es crear una enciclopedia digital centrada en el desarrollo de software, una disciplina fundamental en la era digital. La enciclopedia debería servir como un recurso educativo integral que cubra temas clave como métodos de desarrollo, lenguajes de programación y herramientas industriales. Para lograr este objetivo, las actividades se dividen en varias fases importantes que impulsan la ejecución del proyecto de manera organizada y eficiente. Primero, en la fase de análisis, el equipo debe identificar y documentar los requisitos del sistema, determinar qué información incluir y cómo estructurarla. La fase de diseño incluye la preparación de diagramas

técnicos y la planificación de la base de datos, que son elementos clave para el desarrollo eficaz de una enciclopedia. Durante la fase de construcción, el sistema se implementará para garantizar que todas las funciones funcionen y que el contenido sea accesible y esté bien organizado. Finalmente, la fase de transición implica probar y lanzar el sistema en un entorno en línea, asegurando que la enciclopedia sea accesible para el público objetivo. Con los servicios de Azure DevOps, este proceso será más manejable, lo que permitirá una planificación detallada y una colaboración eficaz entre los miembros del equipo. Este enfoque garantiza el progreso del proyecto, reduce el riesgo y aumenta la calidad del producto final.

## **Justificación**

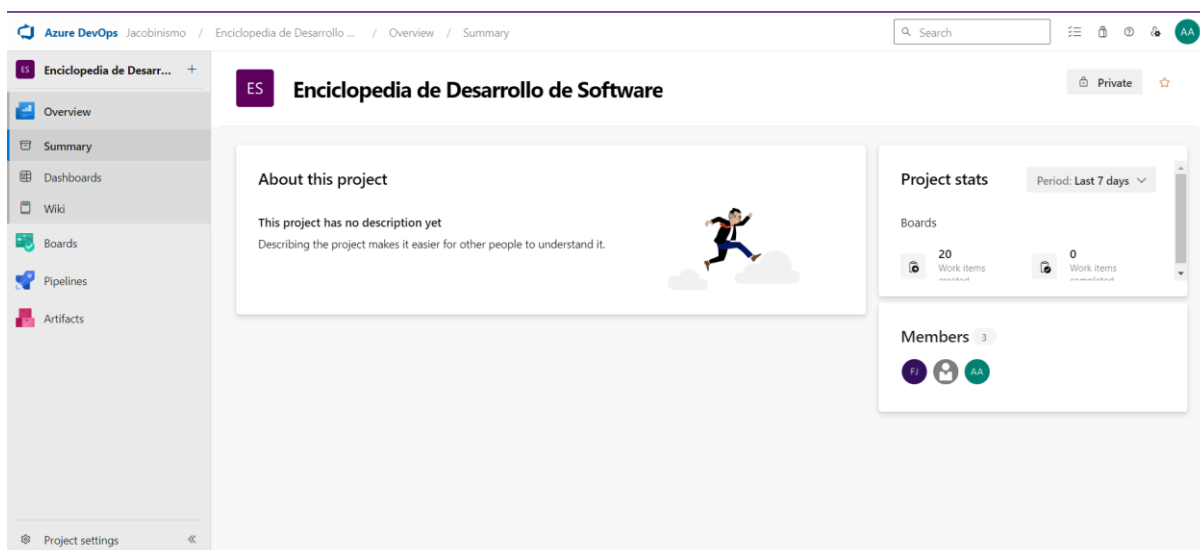
El uso de los servicios de Azure DevOps para crear una enciclopedia digital de desarrollo de software proporcionó varias ventajas importantes que justificaron su elección como solución para este proyecto. En primer lugar, Azure DevOps facilita la gestión de proyectos con herramientas integradas que le permiten planificar, realizar un seguimiento y coordinar tareas de manera eficiente. La plataforma permite a los equipos colaborar instantáneamente, asignar tareas y monitorear el progreso, lo cual es esencial para proyectos complejos como las enciclopedias. Además, Azure DevOps admite el método Scrum, que es adecuado para la naturaleza iterativa del desarrollo de software.

El método Scrum ayuda a dividir los proyectos en Sprint manejables, asegurando que el equipo pueda entregar partes del sistema funcional de forma incremental. Esto permite una mayor flexibilidad para adaptarse a nuevos comentarios y requisitos. La integración de repositorios y herramientas de control de versiones en Azure DevOps garantiza que todo el código y los archivos estén centralizados y versionados correctamente. Esto es esencial para mantener la alineación y promover la colaboración entre los miembros del equipo.

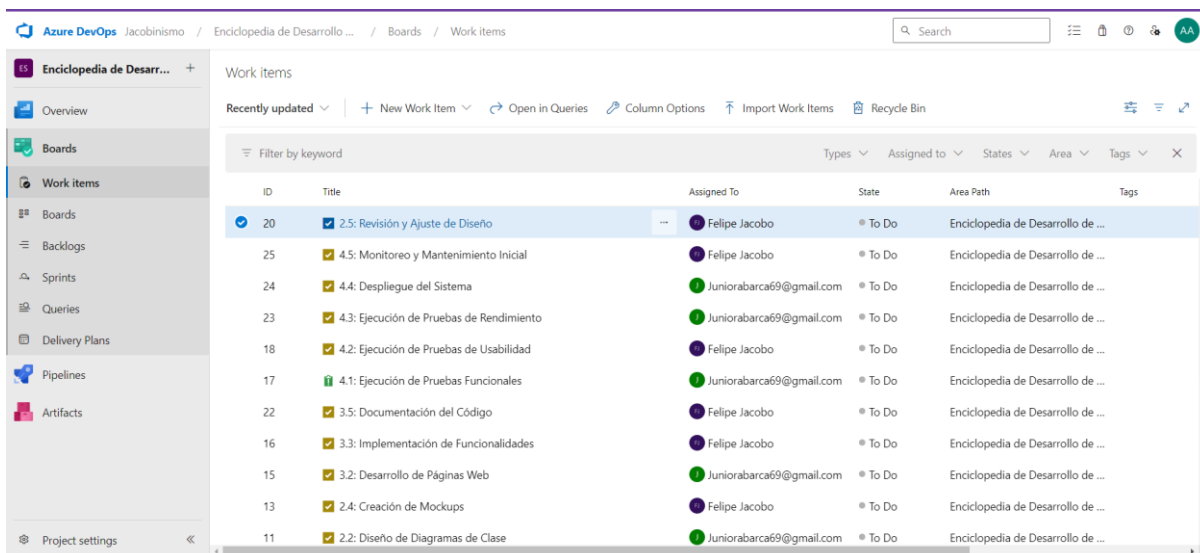
Finalmente, Azure DevOps permite la integración con otros servicios de Azure, como implementaciones en la nube, lo que facilita la publicación y el mantenimiento de enciclopedias en un entorno accesible para el usuario. Azure DevOps proporciona soluciones integrales que optimizan la planificación, el desarrollo y la administración de proyectos para ofrecer resultados finales de alta calidad.

## Capturas de pantalla

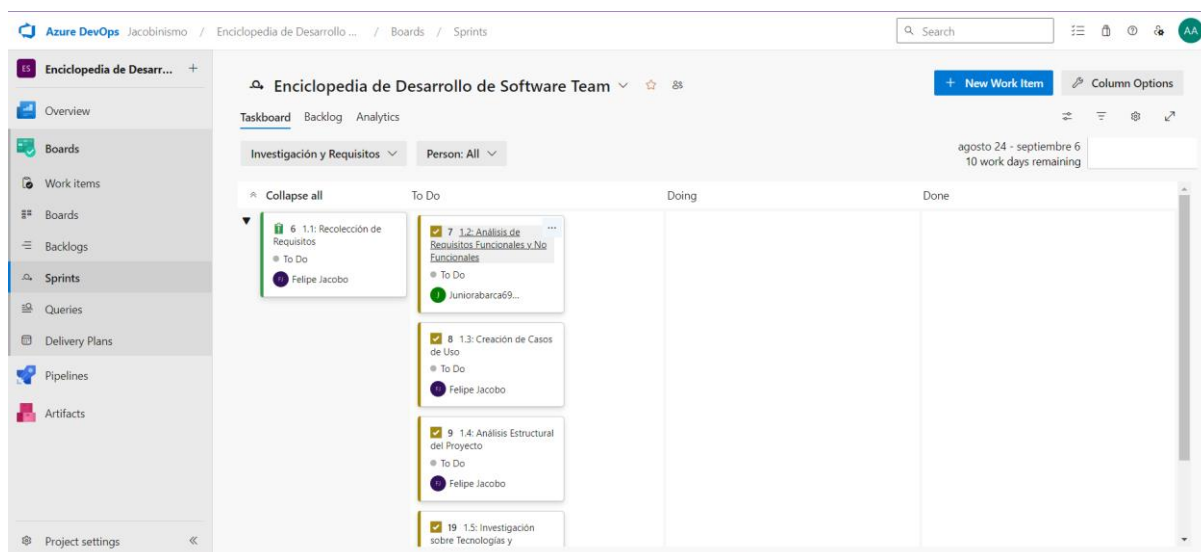
Creación y configuración inicial del proyecto "Enciclopedia de Desarrollo de Software" en Azure DevOps, realizado en conjunto con Luis Felipe Jacobo.



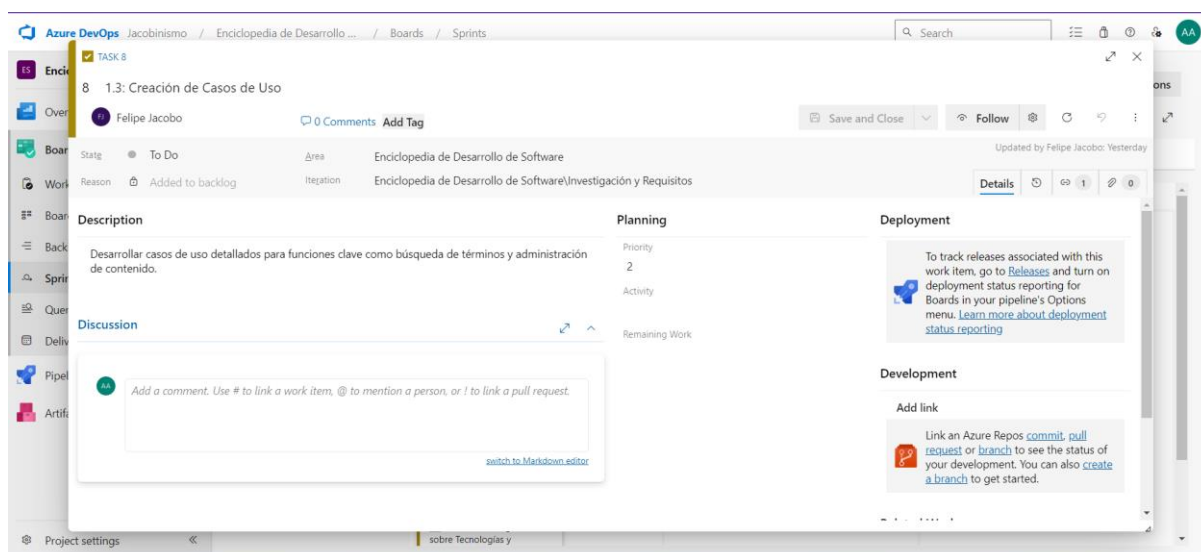
Tareas asignadas en el proyecto "Enciclopedia de Desarrollo de Software" en Azure DevOps, con cada tarea delegada a los miembros del equipo.



Captura de pantalla que muestra la planificación del Sprint, incluyendo todas las tareas a realizar en el proyecto.



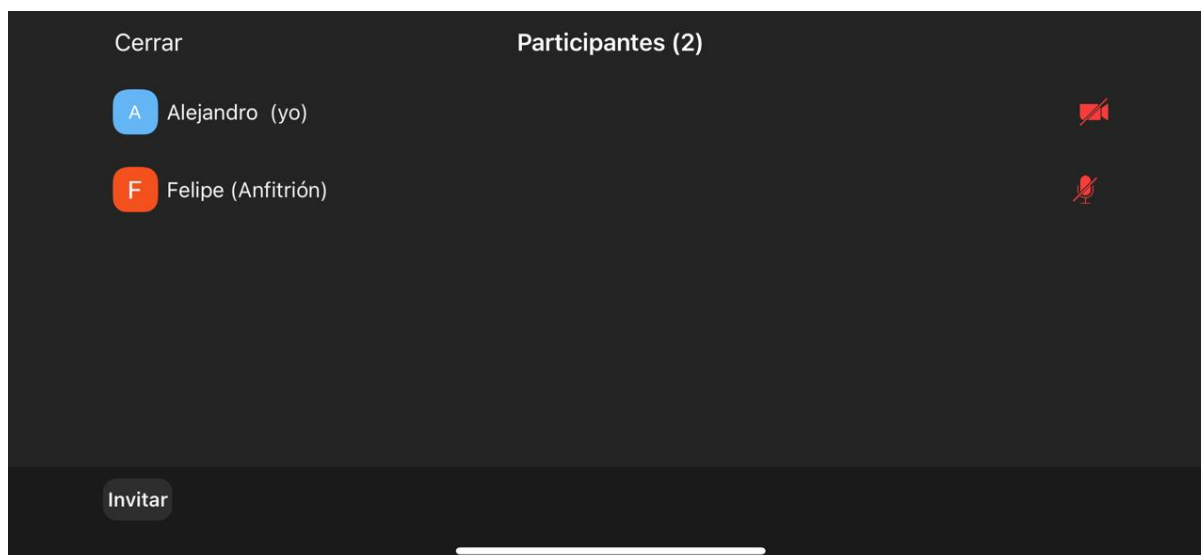
Captura de pantalla que muestra la descripción detallada de las tareas asignadas dentro del proyecto "Enciclopedia de Desarrollo de Software" en Azure DevOps.



Reunión en Zoom donde se dialogó sobre los temas del proyecto y las tareas asignadas a cada uno de los integrantes sobre el tema "Enciclopedia de Desarrollo de Software". En la sesión, Luis líder del proyecto y yo revisamos los requisitos del proyecto, asignamos tareas clave. Durante la reunión, utilizamos Azure DevOps para organizar el trabajo, asegurando que cada tarea estuviera claramente definida y alineada con nuestros objetivos.

The screenshot shows a Zoom meeting window with the Azure DevOps interface in the background. The Azure DevOps interface is for the 'Enciclopedia de Desarrollo de Software' project. It features a Kanban board with columns for 'To Do', 'Doing', and 'Done'. The 'To Do' column contains several tasks, including '1.1: Recolección de Requisitos', '1.2: Análisis de Requisitos Funcionales y No Funcionales', '1.3: Creación de Casos de Uso', and '1.4: Análisis Estructural del Proyecto'. The 'Doing' column is currently empty. The 'Done' column is also empty. A list of work items is displayed at the bottom of the screen, showing details for each task, including its ID, title, assigned to, and state.

ID	Title	Assigned To	State
6	1.1: Recolección de Requisitos	Felipe Jacobo	To Do
7	1.2: Análisis de Requisitos Funcionales y No Funcionales	Juniorabarca69@gmail.com	To Do
8	1.3: Creación de Casos de Uso	Felipe Jacobo	To Do
9	1.4: Análisis Estructural del Proyecto	Felipe Jacobo	To Do
19	1.5: Investigación sobre Tecnologías y Herramientas	Juniorabarca69@gmail.com	To Do
10	2.1: Diseño de Diagramas de Secuencia	Felipe Jacobo	To Do
11	2.2: Diseño de Diagramas de Clase	Juniorabarca69@gmail.com	To Do



## Explicación breve de cada tarea

### Sprint 1: Investigación y Requisitos

#### Tarea 1.1: Recolección de Requisitos

Realizar sesiones de brainstorming para definir qué conceptos y metodologías se incluirán en la enciclopedia.

#### Tarea 1.2: Análisis de Requisitos Funcionales y No Funcionales

Documentar los requisitos funcionales (como módulos de búsqueda y gestión de contenido) y no funcionales (como diseño responsivo y seguridad).

#### Tarea 1.3: Creación de Casos de Uso

Desarrollar casos de uso detallados para funciones clave como búsqueda de términos y administración de contenido.

#### Tarea 1.4: Análisis Estructural del Proyecto

Definir la estructura general del proyecto, incluyendo la navegación y las secciones principales de la enciclopedia.

#### Tarea 1.5: Investigación sobre Tecnologías y Herramientas



Investigar las tecnologías y herramientas adecuadas para el desarrollo del sistema, como lenguajes de programación, frameworks, y herramientas de gestión.

## **Sprint 2: Diseño**

### **Tarea 2.1: Diseño de Diagramas de Secuencia**

Crear diagramas de secuencia que muestren la interacción entre el usuario y el sistema para procesos clave.

### **Tarea 2.2: Diseño de Diagramas de Clase**

Diseñar diagramas de clase que representen las principales entidades del sistema, como términos y usuarios.

### **Tarea 2.3: Diseño de la Base de Datos**

Definir la estructura de la base de datos, incluyendo tablas y relaciones entre ellas.

### **Tarea 2.4: Creación de Mockups**

Diseñar prototipos visuales de las páginas principales de la enciclopedia, como la página de inicio y formularios de gestión.

### **Tarea 2.5: Revisión y Ajuste de Diseño**

Revisar y ajustar los diseños y diagramas basándose en el feedback recibido del equipo o stakeholders.

## **Sprint 3: Desarrollo**

### **Tarea 3.1: Implementación de la Base de Datos**

Crear y configurar la base de datos según el diseño definido en el Sprint 2.

### **Tarea 3.2: Desarrollo de Páginas Web**

Implementar las páginas web basadas en los mockups diseñados.

### **Tarea 3.3: Implementación de Funcionalidades**

Programar y poner en funcionamiento las funcionalidades de búsqueda, gestión de contenido y otras características clave.

### **Tarea 3.4: Integración de Componentes**

Integrar los diferentes componentes del sistema, asegurando que interactúen correctamente entre sí.

### **Tarea 3.5: Documentación del Código**

Crear documentación técnica del código para facilitar el mantenimiento y futuras modificaciones.

## **Sprint 4: Pruebas y Despliegue**

### **Tarea 4.1: Ejecución de Pruebas Funcionales**

Realizar pruebas funcionales para asegurar que todas las características del sistema funcionen según lo esperado.

### **Tarea 4.2: Ejecución de Pruebas de Usabilidad**

Realizar pruebas de usabilidad para asegurarse de que la interfaz sea intuitiva y fácil de usar para los usuarios.

### **Tarea 4.3: Ejecución de Pruebas de Rendimiento**

Comprobar que el sistema cargue rápidamente y responda bien bajo diferentes condiciones de carga.

### **Tarea 4.4: Despliegue del Sistema**

Subir el sistema a un servicio de hosting y verificar que esté accesible en Internet.

#### **Tarea 4.5: Monitoreo y Mantenimiento Inicial**

Monitorear el sistema post-despliegue para identificar y solucionar posibles problemas iniciales.

### **Conclusión**

Durante el desarrollo de la actividad en conjunto con los integrantes del equipo se analizó y planificó el desarrollo del proyecto así como los requerimientos para la elaboración satisfactoria de la actividad.

Se utilizó DevOps para dicha planificación, ya que esta herramienta ofrece una nueva forma de trabajar, un cambio cultural que tiene una profunda repercusión tanto para los equipos como para las organizaciones en las que trabajan.

Con el modelo DevOps se rompe el aislamiento de los equipos de desarrollo y operaciones. Estos equipos se combinan en uno. En este caso, ingenieros con una variedad de habilidades multidisciplinarias trabajarán durante todo el ciclo de vida de la aplicación, desde las fases de desarrollo y prueba hasta la implementación y las operaciones.

Los equipos de DevOps utilizan herramientas para automatizar y acelerar los procesos, haciéndolos más confiables. Al utilizar la cadena de herramientas de DevOps, los equipos pueden resolver desafíos clave de DevOps, como la integración continua, la entrega continua, la automatización y la colaboración.

## Referencias

1. Canive, T. (2020, enero 14). ¿Cómo hacer un sprint en Scrum para proyectos ágiles? Gestor de proyectos online. <https://www.sinnaps.com/blog-gestion-proyectos/sprint-en-scrum>
2. Félix, F. (2022, diciembre 31). 15 buenas prácticas para proyectos de desarrollo de software. Velneo.com. <https://www.velneo.com/blog/15-buenas-practicas-proyectos-desarrollo-software>
3. Insitech. (2024, enero 9). Fases y mejores prácticas para la gestión de proyectos de software. BLOG - Insitech - BMC Partner - Servicios de consultoría en TI. <https://go.insitech.com.mx/fases-y-mejores-practicas-para-la-gestion-de-proyectos-de-software/>
4. Laoyan, S. (2024, enero 23). La importancia de la planificación de sprints en las metodologías ágiles. Asana. <https://asana.com/es/resources/sprint-planning-meeting>
5. Pérez, A. (2021, abril 25). Las 5 etapas en los “Sprints” de un desarrollo Scrum. OBS Business School. <https://www.obsbusiness.school/blog/las-5-etapas-en-los-sprints-de-un-desarrollo-scrum>
6. ¿Qué es la gestión de proyectos de software? (s/f). Wrike.com. Recuperado el 25 de agosto de 2024, de <https://www.wrike.com/es/project-management-guide/faq/que-es-la-gestion-de-proyectos-de-software/>
7. Rehkopf, D. M. (s/f). Todo lo que necesitas saber sobre los sprints de scrum. Atlassian. Recuperado el 25 de agosto de 2024, de <https://www.atlassian.com/es/agile/scrum/sprints>
8. Saavedra, J. A. (2023, agosto 14). Qué son los Sprints en la programación: características, ventajas y etapas. Ebac. <https://ebac.mx/blog/que-son-los-sprints>