**1. Project Justification (1 minuto)**

**Texto a decir:** "Good day everyone. Today I am excited to present my project EcoVital, a mobile application designed to enhance health and wellness. EcoVital stands out by incorporating an advanced chatbot powered by artificial intelligence, specifically utilizing the OpenAI API. This intelligent chatbot provides personalized health-related advice, answering users' questions and offering recommendations suited to their individual needs. Additionally, the app includes features like personalized reminders for important health-related actions and a system to track progress. Furthermore, EcoVital ensures user data security with secure login and registration processes. The project was chosen to leverage the latest AI technology to address the growing demand for accessible and personalized health management solutions. By combining AI technology with user-centric design, EcoVital aims to create a seamless and supportive experience for all users, promoting overall well-being."

**2. Planificación y Análisis de Requisitos + Diagramas (2 minutos)**

**Texto a decir:** "Utilizando el método de diagrama de Scrum, el proyecto se dividió en seis fases principales: Estudio de Viabilidad, Análisis, Diseño, Implementación, Pruebas y Despliegue y Documentación.

Durante la fase de análisis, se crearon los diagramas de casos de uso y clases, especificando las características del chatbot. Los requisitos funcionales incluyen un chatbot inteligente, recordatorios personalizados y seguimiento de metas de salud. Los requisitos no funcionales se centran en la usabilidad, rendimiento, seguridad y escalabilidad de la aplicación.

En el diagrama de casos de uso, se visualizan las interacciones del usuario con la aplicación, destacando la interacción con el chatbot, la configuración de recordatorios y el seguimiento de metas.

El diagrama de clases de EcoVital se organiza en cuatro categorías principales:

1. **Models**:
   * **UserInfo**: Depende de SecurityQuestion y ChangePasswordRequest, y se relaciona con UserGoals y ActivityRecords para gestionar la información del usuario, metas y actividades.
2. **Services**:
   * **ILoginRepository**: Define métodos como Login, Register y ChangePassword.
   * **OpenAIService**: Gestiona las interacciones del chatbot.
   * **LoadingService**: Maneja la interfaz de carga durante operaciones asíncronas.
3. **ViewModels**:
   * Todos heredan de BaseViewModel para gestionar el estado de la aplicación.
   * Ejemplos incluyen ChangePasswordViewModel y ForgotPasswordPageViewModel.
4. **Views**:
   * Las vistas como AboutPage, ChangePasswordPage y ChatBotPage se relacionan con BaseViewModel mediante el patrón MVVM, actualizando dinámicamente la interfaz."

**Texto a decir:** "Ahora, brevemente, el diagrama MER (Modelo Entidad-Relación) de nuestra aplicación:

* **UserInfos**: Almacena la información básica del usuario como UserId, UserName, Email y Password.
* **UserGoals**: Relacionado con UserInfos para rastrear las metas del usuario.
* **ActivityRecords**: Registra las actividades realizadas por el usuario.
* **SecurityQuestions**: Vinculado a UserInfos para la recuperación de contraseñas.
* **Feedback**: Permite a los usuarios enviar sus opiniones sobre la aplicación.

Este diagrama nos ayuda a estructurar la base de datos de manera eficiente y segura."

**3. Desarrollo de la App (4 minutos, incluyendo videos)**

**Texto a decir y tiempo para videos:** **1. Control y Autenticación de Usuarios (1 minuto):** "La aplicación implementa un sistema seguro de registro e inicio de sesión con cifrado SHA-256. Los usuarios pueden recuperar su contraseña mediante preguntas de seguridad." *Video de 44 segundos sobre el sistema de autenticación y registro.*

**2. Desarrollo del Chatbot (1 minuto):** "El chatbot proporciona respuestas personalizadas utilizando la API de OpenAI, basado en el modelo GPT-3.5-turbo." *Video de 26 segundos sobre la interacción del chatbot.*

**3. Seguimiento de Metas (1 minuto):** "Los usuarios pueden establecer y seguir sus metas de salud con indicadores visuales de progreso." *Video de 34 segundos sobre el seguimiento de metas.*

**4. Sistema de Recordatorios (1 minuto):** "Los usuarios pueden configurar recordatorios para diversas actividades de salud." *Video de 22 segundos sobre el sistema de recordatorios.*

**5. Sistema de Contacto y Feedback (0.5 minutos):** "La aplicación incluye un sistema de contacto y feedback para mejorar continuamente la experiencia del usuario." *Video de 18 segundos sobre el sistema de contacto y feedback.*

**4. Pruebas, Implantación y Documentación (3 minutos)**

**Texto a decir:** "En la fase de pruebas, se realizaron pruebas unitarias e integrales para asegurar el correcto funcionamiento de cada componente. También se realizaron pruebas de usabilidad para refinar la experiencia del usuario.

Para la implantación, se generaron los archivos de distribución y se publicaron en tiendas de aplicaciones. La documentación técnica incluye el código fuente en un repositorio de GitHub y la documentación de la API. Además de que ha sido generada gracias a la herramienta DocFX, herramienta de generación de documentación de código fuente para proyectos .NET en concreto.

Los manuales de usuario proporcionan una guía de inicio rápido y un manual completo para resolver problemas.

El proceso de recuperación de contraseña y la interacción del chatbot están bien documentados, asegurando que los usuarios tengan una experiencia fluida y segura al usar la aplicación."

**Resumen y Conclusión (1 minuto)**

**Texto a decir:** "En conclusión, EcoVital está diseñado para proporcionar una plataforma interactiva y fácil de usar para gestionar la salud y el bienestar de manera eficiente. Aprovechando tecnologías avanzadas de IA y manteniendo un fuerte enfoque en la experiencia del usuario, EcoVital tiene como objetivo promover hábitos saludables y ofrecer apoyo continuo a sus usuarios. Muchas gracias por su atención."