**Caso de Uso: Aplicación de Gestión de Tareas**

**Descripción del Proyecto:**

Desarrollar un proyecto que cree y comparta vía correo electrónico una imagen con los resultados de las loterías del día, adicional una aplicación web simple de administración de los parámetros. Los usuarios deben poder cambiar la imagen de fondo, el horario de ejecución, correos de envío y demás parámetros que considere útiles.

**Requerimientos Funcionales:**

1. **Creación y envío de la imagen:**
   * Se debe crear una imagen que traiga los resultados de las loterías del día (Resultado Prueba.png) para ser compartida posteriormente por las redes sociales de Apostar. Este archivo debe ser compartido a los correos destinados para la prueba usar (Direccion.analitica@apostar.com.co)
   * Adjunto encontrará la imagen del fondo (fondo.png) que se utiliza esta imagen también debe ser parametrizable pues periódicamente cambian los promocionales que se encuentran en el encabezado y pie de página**.**
   * Para consultar los resultados de las loterías utilice el siguiente Endpoint: [resultadosloterias.apostar.com.co/api/test-db/](https://resultadosloterias.apostar.com.co/api/test-db/)
   * La imagen se debe enviar por correo de manera programada todos los días a las 23:50 con los resultados de las loterías del día.
2. **UI/UX:**
   * La interfaz debe ser simple, limpia y fácil de usar.
   * Opcional: Implementar un diseño responsivo que se vea bien en dispositivos móviles.
3. **Persistencia de Datos:**
   * Los datos deben ser persistentes (usando una base de datos simple como SQLite o una base de datos en memoria).
4. **Despliegue:**
   * La aplicación debe estar contenida en un entorno Dockerizado.
   * Incluir un Dockerfile y un archivo docker-compose.yml para facilitar el despliegue.
5. **Pruebas:**
   * Implementar pruebas unitarias básicas para las principales funcionalidades de la aplicación.

**Criterios de Evaluación:**

1. **Calidad del Código:**
   * Limpieza y organización del código.
   * Uso adecuado de buenas prácticas de desarrollo (nombres de variables, comentarios, modularidad).
   * Correcto manejo de errores.
2. **Cumplimiento de Requerimientos:**
   * Completitud de las funcionalidades solicitadas.
   * Implementación correcta de las tareas programadas.
3. **Uso de Git:**
   * El candidato debe hacer uso de un repositorio Git para su desarrollo, con commits claros y significativos.
4. **Documentación:**
   * Incluir un archivo README.md que explique cómo ejecutar la aplicación, cómo está estructurada, y cualquier consideración adicional.
5. **Pruebas y Despliegue:**
   * Correcta implementación y ejecución de las pruebas unitarias.
   * Facilidad para levantar la aplicación en un entorno Dockerizado.

**Entregables:**

1. Código fuente en un repositorio Git (preferiblemente en GitHub o GitLab).
2. Instrucciones claras en un archivo README.md para levantar la aplicación.
3. Breve explicación de las decisiones técnicas tomadas y posibles mejoras futuras.