**Chequeo Front-End:**

**1. Diseño y UX/UI**

* **¿El diseño es consistente en todas las pantallas (colores, tipografía, iconografía)?**
* **Interactividad:** ¿Las interacciones (clics, desplazamientos, hover, etc.) funcionan correctamente?
* ¿El estilo visual sigue las guías de diseño (por ejemplo, Material Design, Human Interface Guidelines)?
* **Accesibilidad:** ¿El sitio es accesible para personas con discapacidades? (uso de lector de pantalla, navegación por teclado, contraste, etc.)
* **Responsividad:** ¿La interfaz de usuario se adapta adecuadamente a diferentes dispositivos y tamaños de pantalla? (Móvil, tablet, escritorio)
* **Flujo de usuario:** ¿Los usuarios pueden completar las tareas fácilmente y de manera intuitiva?
* ¿El diseño es consistente en todas las pantallas (colores, tipografía, iconografía)?
* ¿El diseño es visualmente atractivo y moderno?
* ¿El uso de imágenes, gráficos y videos está equilibrado y no sobrecarga al usuario?

**Claridad en la jerarquía visual:**

* ¿Los elementos más importantes (botones, llamadas a la acción) están destacados correctamente?
* ¿La jerarquía visual ayuda al usuario a entender de inmediato dónde debe hacer clic o qué debe hacer a continuación?

**2. Rendimiento**

* **Velocidad de carga:** ¿Las páginas se cargan en un tiempo razonable?
* **Optimización de imágenes:** ¿Las imágenes están optimizadas para no afectar el tiempo de carga?
* **Uso de recursos:** ¿El uso de recursos como JavaScript y CSS está optimizado para evitar sobrecargar el navegador?

**Facilidad de navegación:**

* ¿Es fácil para los usuarios encontrar lo que buscan?
* ¿La estructura de navegación es intuitiva y predecible?
* ¿El menú está organizado y accesible en todas las pantallas?

**Compatibilidad con tecnologías de asistencia:**

* ¿La aplicación es compatible con lectores de pantalla y otras tecnologías de asistencia?
* ¿El contenido es accesible para personas con discapacidades (color, tamaño de texto, control por teclado, etc.)?

**Privacidad del usuario:**

* ¿El usuario tiene control sobre sus datos (por ejemplo, la opción de borrar su cuenta o cambiar configuraciones de privacidad)?
* ¿Las políticas de privacidad están disponibles y son fáciles de entender?

**3. Funcionalidad**

* **Validación de formularios:** ¿Los formularios validan correctamente las entradas del usuario? (campos obligatorios, formatos de correo, contraseñas, etc.)
* **Errores y mensajes:** ¿Los mensajes de error son claros, concisos y útiles para el usuario?
* **Integración con el back-end:** ¿Los datos se obtienen y muestran correctamente desde el servidor?

**4. Código**

* **HTML/CSS limpio y semántico:** ¿El código HTML y CSS es limpio, bien estructurado y fácil de mantener?
* **JS sin errores:** ¿No hay errores en la consola del navegador? ¿El JavaScript está bien organizado?
* **Buenas prácticas:** ¿Se siguen las mejores prácticas en cuanto a accesibilidad, usabilidad y estructura del código?

**5. Pruebas**

* **Pruebas funcionales:** ¿Se han realizado pruebas funcionales en las principales funcionalidades del front-end?
* **Pruebas en navegadores:** ¿El producto ha sido probado en todos los navegadores y dispositivos relevantes?
* **Pruebas de usabilidad:** ¿Se han realizado pruebas con usuarios reales para verificar la facilidad de uso?

**Chequeo Back-End:**

**1. Funcionalidad**

* **Lógica de negocio: ¿La lógica de negocio en el back-end se comporta como se espera?**
* **Gestión de errores: ¿El sistema maneja correctamente los errores y excepciones (por ejemplo, 404, 500)?**
* **Validación de datos: ¿Los datos se validan adecuadamente antes de ser procesados o almacenados?**

**2. Seguridad**

* **Autenticación y autorización: ¿El sistema tiene un mecanismo seguro de autenticación y autorización de usuarios? (JWT, OAuth, sesiones, etc.)**
* **Protección contra inyecciones: ¿El sistema es resistente a ataques de inyección (SQL, XSS, CSRF)?**
* **Cifrado: ¿Se cifran las contraseñas y otros datos sensibles correctamente (por ejemplo, usando bcrypt, AES)?**
* **Revisión de dependencias: ¿Se han revisado las dependencias de la aplicación para asegurarse de que no haya vulnerabilidades conocidas?**

**3. Rendimiento**

* **Escalabilidad: ¿El sistema puede escalar horizontalmente si es necesario?**
* **Uso eficiente de la base de datos: ¿Las consultas a la base de datos están optimizadas para rendimiento? ¿Se usan índices correctamente?**
* **Control de carga: ¿El sistema puede manejar la carga de usuarios concurrentes de manera efectiva (caché, balanceo de carga, etc.)?**

**4. Integraciones**

* **API's: ¿Las APIs funcionan correctamente (GET, POST, PUT, DELETE)? ¿El formato de datos (JSON, XML) es consistente y correcto?**
* **Integración con servicios externos: ¿Las integraciones con servicios de terceros (por ejemplo, pasarelas de pago, servicios de correo electrónico) están funcionando sin problemas?**
* **Manejo de fallos en integraciones externas: ¿El sistema maneja correctamente los errores provenientes de servicios externos?**

**5. Base de datos**

* **Estructura de la base de datos: ¿La estructura de la base de datos es adecuada y normalizada?**
* **Integridad de datos: ¿Se garantiza la integridad referencial y la consistencia de los datos?**
* **Backups: ¿Hay un sistema adecuado de copias de seguridad y recuperación de datos?**