

Control de documento

| | |
|--|---|
| Nombre del proyecto | LensFrame by Swiftcode |
| Cierre de iteración | I-09 10/11/23 |
| Generado por | Ivan Osmar Vasquez Flores Marcos Eduardo Solis Cenicerros |
| Aprobado por | Oswaldo Alfonso Hernandez Campos Carlos Francisco Babún Ravelo Alejandro García Alférez |
| Alcance de la distribución del documento | Control interno para todo el proyecto. |

Índice

| | |
|---|-----------|
| Sobre este documento | 3 |
| Resumen de la Iteración. | 4 |
| Identificación. | 4 |
| Hitos especiales | 5 |
| Artefactos y evaluación | 6 |
| Riesgos y problemas. | 9 |
| Notas y observaciones | 13 |
| Asignación de recursos | 14 |
| Anexos | 15 |
| Glosario de términos | 22 |
| Significado de los elementos de la notación gráfica | 23 |

Sobre este documento

La calidad se logra por medio de la revisión constante de las actividades que conducen desde la idea al producto. Al momento del cierre de una iteración es buen momento para hacer un alto, y evaluar lo logrado, los problemas encontrados y los retos a enfrentar.

El presente documento marca el final de la iteración [\[I - 09\]](#), y contiene una evaluación de los artefactos y actividades realizadas durante la misma.

Se recogen también las impresiones y observaciones hechas durante el desarrollo de la iteración, así como el esfuerzo invertido en cada una de las disciplinas involucradas.

Selección de la plataforma en la cual el desarrollo del proyecto se está llevando a cabo.

Resumen de la Iteración.

Identificación.

| Código de la iteración | Fase a la que pertenece | Fecha de inicio | Fecha de cierre | Comentarios |
|------------------------|-------------------------|-----------------|-----------------|--|
| I - 09 | INICIO | 06 nov 2023 | 10 nov 2023 | Mantenimiento Continuo: Actualizar la aplicación regularmente para corregir errores, agregar nuevas características y mantener la compatibilidad con nuevos dispositivos y sistemas operativos. |
| | | | | |

Hitos especiales

1. Identificación y Priorización de Problemas:

Identificación y Priorización de Problemas, mediante la revisión detallada de informes de errores y la clasificación de estos según su criticidad. Este proceso permitirá centrarse en resolver aquellos problemas que afecten significativamente la experiencia del usuario o la estabilidad del software.

2. Corrección de Errores:

Implementación de soluciones efectivas para los problemas identificados. Esta fase incluirá pruebas internas para garantizar la eficacia de las soluciones y la estabilidad general del sistema. Al abordar los errores de manera proactiva, se busca mantener la aplicación en un estado óptimo y asegurar una experiencia positiva para los usuarios.

3. Adición de Nuevas Características y Mejoras:

Con el objetivo de mantener la aplicación fresca y relevante, se contempla la evaluación de solicitudes de características y sugerencias de los usuarios para identificar oportunidades de mejora. La implementación de nuevas características busca enriquecer la funcionalidad de la aplicación y adaptarla a las necesidades cambiantes de los usuarios.

4. Garantía de Compatibilidad:

Se verifica la adaptación de la aplicación a las últimas actualizaciones de sistemas operativos y dispositivos móviles. Esto implica ajustes necesarios para asegurar que la aplicación mantenga su funcionalidad en un entorno tecnológico en constante evolución.

5. Pruebas Finales y Documentación:

Finalmente, se llevarán a cabo diversas pruebas, una etapa crítica para garantizar la estabilidad y el rendimiento de la aplicación después de implementar correcciones y mejoras. La documentación del usuario se actualizará para reflejar las últimas mejoras y proporcionar información relevante sobre las nuevas características.

Artefactos y evaluación

| Artefacto | Aspecto a evaluar | Evaluación % | Comentarios |
|-----------|--------------------------------|--------------|---|
| ICF-33 | Informe de Errores Priorizados | 85% | El informe detalla de manera exhaustiva cada error identificado, proporcionando información clara sobre su impacto y prioridad. La priorización se realiza con precisión, asegurando que se abordan primero los problemas críticos para la experiencia del usuario. |
| | | | |

| Artefacto | Aspecto a evaluar | Evaluación % | Comentarios |
|-----------|--------------------------------------|--------------|--|
| ICF-34 | Registro de Soluciones Implementadas | 75% | El registro documenta de manera clara las soluciones implementadas para cada error identificado. Se verifica la eficacia de las soluciones a través de pruebas internas, aunque algunos detalles podrían ser más detallados para una comprensión completa. |
| | | | |

| Artefacto | Aspecto a evaluar | Evaluación % | Comentarios |
|-----------|--|--------------|--|
| ICF-35 | Documento de Características y Mejoras | 80% | El documento detalla nuevas características y mejoras, proporcionando una comprensión clara de su inclusión. Se destaca la alineación con las necesidades del usuario, aunque se sugiere mayor énfasis en la justificación de la inclusión de cada elemento. |
| | | | |

| Artefacto | Aspecto a evaluar | Evaluación % | Comentarios |
|-----------|------------------------------|--------------|--|
| ICF-36 | Evaluación de Compatibilidad | 85% | El informe documenta las evaluaciones de compatibilidad de manera completa. Se verifica que la aplicación se ajusta correctamente a las actualizaciones, aunque algunos ajustes adicionales podrían considerarse para garantizar una compatibilidad total. |
| | | | |

| Artefacto | Aspecto a evaluar | Evaluación % | Comentarios |
|-----------|----------------------------------|--------------|---|
| ICF-37 | Documentación de Pruebas Finales | 80% | La documentación refleja de manera clara los resultados de las pruebas finales y se actualiza de manera efectiva. Se destaca la estabilidad general de la aplicación, aunque se sugiere una mayor claridad en algunos puntos de la documentación. |
| | | | |

Riesgos y problemas.

ICF-33 Informe de errores priorizados:

El informe detalla de manera exhaustiva cada error identificado, proporcionando información clara sobre su impacto y prioridad. La priorización se realiza con precisión, asegurando que se abordan primero los problemas críticos para la experiencia del usuario.

Posibles riesgos:

1. **Falta de Información Completa:** Existe la posibilidad de que el equipo no recopile información completa sobre algunos errores, lo que podría afectar la exhaustividad del informe. Esto podría resultar en una priorización inexacta y pérdida de tiempo al abordar problemas sin comprender completamente su alcance.
2. **Dificultad en la Evaluación de Impacto:** Podría haber dificultades en la evaluación del impacto de algunos errores, llevando a una priorización inexacta. Esto podría resultar en la resolución de problemas que no se alinean completamente con las expectativas de los usuarios y las necesidades del proyecto.
3. **Cambios en la Prioridad durante el Proceso:** Cambios inesperados en la prioridad de ciertos problemas podrían ocurrir durante la recopilación de información. Esto podría dificultar la consistencia en la priorización y la reasignación eficiente de recursos.
4. **Retrasos en la Identificación de Errores Críticos:** Errores críticos podrían no identificarse oportunamente, posiblemente debido a limitaciones en las pruebas o falta de retroalimentación de los usuarios. Esto podría resultar en retrasos en la resolución de problemas que afectan significativamente la experiencia del usuario.
5. **Desalineación con Expectativas del Usuario:** Puede haber diferencias en la percepción de la gravedad de los errores entre el equipo y los usuarios. Esto podría resultar en una priorización incorrecta que no refleje las preocupaciones y expectativas reales de los usuarios.
6. **Dependencia de la Retroalimentación de Usuarios:** La calidad del informe podría depender en gran medida de la retroalimentación de los usuarios. Si la retroalimentación de los usuarios es limitada, el informe puede carecer de diversidad y completitud.
7. **Complejidad en la Definición de Prioridades:** La diversidad de errores puede hacer que la definición de prioridades sea compleja y subjetiva. Esto podría resultar en dificultades para asignar recursos de manera efectiva y abordar los problemas más críticos primero.

Riesgo Asociado: RIE-2 (Conflictos Laborales y Gestión Interna)

RIE-2: Una mala toma de decisiones, falta de liderazgo efectivo o una cultura empresarial tóxica o descontento generalizado entre los empleados.

Riesgo Asociado: RIE-18 (Problemas de rendimiento/Escalabilidad)

RIE-18: A medida que más usuarios utilicen la aplicación, pueden surgir problemas de rendimiento, como ralentizaciones o bloqueos. Se debe monitorear y optimizar el rendimiento de la aplicación para garantizar una experiencia de usuario satisfactoria.

ICF-34 Registro de Soluciones Implementadas:

El registro documenta de manera clara las soluciones implementadas para cada error identificado. Se verifica la eficacia de las soluciones a través de pruebas internas, aunque algunos detalles podrían ser más detallados para una comprensión completa.

Posibles riesgos:

1. **Ineficacia de las Soluciones:** Existe el riesgo de que las soluciones implementadas no sean efectivas para abordar completamente los errores identificados. Esto podría comprometer la calidad general del software, generando posibles problemas de rendimiento, seguridad o compatibilidad que afectarían negativamente la experiencia del usuario.
2. **Falta de Pruebas Exhaustivas:** La realización de pruebas internas que no cubran todos los escenarios posibles podría resultar en soluciones no probadas adecuadamente. Este riesgo podría introducir nuevos problemas o no abordar completamente los errores existentes, afectando la estabilidad y confiabilidad del software.
3. **Documentación Insuficiente:** La falta de claridad o detalles suficientes en la documentación de las soluciones implementadas podría dificultar la comprensión para otros miembros del equipo. Esto podría afectar la colaboración y la mantenibilidad a largo plazo, ya que la falta de información detallada podría generar malentendidos y decisiones incorrectas durante futuras actualizaciones o mantenimientos.
4. **No Cumplimiento con Estándares o Requisitos:** Si las soluciones implementadas no cumplen con los estándares de codificación o los requisitos del proyecto, existe el riesgo de problemas de compatibilidad, seguridad o rendimiento. Esto podría comprometer la

integridad del sistema y generar dificultades en la adaptación a cambios futuros o integración con otros componentes.

5. **Vulnerabilidades de Seguridad:** La implementación de soluciones podría introducir inadvertidamente vulnerabilidades de seguridad, exponiendo al sistema a amenazas externas. Este riesgo podría comprometer la confidencialidad, integridad y disponibilidad de los datos, generando un impacto significativo en la seguridad general del sistema.
6. **Dependencia de Recursos Específicos:** Si las soluciones implementadas dependen de recursos específicos que no están disponibles en todas las configuraciones, se podría generar una falta de portabilidad. Este riesgo afectaría la compatibilidad y accesibilidad del software en diferentes entornos, limitando su aplicabilidad y alcance.
7. **Falta de Escalabilidad:** Si las soluciones implementadas no escalan adecuadamente con el crecimiento de la aplicación, existe el riesgo de problemas de rendimiento a largo plazo. Esto afectaría la experiencia del usuario, generando posibles ralentizaciones o bloqueos a medida que la aplicación se expande en complejidad y volumen de datos.

Riesgo Asociado: RIE-11 (Falta de calidad del producto o servicio)

RIE-11: Una mala toma de decisiones, falta de liderazgo efectivo o una cultura empresarial tóxica o descontento generalizado entre los empleados.

Riesgo Asociado: RIE-18 (Problemas de rendimiento/Escalabilidad)

RIE-18: A medida que más usuarios utilicen la aplicación, pueden surgir problemas de rendimiento, como ralentizaciones o bloqueos. Se debe monitorear y optimizar el rendimiento de la aplicación para garantizar una experiencia de usuario satisfactoria.

ICF-35 Documento de Características y Mejoras:

8. **Ambigüedad en los Requisitos:** Descripciones vagas o ambiguas de las características pueden llevar a malentendidos y a la implementación incorrecta de funcionalidades.
9. **Alcance Mal Definido:** Si el alcance del proyecto no se define claramente, es probable que haya desviaciones durante el desarrollo, lo que podría afectar los plazos y los recursos.

10. Cambios Constantes en los Requisitos: Cambios frecuentes en los requisitos pueden causar retrasos, aumento de costos y dificultades para mantener la coherencia en el desarrollo.

11. Expectativas No Gestionadas: Si las expectativas de los stakeholders no se gestionan adecuadamente, pueden surgir descontento y malentendidos sobre las capacidades y el rendimiento de la aplicación.

Riesgo Asociado: RIE-19 (Comunicación deficiente)

RIE-19: En casi todas las etapas del proceso de desarrollo del producto, es necesario comunicar las ideas entre los receptores de la información (los clientes) y los creadores de esa misma información (los proveedores).

Riesgo Asociado: RIE-11 (Falta de calidad del producto o servicio)

RIE-11: Una mala toma de decisiones, falta de liderazgo efectivo o una cultura empresarial tóxica o descontento generalizado entre los empleados.

ICF-36 Evaluación de Compatibilidad:

12. Inconsistencias en Versiones de Software: Diferencias significativas entre las versiones de software utilizadas para el desarrollo y las versiones en los entornos de prueba o producción pueden resultar en comportamientos inesperados.

13. Variaciones de Hardware: Diferencias en el hardware, como variaciones en la potencia de la CPU, la cantidad de memoria RAM o la capacidad de almacenamiento, pueden afectar el rendimiento y la estabilidad de la aplicación.

14. Problemas de Sistema Operativo: Incompatibilidades con sistemas operativos específicos pueden surgir, especialmente si la aplicación se desarrolla y prueba en un sistema operativo y se despliega en otro.

Riesgo Asociado: RIE-2 (Conflictos Laborales y Gestión Interna)

RIE-2: Una mala toma de decisiones, falta de liderazgo efectivo o una cultura empresarial tóxica o descontento generalizado entre los empleados.

Riesgo Asociado: RIE-11 (Falta de calidad del producto o servicio)

RIE-11: Una mala toma de decisiones, falta de liderazgo efectivo o una cultura empresarial tóxica o descontento generalizado entre los empleados.

ICF-37 Documentación de Pruebas Finales:

15. Omisión de Escenarios de Uso Importantes: No incluir casos de prueba que cubran situaciones críticas o escenarios de uso importantes puede llevar a la falta de cobertura y a la posible omisión de errores significativos.

16. Falta de Colaboración con Usuarios Finales: No involucrar a los usuarios finales en el proceso de prueba puede resultar en la falta de validación de la usabilidad y la funcionalidad real de la aplicación.

17. Cambios no documentados: Si se realizan cambios en la aplicación durante las pruebas finales y estos no se documentan adecuadamente, la documentación puede volverse obsoleta y no reflejar la realidad de la aplicación.

Riesgo Asociado: RIE-4 (Problema de licencias de las APIs)

RIE-4: Las APIs, fundamentales para la comunicación entre sistemas, pueden enfrentar diversos problemas. Cambios sin retrocompatibilidad y falta de documentación pueden causar dificultades en las actualizaciones.

Riesgo Asociado: RIE-15 (Cambios en la administración)

RIE-15: Las decisiones inconsistentes, los reajustes organizativos y la falta de compromiso pueden resultar en aumentos de costos y retrasos en el proyecto. Además, la comunicación y la confianza de los interesados pueden resentirse, y la resistencia al cambio puede dificultar la implementación efectiva de nuevas estrategias. Mitigar estos riesgos requiere una transición cuidadosa, comunicación clara y un enfoque en la continuidad y compromiso del equipo, así como la gestión proactiva del cambio y la participación de los interesados.

Notas y observaciones

1. Identificación y Priorización de Problemas:

Durante esta fase crítica, se destaca la eficacia del proceso de identificación y priorización de problemas, asegurando una atención inmediata a los aspectos más críticos. La revisión detallada de informes de errores y la clasificación según su criticidad permiten al equipo centrarse en resolver aquellos problemas que afecten significativamente la experiencia del usuario o la estabilidad del software. Sin embargo, para mejorar aún más este proceso, se sugiere un monitoreo constante para detectar posibles cambios inesperados en la prioridad durante el desarrollo. Este enfoque proactivo garantiza que el equipo pueda adaptarse ágilmente a las fluctuaciones en la gravedad de los problemas, manteniendo la flexibilidad necesaria para abordar los desafíos en constante evolución.

2. Corrección de Errores:

La fase de corrección de errores destaca la importancia de abordar proactivamente los problemas identificados, incluyendo pruebas internas para garantizar la eficacia de las soluciones. Se implementan soluciones efectivas para cada error identificado, y se verifica su eficacia mediante pruebas internas. Sin embargo, para optimizar este proceso, se recomienda una revisión continua

para evitar posibles retrasos en la identificación de errores críticos. Esta sugerencia apunta a mantener la agilidad del equipo, asegurando que los problemas críticos se identifiquen oportunamente, evitando así impactos negativos en la experiencia del usuario.

3. Adición de Nuevas Características y Mejoras:

En esta sección, se destaca la alineación con las necesidades del usuario en la incorporación de nuevas características y mejoras. Se evalúan solicitudes y sugerencias de los usuarios para identificar oportunidades de mejora, asegurando que la implementación de nuevas características enriquezca la funcionalidad de la aplicación y se adapte a las necesidades cambiantes de los usuarios. No obstante, para fortalecer aún más la fundamentación detrás de cada adición, se sugiere un mayor énfasis en justificar la inclusión de cada elemento y diversificar la retroalimentación de los usuarios. Esto busca garantizar que cada mejora no solo responda a las demandas inmediatas, sino que esté respaldada por un análisis sólido y una comprensión completa de las expectativas del usuario.

4. Garantía de Compatibilidad:

La evaluación de compatibilidad es descrita como completa, asegurando la adaptación a las últimas actualizaciones y estándares tecnológicos. Se verifica que la aplicación se ajuste correctamente a los cambios en el entorno tecnológico y se proyectan posibles ajustes adicionales. Aunque se reconoce este logro, se señalan oportunidades para ajustes adicionales que refuercen la compatibilidad total. Este enfoque proactivo subraya la importancia de prepararse para futuras evoluciones tecnológicas, asegurando que la aplicación no solo sea funcional en el presente, sino que también esté lista para los desafíos tecnológicos que puedan surgir.

5. Pruebas Finales y Documentación:

La sección de pruebas finales destaca la estabilidad general de la aplicación, aunque se sugiere una mayor claridad en algunos puntos de la documentación. Se llevan a cabo diversas pruebas críticas para garantizar la estabilidad y el rendimiento de la aplicación después de implementar correcciones y mejoras. La documentación se actualiza de manera efectiva para reflejar las últimas mejoras y proporcionar información relevante sobre las nuevas características. Sin embargo, para mejorar aún más la comprensión efectiva, se recomienda ajustes en la presentación de datos y una revisión exhaustiva. Este enfoque proactivo no solo beneficia al equipo de desarrollo internamente, sino que también facilita la comunicación con otros departamentos y partes interesadas externas, asegurando una interpretación clara de la calidad de la aplicación.

6. Informe de Errores Priorizados:

El informe de errores priorizados se destaca por su exhaustividad al detallar cada error identificado, proporcionando información clara sobre su impacto y prioridad. Este enfoque meticuloso garantiza que los problemas críticos para la experiencia del usuario se aborden con la máxima prioridad. La priorización precisa es esencial para asignar eficientemente recursos y resolver los desafíos que más impactan positivamente en la aplicación. La introducción de este informe sirve como una herramienta estratégica que no solo documenta los problemas, sino que también orienta la misión del equipo hacia la resolución efectiva y eficiente de los problemas identificados.

7. Registro de Soluciones Implementadas:

El registro de soluciones implementadas desglosa claramente las respuestas a cada error identificado y verifica la eficacia de las soluciones a través de pruebas internas. Aunque destaca la importancia de las soluciones implementadas, sugiere que algunos detalles podrían ser más explícitos para una comprensión completa. Este registro no solo actúa como un historial de

correcciones, sino que también sirve como una referencia valiosa para el equipo de desarrollo, proporcionando una visión histórica de los desafíos encontrados y las soluciones aplicadas.

8. Documento de Características y Mejoras:

El documento de características y mejoras juega un papel crucial al detallar nuevas adiciones a la aplicación. Destaca la alineación con las necesidades del usuario, aunque se sugiere un mayor énfasis en justificar la inclusión de cada elemento. Este documento no es simplemente un catálogo de cambios; es una expresión dinámica de la evolución continua de la aplicación. Su énfasis en la justificación sólida refuerza el compromiso con la excelencia y la relevancia en cada paso que la aplicación da hacia adelante.

9. Evaluación de Compatibilidad:

La evaluación de compatibilidad documenta exhaustivamente la adaptación de la aplicación a las últimas actualizaciones y estándares. Aunque se verifica la compatibilidad actual, se identifican áreas donde podrían considerarse ajustes adicionales. Este informe no solo garantiza que la aplicación funcione con las actualizaciones actuales, sino que también está preparada para innovaciones futuras. Es una guía estratégica que equilibra la adaptación a las tecnologías actuales con la preparación para los avances futuros.

10. Documentación de Pruebas Finales:

La documentación de pruebas finales refleja claramente los resultados de las pruebas y se actualiza de manera efectiva. A pesar de destacar la estabilidad general de la aplicación, se sugiere una mayor claridad en algunos puntos. Este documento no solo sirve como una herramienta de referencia interna, sino que también es una narrativa que cuenta la historia de la calidad de la

aplicación. Al revisar y optimizar la claridad en la documentación, se construye para el presente y se proporciona una comprensión clara de su calidad, asegurando una aplicación estable y una interpretación transparente de su rendimiento.

Asignación de recursos

| Rol | Horas-Hombre | Desempeñado por | Observaciones |
|-----------------------|--------------|-----------------------------|---|
| Analista/Investigador | 1 hrs | Marcos E. Solis Ceniceros | Análisis de requerimientos necesarios para el desarrollo del proyecto e investigación general que abre el panorama de desarrollo. |
| Programación | 3 hrs | Oswaldo A. Hernandez Campos | Desarrollo continuo de la aplicación, creacion de las interfaces de la aplicacion con sus funcionalidades |

| | | | |
|---------------------------------------|-------|-------------------------------|--|
| Investigador | 1hrs | Iván O. Vásquez Flores | Investigación general que abre el panorama de desarrollo. |
| Analista | 1hrs | Alejandro García Alférez | Investigación general que abre el panorama de desarrollo. |
| Equipo de desarrollo/ Investigador | 1 hrs | Carlos Francisco Babún Ravelo | Crear/programar las funciones del producto e investigación general que abre el panorama de desarrollo. |

Anexos

Anexo 1 (ICF-33): Informe de Errores Priorizados.

La creación del **Informe de Errores Priorizados** representa un componente esencial en nuestra estrategia de desarrollo de la aplicación. Este informe no solo documenta los problemas encontrados, sino que también establece un enfoque meticuloso para priorizarlos. La finalidad de este proceso es garantizar que nuestro equipo se centre en la resolución de los problemas que tienen el mayor impacto en la experiencia del usuario. A través de esta metodología, buscamos mejorar la calidad y confiabilidad de la aplicación de manera eficiente y efectiva.

Durante el desarrollo de la aplicación, hemos implementado un riguroso sistema de seguimiento de errores para identificar y abordar posibles problemas. El Informe de Errores Priorizados, es una representación detallada de los contratiempos encontrados y proporciona una guía clara para la asignación de recursos y la resolución efectiva de problemas. Cada error documentado incluye información sobre su naturaleza, contexto y, lo más importante, su impacto potencial en la experiencia del usuario.

La priorización de errores es un paso crucial en nuestro proceso. En este informe, cada problema se clasifica en función de su gravedad y su influencia en la usabilidad general de la aplicación. Nos aseguramos de que los problemas críticos, aquellos que podrían afectar significativamente la experiencia del usuario, se aborden con la máxima prioridad. Este enfoque nos permite concentrar nuestros esfuerzos en resolver los desafíos que más importan, garantizando así una mejora sustancial y continua de la calidad de la aplicación.

El **Informe de Errores Priorizados** no solo es un documento técnico, sino una herramienta estratégica que impulsa nuestra misión de ofrecer una aplicación sin problemas y altamente funcional. A través de este proceso, hemos establecido un marco sólido para abordar los desafíos de manera sistemática, mejorando la eficiencia de nuestro equipo y la satisfacción del usuario. Continuaremos utilizando esta metodología para mantener y elevar constantemente los estándares de nuestra aplicación.

Anexo 2 (ICF-34): Registro de Soluciones Implementadas.

El **Registro de Soluciones Implementadas** es un componente integral de nuestro enfoque en el desarrollo de la aplicación. Este registro no sólo detalla las soluciones aplicadas a errores específicos, sino que también proporciona una visión transparente de cómo abordamos y resolvemos los desafíos técnicos. En este proceso, destacamos la importancia de la verificación de la eficacia de las soluciones a través de pruebas internas, asegurando así que las correcciones implementadas sean efectivas y contribuyan a la mejora continua de nuestra aplicación.

Durante el ciclo de desarrollo de la aplicación, enfrentamos diversos desafíos que requieren soluciones efectivas y rápidas. El Registro de Soluciones Implementadas sirve como un documento detallado que no solo enumera las soluciones aplicadas a problemas específicos, sino que también valida la eficacia de estas soluciones mediante pruebas internas meticulosas.

Cada entrada en este registro presenta una descripción clara del problema inicial, la lógica detrás de la solución propuesta y detalles sobre la implementación real. Aunque se realiza una verificación exhaustiva de la eficacia de las soluciones mediante pruebas internas, reconocemos que algunos detalles pueden requerir una explicación más detallada para garantizar una comprensión completa.

El enfoque clave en este proceso es garantizar que cada solución implementada no solo resuelva el problema inmediato, sino que también contribuya a la estabilidad general y al rendimiento de la aplicación. Este registro actúa como una herramienta de referencia valiosa para el equipo de desarrollo, brindando una visión histórica de los desafíos encontrados y las soluciones aplicadas.

El **Registro de Soluciones Implementadas** representa no solamente un historial de correcciones, sino un testimonio de nuestro compromiso con la excelencia técnica. A través de este proceso, hemos creado un marco sólido para abordar problemas de manera sistemática y para garantizar que nuestras soluciones no solo sean rápidas sino también efectivas. Este registro continuará siendo una herramienta vital para nuestro equipo mientras trabajamos hacia la mejora constante de la calidad y el rendimiento de nuestra aplicación.

Anexo 3 (ICF-35): Documento de Características y Mejoras.

El **Documento de Características y Mejoras** no solo sirve como un catálogo de los avances más recientes en nuestra aplicación, sino que también es un reflejo de nuestra dedicación para ofrecer una experiencia del usuario que se ajuste de manera precisa a sus necesidades y expectativas. En este contexto, se enfatiza la importancia de justificar de manera clara y precisa la inclusión de cada

nueva característica o mejora, asegurando una toma de decisiones informada y alineada con las metas estratégicas del proyecto.

Este documento desglosa meticulosamente las nuevas características y mejoras implementadas en nuestra aplicación, brindando una visión detallada de cada elemento añadido. Nos esforzamos por alinear cada incorporación con las necesidades identificadas del usuario, asegurando que cada mejora contribuya significativamente a la experiencia general.

Cada entrada en este documento no solo describe la funcionalidad recién agregada, sino que también destaca la justificación detrás de su inclusión. Esta justificación no solo se centra en la mejora técnica, sino que se extiende a la alineación estratégica con los objetivos del proyecto y las expectativas del usuario.

Si bien reconocemos la importancia de resaltar la alineación con las necesidades del usuario, hemos identificado la oportunidad de otorgar un mayor énfasis en la justificación de cada elemento incluido. Este enfoque garantiza que cada característica y mejora no solo sea impulsada por la demanda del usuario, sino que también esté respaldada por un análisis sólido que respalde su inclusión.

Este documento, por lo tanto, no solo es un inventario de nuestras últimas innovaciones, sino una herramienta estratégica que guía nuestras decisiones de desarrollo. Actúa como un recurso vital para la toma de decisiones informada, asegurando que cada mejora tenga un propósito claro y contribuya al crecimiento sostenible de nuestra aplicación.

El **Documento de Características y Mejoras** no solo es un documento estático que detalla cambios; es una expresión dinámica de nuestra evolución continua. A medida que trabajamos para mejorar y expandir nuestra aplicación, este documento se mantiene como un faro que ilumina el camino hacia una experiencia del usuario más rica y satisfactoria. Su énfasis en la justificación sólida

refuerza nuestro compromiso con la excelencia y la relevancia en cada paso que damos hacia adelante.

Anexo 4 (ICF-36): Evaluación de Compatibilidad.

La **Evaluación de Compatibilidad** es crucial para asegurar que nuestra aplicación no simplemente se ajuste a los avances tecnológicos actuales, sino que esté preparada para futuras actualizaciones. Este informe exhaustivo refleja nuestro compromiso con la sostenibilidad y la interoperabilidad, aunque señala oportunidades para ajustes adicionales que refuercen la compatibilidad total de la aplicación.

Este informe aborda cada aspecto de la compatibilidad de nuestra aplicación con las últimas actualizaciones y estándares tecnológicos. Cada evaluación se realiza con precisión, asegurando que nuestra aplicación se ajuste correctamente a los cambios en el entorno tecnológico.

Cada sección documenta la compatibilidad actual y proyecta posibles ajustes adicionales. Reconocemos la necesidad de no simplemente satisfacer los estándares existentes, sino de anticipar y prepararse para las futuras evoluciones tecnológicas. El informe actúa como un mapa estratégico que guía nuestros esfuerzos para mantener una aplicación alineada con la vanguardia tecnológica.

Aunque resalta nuestros esfuerzos actuales para garantizar la compatibilidad, identifica áreas donde podrían considerarse ajustes adicionales. Estos ajustes no solo se centran en cumplir con los requisitos técnicos, sino en fortalecer la aplicación para una compatibilidad total. Este enfoque proactivo asegura que nuestra aplicación no simplemente funcione bien con las actualizaciones actuales, sino que esté preparada para las innovaciones futuras.

La **Evaluación de Compatibilidad** es un testimonio de nuestro compromiso con la longevidad y la relevancia continua de nuestra aplicación. Este documento es una guía estratégica que equilibra la adaptación a las tecnologías actuales con la preparación para los avances futuros. A medida que

avanzamos, consideraremos cuidadosamente los ajustes sugeridos para garantizar que nuestra aplicación sea compatible y de igual manera, líder en la escena tecnológica en constante cambio.

Anexo 5 (ICF-37): Documentación de Pruebas Finales.

La **Documentación de Pruebas Finales** se erige como un compendio crucial que ilustra la robustez y fiabilidad de nuestra aplicación al llegar al término del proceso de desarrollo. Este informe no únicamente evidencia nuestro compromiso con la calidad, sino que señala áreas donde la claridad puede ser perfeccionada para lograr una comprensión más efectiva.

Este expediente presenta con total claridad los desenlaces de las pruebas finales, enfatizando la fortaleza y estabilidad general de nuestra aplicación. Cada segmento se actualiza de manera efectiva para reflejar con precisión el estado final de la aplicación, garantizando que todos los aspectos sean debidamente registrados.

A pesar de destacar la estabilidad, también se reconocen oportunidades para perfeccionar la claridad en algunos puntos de la documentación. Esto implica ajustes en la presentación de datos, así como una revisión exhaustiva para asegurar que inclusive los detalles más técnicos sean accesibles para todos los miembros del equipo.

La documentación no funciona únicamente como una herramienta de referencia interna; es una narrativa que cuenta la historia de la calidad de nuestra aplicación. Al centrarnos en mejorar la claridad, no favorecemos únicamente a nuestro equipo de desarrollo, sino que también facilitamos la comunicación con otros departamentos y partes interesadas externas.

La **Documentación de Pruebas Finales** constituye nuestra última afirmación acerca de la confiabilidad de la aplicación. Al revisar y optimizar la claridad en la documentación, estamos construyendo no solo para el presente, también proporcionamos una comprensión clara de su calidad. Este enfoque proactivo asegura que, al final del día, entregamos una aplicación estable y ofrecemos una comprensión clara de su calidad.

Glosario de términos

Términos generales

1. CPU: Unidad central de procesamiento, componente de hardware que realiza cálculos.
2. Datos Visuales: Información obtenida a través de imágenes o grabaciones.
3. Capacidad de Respuesta: La velocidad y eficacia con la que la aplicación reacciona a las acciones del usuario.
4. Eficiencia Operativa: La capacidad de realizar tareas de manera eficaz y sin desperdicio de recursos.
5. Manejo de Errores y Excepciones: El manejo de errores y excepciones es una parte crucial de la programación que se ocupa de cómo gestionar situaciones inesperadas o errores que pueden surgir durante la ejecución de un programa.
6. Escalabilidad: Capacidad de la aplicación para crecer y adaptarse a mayores demandas.
7. Optimizaciones: Mejoras realizadas para aumentar el rendimiento y eficiencia de la aplicación.
8. Interacciones de Usuarios: Maneras en que los usuarios se relacionan con la aplicación.
9. Experiencia de Usuario: La calidad de la interacción del usuario con la aplicación.
10. Evaluación de Resultados: Análisis de los datos de prueba para obtener conclusiones.
11. Notación gráfica: La notación es la parte gráfica que se ve en los modelos y representa la sintaxis del lenguaje de modelado. Por ejemplo, la notación del diagrama de clases define cómo se representan los elementos y conceptos como son: una clase, una asociación y una multiplicidad.
12. SCRUM: Marco de trabajo ágil para la gestión de proyectos de desarrollo de software. Se centra en la flexibilidad y adaptabilidad a medida que evolucionan los requisitos del proyecto.
13. Depuración (Debugging): Proceso de identificación y corrección de errores en el código fuente de un programa para garantizar su correcto funcionamiento.

14. Integración Continua: Práctica de fusionar automáticamente los cambios realizados por varios desarrolladores en un repositorio compartido varias veces al día. Busca detectar y resolver problemas de integración de manera temprana.
15. Despliegue Continuo: Automatización del proceso de liberación de software, permitiendo la entrega de versiones nuevas de una aplicación de manera rápida y eficiente.
16. API (Interfaz de Programación de Aplicaciones): Conjunto de reglas y protocolos que permiten que diferentes aplicaciones se comuniquen entre sí. Facilita la integración de funciones de una aplicación en otra.
17. Framework (Marco de Trabajo): Estructura conceptual y técnica que proporciona una base para el desarrollo de software. Incluye bibliotecas, herramientas y estándares que facilitan la creación y mantenimiento de aplicaciones.
18. Git: Sistema de control de versiones distribuido que permite el seguimiento de cambios en el código fuente durante el desarrollo de software. Facilita la colaboración entre desarrolladores.

Significado de los elementos de la notación gráfica



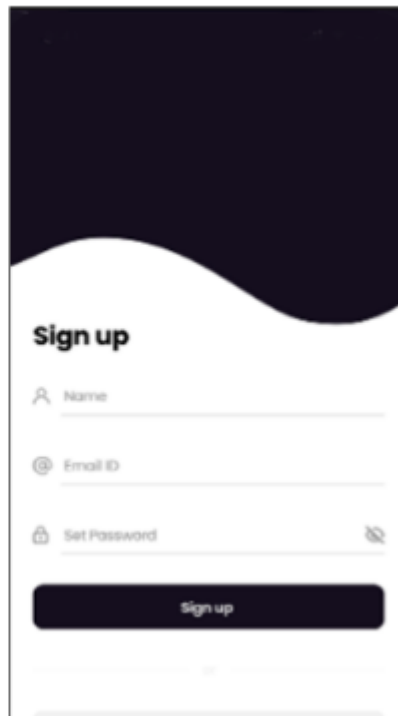
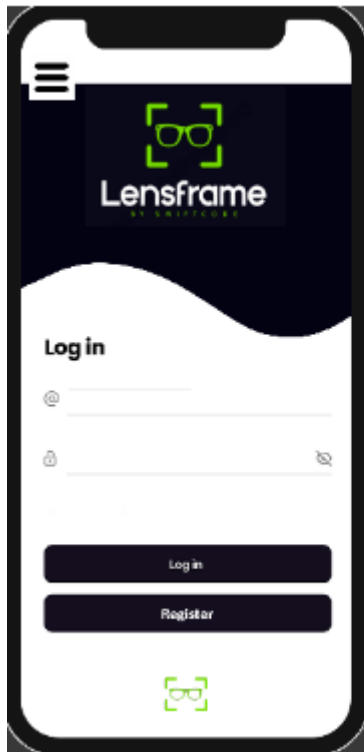
Se ajustó la interfaz en la cual los iconos han sido cambiados por unas categorías ya establecidas en las cuales el usuario elegirá cuál será su mejor opción.

- se añade el apartado de usuario de login/register
- También un menú de búsqueda.



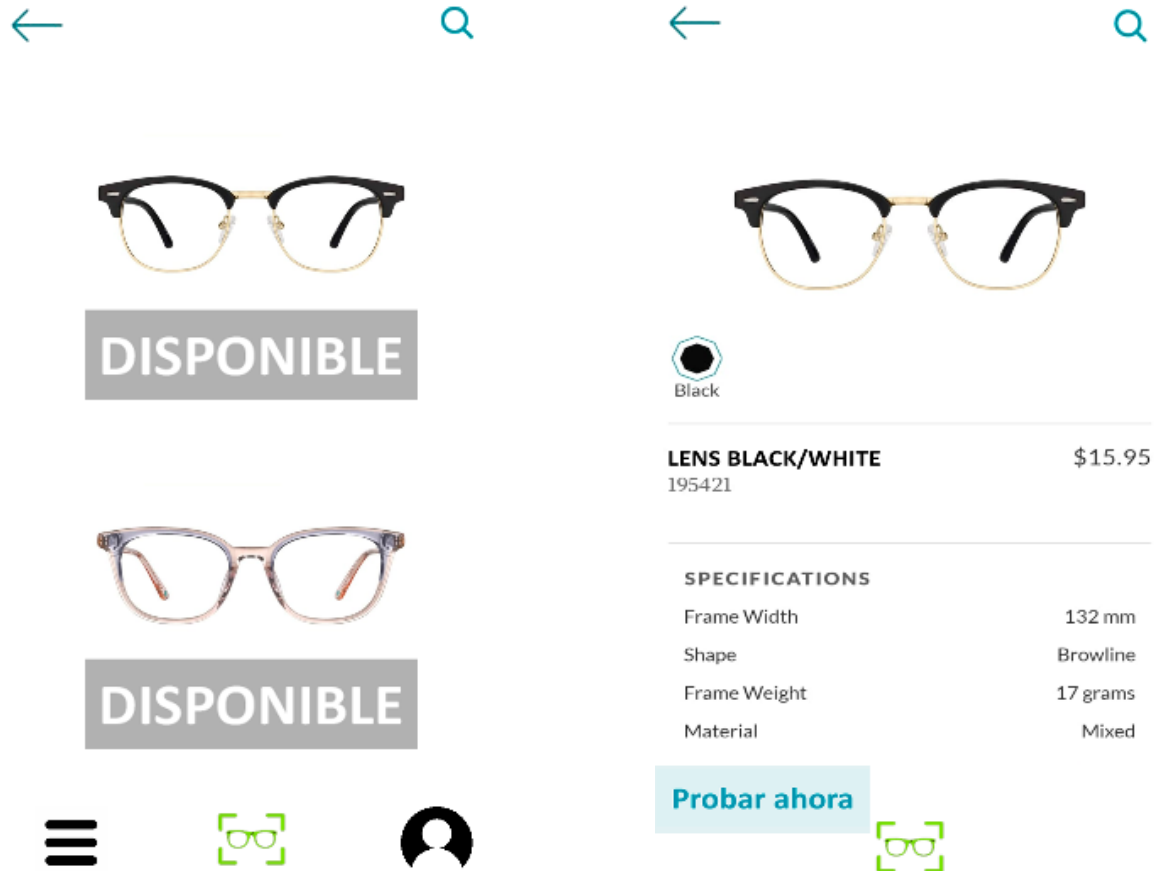
Apartado de menú de búsqueda:

En la cual el usuario tiene la opción de scrollear todo el catálogo disponible en la aplicación, sin necesidad de tener que ir específicamente a la categoría y buscarlo dentro de ella.



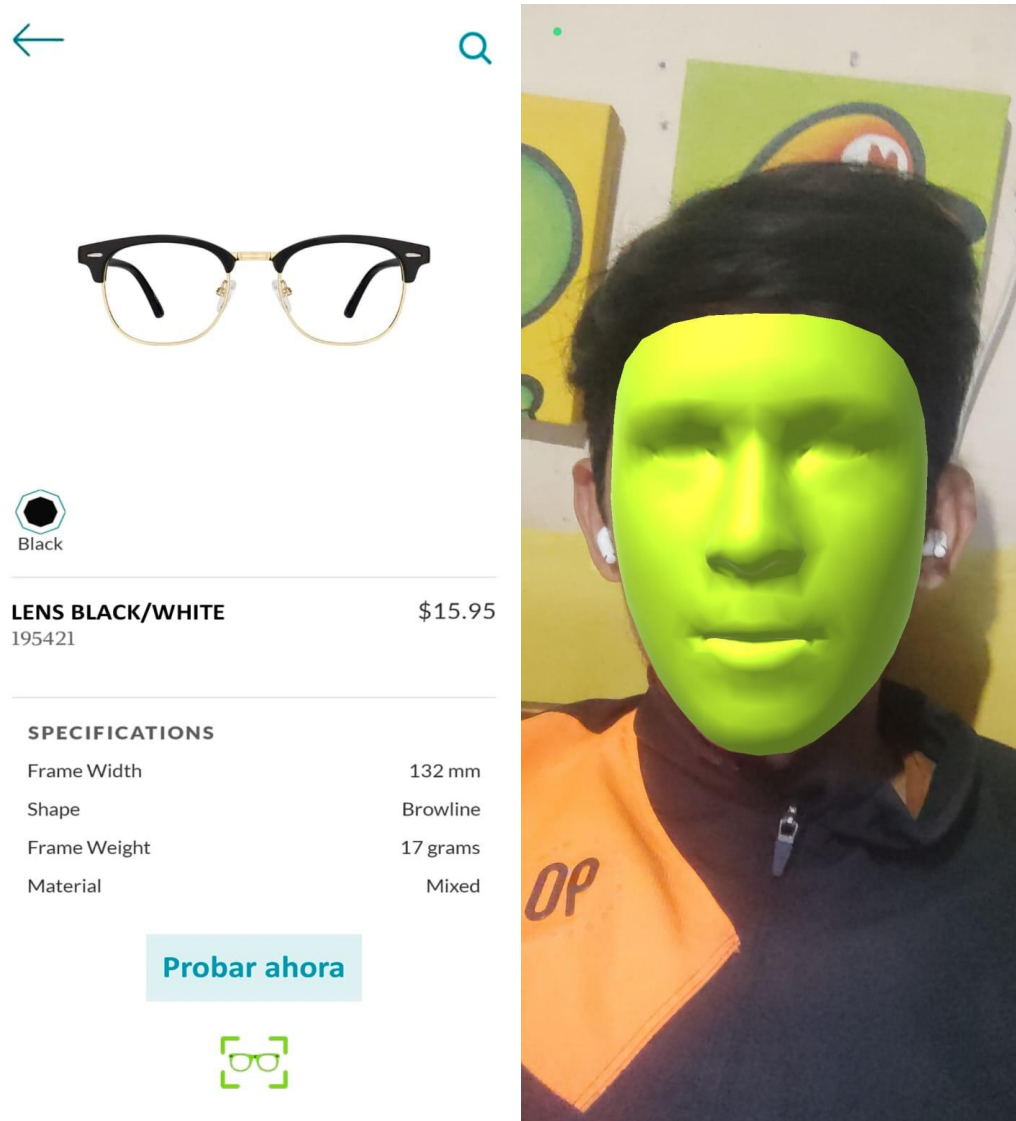
Apartado para la cuenta del usuario:

Tiene un menú con una interfaz de login y otra que incluye el registro de la cuenta, estas se almacenarán en una base de datos con la finalidad de que el usuario tenga un mejor control de sus productos comprados o un seguimiento.



Como tal se muestra la categoría y se muestran los modelos de dicha categoría, el usuario elegirá cual le interesa más, una vez seleccionado el modelo deseado, este se abrirá con un menú el cual consta tanto de la imagen del lente seleccionado, como el color disponible, a su vez se mostrará el nombre, código y precio de dicho producto, acompañado de sus respectivas especificaciones y un botón de probar los lentes para que el usuario califique si es el producto deseado.

Como tal se siguen ajustando ajustes en la interfaz para lograr tener una interfaz limpia, cálida, intuitiva y sobre todo cómoda para cualquier usuario.



Por ultimo añadido temporal, se crea la funcion en la cual el usuario podra probar los lentes de manera virtual utilizando la funcion trackface que es la que se encarga del mapeo de la persona con los lentes, en este momento se logra la implementación del tracking pero en el transcurso de la semana se planea la implementación completa y lista para la experiencia principal de la aplicacion.

Informe de Errores Priorizados

| Aspecto Esencial | Descripción |
|-----------------------------------|--|
| Calidad de la Captura de Imágenes | Asegurarse de que las imágenes utilizadas para el reconocimiento facial sean de alta calidad y bien iluminadas y evaluar la resolución y la nitidez de las imágenes para garantizar una precisión adecuada en el reconocimiento. |
| Manejo de Errores y Excepciones | Implementar un manejo adecuado de errores para proporcionar mensajes claros y orientación al usuario en caso de problemas |
| Pruebas de Usuario | Realizar pruebas de usuario para obtener retroalimentación sobre la usabilidad y la experiencia general con la aplicación. |

Registro de Soluciones Implementadas

| Aspecto Esencial | Descripción |
|--|--|
| Requisitos del Sistema | Asegurarse de que la aplicación cumple con los requisitos mínimos del sistema, tanto en términos de hardware como de software. Esto puede incluir especificaciones de la cámara, capacidad de procesamiento y compatibilidad con las versiones de Unity y el entorno de ejecución. |
| Implementación del Reconocimiento Facial | Seleccionar una biblioteca o API de reconocimiento facial que se adapte a nuestras necesidades, integrar estas herramientas en Unity y seguir las mejores prácticas de implementación. |
| Manejo de Datos del Usuario | Implementar un sistema seguro para almacenar y gestionar los datos faciales de los |

| | |
|--|--|
| | usuarios, cumpliendo con las regulaciones de privacidad y protección de datos. |
|--|--|

Documento de Características y Mejoras

| Aspecto Esencial | Descripción |
|-------------------------------------|---|
| Algoritmos de Reconocimiento Facial | Especificar los algoritmos de reconocimiento facial que se planean implementar. Se podrían incluir técnicas de aprendizaje profundo, reconocimiento de patrones, etc. |
| Mejoras y Actualizaciones Futuras | Planificar posibles mejoras y actualizaciones para la aplicación en el futuro. Esto podría incluir la integración de nuevas características o la adaptación a avances tecnológicos. |
| Pruebas y Evaluación | Definir un plan de pruebas que abarque la funcionalidad, rendimiento y seguridad de la aplicación. Especificar los criterios de éxito y las métricas de rendimiento. |

Evaluación de Compatibilidad

| Aspecto Esencial | Descripción |
|-------------------------------------|---|
| Licencias y Políticas de Privacidad | Revisar las licencias asociadas con las bibliotecas de reconocimiento facial y cualquier SDK utilizado. Asegúrate de cumplir con los términos de uso y considera las implicaciones de privacidad. |
| Documentación y Soporte Técnico | Contar con documentación detallada de la integración de la biblioteca de reconocimiento facial. Verificar la disponibilidad de soporte técnico en caso de problemas durante el desarrollo. |

Documentación de Pruebas Finales

| Aspecto Esencial | Descripción |
|-------------------------------------|--|
| Documentación de Pruebas de Usuario | Proporcionar un conjunto de casos de prueba detallados para que los evaluadores realicen pruebas exhaustivas. Incluir escenarios específicos de reconocimiento facial y verificar la respuesta de la aplicación. |
| Optimización de Recursos | Optimizar el uso de recursos del sistema para garantizar un rendimiento eficiente. Esto incluye la gestión de memoria, la carga de texturas y cualquier otro recurso utilizado por la aplicación. |
| Documentación del Código | Proporcionar documentación completa del código fuente, incluyendo comentarios detallados y descripciones de las posibles funciones clave. |

| Atributo | NIVEL | VALOR |
|------------|-------|-------|
| DATA | N | 0.94 |
| CPLX | VL | 0.7 |
| TIME | N | 1 |
| STOR | L | 0.87 |
| VIRT | L | 0.87 |
| TURN | N | 1 |
| ACAP | N | 1 |
| AEXP | L | 1.13 |
| PCAP | L | 1.13 |
| VEXP | N | 1 |
| LEXP | L | 1.07 |
| MODP | L | 1.1 |
| TOOL | N | 1 |
| SCED | N | 1 |
| M = 0.7485 | | |