**Guía2. Desarrollo Proyecto APT**

**Asignatura Capstone**

|  |
| --- |
| **1. Resumen avance Proyecto APT** |
| A continuación, encontrarás distintos campos que deberás completar con la información solicitada. |

|  |  |
| --- | --- |
| Resumen de avance proyecto APT | Durante el desarrollo del proyecto LibreBit, se ha avanzado conforme a lo planificado en la Fase 1, cumpliendo la mayoría de los objetivos específicos y realizando ajustes menores en el alcance y la metodología.  Se completó la configuración del entorno de trabajo con Flutter, Firebase y Hive, junto con el diseño de la arquitectura y del prototipo visual en Figma. Asimismo, se implementó el sistema de autenticación y la generación de billeteras con frase semilla, además de la sincronización de datos entre Firebase y Hive.  También se desarrollaron las funciones de recepción de pagos en Bitcoin y USDT, alcanzando uno de los objetivos principales del proyecto. Actualmente, el proyecto se encuentra en la etapa de pruebas técnicas y optimización, donde se han corregido errores de integración y mejorado la seguridad mediante encriptación AES y validación de direcciones.  Como ajuste, se incorporó la creación de un PIN de seguridad que permite acceder a la billetera de forma rápida y protegida sin necesidad de ingresar la frase semilla en cada inicio. Este cambio busca mejorar la experiencia del usuario y aumentar la protección de la información, manteniendo la practicidad y accesibilidad que caracterizan al proyecto.  En síntesis, el proyecto avanza de forma estable, con las funciones principales operativas y las pruebas finales en curso antes de la entrega definitiva. |
| Objetivos | Durante la Fase 1 del proyecto LibreBit, los objetivos principales estuvieron orientados al desarrollo del frontend de la aplicación y a la implementación de las primeras funcionalidades esenciales para la creación y gestión de billeteras digitales.  En primer lugar, se desarrolló la interfaz gráfica principal utilizando el framework Flutter, priorizando un diseño moderno, responsivo y centrado en la experiencia del usuario. Esta etapa incluyó la creación de los componentes visuales y de navegación base que estructuran la aplicación, tales como la barra inferior, los botones de acción y la organización de las pantallas principales.  Uno de los objetivos más relevantes fue la implementación del sistema de autenticación y registro que permite la creación de billeteras mediante frase semilla. Esta funcionalidad fue diseñada para garantizar un acceso seguro y sin custodia, alineado con los principios de privacidad y control total del usuario sobre sus fondos.  Asimismo, se desarrollaron las pantallas funcionales que componen el núcleo de la aplicación:   * Pantalla “Recibir”: encargada de generar y mostrar direcciones de recepción en Bitcoin y USDT, junto con la opción de copiar o visualizar el código QR. * Pantalla “Portafolio”: destinada a mostrar el balance general del usuario y los valores actuales de las criptomonedas disponibles en la billetera. * Pantalla “Mercado”: diseñada para visualizar precios actualizados, variaciones de valor y oportunidades de intercambio entre activos. * Pantalla “Perfil”: implementada para que el usuario pueda visualizar y editar su nombre de usuario, foto de perfil y datos sincronizados con la nube mediante Firebase.   Cada uno de estos módulos fue integrado de forma coherente en la estructura del frontend, logrando un entorno funcional, visualmente uniforme y conectado con la base de datos local y remota.  En síntesis, los objetivos de la Fase 1 se centraron en construir la base visual, funcional y estructural de LibreBit, asegurando la correcta interacción entre las pantallas, la autenticación de usuarios y la generación de billeteras, sentando las bases para las etapas siguientes de desarrollo e integración de redes de pago. |
| Metodología | El proyecto LibreBit se desarrolla utilizando la metodología ágil Kanban, adaptada al hecho de que el desarrollo está a cargo de un único integrante. Este enfoque resulta especialmente adecuado, ya que permite visualizar las tareas, priorizarlas y gestionar el flujo de trabajo de manera flexible, evitando sobrecargas y asegurando un avance constante.  El tablero Kanban se organiza en columnas que representan los estados de las actividades: Pendiente, En Progreso, En Revisión y Completado. Cada actividad específica, como la generación de la billetera con frase semilla, la integración con Firebase, el consumo de APIs de blockchain o la creación de pantallas en Flutter, se representa mediante una tarjeta en el tablero. Esto permite mantener un control visual del progreso y detectar posibles cuellos de botella en el proceso.  La aplicación de Kanban en LibreBit permite gestionar eficientemente las tareas técnicas y de documentación, equilibrar el ritmo de trabajo y facilitar la entrega incremental de resultados. Al tratarse de un proyecto individual, este enfoque ágil otorga autonomía en la priorización de actividades y garantiza que cada funcionalidad crítica —como seguridad, usabilidad e integración multi-cripto— sea completada oportunamente.  De esta forma, Kanban se consolida como un marco práctico que impulsa la organización personal, la productividad y la calidad del producto final. |
| Evidencias de avance | Durante la Fase 1 del proyecto LibreBit, se completó el desarrollo del frontend de la aplicación, implementado en Flutter con un diseño moderno, responsivo y centrado en la experiencia del usuario.  Se desarrollaron las pantallas principales —Home, Recibir, Portafolio, Mercado y Perfil—, integradas con la base de datos local Hive y el servicio en la nube Firebase Firestore para garantizar la persistencia y sincronización de la información.  También se implementó el sistema de autenticación y registro mediante frase semilla, permitiendo la creación y recuperación de billeteras de forma segura. Además, se añadió un PIN de acceso cifrado para mejorar la protección y facilitar el ingreso rápido del usuario.  Como resultado, la aplicación cuenta con una interfaz funcional y estable, con las principales características operativas y listas para la fase de pruebas técnicas. |

|  |
| --- |
| **2. Monitoreo del Plan de Trabajo** |
| Examina cuidadosamente tu plan de trabajo, enfocándote especialmente en la columna de estado de avance y ajustes. |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Plan de Trabajo | | | | | | | |
| Competencia o unidades de competencias | Actividades | Recursos | Duración de la actividad | Responsable[[1]](#footnote-1) | Observaciones | Estado de avance | Ajustes |
| Desarrollo, adaptación e integración de sistemas computacionales | Configuración inicial del entorno de trabajo (Flutter, Firebase, Hive). | Computador personal, conexión a internet, software libre. | 2 semanas. | Mirko González. | Se presentó compatibilidad entre versiones de dependencias, resuelto mediante actualización manual. | Completado. | Ninguno. |
| Desarrollo de soluciones tecnológicas | Diseño de arquitectura y creación de la interfaz base en Figma (Home, Enviar, Recibir y Perfil). | Figma, Flutter, PC personal.. | 2 semanas. | Mirko González. | Dificultad inicial para definir el flujo de navegación entre pantallas, solucionado mediante esquema visual previo. | Completado. | Se ajustaron colores y jerarquía visual para mejorar la usabilidad. |
| Integración de tecnologías de información | Implementación de autenticación, generación de billeteras (BIP39) y sincronización Firebase–Hive. | Flutter/Dart, librería bip39, Firebase, Hive. | 3 semanas. | Mirko González. | Hubo dificultades de sincronización de datos entre Firebase y Hive; resuelto con pruebas iterativas. | Completado. | Se optimizó la lógica de almacenamiento local y respaldo en nube. |
| Seguridad de sistemas computacionales | Implementación de PIN de seguridad para acceso rápido y protegido a la billetera. | Flutter Secure Storage, librerías de cifrado AES. | 2 semanas. | Mirko González. | Requiere pruebas de persistencia del PIN tras reinicio de app. | Completado. | Se incorporó cifrado adicional y validación en memoria temporal. |
| Desarrollo e integración de sistemas de pago | Desarrollo de la funcionalidad de envío mediante Lightning Network. | Flutter, API Lightning, conexión a nodo público. | 4 semanas. | Mirko González. | El mayor reto ha sido la integración del backend de Lightning y la validación de pagos instantáneos. | En espera. | Se configuró un entorno de pruebas para Lightning y validaciones fuera de línea. |

|  |
| --- |
| **3. Ajustes a partir del monitoreo** |
| Profundiza en las observaciones de tu plan de trabajo. Analiza las actividades planificadas y señala qué aspectos facilitaron u obstaculizaron la ejecución del plan. Plantea cómo abordaste y/o abordarás los obstáculos. Por último, señala los ajustes que realizaste al plan de trabajo a partir de este análisis. |

|  |
| --- |
| Factores que han facilitado y/o dificultado el desarrollo de mi plan de trabajo:  El desarrollo del proyecto LibreBit ha avanzado de manera constante gracias a una planificación clara basada en la metodología Kanban, que ha permitido organizar las tareas y mantener un flujo de trabajo ordenado. El uso de herramientas como Flutter, Firebase y Hive ha facilitado la integración de los componentes y la sincronización de datos entre el almacenamiento local y la nube.  Entre las principales dificultades se destacaron las diferencias de compatibilidad entre versiones de librerías y dependencias, las cuales fueron resueltas mediante actualizaciones manuales y pruebas iterativas.  Asimismo, la incorporación de un PIN de acceso cifrado representó un reto técnico menor, superado con el uso de almacenamiento seguro y validaciones adicionales.  En conjunto, estos factores han permitido mantener el proyecto dentro del cronograma establecido, garantizando estabilidad y un avance progresivo hacia los objetivos definidos. |

|  |
| --- |
| Actividades ajustadas o eliminadas: Durante el desarrollo del proyecto LibreBit, se realizaron algunos ajustes menores al plan de trabajo con el objetivo de mejorar la eficiencia y optimizar la experiencia del usuario.  En primer lugar, se incorporó la creación de un PIN de seguridad para permitir un acceso rápido a la billetera sin necesidad de ingresar la frase semilla en cada inicio, reforzando tanto la usabilidad como la protección de los datos personales.  Asimismo, se decidió concentrar los esfuerzos en la integración con la red Bitcoin, dejando la incorporación de otras redes para fases posteriores. Esta medida permitió optimizar los tiempos de desarrollo y garantizar la estabilidad general del sistema.  Estos ajustes no afectaron el avance global del proyecto, ya que la planificación inicial y la aplicación de una metodología ágil han permitido mantener el trabajo dentro de los plazos establecidos. |

|  |
| --- |
| Actividades que no has iniciado o están retrasadas:La única actividad que aún no ha sido iniciada corresponde al desarrollo de la funcionalidad de envío de pagos, la cual se encuentra planificada para la siguiente fase del proyecto.  Esta etapa requiere una configuración específica y pruebas controladas para asegurar la estabilidad y seguridad del sistema antes de su implementación final.  El trabajo se ejecutará una vez finalizadas las pruebas y validaciones de las funciones ya desarrolladas, priorizando la continuidad lógica del proceso de integración y manteniendo el cumplimiento del cronograma general del proyecto. |

1. En caso de que el Proyecto APT sea grupal, en esta columna deben indicar el nombre de los responsables de cada tarea o actividad. Esto posteriormente permitirá diferenciar la evaluación por cada integrante. [↑](#footnote-ref-1)