

LOS GAMERS



MATRIZ RACI

Proyecto: **CRIPTONET**

Fecha: **17 de may. de 24**

Cliente: **Ninguno**

Patrocinador: **Ninguno**

Gerente del Proyecto: **Arturo Badillo**

		Edison	Omar	Jesus	Bryan	Nigell	Santiago	Arturo	Kevin
ID	Descripción	Análisis		Diseño		Construcción		PMO	
1.	Criponet								
1.1.	Planificación del Proyecto			C	C	I			
1.1.1.	Definir el alcance, objetivos y requisitos del proyecto CriptoNet	R	R					A	A
1.1.2.	Crear la estructura de desglose de trabajo (WBS) y el cronograma detallado	R	R						
1.1.3.	Estudio de factibilidad	R	R						
1.1.4.	Acta constitutiva	R	R						
1.1.5.	Plan de gestión del proyecto	R	R						
1.1.7.	Desarrollar un plan de gestión de riesgos	R	R						
1.2.	Seguimiento y Control			C	C	I		A	A
1.2.1.	Realizar reuniones semanales de estado con el equipo de proyecto							R	R
1.2.2.	Monitorear el progreso del proyecto y compararlo con el cronograma establecido							R	R
1.2.3.	Identificar y mitigar riesgos a medida que surjan durante el desarrollo							R	
1.2.4.	Gestionar y controlar las solicitudes de cambios y los cambios aprobados								
1.3.	Gestión de Comunicaciones			I		C	C	A	A
1.3.1.	Establecer canales de comunicación eficientes entre los miembros del equipo y los stakeholders							R	
1.3.3.	Gestionar y mantener actualizada la documentación del proyecto								R
2.	Análisis y Diseño								
2.1.	Definición de Requisitos			I				C	C
2.1.1.	Requisitos funcionales	A	A						
2.1.1.1.	Especificación caso de uso "Iniciar sesión"					R	R		
2.1.1.1.	Especificación caso de uso "Registrarse"					R	R		
2.1.1.1.	Especificación caso de uso "Administrar publicaciones"					R	R		
2.1.1.1.	Especificación caso de uso "Administrar criptomonedas"					R	R		
2.1.1.1.	Especificación caso de uso "Administrar recompensas"					R	R		
2.1.2.	Requisitos no funcionales	A	A						
2.1.2.1.	Requisitos de rendimiento			R	R				
2.1.2.2.	Requisitos de seguridad			R	R				
2.1.2.3.	Requisitos de escalabilidad			R	R				
2.1.2.4.	Requisitos de usabilidad			R	R				
2.1.3.	Priorizar y validar requisitos	A	A						
2.1.3.1.	Realizar talleres con stakeholders					R			
2.1.3.2.	Revisar y aprobar requisitos							R	
2.2.	Diseño de Arquitectura					C	C	I	I
2.2.1.	Diseñar arquitectura de software			A	A				
2.2.1.1.	Especificación diagrama de clases "Iniciar sesión"	R	R						
2.2.1.2.	Especificación diagrama de clases "Registrarse"	R	R						
2.2.1.3.	Especificación diagrama de clases "Administrar publicaciones"	R	R						
2.2.1.4.	Especificación diagrama de clases "Administrar criptomonedas"	R	R						
2.2.1.5.	Especificación diagrama de clases "Administrar recompensas"	R	R						
2.2.1.6.	Definir patrones de diseño			R	R				
2.2.1.7.	Diseñar arquitectura de capas			R	R				
2.2.2.	Diseñar arquitectura de criptomonedas			A	A				
2.2.2.1.	Seleccionar protocolos de criptomonedas	R							
2.2.2.2.	Diseñar integración con protocolos					R			
2.2.2.3.	Diseñar módulos de transacciones					R			
2.2.3.	Definir protocolos y tecnologías a utilizar				A		A		
2.2.3.1.	Seleccionar tecnologías frontend	R							
2.2.3.2.	Seleccionar tecnologías backend	R							
2.2.3.3.	Seleccionar proveedores de nube	R							
2.3.	Diseño de Bases de Datos	I	I					C	C
2.3.1.	Diseñar modelo de datos				A				
2.3.1.1.	Identificar entidades y relaciones						R		
2.3.1.2.	Normalizar modelo de datos						R		
2.3.2.	Diseñar esquemas de bases de datos				A				
2.3.2.1.	Diseñar esquemas relacionales			R	R				
2.3.2.2.	Diseñar esquemas no relacionales			R	R				
2.3.3.	Definir estructuras de datos y relaciones			A					
2.3.3.1.	Definir tablas y campos			R	R				
2.3.3.2.	Definir índices y claves primarias			R	R				
2.3.3.3.	Definir relaciones y llaves foráneas			R	R				
2.4.	Diseño de Interfaces de Usuario	C	I						
2.4.1.	Diseñar wireframes y mockups			R	A		R		
2.4.1.1.	Diseñar wireframes de baja fidelidad			R			R		
2.4.1.2.	Diseñar mockups de alta fidelidad				R	R			
2.4.2.	Definir estándares de diseño				A				
2.4.2.1.	Definir guías de estilo			R					
2.4.2.2.	Definir patrones de diseño UI/UX			R		R			
2.4.3.	Diseñar flujos de navegación			A					
2.4.3.1.	Mapear flujos de usuario			R		R			
2.4.3.2.	Diseñar diagramas de navegación			R		R			
3.	Desarrollo								
3.1.	Frontend				C			I	I
3.1.1.	Implementar interfaz de usuario	A	A						
3.1.1.1.	Implementación caso de uso "Iniciar sesión"	R		R			R		
3.1.1.2.	Implementación caso de uso "Registrarse"	R		R			R		
3.1.1.3.	Implementación caso de uso "Administrar publicaciones"	R		R			R		
3.1.1.4.	Implementación caso de uso "Administrar criptomonedas"	R		R			R		
3.1.1.5.	Implementación caso de uso "Administrar recompensas"	R		R			R		
3.1.2.	Integrar con frameworks y librerías			A		R	R		
3.2.	Backend	C	C					I	I
3.2.1.	Implementar lógica de negocio							A	A
3.2.1.1.	Desarrollar servicios y controladores			R		R	R		
3.2.1.2.	Implementar reglas de negocio			R		R	R		
3.2.2.	Integración con Base de Datos					A			
3.2.2.1.	Desarrollar capa de acceso a datos	R					R		
3.2.2.2.	Implementar consultas y operaciones	R					R		

3.2.3.	Integración con Criptomonedas			A					
3.2.3.1.	Implementar protocolos de criptomonedas				R	R			
3.2.3.2.	Desarrollar módulos de transacciones				R	R			
3.2.4.	Implementación caso de uso "Iniciar sesión"						R	R	A
3.2.5.	Implementación caso de uso "Registrarse"						R	R	A
3.2.6.	Implementación caso de uso "Administrar publicaciones"						R	R	A
3.2.7.	Implementación caso de uso "Administrar criptomonedas"						R	R	A
3.2.8.	Implementación caso de uso "Administrar recompensas"						R	R	A
3.3.	Desarrollo de Billetera Digital	C	C					I	I
3.3.1.	Implementar funcionalidades de billetera					R			
3.3.2.	Integrar con protocolos de criptomonedas					R			
3.3.3.	Implementar medidas de seguridad					R			
4.	Integración y Pruebas		C					I	I
4.1.	Integración de Componentes	A							
4.1.1.	Integrar frontend y backend			R	R	R	R		
4.1.2.	Integrar con bases de datos			R	R	R	R		
4.1.3.	Integrar con servicios externos - Configurar conexiones con servicios de criptomonedas					R	R		
4.3.	Pruebas unitarias				A	A			
4.3.2.	Prueba unitaria caso de uso "Iniciar sesión"							R	R
4.3.3.	Prueba unitaria caso de uso "Registrarse"							R	R
4.3.4.	Prueba unitaria caso de uso "Administrar publicaciones"							R	R
4.3.5.	Prueba unitaria caso de uso "Administrar criptomonedas"							R	R
4.3.6.	Prueba unitaria caso de uso "Administrar recompensas"							R	R
4.2.	Pruebas de Integración				A	A			
4.2.1.	Desarrollar casos de prueba de integración							R	R
4.2.2.	Ejecutar pruebas de integración							R	R
4.2.3.	Corregir defectos identificados							R	R
4.3.	Pruebas de Aceptación				A				
4.3.1.	Desarrollar casos de prueba de aceptación			R					
4.3.2.	Ejecutar pruebas de aceptación			R					
4.3.3.	Corregir defectos identificados			R					
4.4.	Pruebas de Rendimiento				A				
4.4.1.	Definir escenarios y métricas de rendimiento	R	R						
4.4.2.	Ejecutar pruebas de rendimiento	R	R						
4.4.3.	Analizar resultados y optimizar	R	R						
4.5.	Pruebas de Seguridad					A			
4.5.1.	Definir alcance y plan de pruebas de seguridad	R	R						
4.5.2.	Ejecutar pruebas de seguridad	R	R						
4.5.3.	Analizar resultados y corregir vulnerabilidades	R	R						
5.	Implementación		C	C	I			I	
5.1.	Preparación del Entorno de Producción					A			
5.1.1.	Aprovisionar infraestructura de producción						R		R
5.1.2.	Configurar entorno de producción						R		R
5.1.3.	Implementar monitoreo y alertas						R		R
5.2.	Migración de Datos					A			
5.2.1.	Planificar migración de datos	R							R
5.2.2.	Ejecutar migración de datos	R					R		
5.2.3.	Validar datos migrados	R							
5.3.	Despliegue de Aplicación					A			
5.3.1.	Preparar artefactos de despliegue	R					R		
5.3.2.	Ejecutar despliegue						R		
5.3.3.	Verificar despliegue	R					R		
5.4.	Pruebas de Aceptación en Producción					A			
5.4.1.	Ejecutar pruebas de aceptación	R					R		R
5.4.3.	Obtener aprobación final	R					R		R