



Rúbrica/KPI

Integrantes:

David Mardones

Arturo Rojas

Maria Morales

Rúbrica / KPI – Fase 2 (Sprint 2: Fase de Diseño)

| % Avance | Funcionalidades / Hitos | Justificación del Porcentaje | Evidencia/Recursos | Estado esperado |
| --- | --- | --- | --- | --- |
| 10% | Diagramas de flujo (navegación principal y registro de huevos) | Marca el 10% porque establece la base del sistema, define los procesos principales y flujos de navegación. Requiere tiempo de planificación y análisis de casos, con complejidad baja, pero fundamental para guiar el desarrollo posterior. | Trello, Figma | Completado |
| 20% | Prototipos interactivos (login, menú, registro, historial) | Marca el 20% ya que permite visualizar la interfaz y experiencia del usuario antes de la codificación. Requiere tiempo de diseño y validación de prototipos, con complejidad media, porque implica decidir la lógica de interacción y consistencia visual. | Figma, Trello | Completado |
| 30% | Modelo de Base de Datos (Firebase Firestore) | Marca el 30% porque define la estructura de datos que soporta todas las funcionalidades. Requiere investigación sobre manejo de colecciones, documentos y relaciones en Firebase, lo que aumenta la complejidad y consume tiempo significativo. | Firebase, Trello, documentación | Completado |
| 40% | Vistas 4+1 (lógica, desarrollo, procesos, física, escenarios) | Marca el 40% ya que integra arquitectura, lógica y procesos, conectando diagramas, prototipos y modelo de datos. Requiere tiempo de planificación detallada y complejidad media-alta para asegurar que las vistas cumplan con la lógica funcional del sistema. | Figma, Visual Studio Code, Trello | Completado |
| 50% | Configuración de Firebase (Auth, Firestore, Storage) + Desarrollo de login y autenticación + CRUD de usuarios + CRUD de huevos | Marca el 50% porque implementa la funcionalidad central del proyecto: autenticación, gestión de usuarios y CRUD de huevos. Es crítico, ya que sin esta base las demás funcionalidades no pueden implementarse. Requiere mucho tiempo y alta complejidad técnica, ya que se necesitó investigar y configurar correctamente Firebase y la integración con Angular/Ionic. | Trello, GitHub, Firebase, Visual Studio Code | Completado |
| 60% | Visualización de historial | Marca el 60% porque permite consultar y seguir los datos del CRUD de huevos, agregando funcionalidad básica de seguimiento. Requiere tiempo de desarrollo adicional y complejidad media, ya que implica consultas, filtrado y presentación correcta de datos. | Trello, GitHub, prototipos interactivos | No iniciado |
| 70% | Control de inventario / calidad de huevos | Marca el 70% porque integra operaciones sobre los datos, como inventario y control de calidad. Requiere tiempo considerable y alta complejidad, pues implica validaciones, reglas de negocio y manipulación de datos existentes. | Trello, Firebase, documentación técnica | No iniciado |
| 80% | Roles y permisos de usuario | Marca el 80% porque asegura seguridad y control de acceso, vital para proteger la información del sistema. Requiere tiempo de desarrollo y complejidad técnica alta, configurando roles, permisos y restricciones en Firebase y la aplicación. | Trello, Firebase, documentación de seguridad | No iniciado |
| 90% | Notificaciones y alertas | Marca el 90% porque agrega valor funcional avanzado, permitiendo avisos automáticos sobre eventos críticos en inventario o calidad. Requiere tiempo de implementación y complejidad media-alta, integrando la lógica de eventos y pruebas. | Trello, Firebase, código de alertas y pruebas | No iniciando |
| 100% | Validación de datos, pruebas unitarias e integración, documentación técnica parcial y revisión general / Sprint Review | Marca el 100% porque finaliza la Fase 2, asegurando que todas las funcionalidades estén integradas, validadas y documentadas, listas para la entrega. Requiere tiempo de pruebas y ajustes finales, con complejidad alta para garantizar calidad y cumplimiento de objetivos. | GitHub, Trello, pruebas de integración, documentación técnica | No iniciado |

Este análisis o rúbrica fue desarrollado exclusivamente para la Fase 2 (Diseño y Desarrollo) del proyecto. En esta etapa se evaluaron detalladamente cada una de las tareas planificadas, considerando el tiempo de dedicación, el nivel de complejidad técnica y el esfuerzo requerido para su implementación.

A partir de estos criterios, se estableció un porcentaje de avance representativo para cada hito, reflejando de manera más precisa el progreso real del proyecto en función del trabajo ejecutado y no solo de la cantidad de tareas completadas.

Tabla de Avance General del Proyecto

| % Avance | Fase/Funcionalidades/Hitos | Justificación del Porcentaje | Evidencia/ Recursos | Estado esperado |
| --- | --- | --- | --- | --- |
| 10% | Sprint 1 – Planificación inicial (Sprint Planning, Análisis del caso, Mapa Mental, Mapa de Actores) | Marca el 10% porque corresponde a la definición del proyecto, actores y objetivos. Es una etapa esencial para comprender el contexto, aunque de baja complejidad técnica. | Trello, documentación(Drive), Canvas | Completado |
| 20% | Visión del Proyecto + 4 Pilares, Squad y Responsabilidades, Épicas, Historias de Usuario, Product Backlog, Sprint Backlog, Reunión retrospectiva e Impediment Log | Llega al 20% porque consolida la planificación, estructura del equipo, y definición de backlog. Complejidad baja-media, pero clave para orientar el desarrollo. | Trello, documentación de planificación(Drive) | Completado |
| 30% | Diagramas de flujo (navegación principal y registro de huevos) | Marca el 30% ya que representa el inicio de la fase de diseño, estableciendo la lógica de navegación principal. Requiere análisis y modelamiento funcional. | Trello, StarUML | Completado |
| 40% | Prototipos interactivos (login, menú, registro, historial) | Llega al 40% porque define visualmente la aplicación. Es de complejidad media y permite validar la interfaz y experiencia de usuario antes de programar. | Figma, Trello | Completado |
| 45% | Modelo de Base de Datos (Firebase Firestore) | Sube a 45% ya que define la estructura técnica que soporta todas las funcionalidades. Implica análisis técnico y comprensión de la arquitectura NoSQL. | Firebase,Trello,Drive | Completado |
| 50% | Vistas 4+1 (lógica, desarrollo, procesos, física, escenarios) | Representa el cierre del diseño y arquitectura. Conecta los componentes de negocio, aplicación y sistema. Requiere coordinación y comprensión técnica. | StarUML,Trello,Drive | Completado |
| 58% | Configuración de Firebase (Auth, Firestore, Storage), desarrollo de login/autenticación y CRUD de usuarios y huevos | Marca el 58% ya que consolida la funcionalidad principal del sistema. Alta complejidad técnica: configuración, conexión e integración con Firebase y Angular/Ionic. | Github,Trello,Visual Studio Code | Completado |
| 65% | Visualización de historial | Permite consultar datos del CRUD, con filtrado y orden. Complejidad media, requiere consultas dinámicas y control de interfaz. | GitHub,Visual Studio Code | No iniciado |
| 70% | Control de inventario / calidad de huevos | Incrementa el nivel de procesamiento de datos, agregando validaciones y lógica de negocio. Complejidad alta. | Firebase,Trello,documentación | No iniciado |
| 80% | Roles y permisos de usuario | Fundamental para la seguridad y control de accesos. Complejidad alta, ya que requiere configurar políticas y restricciones en Firebase y el frontend. | Firebase, Trello, Drive | No iniciado |
| 90% | Notificaciones y alertas | Añade valor avanzado al sistema, gestionando eventos automáticos y avisos críticos. Complejidad media-alta. | Visual Studio Code, Firebase, Trello, Pruebas | No iniciado |
| 95% | Validación de datos y reglas de negocio, pruebas unitarias e integración | Casi cierre del desarrollo: garantiza estabilidad, confiabilidad y cumplimiento de los requerimientos. Complejidad alta. | Pruebas internas, Github | No iniciado |
| 100% | Sprint 3 – Presentación final, documentación técnica, revisión y entrega formal | Marca el 100% del proyecto. Incluye manual de usuario, revisión general, retrospectiva final y presentación. Complejidad administrativa y de documentación. | Documentación final(Drive), presentación, Trello | No iniciado |

El presente análisis se elaboró considerando todas las fases del proyecto, desde la planificación inicial hasta la presentación final. Cada tarea fue evaluada en función del tiempo requerido, nivel de complejidad técnica y esfuerzo necesario, lo que permitió asignar un porcentaje de avance realista a cada hito, reflejando el progreso alcanzado hasta el momento.

* **Fase 1 -Planificación (10-20%)**

Incluye Sprint Planning, análisis del caso, mapas, backlog, squad y responsabilidades. Baja complejidad, pero esencial para establecer la base del proyecto

* **Fase 2 - Diseño y Desarrollo (30-58%)**

Comprende diagramas de flujo, prototipos, modelo de base de datos, vistas 4+1, configuración de Firebase, login y CRUD de usuarios y huevos. Complejidad media-alta a alta; constituye la funcionalidad central del sistema.

***Funcionalidades pendientes:***

* Visualización de historial (65%)
* Control de inventario / calidad de huevos (70%)
* Roles y permisos de usuario (80%)
* Notificaciones y alertas (90%)
* Validación de datos y reglas de negocio, pruebas e integración (95%)
* **Fase 3 - Presentación final**

El cierre del proyecto incluye la presentación, documentación final y entrega formal, correspondiente al **100% del avance**. Esta fase tiene complejidad administrativa y de documentación.

Este análisis refleja que, aunque más de la mitad de las tareas están completadas en número, el avance real ponderado por complejidad aún indica que queda cerca del 42% del trabajo, principalmente en funcionalidad críticas, validaciones pruebas y documentación final

* Fase 1(Planificación): Completada 20%
* Fase 2(Diseño y Desarrollo):Completada hasta Crud de Huevos 58%
* Fase 3(Presentación): No Iniciada
* Avance total del proyecto: 58%, considerando dificultades y esfuerzo.