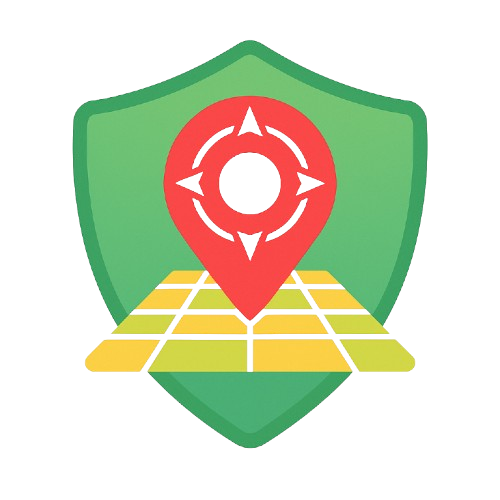
***"Año de la recuperación y consolidación de la economía peruana"***

**FACULTAD DE INGENIERÍA**

**TEMA:**

**TP2.E8 Entrega de sprint 02**

**Curso:**

**Taller de proyectos 2**

**NRC: 17185**

**Docente:**

**Americo Estrada Sanchez**

**Alumno:**

* Joan Nicole Cordova Bustamante
* Mijhael Amilkar Mejia Ballona
* Deyvid Brayan Vargaya Coaquira

**CUSCO -  PERÚ**

**2025**

Sprint 02 - WaykiSafe

**Fecha:** 01/10/2025

**Versión:** 1.0

# 1. Introducción

Este documento detalla el Sprint Backlog del equipo de desarrollo para el primer Sprint de Waykisafe. Se incluyen las historias de usuario seleccionadas del Product Backlog, sus tareas específicas, los responsables y la estimación de esfuerzo en horas.

# 2. Criterios de estimación de esfuerzo

Para estimar el esfuerzo de cada tarea, se utilizó la técnica Planning Poker, donde el el desarrollador discutió cada tarea y asignó una cantidad de horas basada en:

● Complejidad técnica

● Dependencias con otras tareas

● Experiencia previa en tareas similares

● Posibles riesgos o bloqueos

Cada tarea se estimó en consenso utilizando la escala de horas-persona

# 3. Historias de usuario seleccionadas para el Sprint Número 02

### Historia de Usuario 2.1: Pantalla de bienvenida personalizada

**Descripción:** Como turista, quiero visualizar una pantalla de bienvenida con mi nombre, para sentir que la aplicación está personalizada.

**Tareas:**

| **ID** | **Tarea** | **Responsable** | **Estimación (horas)** | **Estado** |
| --- | --- | --- | --- | --- |
| 2.1.1 | Diseñar la interfaz de bienvenida | UX/UI Designer | 4 | Pendiente |
| 2.1.2 | Implementar mensaje dinámico con nombre del usuario | Frontend Dev | 5 | Pendiente |
| 2.1.3 | Conectar datos de Firebase para personalización | Backend Dev | 6 | Pendiente |
| 2.1.4 | Pruebas de personalización y rendimiento | QA Engineer | 5 | Pendiente |

### Historia de Usuario 2.2: Validación de cuenta segura

**Descripción:** Como turista, quiero que la aplicación valide mi cuenta de manera segura, para proteger mi información personal.

| **ID** | **Tarea** | **Responsable** | **Estimación (Horas)** | **Estado** |
| --- | --- | --- | --- | --- |
| 2.2.1 | Configurar autenticación segura (tokens JWT/Firebase) | Backend Dev | 8 | Pendiente |
| 2.2.2 | Implementar cifrado de contraseñas en base de datos | Backend Engineer | 6 | Pendiente |
| 2.2.3 | Validar sesión activa en segundo plano | Frontend Dev | 5 | Pendiente |
| 2.2.4 | Pruebas de seguridad (penetration test básico) | QA Engineer | 7 | Pendiente |

**Tareas:**

**4. Evidencias de Código**

A continuación, se incluirán fragmentos del código implementado para las historias de usuario.

**4.1 Historia de Usuario 2.1**

**Archivo**: app/src/main/java/com/example/waykisafe/Bienvenido.java

| **FirebaseAuth - usuario actual (líneas 29-46)**  super.onCreate(savedInstanceState);  setContentView(R.layout.activity\_bienvenido);  txtWelcome = findViewById(R.id.txtWelcome);  btnNext = findViewById(R.id.btnInfo);  mAuth = FirebaseAuth.getInstance();  db = FirebaseFirestore.getInstance();  FirebaseUser currentUser = mAuth.getCurrentUser();  if (currentUser != null) {  String uid = currentUser.getUid();  db.collection("usuarios").document(uid).get()  .addOnSuccessListener(documentSnapshot -> {  if (documentSnapshot.exists()) {  String nombre = documentSnapshot.getString("nombre");  String apellido = documentSnapshot.getString("apellido");  **Firestore - obtener documento usuario (líneas 30-55)**    txtWelcome = findViewById(R.id.txtWelcome);  btnNext = findViewById(R.id.btnInfo);  mAuth = FirebaseAuth.getInstance();  db = FirebaseFirestore.getInstance();  FirebaseUser currentUser = mAuth.getCurrentUser();  if (currentUser != null) {  String uid = currentUser.getUid();  db.collection("usuarios").document(uid).get()  .addOnSuccessListener(documentSnapshot -> {  if (documentSnapshot.exists()) {  String nombre = documentSnapshot.getString("nombre");  String apellido = documentSnapshot.getString("apellido");  if (nombre == null) nombre = "";  if (apellido == null) apellido = "";  String saludo = "¡Hola, " + nombre + " " + apellido + "!";  txtWelcome.setText(saludo);  } else {  txtWelcome.setText("¡Hola, Usuario!");  }  **UI - saludo personalizado setText (líneas 46-64)**  if (documentSnapshot.exists()) {  String nombre = documentSnapshot.getString("nombre");  String apellido = documentSnapshot.getString("apellido");  if (nombre == null) nombre = "";  if (apellido == null) apellido = "";  String saludo = "¡Hola, " + nombre + " " + apellido + "!";  txtWelcome.setText(saludo);  } else {  txtWelcome.setText("¡Hola, Usuario!");  }  })  .addOnFailureListener(e -> {  Log.e("Bienvenido", "Error al obtener datos", e);  txtWelcome.setText("¡Hola, Usuario!");  });  } else {  txtWelcome.setText("¡Hola, Usuario!");  }  btnNext.setOnClickListener(v -> {  Intent intent = new Intent(Bienvenido.this, MapsActivity.class);  startActivity(intent);  }); |
| --- |

**4.2 Historia de Usuario**

|  |
| --- |

**5. Burndown Chart**

# 6. Análisis del Sprint 7. Conclusiones