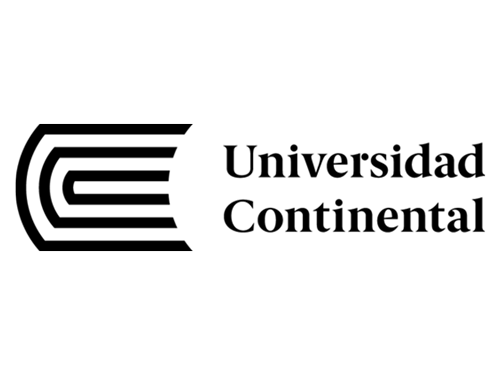
***"Año de la recuperación y consolidación de la economía peruana"***



**FACULTAD DE INGENIERÍA**

**TEMA:**

**TP2.E8 Entrega de sprint 03**

**Curso:**

**Taller de proyectos 2**

**NRC: 17185**

**Docente:**

**Americo Estrada Sanchez**

**Alumno:**

* [**CHRISTIAN CASTRO USTO**](mailto:73024009@continental.edu.pe)

**CUSCO -  PERÚ**

**2025**

Sprint 03 - SLEEPAPP

**Fecha:** 14/04/2025

**Autor: Christian Castro Usto**

**Versión:** 1.0

Sprint 03 – SleepApp

# 1. Introducción

Durante el Sprint 03 se trabajará en el desarrollo de funcionalidades clave que mejorarán la experiencia de uso de la aplicación SleepApp. En esta etapa se priorizará la **configuración y gestión avanzada de alarmas**, así como el **acceso intuitivo al menú de navegación**, permitiendo a los usuarios personalizar sus horarios de alerta de acuerdo con sus rutinas, y desplazarse fácilmente entre las distintas secciones de la aplicación. Se abordarán tareas como la selección de hora para alarmas, personalización de tonos, validación de funcionamiento offline, así como pruebas unitarias y de integración. Asimismo, se desarrollará el diseño e implementación del menú de navegación, incluyendo animaciones, configuración de rutas internas y validación de sesión para asegurar que solo los usuarios autenticados puedan acceder a las funcionalidades internas.

# 3. Lineamientos de Diseño

Durante el **Sprint 02**, se desarrollará una versión más completa de la aplicación móvil **SleepApp**, enfocada en incorporar funcionalidades relacionadas con la **rutina diaria del usuario** y la **configuración de alarmas personalizadas**. El objetivo de este sprint será permitir que los usuarios gestionen su planificación diaria (horas de estudio, trabajo, recreación y descanso), y que la aplicación pueda utilizar esta información para futuras recomendaciones sobre el sueño:

* Registro y gestión de rutina diaria del usuario
* Edición y modificación de la rutina registrada
* Almacenamiento local de datos para permitir el uso sin conexión
* Diseño de interfaces para rutina y alarma

Estas funcionalidades permitirán que en los siguientes sprints se integre el módulo de análisis de patrones de sueño, alarmas inteligentes, y recomendaciones personalizadas, basadas en los datos que el usuario haya proporcionado a través de estas rutinas.

# 2. Objetivo del Sprint

Se implementarán las funcionalidades de configuración y gestión de alarmas personalizadas, permitiendo a los usuarios seleccionar la hora de la alarma y personalizar los tonos de notificación. Asimismo, se desarrollará el funcionamiento sin conexión mediante almacenamiento local, y se diseñará e integrará un menú de navegación funcional que estará disponible solo para usuarios autenticados.

# 3. Historias de Usuario y Tareas

* Historia de Usuario 3.1: Configuración y gestión de alarma.

Descripción: Como usuario, quiero configurar alarmas personalizadas según mi rutina, para recibir en el momento adecuado.

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Tarea | Responsable | Horas |
| 3.1.2 | Implementar selección de hora de alarma | Frontend Dev | 6 |
| 3.1.3 | Permitir personalización de tonos de alarmas | Backend Dev | 6 |
| 3.1.4 | Implementar funcionamiento sin conexión | Backend Engineer | 5 |
| 3.1.5 | Pruebas unitarias y de integración | QA Enginner | 6 |

* Historia de Usuario 4.1: Acceder al menú de Navegación

Descripción: Como usuario, quiero acceder fácilmente al menú de navegación de la aplicación para moverme entre las diferentes funcionalidades sin dificultad.

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Tarea | Responsable | Horas |
| 4.1.1 | Diseñar la interfaz del menú de navegación | UX/UI Designer | 4 |
| 4.1.2 | Agregar animaciones y transiciones para una mejor UX | Frontend Dev | 4 |
| 4.1.3 | Configurar rutas y vinculación de cada sección | Backend Dev | 4 |
| 4.1.4 | Implementar validación de sesión (el menú solo es visible si el usuario está autenticado) | Backend Engineer | 6 |
| 2.1.5 | Pruebas de navegación y usabilidad | QA Enginner | 6 |

# 4. Evidencias de Código

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

# 4.1 Historia de Usuario 03

* Implementación selección de hora de alarma

<LinearLayout  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical"  
 android:padding="16dp"  
 android:background="@drawable/fondo">  
  
  
 <!-- Selector de la hora para la alarma -->  
 <TimePicker  
 android:id="@+id/timePickerAlarm"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:timePickerMode="clock"  
 android:layout\_gravity="center"/>  
  
 <!-- Texto que muestra la hora sugerida para acostarse -->  
 <TextView  
 android:id="@+id/tvRecommendedSleepTime"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Hora sugerida para acostarse"  
 android:layout\_gravity="center"  
 android:padding="10dp"/>  
  
 <!-- Texto que muestra la hora para dormir máxima -->  
 <TextView  
 android:id="@+id/tvMaxSleepTime"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Hora para dormir máxima"  
 android:layout\_gravity="center"  
 android:padding="10dp"/>  
  
 <!-- Botón para seleccionar el tono de alarma -->  
 <Button  
 android:id="@+id/btnSelectTone"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Seleccionar Tono de Alarma"  
 android:layout\_gravity="center"/>  
  
 <!-- Botón para confirmar la alarma -->  
 <Button  
 android:id="@+id/btnSetAlarm"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Confirmar Alarma"  
 android:layout\_gravity="center"/>  
 <Button  
 android:id="@+id/btnCancelarAlarma"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:backgroundTint="#F44336"  
 android:text="Cancelar"  
 android:layout\_gravity="center"/>  
  
</LinearLayout>

* Permitir personalización de tonos de alarmas

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

* Implementar funcionamiento sin conexión
* // Configurar la alarma al hacer clic en el botón  
   btnSetAlarm.setOnClickListener(new View.OnClickListener() {  
   @Override  
   public void onClick(View v) {  
   int hour = timePickerAlarm.getHour();  
   int minute = timePickerAlarm.getMinute();  
    
   if (selectedToneUri != null) {  
   // Guardar la alarma en la base de datos con el tono seleccionado  
   Calendar calendar = Calendar.*getInstance*();  
   calendar.set(Calendar.*HOUR\_OF\_DAY*, hour);  
   calendar.set(Calendar.*MINUTE*, minute);  
   calendar.set(Calendar.*SECOND*, 0);  
    
   // Programar la alarma  
   Intent intent = new Intent(RegistrarAlarma.this, AlarmReceiver.class);  
   intent.putExtra("ALARM\_TONE", selectedToneUri.toString());  
    
   PendingIntent pendingIntent = PendingIntent.*getBroadcast*(  
   RegistrarAlarma.this, 0, intent, PendingIntent.*FLAG\_UPDATE\_CURRENT* | PendingIntent.*FLAG\_IMMUTABLE*);  
    
   AlarmManager alarmManager = (AlarmManager) getSystemService(*ALARM\_SERVICE*);  
   if (alarmManager != null) {  
   alarmManager.setExact(AlarmManager.*RTC\_WAKEUP*, calendar.getTimeInMillis(), pendingIntent);  
   Toast.*makeText*(RegistrarAlarma.this, "Alarma configurada para " + hour + ":" + minute, Toast.*LENGTH\_SHORT*).show();  
    
   // Enviar Intent a estadisticas  
   Intent updateIntent = new Intent(RegistrarAlarma.this, MenuPrincipal.class);  
   updateIntent.putExtra("UPDATE\_GRAPH", true);  
   startActivity(updateIntent); // O usa un método diferente si deseas regresar a la actividad actual  
   }  
   } else {  
   Toast.*makeText*(RegistrarAlarma.this, "Por favor, selecciona un tono de alarma", Toast.*LENGTH\_SHORT*).show();  
   }  
   }  
   });  
   btnCancelarAlarma.setOnClickListener(new View.OnClickListener() {  
   @Override  
   public void onClick(View view) {  
   //Regresar al menu principal  
   Intent Regresar = new Intent(RegistrarAlarma.this,MenuPrincipal.class);  
   startActivity(Regresar);  
   }  
   });  
    
    
  }
* Pruebas unitarias y de integración

Captura de pantalla de computadora

Descripción generada automáticamente

Pantalla de juego de computadora

Descripción generada automáticamente con confianza media

# 4.2 Historia de Usuario 04

* Diseñar la interfaz del menú de navegación
* <?xml version="1.0" encoding="utf-8"?>  
  <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  
   xmlns:app="http://schemas.android.com/apk/res-auto"  
   xmlns:tools="http://schemas.android.com/tools"  
   android:id="@+id/main"  
   android:layout\_width="match\_parent"  
   android:layout\_height="match\_parent"  
   android:background="@drawable/fondo"  
   tools:context=".MenuPrincipal">  
    
    
   <ImageButton  
   android:id="@+id/btnSalir"  
   android:layout\_width="wrap\_content"  
   android:layout\_height="wrap\_content"  
   android:background="#00FFFFFF"  
   app:layout\_constraintBottom\_toBottomOf="parent"  
   app:layout\_constraintEnd\_toEndOf="parent"  
   app:layout\_constraintHorizontal\_bias="0.847"  
   app:layout\_constraintStart\_toStartOf="parent"  
   app:layout\_constraintTop\_toTopOf="parent"  
   app:layout\_constraintVertical\_bias="0.076"  
   app:srcCompat="@drawable/iconosalida" />  
    
   <ImageButton  
   android:id="@+id/perfil"  
   android:layout\_width="60dp"  
   android:layout\_height="65dp"  
   android:background="#00FFFFFF"  
   app:layout\_constraintBottom\_toBottomOf="parent"  
   app:layout\_constraintEnd\_toEndOf="parent"  
   app:layout\_constraintHorizontal\_bias="0.133"  
   app:layout\_constraintStart\_toStartOf="parent"  
   app:layout\_constraintTop\_toTopOf="parent"  
   app:layout\_constraintVertical\_bias="0.067"  
   app:srcCompat="@drawable/usuario" />  
    
   <TextView  
   android:id="@+id/Bienvenido"  
   android:layout\_width="247dp"  
   android:layout\_height="40dp"  
   android:text="BIENVENIDO &quot;CHRISTIAN&quot;"  
   android:textSize="20sp"  
   android:textStyle="bold"  
   app:layout\_constraintBottom\_toBottomOf="parent"  
   app:layout\_constraintEnd\_toEndOf="parent"  
   app:layout\_constraintStart\_toStartOf="parent"  
   app:layout\_constraintTop\_toTopOf="parent"  
   app:layout\_constraintVertical\_bias="0.205" />  
    
   <CalendarView  
   android:id="@+id/calendarView"  
   android:layout\_width="317dp"  
   android:layout\_height="274dp"  
   app:layout\_constraintBottom\_toBottomOf="parent"  
   app:layout\_constraintEnd\_toEndOf="parent"  
   app:layout\_constraintStart\_toStartOf="parent"  
   app:layout\_constraintTop\_toTopOf="parent"  
   app:layout\_constraintVertical\_bias="0.452" />  
    
   <TextClock  
   android:id="@+id/textClock"  
   android:layout\_width="170dp"  
   android:layout\_height="51dp"  
   android:textSize="34sp"  
   app:layout\_constraintBottom\_toBottomOf="parent"  
   app:layout\_constraintEnd\_toEndOf="parent"  
   app:layout\_constraintHorizontal\_bias="0.568"  
   app:layout\_constraintStart\_toStartOf="parent"  
   app:layout\_constraintTop\_toTopOf="parent"  
   app:layout\_constraintVertical\_bias="0.747" />  
    
   <ImageButton  
   android:id="@+id/btnRutina"  
   android:layout\_width="91dp"  
   android:layout\_height="70dp"  
   android:layout\_marginStart="28dp"  
   android:background="#00FFFFFF"  
   app:layout\_constraintBottom\_toBottomOf="parent"  
   app:layout\_constraintEnd\_toStartOf="@+id/textView3"  
   app:layout\_constraintHorizontal\_bias="0.315"  
   app:layout\_constraintStart\_toStartOf="parent"  
   app:layout\_constraintTop\_toTopOf="parent"  
   app:layout\_constraintVertical\_bias="0.888"  
   app:srcCompat="@drawable/rutina" />  
    
   <ImageButton  
   android:id="@+id/btnEstadistica"  
   android:layout\_width="95dp"  
   android:layout\_height="76dp"  
   android:background="#00FFFFFF"  
   app:layout\_constraintBottom\_toBottomOf="parent"  
   app:layout\_constraintEnd\_toEndOf="parent"  
   app:layout\_constraintHorizontal\_bias="0.693"  
   app:layout\_constraintStart\_toStartOf="parent"  
   app:layout\_constraintTop\_toTopOf="parent"  
   app:layout\_constraintVertical\_bias="0.896"  
   app:srcCompat="@drawable/estadistic" />  
    
   <ImageButton  
   android:id="@+id/botonMostrarAlarma"  
   android:layout\_width="75dp"  
   android:layout\_height="78dp"  
   android:background="#00FFFFFF"  
   app:layout\_constraintBottom\_toBottomOf="parent"  
   app:layout\_constraintEnd\_toEndOf="parent"  
   app:layout\_constraintHorizontal\_bias="0.407"  
   app:layout\_constraintStart\_toStartOf="parent"  
   app:layout\_constraintTop\_toTopOf="parent"  
   app:layout\_constraintVertical\_bias="0.898"  
   app:srcCompat="@drawable/clock" />  
    
   <ImageButton  
   android:id="@+id/btnVideos"  
   android:layout\_width="98dp"  
   android:layout\_height="80dp"  
   android:background="#00FFFFFF"  
   app:layout\_constraintBottom\_toBottomOf="parent"  
   app:layout\_constraintEnd\_toEndOf="parent"  
   app:layout\_constraintHorizontal\_bias="1.0"  
   app:layout\_constraintStart\_toStartOf="parent"  
   app:layout\_constraintTop\_toTopOf="parent"  
   app:layout\_constraintVertical\_bias="0.901"  
   app:srcCompat="@drawable/yoga" />  
    
  </androidx.constraintlayout.widget.ConstraintLayout>
* Agregar animaciones y transiciones para una mejor UX

<ImageButton  
 android:id="@+id/btnRutina"  
 android:layout\_width="91dp"  
 android:layout\_height="70dp"  
 android:layout\_marginStart="28dp"  
 android:background="#00FFFFFF"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toStartOf="@+id/textView3"  
 app:layout\_constraintHorizontal\_bias="0.315"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.888"  
 app:srcCompat="@drawable/rutina" />

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

* Configurar rutas y vinculación de cada sección
* botonRutina.setOnClickListener(new View.OnClickListener() {  
   @Override  
   public void onClick(View view) {  
   Intent botonRutina = new Intent(MenuPrincipal.this, registrorutina.class);  
   startActivity(botonRutina);  
   }  
  });  
    
  botonAlarma.setOnClickListener(new View.OnClickListener() {  
   @Override  
   public void onClick(View view) {  
   Intent botonAlarma = new Intent(MenuPrincipal.this, RegistrarAlarma.class);  
   startActivity(botonAlarma);  
   }  
  });  
    
  botonEstadistica.setOnClickListener(new View.OnClickListener() {  
   @Override  
   public void onClick(View view) {  
   Intent botonEstadistica = new Intent(MenuPrincipal.this, estadisticas.class);  
   startActivity(botonEstadistica);  
   }  
  });  
    
  botonVideos.setOnClickListener(new View.OnClickListener() {  
   @Override  
   public void onClick(View view) {  
   Intent botonVideos = new Intent(MenuPrincipal.this, VideosActivity.class);  
   startActivity(botonVideos);  
   }  
  });
* Implementar validación de sesión (el menú solo es visible si el usuario está autenticado)

// Guardar el DNI del usuario actual para futuras consultas  
String dniUsuarioActual = cursor.getString(cursor.getColumnIndex("dni"));  
Intent ventanaPrincipal = new Intent(MainActivity.this, MenuPrincipal.class);  
ventanaPrincipal.putExtra("dniUsuarioActual", dniUsuarioActual); // Pasar el DNI a la siguiente actividad  
startActivity(ventanaPrincipal);

* Pruebas de navegación y usabilidad

Captura de pantalla de computadora

Descripción generada automáticamente

Pantalla de juego de computadora

Descripción generada automáticamente con confianza media

# 5. Burndown Chart

El siguiente gráfico representa el progreso del sprint, comparando las horas restantes con los días de ejecución.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| SPRINT BURNDOWN CHART | | | | | | | | | | | | |
|  |
| Backlog ID | Historia de Usuario | INICIAL ESTIMADO | 21-Abr | 22-Abr | 23-Abr | 24-Abr | 25-Abr | 26-Abr | 27-Abr | 28-Abr | 29-Abr | 30-Abr |  |
| DIA 0 | DIA 1 | DIA 2 | DIA 3 | DIA 4 | DIA 5 | DIA 6 | DIA 7 | DIA 8 | DIA 9 | DIA 10 |  |
| 3.1.2 | Historia de usuario 3.1 Configuración y Gestión de alarma | 6 | 2 | 2 | 1 | 1 |  |  |  |  |  |  |  |
| 3.1.3 | 6 | 2 | 1 | 1 | 1 | 1 |  |  |  |  |  |  |
| 3.1.4 | 5 |  |  |  | 1 | 1 | 1 | 1 | 1 |  |  |  |
| 3.1.5 | 6 |  |  | 1 | 1 | 1 | 1 | 1 | 1 |  |  |  |
| 4.1.1 | Historia de Usuario 4.1 Acceder al meú de navegación | 6 |  | 1 |  | 1 |  |  | 1 | 1 | 2 |  |  |
| 4.1.2 | 4 |  |  |  |  | 1 | 1 | 1 | 1 |  |  |  |
| 4.1.3 | 4 |  |  |  |  |  |  | 1 | 1 | 1 | 1 |  |
| 4.1.4 | 6 | 2 | 2 |  |  | 1 | 1 |  |  |  |  |  |
| 4.1.5 | 6 |  |  |  |  | 1 | 1 | 1 | 1 | 1 | 1 |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Tiempo Trabajado | | 49 | 43 | 37 | 34 | 29 | 23 | 18 | 12 | 6 | 2 | 0 |  |
| Tiempo Estimado | | 49 | 44 | 39 | 34 | 29 | 25 | 20 | 15 | 10 | 5 | 0 |  |

# 6. Análisis del Sprint

# Durante el Sprint 03, se espera que el equipo mantenga un desempeño coordinado en el diseño e implementación de las funcionalidades relacionadas con la configuración de alarmas personalizadas y el acceso al menú de navegación de la aplicación. Se reforzará la colaboración entre diseño, frontend y backend para lograr una experiencia de usuario fluida y funcional. Uno de los principales desafíos anticipados será la implementación del funcionamiento sin conexión para las alarmas, así como la validación de sesión para mostrar el menú solo a usuarios autenticados. También se prestará especial atención a la integración de animaciones en el menú de navegación y la personalización de tonos de alarma, lo cual puede requerir un mayor número de pruebas y ajustes. Con base en la experiencia del sprint anterior, se priorizará la revisión temprana de las dependencias técnicas, una mejor planificación de tiempos y una definición precisa de los casos de prueba, con el fin de minimizar riesgos y bloqueos durante el desarrollo.

# 7. Conclusiones

El Sprint 03 permitirá consolidar funcionalidades clave de la aplicación SleepApp, enfocadas en la personalización de la experiencia del usuario a través de la configuración de alarmas y la navegación fluida por la aplicación. Se buscará avanzar de forma significativa en la implementación de alarmas personalizadas según la rutina del usuario, la gestión offline de estas configuraciones, así como en el diseño e integración del menú de navegación. Estos avances fortalecerán la usabilidad general de la aplicación y prepararán el terreno para incorporar funcionalidades más avanzadas en los siguientes sprints, como las alarmas inteligentes y el análisis automático de patrones de sueño.