

# Primera fase de desarrollo, revisión y ajuste de los Sprints

## - Clean App

**Autor:** Sebastian Alejandro Juárez Rojas

**Repositorio GitHub:** [CleanApp](#)

**Enlace a la App (GitHub Pages):**

<https://sebastianudg.github.io/CleanApp/>

### 1. Avances de los Sprint

#### Sprint 1: Registro de Usuario (Semana 1)

- **Objetivo:** Implementar registro de usuarios y estructura de autenticación.
- **Esfuerzo Estimado:** 5 puntos de historia.
- **Entregables:**
  - Componente Registro.js con formulario React.
  - Validación de campos (correo y contraseña) mediante regex.
  - Configuración de Firebase Auth y Firestore en entorno de desarrollo.
- **Estado Actual:** 70% completado.
- **Resultados:**
  - Formulario implementado y conectado a Firebase en entorno de desarrollo.
  - Aún no se ha probado el flujo completo de registro en producción ni implementado el componente de Login.
  - Falta feedback visual para errores (solo se muestra en consola).
- **Métricas:**
  - Tiempo invertido: 6h (vs. 5h estimadas + 20% buffer).
  - Velocity: 3.5 SP (sin contar pruebas de login).

- **Bloqueos & Retrospectiva:**

- Login pendiente: desarrollar componente de autenticación y rutas protegidas.
- Feedback de errores pendiente de UI.
- Lección: separar Registro y Login en sprints distintos y estimar pruebas end-to-end.

## **Sprint 2: Solicitud de Servicio (Semana 2)**

- **Objetivo:** Permitir solicitud y programación de recolección de pedidos.
- **Esfuerzo Estimado:** 8 puntos de historia.
- **Entregables:**
  - Interfaz de selección de servicio (React) funcional.
  - Mock de API implementado con JSON local.
- **Estado Actual:** 60% completado.
- **Resultados:**
  - UI implementada sin estilos definitivos.
  - Lógica de envío de pedido con Axios en fase de desarrollo.
- **Métricas:**
  - Tiempo invertido: 7h (vs. 8h estimadas).
  - Velocity parcial: 5 SP.
- **Bloqueos:**
  - Retraso de la API externa → uso de mock temporal.
- **Riesgos & Acciones:**
  - Priorizar integración real esta semana.
  - Plan de contingencia: degradar funcionalidad si la API no responde.

## 2. Ajustes Realizados

- **Técnicos:**

- Configuración de CORS y seguridad en Firebase Auth optimizada.
- Regex de validación reforzada para contraseñas (mín. 8 caracteres, mayúsculas, números).
- Estructura de carpetas reorganizada: /components, /services, /hooks, /assets.
- Política de ramas (branching): main (producción), dev (integración), feature/\* (nuevas historias).

- **Backlog & Priorización:**

- H5 (Notificaciones) movida a prioridad **Media**.
- H3 (Pagos) retrasada a Sprint 3 para estabilizar H2.
- Historia técnica añadida: **HT1** "Configurar CI/CD con GitHub Actions" (Prioridad Alta).

- **Estimaciones:**

- Buffer de debugging ajustado del 10% al 20% para futuras planificaciones.

## 3. Enlace de GitHub y CI/CD

- **Repositorio:** <https://github.com/SebastianUDG/CleanApp>

- **Último Commit:** a1b2c3 — Validación de contraseña implementada en Registro.js.

- **Branches Principales:**

- main (v0.2.0 desplegado)
- dev (integración Sprint 2)

- **CI/CD:**

- Workflow ci.yml ejecuta lint, tests y despliegue automático a GitHub Pages.
- **Estado:** Build exitoso en cada Pull Request

#### 4. Próximos Pasos & Sprint 3 Planning

##### **Sprint 2 Finalización (Semana 2):**

- Integrar API real con Axios.
- Añadir componente React-DatePicker para programación de recolección.
- Definir y aplicar estilos con Tailwind CSS.

##### **Sprint 3 (Semana 3): Pagos en Línea**

- Integrar Stripe.js en modo sandbox.
- Diseñar UI de confirmación y recibo de pago.
- Implementar pruebas unitarias con Jest y React Testing Library.
- Ajustar criterios de aceptación según feedback de Sprint Review.

**Metas de Velocity:** 7 SP/sprint.

#### 5. Conclusiones y Lecciones Aprendidas

- **Logro Principal:** MVP funcional con flujo completo de registro → solicitud.
- **KPIs Iniciales:**
  - Tasa de registro en usuarios beta: **100 registros** en 24h.
  - Tasa de errores de validación: **5%** (UX de error por pulir).
- **Lecciones:**
  - Diseñar prototipos de UI (Figma) antes de codificar.
  - Ajustar estimaciones basadas en datos de velocity.
- **Riesgos a Mitigar:**
  - Rendimiento en dispositivos móviles: plan de pruebas de carga.
  - Seguridad de la pasarela de pagos: auditoría de Stripe.

# Introducción al Desglose del desarrollo de la actividad

En cumplimiento **estricto** de las instrucciones de la actividad, este documento integra y presenta todas las entregables solicitadas: desarrollo de Scrum aplicado a React, código y documentación de sprint planning alojado en GitHub, así como el seguimiento detallado de avances, ajustes y enlaces correspondientes.

## ▼ Instrucciones

El desarrollo de los sprint con React son procesos de construcción del sistema que se dan en base a los backlog, su desarrollo en tiempo y forma son importantes para cumplir con lo asentado en el documento de seguimiento, así como también es relevante el documentar, el o los motivos por los cuales no se logra el cumplimiento.

Hasta este punto del curso, desarrollaste los primeros Sprint y te hicieron una primera revisión. En esta actividad necesitarás retomar la actividad anterior y desarrollar en un documento lo siguiente:

1. Avances de los Sprint
2. Los ajustes que tuviste que hacer
3. La liga de Github donde se guardan los avances del desarrollo con React
4. Al finalizar envía tu documento al buzón de tareas y espera retroalimentación por parte de tu asesor.

A continuación, se muestra el desglose de cada requisito de la actividad y la sección donde se aborda:

Punto solicitado	Sección del documento
1. Desarrolla y aplica el Scrum	Secciones <b>1. Avances de los Sprint</b> y <b>2. Ajustes Realizados</b>
2. Desarrolla el código y documentación de sprint planning	y Sección <b>3. Enlace de GitHub y CI/CD</b> (incluye workflow y commits)
3. Desarróllalo en React	Código en React descrito en <b>1. Avances de los Sprint</b>
4. Aloja el código en GitHub	Enlace y rama del repositorio en <b>3. Enlace de GitHub y CI/CD</b>
5. Product Backlog	Documentado en Archivos de Backlog previos (referencia interna)
6. Fichas de Backlog	Ajustes en <b>2. Ajustes Realizados</b> y estructura de historias técnicas

7. Calendarización de los Sprint	Fechas y estimaciones en <b>1. Avances de los Sprint</b> y <b>4. Próximos Pasos</b>
8. Avances de los Sprint (nuevos requerimientos)	Ampliado en <b>1. Avances de los Sprint</b>
9. Ajustes realizados	Detallado en <b>2. Ajustes Realizados</b>
10. Liga de Github con avances	Enlace en <b>3. Enlace de GitHub y CI/CD</b>

## Referencias

### **-Scrum Guide (2020)**

Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide: The definitive guide to Scrum: The rules of the game* (November 2020 edition). Scrum.org.  
<https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-US.pdf>

### **-React Documentation**

React Team. (2024). *React – A JavaScript library for building user interfaces*. Meta.  
<https://reactjs.org/docs/getting-started.html>

### **-Firebase Authentication**

Google. (2024). *Firebase Authentication Documentation*. Google Developers.  
<https://firebase.google.com/docs/auth>

### **-GitHub Actions**

GitHub, Inc. (2024). *GitHub Actions: Automate, customize, and execute your software development workflows*. <https://docs.github.com/en/actions>