

CONSTRUCCIÓN DEL SOFTWARE

App Prográmate en Medellín Versión 1.0 2025

**Cristian David López Cardona
Dayana García Rodríguez
Brian Alexis Cadavid Castaño**

Institución Universitaria Pascual Bravo

Facultad de Ingeniería

Profesora Claudia María Giraldo Flórez

3.1 SELECCIÓN DE TECNOLOGÍAS

Elicitación de Requerimientos	Análisis	Diseño de Sistema	Diseño Detallado	Implementación	Testing
W BuenasPra...	W Diseño ...	W Diseño d...	W Diseño d...	W Diseño de ...	

3.2 CONTROL DE VERSIONES DEL CÓDIGO

Versión 0.1.0 con funcionalidades básicas como lo son Loguin de registro, Login de acceso, botones de accesos a las actividades y la geolocalización.

3.3 Calidad del Código

Criterio	Descripción	Métrica	Valor obtenido	Comentarios
Fiabilidad	Capacidad del software para funcionar sin fallos bajo condiciones determinadas	Número de errores reportados durante pruebas	1	El sistema es estable en la mayoría de escenarios.
Usabilidad	Facilidad de uso y comprensión por parte del usuario final.	Tiempo promedio para realizar una tarea + feedback del usuario	90	Interfaz intuitiva; se recomienda mejorar accesibilidad para personas mayores.

Eficiencia	Uso óptimo de recursos (memoria, tiempo de respuesta).	Tiempo promedio de respuesta	1.2 segundos	El rendimiento es aceptable, aunque se puede optimizar carga inicial.
Mantenibilidad	Facilidad para modificar y actualizar el código fuente.	Tiempo estimado para aplicar cambios simples	20 minutos	Arquitectura modular facilita el mantenimiento.
Portabilidad	Capacidad del software de ejecutarse en diferentes entornos.	Sistemas operativos compatibles probados	80	Adaptado para navegadores móviles y de escritorio.
Seguridad	Protección contra accesos no autorizados y vulnerabilidades.	Número de vulnerabilidades detectadas	10	Se validan entradas y se evita exposición de datos sensibles.
Compatibilidad			80	

	Interacción correcta con otros sistemas y tecnologías.	Navegadores y dispositivos compatibles		Compatible con los navegadores más comunes.
Desempeño	Capacidad del sistema de responder eficientemente bajo carga.	Número de usuarios simultáneos soportados	25	Sin retrasos ni cuelgues con 25 usuarios concurrentes.
Escalabilidad	Capacidad del sistema para crecer sin perder rendimiento.	Posibilidad de añadir nuevas funciones sin reestructurar código	100	Arquitectura modular permite añadir nuevas vistas y funciones fácilmente.

3.4 DESARROLLO DE COMPONENTES REUTILIZABLES

DIAGRAMA UML POR COMPONENTES

Diagrama UML

