

1. Introducción:

Este Plan de pruebas está diseñado para definir el alcance, el enfoque, los recursos y el cronograma de todas las actividades de prueba del proyecto Bienestar Universitario.

El plan identifica los elementos que se probarán, las características que se probarán, los tipos de pruebas que se realizará, el personal responsable de las pruebas, los recursos y el cronograma necesarios para completar las pruebas y los riesgos asociados con el plan.

1.1 Alcance

1.1.1 Dentro del Alcance

Todas las funciones de la aplicación Bienestar Universitario que se definieron en los requisitos del software deben ser probadas:

Nombre de la funcionalidad	Roles aplicables	Descripción
Gestión de actividades CADI	Estudiantes, Administradores CADI	Publicación, consulta y actualización de actividades artísticas y deportivas con calendario interactivo y retroalimentación.
Gestión de torneos deportivos y PSU	Estudiantes, Coordinadores	Inscripción digital, creación de equipos, validación de elegibilidad, seguimiento de proyectos sociales y chatbot de asistencia.
Gestión de citas psicológicas	Estudiantes	Reserva, cancelación y reprogramación de citas; visualización de

		disponibilidad en tiempo real.
Gestión de horarios e información	Estudiantes	Calendario personalizado y unificado, notificaciones automáticas y recordatorios.
Analítica y Reportes	Administradores	Consolidación de datos, análisis de comportamiento, generación de métricas, exportación de reportes y registro de asistencia.

1.1.2 Fuera del Alcance

Estas características no se pueden probar porque no están incluidas en las especificaciones de requisitos de software:

- Consolidar datos desde archivos externos (CSV, Excel), bases internas y sistemas externos (Banner, Gestión Humana).

- Limpiar y estandarizar datos (nombres, fechas, formatos), validando estructura y eliminando duplicados.

- Descargar graficas en Excel

- Registrar asistencia por identificación, Excel

- Geolocalización.

- Pruebas de seguridad avanzada y ataques externos.

- Contingencia y sincronización (Módulo 5)

1.2 Objetivo de Calidad

- Verificar el correcto funcionamiento de las funcionalidades principales de la aplicación Bienestar Universitario.

- Asegurar que cada funcionalidad proporcione una experiencia de usuario fluida, intuitiva y sin errores.

- Validar que el sistema responde correctamente ante entradas válidas e inválidas,

manteniendo la integridad de los datos.

- Confirmar que la navegación entre las distintas secciones de la aplicación es consistente, con tiempos de respuesta adecuados.

1.3 Roles y Responsabilidades

No	Rol (Scrum)	Tareas
1	Product Owner	Definir criterios de aceptación en historias de usuario. Priorizar funcionalidades y pruebas en el backlog.
2	Scrum Master	Facilitar la organización del equipo en pruebas. Eliminar impedimentos para la ejecución de pruebas
3	Development Team	Diseñar y ejecutar pruebas unitarias, de integración y aceptación. Reportar defectos. Garantizar calidad continua.
4	Usuarios piloto / Stakeholders	Participar en pruebas de aceptación durante las revisiones de Sprint. Dar retroalimentación de usabilidad.

2. Metodología de prueba

2.1 Niveles de prueba

En el proyecto Bienestar Universitario, se deben realizar 6 tipos de pruebas:

- Pruebas de unidad: Verificación de funcionalidad correcta de los módulos (ej: validación de inscripción en torneos).
- Pruebas de resistencia: Determinar la resistencia del sistema (ej: miles de eventos en calendario).
- Pruebas de regresión: Asegurar que nuevas funciones no rompen funcionalidades existentes.
- Pruebas de desempeño: Simular más de 1000 usuarios concurrentes consultando mapas o inscribiéndose.
- Pruebas de concurrencia: Evaluar cómo maneja múltiples usuarios en la misma funcionalidad simultáneamente.
- Pruebas de volumen: Validar rendimiento con grandes cantidades de datos (más de 50 mil registros).

2.2 Clasificación de errores

Alta: Fallo en funcionalidades críticas (ej: no se pueden reservar citas o inscribir en torneos).

Media: Fallos menores que no bloquean el uso (ej: un mapa que no muestra ubicaciones favoritas).

Baja: Errores visuales o de texto (ej: un nombre de actividad mal escrito).

2.4 Criterios de suspensión y requisitos de reanudación

Si el sistema presenta errores críticos que impiden probar funcionalidades clave, se suspenderán las pruebas hasta que el equipo de desarrollo entregue una nueva versión con correcciones validadas.

2.5 Completitud de la prueba

- Tiempo de respuesta ≤ 2 segundos en búsquedas y consultas.
- Soporte para más de 1000 usuarios concurrentes.
- 95% de los casos de prueba ejecutados exitosamente.

2.6 Tareas del proyecto, estimación y cronograma

Tarea	Miembros	Esfuerzo estimado
Elaborar el plan de pruebas	Product Owner + Development Team	50 Horas-hombre
Diseñar las pruebas	Development Team	100 Horas-hombre
Ejecución de las pruebas	Development Team	70 Horas-hombre
Informe de las pruebas	Development Team + PO	15 Horas-hombre
Corrección y/o aceptación de las pruebas	Development Team + PO	100 Horas-hombre
Entrega de las pruebas	Scrum Master + PO	20 Horas-hombre

Total: 355 Horas-hombre

3. Entregables de prueba

Antes de la fase de prueba:

- Documento de planes de prueba.
- Documento de casos de prueba.
- Especificaciones de diseño de pruebas.

Durante la prueba:

- Registros de ejecución.
- Datos de prueba.
- Reporte de defectos.
- Matriz de trazabilidad de pruebas.

Al finalizar los ciclos de prueba:

- Informe final de pruebas.
- Informe de defectos.

- Manual de instalación/configuración.
- Documentación de la versión.

4. Necesidades de recursos y ambientes

4.1 Herramientas de prueba

Número	Recursos	Descripción
1	Equipos	Teléfono móvil iOS o Android con acceso a internet, cámara y periféricos. PC con Windows 11 o superior, al menos 4GB RAM y procesador Intel i5.
2	Software	OS Android 9.0+, iOS 15.0+, Windows 11+. API de mapas y multimedia.
3	Red	LAN Gigabit y una línea de internet de al menos 5 Mb/s para conectividad multimedia.
4	Herramientas de Prueba	Selenium, JMeter, Postman u otra herramienta que genere resultados automáticos en pruebas.

4.2 Entorno de prueba

Requisitos mínimos de hardware y software que se utilizarán para probar la Aplicación:

- Windows 11 o superior

- Android 9.0 o superior
- iOS 15.0 o superior

Beneficios de las pruebas

Tipo de prueba	Funcionalidad	seguridad	usabilidad	fiabilidad	rendimiento	integridad de datos
Pruebas funcionales	X		X	X		
pruebas de integración	X	X		X		X
pruebas de sistema	X	X		X	X	X
Pruebas de aceptación (UAT)	X		X	X		
Pruebas de seguridad (básica)		X		X		
Pruebas unitarias	X			X		
Pruebas de desempeño				X	X	
Pruebas de datos (in-app)				X		X

Indicadores/Métricas de evaluación de calidad

Métrica	Descripción	Fórmula
Densidad de Errores	Número de errores por unidad de tamaño (documentos o software).	Densidad de errores = N° de errores / Tamaño
Densidad de Defectos	Número de defectos encontrados dividido por el tamaño del software.	Densidad de defectos = N° de defectos / Tamaño
Tasa de Fallos	Tiempo medio entre fallos (<i>Mean Time To Failure, MTTF</i>)..	Tasa de fallos = Tiempo total de operación / N° de fallos