

Plan de Pruebas

GRCU Manager

Gestión de Requerimientos y Casos de Uso

4 BYTES

ALVAREZ, Abril

BUTTERFIELD, Nicolas

CARRANZA, Cristian

GAGNA, Martina





El propósito del plan de pruebas es explicitar el alcance, enfoque, recursos requeridos, calendario, responsables y manejo de riesgos de un proceso de pruebas.

Este plan de Pruebas está contemplado dentro del plan SQA para un proyecto dado



Contenido

Contenido

No se encontraron entradas de tabla de contenido.

Plan de Pruebas

Introducción.

El plan de pruebas define las actividades destinadas a garantizar la calidad del sistema de gestión de requerimientos y casos de uso (GRCU Manager). Se evaluarán los módulos principales:

- Autenticación con UARGFlow
- Gestión de proyectos
- Gestión de requerimientos
- Gestión de casos de uso
- Priorización MoSCoW, trazabilidad
- Generación de informes
- Comentarios y adjuntos

Las pruebas contemplarán tanto la funcionalidad del sistema como aspectos de integración, usabilidad y seguridad, ejecutadas en un entorno controlado con participación de alumnos (como desarrolladores y testers) y docentes (como clientes/stakeholders).

Definición de objetivos.

Objetivo general

Las pruebas que se realizarán tendrán como objetivo general encontrar y corregir errores en los componentes individuales y en la integración de los mismos, verificando su cumplimiento con los requerimientos y casos de uso definidos.

Objetivos Específicos

- Validar que las funciones principales (gestionar usuarios, proyectos, requerimientos, casos de uso) operen conforme a las especificaciones.
- Confirmar que la trazabilidad entre requerimientos y casos de uso se genere y visualice correctamente.
- Asegurar que el historial de cambios registre modificaciones con precisión.
- Probar que la autenticación mediante UARGFlow funcione de forma confiable.
- Evaluar la usabilidad del sistema en términos de claridad de interfaces y facilidad de uso.
- Comprobar la generación de informes en los formatos requeridos (PDF, DOCX, XLSX).

Antecedentes y Propósito

Propósito de la Evaluación

El propósito de la evaluación es asegurar que el sistema **GRCU Manager** cumpla con los estándares de calidad definidos para esta iteración del proyecto. Entendemos la calidad como la capacidad del sistema de cumplir con los requisitos funcionales, no funcionales y de diseño, a la vez que proporciona una experiencia confiable y consistente para los usuarios.

Para ello, las pruebas se convierten en una actividad central de aseguramiento de calidad, y nuestro plan busca coordinar dichas actividades en cada iteración. En esta primera etapa, los objetivos principales de la evaluación son:

- Encontrar tantos errores como sea posible en las funcionalidades desarrolladas.
- Verificar el cumplimiento de las especificaciones de diseño planteadas.
- Comprobar que se cumplen los requisitos funcionales y no funcionales definidos en el análisis.
- Identificar problemas relevantes que afecten la trazabilidad de requerimientos y casos de uso.
- Determinar riesgos percibidos sobre la calidad del sistema y retroalimentar al equipo para futuras iteraciones.

Motivadores de la prueba

Los principales motivadores que impulsan la necesidad de pruebas en esta iteración son:

- **Requerimientos funcionales:** gestión de usuarios, gestión de proyectos, registro de requerimientos, vinculación de casos de uso, generación de matriz de trazabilidad.
- **Requerimientos no funcionales:** acceso vía web, autenticación con cuentas Gmail/UNPA mediante UARGFlow.
- **Cambios de requerimientos:** ajustes que puedan surgir a partir de la validación con los docentes.

Objetos a ser Evaluados

Los componentes y aspectos que serán evaluados en esta iteración son los siguientes:

- **Componentes de software:**
 - Módulo de gestión de usuarios.
 - Módulo de gestión de proyectos.
 - Módulo de requerimientos y casos de uso.
 - Módulo de historial de cambios.
 - Generación de matriz de trazabilidad.
- **Componentes de hardware:**

- Pruebas en notebooks de los integrantes del equipo.
- **Características específicas:**
 - Autenticación mediante UARGFlow.
 - Correcto manejo de roles y permisos diferenciados (Administrador, Analista, Stakeholder, Visitante).
 - Trazabilidad entre requerimientos y casos de uso.

Ámbito de las Pruebas

[En esta sección se debe mencionar el conjunto de tareas necesarias para conseguir el objetivo del proyecto, como así también las que fueron descartadas.]

Dentro del Ámbito

Dentro del Ámbito

- Pruebas funcionales de todos los módulos principales.
- Pruebas de integración con UARGFlow.
- Pruebas de generación y exportación de informes.
- Pruebas de interfaz de usuario en navegadores soportados.

Dentro del Ámbito

Se incluye la verificación de:

- Alta, baja, modificación y búsqueda de usuarios.
- Creación, modificación y eliminación de proyectos.
- Registro, edición y eliminación de requerimientos y casos de uso.
- Generación automática de historial de cambios en requerimientos y casos de uso.
- Creación de la matriz de trazabilidad bidireccional.
- Autenticación de usuarios mediante cuentas Gmail/UNPA con UARGFlow.
- Adjuntos y comentarios en requerimientos y casos de uso.

La documentación del proyecto será revisada a través de revisiones técnicas formales (RTF). Los documentos a los que se le van a realizar las RFT son aquellos que fueron expresados como clave en el plan SQA. Por otro lado, cada documento también tendrá una revisión no formal.

Las pruebas que se deben realizar para cumplir con el criterio de calidad son:

- Las pruebas de unidad o unitarias. Estas se van a realizar para evitar que las funciones o métodos que se desarrollen contengan errores lógicos.
- Las pruebas de integración. Estas se van a realizar para evitar los errores que ocurren al

integrar componentes desarrollados por diferentes personas.

- Las pruebas de validación. Estas se van a realizar para comprobar que lo desarrollado es lo que el cliente desea.

La documentación del proyecto GRCU Manager será revisada mediante **revisiones técnicas formales (RTF)**. Además, cada documento contará con una revisión no formal por parte de los integrantes del equipo antes de su entrega.

Para garantizar el cumplimiento de los criterios de calidad, se realizarán los siguientes tipos de pruebas:

- **Pruebas unitarias:** se ejecutarán para detectar errores lógicos en las funciones y métodos desarrollados, como el registro de requerimientos, la asignación de roles y la generación de informes.
- **Pruebas de integración:** se aplicarán para verificar la correcta comunicación entre los módulos del sistema (gestión de proyectos, requerimientos, casos de uso, trazabilidad) y la integración con el framework UARGFlow para autenticación.
- **Pruebas de validación:** se llevarán a cabo para confirmar que el sistema desarrollado cumple con las necesidades y expectativas de los docentes y alumnos, especialmente en la trazabilidad de requerimientos.

Fuera del Ámbito

Las siguientes pruebas quedan excluidas del ámbito actual, ya que se abordarán en iteraciones futuras o no son prioritarias para la versión inicial:

- Pruebas de carga y estrés (e.g., simulación de miles de usuarios concurrentes), justificadas porque el sistema es para uso académico limitado (no escalabilidad masiva).
- Pruebas de compatibilidad con navegadores obsoletos (e.g., IE), ya que se enfoca en Chrome/Firefox modernos.
- Pruebas de accesibilidad avanzada (e.g., WCAG 2.1 nivel AAA), limitada a nivel AA básico por recursos.
- Pruebas de movilidad (apps móviles), ya que es web desktop-first.
- Pruebas de internacionalización (multi-idioma), ya que es en español para UNPA.

Estas exclusiones permiten enfocarnos en funcionalidad core y calidad básica, alineado con CMMI y TMMi para madurez en pruebas.

Lista de Ideas de las Pruebas

Las ideas de pruebas se generan a partir de los casos de uso y requerimientos, usando técnicas como análisis de flujos (básico/alternativo/excepción) y métricas de calidad.

- *Para autenticación UARGFlow: Verificar login exitoso/fallido, roles asignados.*
- *Para gestión de proyectos: Crear proyecto, agregar usuarios, eliminar sin datos asociados.*
- *Para requerimientos/casos de uso: Registrar, editar, eliminar; priorizar MoSCoW; adjuntar archivos/comentarios.*
- *Para trazabilidad: Generar matriz bidireccional, verificar enlaces.*
- *Para informes: Exportar en PDF/DOCX/XLSX, validar contenido.*
- *Pruebas no funcionales: Usabilidad (encuestas Google Forms), seguridad (roles no autorizados).*

Recursos para identificación: Especificación de Requerimientos, Arquitectura de Software, Casos de Uso. Se resumen en tablas por sprint.

Enfoque de las Pruebas

El enfoque es Ágil, con pruebas integradas en cada sprint (2-4 semanas). Se usan técnicas de TDD (Test-Driven Development) para unitarias y CI/CD para integración. Propósitos: Detectar defectos tempranos, medir métricas (ISO/IEC 25000/TMMi), asegurar cobertura.

Pruebas Unitarias

- *Objetivo: Verificar funciones individuales (e.g., registro de requerimiento).*
- *Descripción: Pruebas automáticas en código.*
- *Técnicas: JUnit/Mocha para validación de lógica; SonarQube para complejidad ciclomática <10.*

Fases:

1. *Criterios de entrada: Código listo en Git branch.*

2. *Ejecución automática en CI.*
3. *Análisis de resultados (cobertura >80%).*
4. *Criterios de salida: Sin fallos críticos, métrica densidad defectos <3/KLOC.*

- *Entorno de prueba: Local en notebooks.*
- *Hardware: Notebooks equipo (Intel i5+, 8GB RAM).*
- *Software: Python, Django.*
- *Configuración: Entorno dev VSC.*
- *Criterios de Éxito: 100% pass, MTBF > tiempo sprint.*
- *Consideraciones Especiales: Integrar con Git.*

Pruebas de Integración

- *Objetivo: Verificar interacción módulos (e.g., requerimientos + trazabilidad).*
- *Descripción: Pruebas bottom-up/top-down.*
- *Técnicas: API testing con Postman, integración continua en Git.*

Fases:

1. *Criterios de entrada: Unitarias aprobadas.*
2. *Ejecutar en entorno staging.*
3. *Verificar logs/errors.*
4. *Criterios de salida: Sin fallos integración, tasa de fallos <5%.*

- *Entorno de prueba: Staging server.*
- *Hardware: Servidor cloud (e.g., Heroku free tier).*
- *Criterios de Éxito: Trazabilidad correcta, MTTR <1 hora.*
- *Consideraciones Especiales: Pruebas de API endpoints.*

Pruebas de Sistema

- *Objetivo: Validar sistema completo (e.g., flujo end-to-end: login -> proyecto -> informe).*
- *Descripción: Black-box testing.*
- *Técnicas: Casos de uso basados en requerimientos.*

Fases:

1. *Criterios de entrada: Integración aprobada.*
2. *Ejecutar en entorno pre-prod.*
3. *Medir usabilidad con Google Forms.*
4. *Criterios de salida: Cumplimiento 95% requerimientos, índice satisfacción >4/5.*

- *Entorno de prueba: Pre-prod similar a prod.*
- *Hardware: Notebooks/docentes.*
- *Software: Browsers Chrome/Firefox, Google Forms.*
- *Configuración: Datos de prueba realistas.*
- *Criterios de Éxito: Sin crashes, cobertura pruebas 85%.*
- *Consideraciones Especiales: Involucrar stakeholders.*

Pruebas de Aceptación

- *Objetivo: Validar con usuarios finales (docentes/alumnos).*
- *Descripción: UAT (User Acceptance Testing).*
- *Técnicas: Escenarios reales, feedback cualitativo.*

Fases:

1. *Criterios de entrada: Sistema estable.*
2. *Sesiones con usuarios.*
3. *Recopilar feedback via Google Forms.*
4. *Criterios de salida: Aprobación stakeholders, porcentaje defectos corregidos 100%.*

- *Entorno de prueba: Prod-like.*
- *Hardware: Dispositivos usuarios.*
- *Software: Sistema deployado.*
- *Configuración: Datos producción simulados.*
- *Criterios de Éxito: Satisfacción cliente >90%.*

- *Consideraciones Especiales: Registrar.*

Pruebas de Usabilidad y Seguridad

- *Objetivo: Evaluar UX y protección (roles, auth).*
- *Descripción: Heurísticas Nielsen para usabilidad; OWASP para seguridad.*
- *Técnicas: Encuestas Google Forms, scans SonarQube.*

Fases: Integradas en sprints.

<Identificador del tipo de prueba>

Objetivo:	<i>[Objetivo del tipo de prueba señalado.]</i>
Descripción:	<i>[Descripción breve del tipo de prueba.]</i>
Técnicas:	<i>[Defina el conjunto de procedimientos necesarios para que la prueba se llevada a cabo de manera correcta.]</i>
Fases:	<i>[1. Criterios de entrada: inicio de la fase 2. Fase 1 3. Fase 2 4. Criterios de salida: fin de la fase 2]</i>
Entorno de prueba:	<i>[Describa el entorno de prueba, los servicios básicos y configuraciones de la infraestructura requeridos por la prueba.]</i>
Hardware:	<i>[Detalle la configuración de hardware de los equipos necesarios.]</i>
Software:	<i>[Señale todos los componentes de software requeridos.]</i>

Objetivo:	<i>[Objetivo del tipo de prueba señalado.]</i>
Configuración:	<i>[Mencione el conjunto de peculiaridades específicas para la prueba.]</i>
Criterios de Éxito:	<i>[Mencione los aspectos que darán muestra que la prueba funcionará.]</i>
Consideraciones Especiales:	<i>[La prueba puede requerir características especiales, de ser así señale las en este apartado las mismas.]</i>

Diagrama.

De ser necesario para una mejor comprensión realice los diagramas necesarios que ilustren la estrategia de la prueba, como puede ser un diagrama de red, entre otros.]

Herramientas para las Pruebas

[Esta sección describe las distintas clases de herramientas utilizadas en la prueba del sistema.]

Software

[Se han utilizado las siguientes herramientas durante la prueba:

Herramienta_1: se trata de una herramienta de utilidades que puede utilizarse para la instalación, configuración y resolución de problemas de las entidades emisoras.

Herramienta_2: se usa para restaurar y hacer copias de seguridad de archivos.]

Nombre	Ver sión	Descripción

Herramientas de Soporte y Productividad

[Durante las pruebas se utilizaron las siguientes herramientas de supervisión del sistema:

Herramienta_1: permite ver los contadores, las alertas y los registros de rendimiento del sistema.

Herramienta_2: se utiliza para comprobar la coherencia de la base de datos.]

Nombre	Ver sión	Tipo de herramienta	Descripción

Secuencias de Comandos Personalizadas (Script de Pruebas)

[Indique las secuencias de comandos (Script de Pruebas) utilizadas durante las pruebas en las distintas fases del proyecto].

Hardware

[Señale cada uno de los dispositivos físicos que comprenden el sistema de computación a utilizar para la realización del conjunto de pruebas. Lo más recomendable que el sistema simule el ambiente de producción.]

Recurso	Cantidad	Descripción

Configuraciones de Pruebas de ambiente

[Las configuraciones del ambiente de Prueba deben ser provistas y soportadas por este proyecto.]



Nombre de Configuración	Descripción	Implementación de la Configuración Física

Casos de Prueba

[Esta sección detalla las pruebas planeadas a utilizar para comprobar el sistema y garantizar la consecución de sus objetivos con los que fue diseñado. Para realizar estas pruebas se debe llevar a cabo las configuraciones necesarias en el entorno de prueba, cada caso de prueba puede requerir un conjunto de especificaciones determinadas.

Para cada caso de prueba planteado llene una tabla como la presentada a continuación:

ID/Nombre/Sistema/Proyecto:		Nivel de Prueba:		
ID Caso de Uso:		Tipo(s) de Pruebas(s):		
ID Requerimiento: (Si es Caso de Uso no Funcional)		Ambiente de Prueba: (Ubicación)		
ID/Nombre Escenario:		Autor del Caso de Prueba:		
ID/Nombre Caso de Prueba:		Nombre del Probador:		
Versión del Caso de Prueba:		Fecha de Creación:	Fecha de Ejecución:	
Condición(es) para que se ejecute el Caso de Prueba:				
Para la Ejecución del Caso de Prueba:				
Nro. Paso Flujo	Condición	Valor(es)	Resultado Esperado	Resultado Obtenido
Criterios de Aprobación del Caso de Prueba:				
Decisión de Aprobación del Caso de Prueba: Aprobó: ____ Fallo: ____ (marque con x el resultado)				
Fecha de Aprobación del Caso de Prueba: _____				

Prioridades

Casos de Prueba por Características de Prioridad

[Establezca esta guía para asegurarse de que las características de alta prioridad están probadas adecuadamente. Enumere las características por prioridad y luego enumere los casos de prueba para cada característica.]

Esenciales

Liste las características a ser evaluadas.

Esperadas

Liste las características a ser evaluadas.

Deseadas

Liste las características a ser evaluadas.]

Casos de Pruebas por Prioridad de Caso de Uso

[Utilice esta guía para asegurarse de que los casos de uso de alta prioridad están probados adecuadamente. Enumere los casos de uso por prioridad y luego enumere los casos de prueba para cada caso de uso.]

Esenciales

Liste los casos de uso a ser evaluadas.

Esperados

Liste los casos de uso a ser evaluadas.

Deseados

Liste los casos de uso a ser evaluadas.]

Flujos de Trabajo de Pruebas

[Para una mejor comprensión realice los diagramas necesarios que ilustren el flujo de trabajo a seguir por el grupo de pruebas en el desarrollo y ejecución del plan de pruebas. Este diagrama puede ser representado de diversas formas dependiendo del proyecto, este puede ser un diagrama de Gantt, un listado de tareas, etc.]



De acuerdo con la naturaleza del proyecto, proporcionar un resumen que explique el proceso que su equipo utiliza para manejar el planeamiento detallado de la tarea y proporcionar una referencia adonde se encuentran los detalles, si es apropiado.]

Entregables

Lista de Entregables de Pruebas

[Especifique en esta sección, los entregables que serán producto de las pruebas a realizar, los cuales serán distribuidos y utilizados por los involucrados en el desarrollo del sistema para brindarles información relevante de los resultados obtenidos y de los avances logrados.]

Entregables	Descripción
<i>[Nombre]</i>	<i>[Proporcione una breve reseña de la forma y del contenido del entregable]</i>

Ficha: Escenarios por Caso de Uso

<ID/Nombre de Caso de Uso>				
ID Escenario	Flujo Básico	Flujo Alternativo 1	Flujo Alternativo 2	Flujo Alternativo (n)



Ficha: Resumen de Ciclos de Prueba

ID del Proyecto/ Nombre:

ID del Ciclo de Prueba:

Fechas para el Ciclo de Prueba: Desde: _____ Hasta:

ID Caso de Uso	ID Caso de Pruebas	Resultados Esperados	Resultados Obtenidos	Observación

Resultados/Observaciones para el Ciclo de Prueba:

Aprobado Ciclo de Prueba por:

CLIENTE

ORGANIZACIÓN

PROBADOR

Ficha: Matriz de Trazabilidad

ID Caso de Uso	ID de Escenario	ID de Caso de Prueba	ID Tipo de Prueba	ID Ciclo de Prueba

[Con herramientas automatizadas, se pueden realizar matrices de las relaciones entre las pruebas.]



Criterio para el Inicio y Fin del Plan de Pruebas

Criterios de Inicio

[Listar las precondiciones y aseveraciones necesarias para iniciar el plan de pruebas:

Condición 1.

Condición 2.

Condición N.]

Criterios de Fin

[Liste los criterios que se emplearan para determinar si la ejecución del plan de prueba estará completa.]

Criterios de Suspensión y Retomo de Actividades

[Especifique los criterios a implementar para determinarse si las pruebas deben ser suspendidas o ser terminadas prematuramente antes de que el plan se haya ejecutado totalmente y bajo qué criterios pueden ser retomadas.]

Criterios para el Lanzamiento

Criterios de Evaluación

[Ver que aspectos serán tomados para determinar que un producto ha superado satisfactoriamente el plan de pruebas y el mismo está listo para pasar a una siguiente fase. El lanzamiento principal de la solución está vinculado a la gravedad y la prioridad de errores sin resolver de acuerdo con los siguientes criterios:]

- *No existen errores sin solucionar de Gravedad 1 o Gravedad 2.*
- *No existen errores sin solucionar de Prioridad 1 o Prioridad 2 de ningún nivel de gravedad.*
- *Todos los casos de prueba del entorno de laboratorio de prueba se han completado satisfactoriamente.]*

Clasificación de los errores

[Establezca los criterios de gravedad y prioridad de los errores utilizados en el laboratorio de pruebas.]

Calificación	Definición de gravedad	Definición de prioridad
1	<i>[El error provoca el bloqueo del sistema o la pérdida de datos.]</i>	<i>[El error debe corregirse lo antes posible. El error bloquea el progreso en esta área]</i>
2	<i>[El error causa problemas graves en la funcionalidad u otros aspectos importantes; el producto se bloquea en casos poco claros.]</i>	<i>[El error debe corregirse antes del lanzamiento del producto]</i>

Resultados de la prueba

[En esta sección se deben indicar las conclusiones de las pruebas realizadas.]

Reportes del problema, escalada y resolución

[Defina cómo los problemas de proceso serán divulgados y extendidos y el proceso que se seguirá para alcanzar la resolución.]

Riesgos

[Enumerar cualquier riesgo que pueda afectar la ejecución de este plan de pruebas, e identificar las estrategias de la mitigación, contingencia y el impacto que pueden generarse por cada riesgo. Para detalles muy específicos emplear el Plan de Gestión de Riesgos, comentando aquí solo los riesgos a manera general.]

Riesgos	Estrategias de Mitigación	Plan de Contingencia	Impacto
<i>Nombre</i>	<i>Estrategia 1.</i> <i>Estrategia 2.</i> <i>Estrategia N.</i>	<i>Actividad 1</i> <i>Actividad 2</i> <i>Actividad N</i>	<i>Indique las áreas que se podrían ver afectadas por el riesgo</i>

Reportes de Problemas y Resolución

[Especificar cómo los problemas de proceso encontrados serán divulgados y extendidos, también se debe indicar el proceso que se seguirá para alcanzar la resolución de dichos problemas.]

Responsabilidades, Personal y Necesidades de Capacitación

Personal y Roles Necesarios

[Detallar los roles requeridos y la cantidad de los mismos, para la realización del plan de pruebas según el esfuerzo necesario para realizar el mismo.]



Roles	Recursos Necesarios	Estado	Responsabilidades Específicas o Comentarios
<i>Nombre de Rol</i>	<i>Cantidad Necesaria</i>	<i>Asignado / Pendiente</i>	<i>Responsabilidad 1. Responsabilidad 2. Responsabilidad N.</i>

Personal y Necesidades de Capacitación

[Especificar en esta sección de ser necesario, si el personal definido en el punto anterior requiere de alguna capacitación para lograr su objetivo, de ser así especifique que clase de capacitación se le dará y la planificación de la misma.]