



**“Proyecto Final Aseguramiento de la calidad del software”**  
**Aplicación publicada en ambiente de producción,**  
**certificando la calidad del software**

**Faculta de Ingeniería en**  
**Sistemas**  
**Décimo semestre**



## Proyecto Aseguramiento de la calidad del software

### Objetivos

1. Proveer a los alumnos información objetiva con relación a la calidad de una aplicación.
2. Elaborar un proyecto de software desde el acta de constitución hasta su publicación en ambiente de producción.
3. Efectuar pruebas a las aplicaciones que pasan al ambiente de producción.

### Normas Internas

1. Realizar grupos de 5 estudiantes, en donde cada uno tendrá las funciones siguientes:
  - a. Scrum Master (Administrador del proyecto).
  - b. Product Owner (Negocio).
    - i. Equipo Scrum:
  - c. Analista de Proyectos.
  - d. Analista de Desarrollo.
  - e. Analista de Aseguramiento de Calidad (QA).
2. Se llevarán a cabo reuniones entre todos los colaboradores del grupo en horario de clase, en donde se levantarán minutas por cada una de estas.
3. El Scrum Master es el encargado de administrar el recurso y los tiempos en las fases del proyecto.
4. Los encargados de cada una de las fases son los siguientes:
  - a. Hoja de ruta: Scrum Master, Product Owner
  - b. Dercas: Scrum Master, Product Owner
  - c. Acta de constitución: Analista de Proyectos, Scrum Master, Product Owner.
  - d. Gestión de procesos de negocio (BPM): Equipo Scrum.
  - e. Casos de uso: Analista de Proyectos, Product Owner.
  - f. Ajustes en casos de uso: Analista de Proyectos, Product Owner.
  - g. Desarrollo: Analista de Desarrollo.
  - h. Pruebas unitarias: Analista de Desarrollo.
  - i. Pruebas funcionales y no funcionales: Analista de Aseguramiento de Calidad.
  - j. Pruebas de seguridad: Analista de desarrollo, Analista de Aseguramiento de Calidad.
  - k. Ajustes al desarrollo: Analista de Desarrollo.
  - l. Segunda iteración de prueba: Analista de Aseguramiento de Calidad.
  - m. Pruebas de seguridad: Analista de Aseguramiento de Calidad.
  - n. Pruebas de aceptación: Product Owner.
  - o. Lecciones aprendidas: Equipo Scrum.
  - p. Implementación: Equipo Scrum.



5. Entrega de documentación.

La documentación por utilizar se encuentra versionada en Canvas, por lo que deberá descargar.

- a. Minutas de reuniones,
- b. Hoja de ruta,
- c. Dercas,
- d. Acta de constitución, firmados y validado,
- e. Cronograma de actividades,
- f. Modelado de procesos BPM,
- g. Casos de uso, firmados y validados,
- h. Informe de pruebas unitarias,
- i. Informe de resultados de pruebas funcionales y no funcionales,
- j. Informe de pruebas de seguridad,
- k. Plantilla de pruebas de aceptación,
- l. Carta de aceptación de negocio,
- m. Receta de publicación en producción,
- n. Informe de control de cambios,
- o. Guía técnica,
- p. Lecciones aprendidas.

6. Valor del proyecto.

- Hoja de ruta: 1 punto zona de actividades.
- Dercas: 2 puntos zona de actividades
- Acta de constitución: 2 puntos zona de actividades.
- Modelado de procesos BPM: 4 puntos zona de actividades
- Casos de uso: 5 puntos zona de actividades.
- Pruebas unitarias: 2 puntos zona de actividades.
- Pruebas funcionales y no funcionales: 2 puntos zona de actividades.
- Pruebas de seguridad: 2 puntos zona de actividades.
- Proyecto finalizado (publicado en producción): 15 puntos del examen final.

## Narrativa

El proyecto es libre en su tecnología (tipo de programación, lenguaje de programación, base de datos, versionamiento, servidores).

### Fases:

**1. Hoja de ruta:** Independientemente del tamaño o el alcance del proyecto, el cronograma de actividades debe ser una parte de su gestión. A la vista de este calendario, el administrador del proyecto puede conocer el momento en que cada actividad se debe llevar a cabo, las tareas que ya se han completado y la secuencia en que cada fase tiene que ser ejecutada. No obstante, hay que tener en cuenta que la precisión de esta herramienta de gestión de proyectos dependerá de la diligencia con que se lleven a cabo las actualizaciones.

Responsable: Scrum Master, Product Owner



**2. Acta de constitución:** documento en el que se define el alcance, los objetivos y los participantes del proyecto. Da una visión preliminar de los roles y responsabilidades, de los objetivos, de los principales interesados y define la autoridad del administrador del proyecto.

Responsable: Analista de Proyectos, Scrum Master, Product Owner

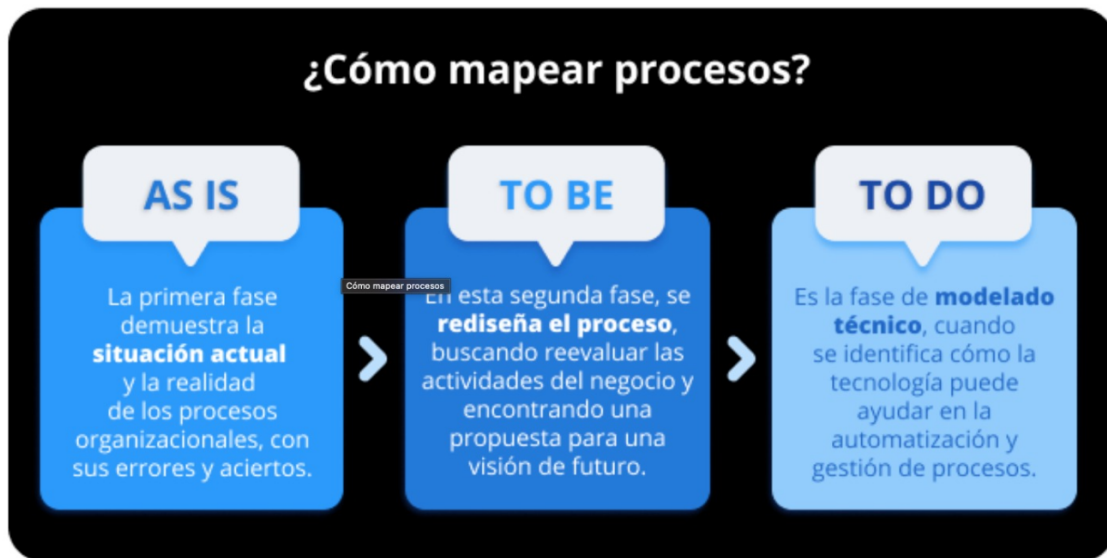
**3. Dercas:** "Documento de Especificaciones, Requerimientos y Criterios" El DERCAS es una metodología de la Ingeniería de software que permite definir los pasos esenciales para el análisis y desarrollo de un proyecto de software. Generalmente se realiza al principio del proyecto, e involucra tanto al arquitecto de software, al administrador del proyecto y a otros miembros del staff, que harán diversas visitas a los clientes para el levantado de requerimientos y así enmarcar en un contexto la solución a desarrollar. La metodología no se centra únicamente en el software, sino en mejorar los procesos de trabajo actuales de la organización y las prácticas en las que se trabaja en el área donde será implementado el sistema. Su equivalente en el idioma inglés es "Analysis and Requirements Specifications", que se traduce como Especificaciones de Análisis y Requerimientos.

Responsable: Scrum Master, Product Owner

- 4.** BPM (Business Process Management) es una metodología de trabajo utilizada tanto por grandes empresas como pymes. Este sistema de gestión de procesos se encarga de controlar el modelado, visibilidad y gestión de los procesos productivos de la empresa. BMP implica adoptar una serie de pasos o acciones que modifican la forma de trabajar de la empresa con el objetivo de mejorar los procesos y facilitar la colaboración con un enfoque hacia el cliente. Este método de trabajo surge en los años 80 del modelo productivo utilizado por la empresa japonesa Toyota. Con el paso del tiempo y la aparición de normas internacionales de buenas prácticas y calidad, se consolidó este sistema basado en la gestión de procesos. Por qué implementar el sistema de gestión de procesos Optimizar los procesos, mejorar la eficiencia, facilitar la toma de decisiones y, en definitiva, ser más competitivos, son los principales motivos por los que las empresas deben implementar un BPM. Cuando se produce una reestructuración empresarial (ampliación, absorción o colaboración) se utiliza la metodología BPM para poder coordinar de forma adecuada los procesos y departamentos de las distintas organizaciones. Muchas empresas empiezan a utilizar BPM cuando quieren conseguir una certificación y necesitan optimizar sus procesos para poder cumplir con las exigencias requeridas en la misma.

Responsable: Equipo Scrum.

## ¿Cuáles son los tipos de mapa de procesos?



**5. Casos de uso:** Independientemente de la metodología que se esté utilizando es la comunicación con el cliente, ya que es fundamental para definir los requerimientos de software porque muchas veces lo que se plantea no es lo que el cliente espera, por lo tanto, se definen formas de presentar al cliente una perspectiva de lo que será el software una vez terminado.

Los casos de uso son levantados los requerimientos por parte de negocio, con el apoyo técnico del analista de proyectos informáticos, el cual debe validarlos y oficializarlos.

Los casos de uso pueden tener varias versiones, según las necesidades y requerimiento de negocio, además de las correcciones realizadas por el analista de negocio.

Responsable: Analista de Proyectos, Product Owner.

*\*\*Deberán ser entregadas todas las versiones del caso de uso realizadas.*

**6. Desarrollo:** Desarrollar un software significa construirlo simplemente mediante su descripción. Esta es una muy buena razón para considerar la actividad de desarrollo de software como una ingeniería. En un nivel más general, la relación existente entre un software y su entorno es clara ya que el software es introducido en el mundo de modo de provocar ciertos efectos en el mismo.

*Para esta fase es necesario contar con las versiones oficiales de los casos de uso.*

Se debe integrar el proyecto en ambiente local, ambiente de pruebas y ambiente de producción. *Para cada publicación deberá generar una receta de publicación en producción, en donde se encuentran las instrucciones para el despliegue.*



**Pruebas unitarias:** consisten en verificar el comportamiento de las unidades más pequeñas de su aplicación. Técnicamente, eso sería una clase o incluso un método de clase en los lenguajes orientados a objetos, y un procedimiento o función en los lenguajes procedimentales y funcionales.

Responsable: Analista de Desarrollo.

**7. Pruebas funcionales y no funcionales, pruebas de aceptación:** Las pruebas son básicamente un conjunto de actividades dentro del desarrollo de software. Dependiendo del tipo de pruebas, estas actividades podrán ser implementadas en cualquier momento de dicho proceso de desarrollo. Existen distintos modelos de desarrollo de software, así como modelos de pruebas. A cada uno corresponde un nivel distinto de involucramiento en las actividades de desarrollo.

- Pruebas funcionales, se deberán mostrar en la plantilla de “informe de resultados”.
- Pruebas no funcionales (pruebas de carga y estrés), se deberán realizar por las herramientas indicadas en clase (selenium, katalon, jmeter, soap, cypress, jest, loadRunner), se mostrarán los resultados en la reunión de control de cambios y se mostrarán los resultados en el “informe de resultados”.

*Las pruebas funcionales y no funcionales deben ser satisfactorias, para que se pueda informar al Product Owner, para la generación de las pruebas de aceptación.*

- Pruebas de aceptación, deberá realizarlo el Product Owner si el resultado de las pruebas es satisfactorio, se indica por medio de una carta de aceptación en donde se agrega la fecha de publicación en producción del proyecto. Dicha carta debe ser informada al Analista de Aseguramiento de Calidad.

Responsable: Analista de aseguramiento de calidad, Product Owner

**8. Pruebas de seguridad:** Las pruebas de seguridad de aplicaciones (AST) son un proceso de identificación, análisis y corrección de las vulnerabilidades de seguridad de una aplicación web.

La seguridad del software es el concepto de implementar mecanismos en la construcción de la seguridad para ayudarla a permanecer funcional (o resistente) a los ataques

Responsable: Analista de Aseguramiento de Calidad.

**9. Control de cambios:** reunión de publicación del proyecto.

- Control de cambios del proyecto: Se realiza el control integrado de cambios del proyecto, debido a que es necesario contar con el proceso de supervisar y controlar los trabajos del proyecto.
- Beneficio del control integrado de cambios: Reduce el riesgo del proyecto. El riesgo del proyecto aumenta si implementamos cambios en el proyecto sin un control adecuado.



- Los objetivos del Control Integrado de Cambios: Establece un método progresivo para identificar y solicitar cambios a las líneas base del proyecto y para determinar el valor y la eficacia de esos cambios.

Responsable: Equipo Scrum.

#### 10. Lecciones aprendidas:

Las lecciones aprendidas en un proyecto son el conocimiento adquirido a través de experiencias, exitosas o no, durante la realización de un proyecto, con el fin de mejorar ejecuciones futuras. Las lecciones aprendidas capturan evidencias e identifican tendencias y relaciones causa-efecto, acotadas a un contexto específico, y sugieren recomendaciones prácticas y útiles para la aplicación o replicación del nuevo conocimiento en otros contextos y en el diseño y/o ejecución de otros proyectos.

***\*\*Para el proyecto, la reunión de control de cambios consistirá en la presentación del proyecto en producción, presentación de documentación, pruebas funcionales y no funcionales y de seguridad.***

Responsable: Equipo Scrum.

#### Proyectos sugeridos

##### Descripción

Debe presentar un software en una plataforma web, la aplicación deben interactuar dentro del modelo de negocio descrito en el siguiente apartado, debe utilizar un motor de bases de datos para almacenar las operaciones descritas.

##### Modelo del Negocio

1. Sistema de gasolinera
2. Sistema de hospital
3. Sistema de hotelería
4. Sistema de restaurante
5. Sistema de ventas online
6. Sistema de supermercado

##### Requerimientos mínimos del Modelo de Negocio

##### 1. Sistema de gasolinera

- Registro y Mantenimiento:
  - Usuarios -Perfiles (mantenimiento, consulta, pedido, ventas, abastecimiento, administrador)
  - Gasolinera
  - Tipos de combustible
  - Contenedores, Unidad de Medida
  - Personal, Puesto
  - Clientes, Proveedores



- Pedidos y ventas (Kardex)
- Suministro y distribución en contenedores (Kardex)
- Reportes
  - Existencia de combustible (búsqueda por id de combustible)
  - Pedidos y Ventas (búsqueda por id de transacción)
  - Kardex (filtro por fecha y por id combustible)
  - Suministro y distribución en contenedores (filtrar por id de contenedor y rango de fechas)

## 2. Sistema de hospital

- Registro y Mantenimiento:
  - Usuarios -Perfiles (consulta médica, pedidos y ventas de medicamento, administrador)
  - Hospital
  - Medicamento, Unidad de Medida
  - Personal, Puesto, Especialización
  - Pacientes, Proveedores
  - Asistencia Médica (clínica, área, habitación)
  - Consulta Médica (Receta y/o tratamientos)
  - Historial clínico
  - Pedido y venta de medicamento (Kardex)
- Reportes
  - Existencia de medicamento (búsqueda por id de medicamento)
  - Pedidos y Ventas (búsqueda por id de transacción)
  - Kardex (filtro por fecha y por id medicamento)
  - Consulta Médica (Filtrar por id consulta y/o paciente)
  - Recetas (Filtrar por id Receta y/o paciente)
  - Historial clínico (filtrar por id paciente)

## 3. Sistema de hotelería

- Registro y Mantenimiento:
  - Usuarios -Perfiles (consulta, reservaciones, administrador)
  - Hotel
  - Personal, Puesto
  - Clientes
  - Habitaciones, servicios varios (parqueo, uso de instalaciones, excursiones, etc.)
  - Reservación de habitación (online y offline)
  - Servicio a la habitación
  - Facturación de hospedaje y alimento (kardex)
  - Promociones
- Reportes
  - Planilla laboral (filtrar por mes y año)
  - Reservaciones (filtrar por id de reservación y/o id cliente)
  - Disponibilidad de habitación (filtrar por id habitación)
  - Facturación (filtrar por id factura)
  - Promociones (filtrar por mes y tipo de servicio)





#### 4. Sistema de restaurante

- Registro y Mantenimiento:
  - Usuarios -Perfiles (consulta, ordenes, administrador)
  - Restaurante
  - Personal, Puesto
  - Clientes, Proveedores
  - Alimento (tipo de alimento, Platos, Menú)
  - Orden de platillo (online y offline)
  - Facturación de ordenes de alimento (kardex)
- Reportes
  - Planilla laboral (filtrar por mes y año)
  - Ordenes de platillo (filtrar por id de orden y/o id cliente)
  - Facturación (filtrar por id factura)
  - Platos del restaurante (filtrar por tipo de comida)

#### 5. Sistema de ventas online

- Registro y Mantenimiento:
  - Usuarios -Perfiles (cliente, administrador)
  - Recuperación de contraseñas
  - Empresa
  - Contactos
  - Catálogo de Artículos (categoría, subcategoría)
  - Clientes, Proveedores
  - Tipos de pago (depósitos, tarjeta (paypal) )
  - Pedidos (Kardex)
  - Carrito de compras (Kardex)
  - Historio de transacciones
- Reportes
  - Existencia de artículos (búsqueda por id)
  - Pedidos y Ventas (búsqueda por id de transacción)
  - Kardex (filtro por fecha y por id articulo)
  - Histórico de transacciones (filtrar por id cliente)

#### 6. Sistema de supermercado

- Registro y Mantenimiento:
  - Usuarios -Perfiles (consulta, cajas de cobro, administrador)
  - Supermercado
  - Catalogo de artículos, Unidad de Medida (categoría, subcategoría)
  - Áreas o pasillos
  - Segmentación de articulo por área o pasillo
  - Código de barra (etiqueta automática de articulo)
  - Cajas de cobro
  - Personal, Puesto
  - Clientes, Proveedores
  - Compras y Ventas (kardex)



- **Reportes**

- Existencia de artículos (búsqueda por id)
- Compras y Ventas (búsqueda por id de transacción y rango de fechas)
- Kardex (filtro por fecha y por id articulo)
- Ventas realizadas (filtrar por día y/o mes, caja de cobro)
- Artículo disponible (filtrar por área o pasillo)