



Universidad de Buenos Aires  
Facultad de Ciencias Económicas



# Aseguramiento de la Calidad de los Sistemas de Información

AR Tema extractado del libro "**Análisis Funcional de Sistemas y Tecnologías de la Información**" de Aníbal M. Mazza Fraquelli - ISBN 978-987-26981-3-3

## 1. Presentación del Tema

El aseguramiento de calidad, conocido como *Quality Assurance* (QA), constituye un conjunto sistemático de actividades orientadas a garantizar que los procesos utilizados para desarrollar productos o servicios cumplan con estándares predefinidos de calidad. En el ámbito de las Tecnologías de la Información (TI), QA adquiere una relevancia central debido a la creciente dependencia organizacional de los sistemas de información, aplicaciones empresariales, plataformas digitales y servicios basados en software.

En términos generales, QA no se limita a la detección de errores, sino que se enfoca en la prevención de defectos a través de la definición, estandarización, monitoreo y mejora continua de procesos. En el contexto del desarrollo de software, QA implica establecer metodologías, criterios, métricas y controles que aseguren que el producto final cumpla con los requisitos funcionales y no funcionales definidos por la organización.

Es fundamental diferenciar QA de *Quality Control* (QC, Control de Calidad). Mientras QA se centra en los procesos que conducen al producto, QC se enfoca

en la verificación del producto final mediante pruebas, inspecciones y validaciones. Esta distinción resulta clave para estudiantes de licenciatura en administración, ya que permite comprender cómo la gestión de la calidad impacta en costos, tiempos, riesgos y competitividad organizacional.

Desde la perspectiva de TI, QA no es un complemento opcional del desarrollo de software; es un componente estructural del gobierno tecnológico y de la gestión estratégica de sistemas de información.

---

## 2. Desarrollo

### 2.1 Definición y alcance del Aseguramiento de Calidad (QA)

El Aseguramiento de Calidad (QA) puede definirse como el conjunto de políticas, procedimientos, estándares y actividades sistemáticas diseñadas para proporcionar confianza en que un producto o servicio cumplirá con los requisitos de calidad establecidos.

En el desarrollo de software, QA abarca:

- Definición de estándares de codificación.
- Revisión de requisitos.
- Gestión de riesgos.
- Auditorías de procesos.
- Métricas de desempeño.
- Evaluación de cumplimiento metodológico.
- Mejora continua.

QA se enmarca dentro de modelos de calidad como:

- ISO 9001 (Sistema de Gestión de Calidad).
- ISO/IEC 25010 (Modelo de Calidad de Producto de Software).
- Capability Maturity Model Integration (CMMI, Modelo de Madurez de Capacidad Integrada).
- ITIL (Biblioteca de Infraestructura de TI).

En estos marcos, QA es un componente transversal que asegura consistencia, repetibilidad y trazabilidad en los procesos.

Desde la administración, QA puede entenderse como un sistema de control preventivo que reduce la variabilidad, minimiza la incertidumbre y mejora la predictibilidad de resultados.

---

## 2.2 Diferencia entre QA y QC (Control de Calidad)

Una de las confusiones más frecuentes en entornos organizacionales es considerar que QA y QC son sinónimos. Sin embargo, conceptualmente y operativamente son distintos.

### 2.2.1 QA (Quality Assurance – Aseguramiento de Calidad)

- Se enfoca en el proceso.
- Es preventivo.
- Busca evitar defectos antes de que ocurran.
- Establece estándares y metodologías.
- Evalúa cómo se trabaja.

Ejemplo en software:

- Definir que todos los requisitos deben pasar por revisión formal.
- Establecer criterios de aceptación.
- Implementar integración continua (*Continuous Integration*, CI).
- Exigir revisión de código (*Code Review*).

QA responde a la pregunta:

¿Estamos desarrollando el software de manera correcta?

---

### 2.2.2 QC (Quality Control – Control de Calidad)

- Se enfoca en el producto.
- Es correctivo.
- Detecta defectos ya existentes.

- Se basa en pruebas e inspecciones.
- Evalúa qué se entregó.

Ejemplo en software:

- Ejecutar pruebas funcionales.
- Realizar pruebas de regresión.
- Detectar errores en la interfaz.
- Validar cumplimiento de requisitos.

QC responde a la pregunta:

¿El producto desarrollado cumple con lo esperado?

## 2.3 Diferencias clave desde la perspectiva administrativa

Dimensión	QA	QC
Enfoque	Proceso	Producto
Naturaleza	Preventiva	Correctiva
Momento de aplicación	Durante todo el ciclo de vida	En etapas de validación
Objetivo	Evitar defectos	Detectar defectos
Impacto económico	Reduce costos futuros	Puede generar retrabajo

Desde la administración, QA tiene un efecto estratégico porque:

- Reduce costos de fallas tardías.
- Disminuye retrabajo.
- Mejora previsibilidad.
- Aumenta confianza del cliente.
- Fortalece gobernanza de TI.

En cambio, depender exclusivamente de QC implica asumir que los errores serán detectados al final, lo que suele resultar más costoso.

## 2.4 Importancia de QA en el desarrollo de software

El desarrollo de software es una actividad compleja, caracterizada por:

- Alta interdependencia entre componentes.
- Cambios frecuentes en requisitos.
- Integración con múltiples sistemas.
- Riesgos técnicos y organizacionales.
- Impacto directo en operaciones del negocio.

En este contexto, la ausencia de QA puede generar:

- Fallas en producción.
- Vulnerabilidades de seguridad.
- Incumplimiento de normativas.
- Pérdida de datos.
- Daño reputacional.

Desde la perspectiva de sistemas de información, un software defectuoso no es solo un problema técnico, sino un riesgo organizacional.

---

## **2.5 QA en el ciclo de vida del desarrollo de software (SDLC)**

El *Software Development Life Cycle* (SDLC, Ciclo de Vida del Desarrollo de Software) incluye fases como:

1. Análisis de requisitos.
2. Diseño.
3. Desarrollo.
4. Pruebas.
5. Implementación.
6. Mantenimiento.

QA debe estar presente en todas las fases.

### **En requisitos:**

- Validación de claridad y completitud.

- Identificación de ambigüedades.

### **En diseño:**

- Revisión de arquitectura.
- Evaluación de escalabilidad.

### **En desarrollo:**

- Estándares de codificación.
- Integración continua.

### **En pruebas:**

- Estrategias de test planificadas.
- Métricas de cobertura.

### **En implementación:**

- Validación de despliegue.
- Planes de contingencia.

### **En mantenimiento:**

- Monitoreo de desempeño.
- Gestión de incidencias.

QA es transversal; QC es una fase específica dentro del ciclo.

---

## **2.6 QA en metodologías ágiles**

En metodologías ágiles como Scrum o Kanban, QA no es una etapa separada, sino una práctica integrada.

Conceptos relevantes:

- Definition of Done (Definición de Terminado).
- Test-Driven Development (TDD, Desarrollo Guiado por Pruebas).
- Continuous Integration (CI, Integración Continua).

- Continuous Delivery (CD, Entrega Continua).

En estos enfoques, QA:

- Participa desde la planificación.
- Define criterios de aceptación.
- Colabora con desarrolladores.
- Automatiza pruebas.
- Monitorea métricas.

La calidad se construye desde el inicio, no se verifica únicamente al final.

---

## 2.7 Impacto económico y estratégico del QA

Desde la teoría de costos de calidad, se distinguen:

- Costos de prevención.
- Costos de evaluación.
- Costos de fallas internas.
- Costos de fallas externas.

QA incrementa costos de prevención, pero reduce costos de fallas externas, que son significativamente más altos.

Ejemplo:

Un error detectado en fase de diseño puede costar 1 unidad.

El mismo error detectado en producción puede costar 100 unidades.

Para estudiantes de administración, esta lógica evidencia que QA es una inversión estratégica, no un gasto operativo.

---

## 2.8 QA y gobernanza de TI

La gobernanza de TI implica asegurar que las tecnologías soporten los objetivos estratégicos del negocio.

QA contribuye a:

- Cumplimiento normativo.

- Gestión de riesgos.
- Transparencia de procesos.
- Trazabilidad.
- Auditoría tecnológica.

En organizaciones reguladas (banca, salud, seguros), QA es esencial para cumplir estándares y evitar sanciones.

---

## 2.9 Ejemplo aplicado

Caso hipotético:

Una empresa implementa un sistema de facturación electrónica sin un proceso formal de QA.

Consecuencias:

- Errores en cálculos impositivos.
- Retrabajo masivo.
- Sanciones regulatorias.
- Pérdida de credibilidad.

Si la empresa hubiese implementado:

- Revisión formal de requisitos.
- Pruebas automatizadas.
- Validaciones fiscales previas.
- Controles de integración.

El riesgo se habría reducido significativamente.

QA, en este contexto, actúa como mecanismo de mitigación de riesgo.

---

## 3. Conclusión

El Aseguramiento de Calidad (QA) es un componente estructural del desarrollo de software y de la gestión moderna de Tecnologías de la Información. Su

enfoque preventivo y sistémico lo diferencia del Control de Calidad (QC), que actúa de manera correctiva sobre el producto final.

Comprender esta diferencia es esencial para futuros profesionales en administración, ya que la calidad del software impacta directamente en:

- Costos.
- Riesgos.
- Reputación.
- Cumplimiento normativo.
- Continuidad operativa.

QA no es una actividad aislada ni una función secundaria; es una práctica transversal que garantiza coherencia entre procesos, tecnología y objetivos estratégicos. En entornos digitales, donde los sistemas de información sostienen la operación organizacional, asegurar la calidad no es opcional: es una condición necesaria para la competitividad y sostenibilidad empresarial.

---

## Preguntas de autoevaluación

1. Explique la diferencia conceptual y operativa entre QA y QC.
  2. ¿Por qué QA se considera un mecanismo preventivo dentro del desarrollo de software?
  3. Analice el impacto económico de detectar un defecto en producción en comparación con detectarlo en fase de diseño.
  4. ¿Cómo contribuye QA a la gobernanza de TI y a la gestión de riesgos organizacionales?
  5. Describa cómo se integra QA dentro de metodologías ágiles como Scrum.
-

## Material de Clases

Compilado por **Aníbal M. Mazza Fraquelli** Doctor de la Universidad de Buenos Aires para el uso de sus clases en la Facultad de Ciencias Económicas de la Universidad de Buenos Aires.

---

### Contenidos de esta página

Los contenidos **aquí incluidos integran desarrollos y escritos propios del autor, así como materiales de terceros (documentos, textos, fragmentos, conceptos, imágenes, esquemas, definiciones u otros recursos)**, los cuales son utilizados a título ilustrativo, explicativo o formativo, respetando la normativa vigente en materia de derechos de autor y citando las fuentes cuando corresponde.

**La selección, organización, adaptación pedagógica y contextualización de los contenidos constituye un trabajo original del autor, orientado a facilitar los procesos de enseñanza y aprendizaje.**

**Este material no persigue fines comerciales y su reproducción, total o parcial, queda limitada al ámbito educativo, debiendo preservarse siempre la mención de la autoría y las fuentes originales.**

---

### Autorización de uso

Se permite la reproducción, comunicación pública, distribución y utilización total o parcial de los contenidos de su material, en formato físico o digital, con fines exclusivamente educativos, académicos o de divulgación, siempre que se respete la integridad del contenido y se incluya la correspondiente referencia a la fuente y a la autoría.

**Las ideas, opiniones e interpretaciones contenidas en este material corresponden exclusivamente al autor.**

**Queda expresamente excluido cualquier uso con fines comerciales.**