**Plan de calidad de software**

Proyecto: Salón De Belleza TYA

Versión: 1.0.

Historial de Revisiones

|  |  |  |  |
| --- | --- | --- | --- |
| **VERSIÓN** | **FECHA** | **AUTOR** | **DESCRIPCIÓN** |
| 1-0 | 03/06/2020 | Grupo | Se comenzó la guía y se ajustaron ciertas cosas |
| 2.0 | 09/06/2020 | Grupo | Se siguió con la realización de la guia |
| 3.0 | 19/06/2020 | Grupo | Se termino la guia, pero se vuelve a revisar y se encuentran errores |
| 4.0 | 24/06/2020 | Grupo | Queda casi lista y se mejora el contenido actual. |
| 5.0 | 25/06/2020 | Grupo | Se hicieron los últimos ajustes del documento |

**CONTENIDO**

Contenido

[1. Introducción: 5](#_Toc43926759)

[2. Objetivo: 5](#_Toc43926760)

[2.1 Objetivos de SQA 6](#_Toc43926761)

[3. Documentos Relacionados 6](#_Toc43926762)

[4. Destinatarios 6](#_Toc43926763)

[5. Administración – Planeación 7](#_Toc43926764)

[5.1 Organización 7](#_Toc43926765)

[5.2 Responsabilidades 8](#_Toc43926766)

[**Rol** 8](#_Toc43926767)

[**Encargado** 8](#_Toc43926768)

[**Funciones** 8](#_Toc43926769)

[**Gerente del proyecto** 8](#_Toc43926770)

[Sebastián Rodriguez 8](#_Toc43926771)

[Se encarga de firmar y verificar los documentos legales 8](#_Toc43926772)

[**Desarrollador de Software** 8](#_Toc43926773)

[Zully Tamayo 8](#_Toc43926774)

[Se encarga de la programación y es la que verifica el correcto funcionamiento. 8](#_Toc43926775)

[**Gerente SQA** 8](#_Toc43926776)

[Juan Moreno 8](#_Toc43926777)

[Es el encargado de SQA, previene posibles errores y hace planes de si llega a pasar un error no se detenga el proyecto, hace que los errores no sean imprevistos 8](#_Toc43926778)

[**Documentación** 8](#_Toc43926779)

[Dayanna Cifuentes 8](#_Toc43926780)

[Se encarga de hacer la documentación necesaria para mitigar el impacto de los errores, y se tenga un claro paso a paso para que todo el equipo sepa que hacer cuando ocurran errores. 8](#_Toc43926781)

[5.3 Cronograma del proyecto 9](#_Toc43926782)

[**5.3.1 recursos** 9](#_Toc43926783)

[5.4 Riesgos del proyecto 10](#_Toc43926784)

[6. Estándares, Practicas, Convenciones y Mediciones 10](#_Toc43926785)

[6.1 Estándares 10](#_Toc43926786)

[7. Métricas de Calidad 10](#_Toc43926787)

[8. Pruebas del Software 11](#_Toc43926788)

[9. Costos Asociados a la Calidad 11](#_Toc43926789)

[10. Reportes De Problemas Y Acciones Correctivas 11](#_Toc43926790)

[11. Auditorias de Calidad 12](#_Toc43926791)

[12. Solicitudes de cambio 12](#_Toc43926792)

[13. Apéndices 12](#_Toc43926793)

[Glosario 13](#_Toc43926794)

[14. Referencias 14](#_Toc43926795)

# Introducción:

SQA como disciplina tiene muchas ventajas, pero las más destacadas e importantes son por ejemplo la reducción de costos, pues hace que minorice la utilización de recursos y por afanes que se puedan dar ante un error; en el sentido que, por querer menos errores, paguen y paguen dinero extra. También mejora notablemente la eficiencia que se puede dar en el desarrollo de software, literalmente si mejora la eficiencia notablemente, lo cual trae de la mano; que disminuye extender plazos en la entrega final del proyecto. Aumente el nivel de satisfacción del cliente, o por lo menos aumenta la posibilidad de que el nivel de satisfacción sea mayor, lo cual es gratificante para ambas partes, pues el cliente no tiene quejas, recomienda la empresa, y la empresa aumenta su reconocimiento y su calidad queda intacta, además es gratificante hacer las cosas bien. Si se tiene un SQA evita en su gran mayoría, por no decir que en su totalidad, el volver hacer un trabajo que ya se hizo, o en otras palabras el retrabajo, pues siendo ordenado el trabajo, dejando claro cada paso y el papel que le corresponde a cada persona, hace que por ejemplo dos personas no hagan una misma cosa, lo cual es pérdida de tiempo, de dinero y se vuelve notable la falta de comunicación dentro de la empresa. Genera mayor confianza dentro de la empresa pues, al hacer que un proyecto salga satisfactoriamente hace que ya se tenga como un paso a paso correcto y si se hace tal cual el resultado sea el mismo, y se pueda hacer en menos tiempo, ya que los empleados van a desarrollar esa habilidad.

El SQA ocupa un papel importante en el proyecto, y debe estar implementado de una manera correcta, puesto que hará que en un futuro hallan perdidas monetarias, o que la confianza del cliente disminuya, es mejor que la confianza desde el comienzo sea buena, y eso lo logramos, dando un producto de calidad, que todo el desarrollo sea trasparente, también manteniendo una muy buena comunicación con el cliente o los clientes, haciéndolo participe en cada fase del proyecto, y que tenga acceso libre a cada fase para que note el profesionalismo y que se dé cuenta como se hacen las cosas, y que vea que cada peso que está invirtiendo vale la pena y que no se le está estafando.

# Objetivo:

* Monitorear correctamente todos los posibles errores que pueda tener el sistema.
* Realizar un estudio general de todo el producto para realizar mejoras al mismo.
* Reunir todo el personal correspondiente para realizar auditorías sobre la calidad del producto.
* Aplicar estándares de calidad de software para obtener certificados de calidad de software.
* Realizar controles correctivos a los errores que surjan por medio de los controles.

## 2.1 Objetivos de SQA

Los principales objetivos del Aseguramiento de la Calidad del Software son los siguientes:

* Mejorar la calidad del software monitoreando apropiadamente tanto los productos de software como el proceso de desarrollo que los genera.
* Asegurar el cumplimiento de los estándares y procedimientos establecidos para el software y el proceso de software establecidos.
* Asegurar que cualquier desviación en el producto, el proceso, o los estándares son elevados a la gerencia para poder resolverlas.

# Documentos Relacionados

Repositorio de los documentos mencionados: <https://drive.google.com/drive/folders/13gSwHQ6clFcheBgt4sebB1QVXb5PVB9R>

|  |
| --- |
| DOCUMENTOS RELACIONADOS |
| Diagrama Casos de Uso |
| MER Crows foot |
| Bpmn |
| Normalización Base de Datos |
| Informe General |
| Manual Usuario |
| Manual Instalación |
| Manual Técnico |
| Plantilla requerimientos IEEE 830 |
| Encuestas de levantamiento de información. |
| Pruebas de migración remota PGSQL |
| Pruebas MySql |

# Destinatarios

El presente documento está destinado a las siguientes personas:

|  |  |
| --- | --- |
| **Lector** | **Sector o Rol** |
| Viviana Andrea Tamayo Martínez | Administradora |

# Administración – Planeación

El presente documento tendrá un grupo SQA, gerente de proyecto, desarrollador y documentadora es importante destacar que todo tiene un protocolo de realización y cada persona tiene un papel especifico la cual hace más fácil la división de responsabilidades.

## 5.1 Organización

Actualmente el equipo de proyecto se reúne virtualmente debido a la crisis sanitaria que está atravesando el país, con el fin de repartir tareas, para revisar y evaluar trabajos, dialogar acerca de problemas con el proyecto o en caso de cambios y demás.

A todas los integrantes del grupo se reparten actividades por igual, unas se encargan del código y otras de documentación, para así llevar una buena organización y en caso de requerir ayuda de todos, se realiza nuevamente una reunión virtual en la que se plantean los problemas y como equipo se busque resolver.

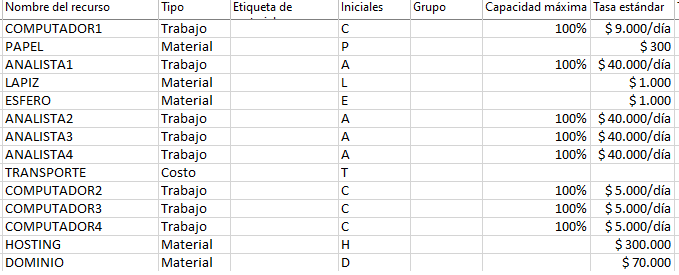
## 5.2 Responsabilidades

|  |  |  |
| --- | --- | --- |
| **Rol** | **Encargado** | **Funciones** |
| **Gerente del proyecto** | Sebastián Rodriguez | Se encarga de firmar y verificar los documentos legales |
| **Desarrollador de Software** | Zully Tamayo | Se encarga de la programación y es la que verifica el correcto funcionamiento. |
| **Gerente SQA** | Juan Moreno | Es el encargado de SQA, previene posibles errores y hace planes de si llega a pasar un error no se detenga el proyecto, hace que los errores no sean imprevistos |
| **Documentación** | Dayanna Cifuentes | Se encarga de hacer la documentación necesaria para mitigar el impacto de los errores, y se tenga un claro paso a paso para que todo el equipo sepa que hacer cuando ocurran errores. |

## Cronograma del proyecto



### **5.3.1 recursos**



## Riesgos del proyecto

Los riesgos negativos que se podrían presentar:

1. Fallas críticas en la base de datos
2. Fallos en el internet local
3. Fallos en el servidor
4. Demoras en respuesta de internet local
5. Información incorrecta
6. Inestabilidad en el sistema
7. Desactualización del navegador utilizado.
8. Mala manipulación del sistema
9. Cambio de personal, a un nuevo personal no capacitado para la manipulación del sistema.
10. Computadores no aptos.
11. Los componentes técnicos tienen vulnerabilidades de seguridad.
12. Los requisitos son incompletos.
13. Falta de mantenimiento.

# Estándares, Practicas, Convenciones y Mediciones

El estándar que se va a aplicar al proyecto va a ser el modelo **ISO 25000**.

## 6.1 Estándares

A continuación, se especifican los estándares para definir diferentes aspectos del proyecto.

* IEEE 830.
* Arquitectura de programación: MVC (Modelo Vista Controlador).
* Modelo de Casos de Uso:
* Modelos de Procesos:
* Proceso para registrarse en el sistema.
* Proceso para loguearse en el sistema.
* Proceso para agendar una cita.
* Proceso para agendar una cita.
* Proceso para recuperar contraseña.
* Estilo de programación: POO (Programación orientada a objetos).

# Métricas de Calidad

Enlace de métricas: <METRICAS.xlsx>

# Pruebas del Software

* Pruebas Jmetter: <salontya.jmx>
  + Documentacion de pruebas Jmetter: [PRUEBAS JMETTER.docx](PRUEBAS%20JMETTER.docx)

# Costos Asociados a la Calidad

|  |  |  |
| --- | --- | --- |
| Rol | Costo | Descripción |
| Empresa de pruebas | 350.000/ semanalmente | Persona encargada a verificar la funcionabilidad, los errores y las posibles debilidades de seguridad. |
| Costos de conexión | 20.000/ semanalmente | Plan de internet local con 10mbs las cuales pueden ser suficientes para un buen funcionamiento |

# Reportes De Problemas Y Acciones Correctivas

**Análisis de problemas**

* Se realizan diferentes tipos de pruebas al aplicativo: Pruebas de estrés, pruebas de base de datos, pruebas hosting etc.
* Se buscan falencias en el sistema, el aplicativo debe tener un tiempo de respuesta óptimo, ya que si tarda más de 1.5 segundos en responder, ya se considera como un punto crítico.
* También se analizan diferentes fallas a nivel de procesos de desarrollo que puedan ahorrar tiempo y presupuesto.
* Se busca que todos los miembros del equipo sean productivos y hagan aportes positivos al proyecto.
* Se verifica que el sistema cumpla con los requisitos ya definidos para evitar un desfase en las expectativas
* Se cuadran sesiones para hablar y pensar las acciones correctivas a realizar, y se da un tiempo determinado para que esas acciones correctivas se realicen.
* Se vuelve a cuadrar una sesión para verificar que las acciones correctivas se hallan hecho y se vuelve a observar el comportamiento del sistema, si en dado caso que el problema siga la sesión se alarga hasta que se pueda pensar que error puede ser.
* Si se corrigen los errores se le da el visto bueno y se programa una sesión cada semana para verificar que no hallan fallos.
* En las diferentes pruebas que se realizaron con el software JMetter se encontraron dos errores a la hora de querer restablecer la contraseña por medio del correo electronico, estos errores surgían porque el tiempo de respuesta por parte del sistema eran de mas de 700 milisegundos, esto ocasionaba que el correo para restablecer la contraseña se demorara mucho tiempo en llegar al usuario. Estos errores se soluciuonaron corrigiendo parte del codigo, asi mismo se logro que el correo de restablecimiento llegara casi instantáneamente el usuario realizara la peticion.

# Auditorias de Calidad

De acuerdo con el PMBOK, una auditoria de calidad es una revisión estructurada e independiente para determinar si las actividades del proyecto cumplen con las políticas, los procesos y los procedimientos del proyecto y de la organización. Los objetivos de una auditoria de calidad son:

• Identificar todas las buenas y mejores prácticas empleadas.

• Identificar todas las diferencias y anomalías.

• Compartir las buenas prácticas introducidas o implementadas en proyectos similares de la organización y/o industria.

• Ofrecer asesoramiento de manera positiva y proactiva, para mejorar la implementación de procesos que ayuden al equipo a incrementar la productividad.

• Resaltar las contribuciones de cada auditoría en la base de datos de lecciones aprendidas de la organización.

El equipo del proyecto realizara las auditorias de calidad en aproximadamente 4 días para hacer más eficiente el cumplimiento del proyecto, esto se hará en un salón que la empresa nos facilite. Se comenzará con una breve introducción por parte del equipo de desarrollo en donde se explicarán los objetivos de la auditoria, a continuación, la persona encargadas de exponer el plan de pruebas y así sucesivamente hasta finalizar con la sustentación del proyecto.

# Solicitudes de cambio

Una vez el equipo de evaluación de software haga el respectivo análisis del proyecto y hayan encontrado errores o inconvenientes que requieran mejoría, se le comunicará al líder del equipo de evaluación los errores encontrados, el líder de calidad de software se comunicará con el líder del proyecto para que esté se ponga en contacto con los desarrolladores del proyecto y así ellos hagan las correcciones respectivas.

# Apéndices

Repositorio Github: <https://github.com/zheta88/salon_tya>

servidor: Heroku

url proyecto salontya: <https://salondebellezatya.herokuapp.com/>

* Prueba de estres: <salontya.jmx>
  + Documentación de pruebas Jmetter: [PRUEBAS JMETTER.docx](PRUEBAS%20JMETTER.docx)

Metricas: <METRICAS.xlsx>

Plan de riesgo: <https://drive.google.com/file/d/1ERgkWdKeck71oqBcTeRbh_IeFn-CDf6k/view?usp=sharing>

# Glosario

SQA: SQA es un set de actividades sistemáticas que aseguran que el proceso del software y productos conformados por requerimientos, estándares, y procedimientos. Los procesos incluyen todas las actividades involucradas en el diseño, codificación, pruebas y mantenimiento; Los productos incluyen software, datos asociados, documentación, y toda la documentación para soporte y reportes. (Ecured, s.f.)

Software: Conjunto de programas y rutinas que permiten a la computadora realizar determinadas tareas.

Apéndices: Los apéndices, son las creaciones del autor, como, por ejemplo, los cuestionarios que sirvieron para recabar información, las fotos que se hayan tomado de determinado entorno, las pruebas que aplico, entre otros. (Tesis y monograficos ideas originles, 2014)

Ieee830: Es un estándar de requerimientos funcionales y no funcionales los cuales debe tener en su gran mayoría implementado en cada software que se haga, es un estándar el cual sirve para dejar muy claro que va a tener y como lo va a hacer el software, entre otras cosas.

MVC: sus siglas significan “modelo vista controlador”, es un estilo de arquitectura de software que separa el código del cual se está trabajando, separa los datos de la aplicación, la interfaz de usuario y la lógica de control en tres componentes distintos.

POO: Programación orientada a objetos, es un paradigma de la programación que viene a innovar la forma de obtener resultados. Es como un estándar dentro del código para hacer lo más reutilizable posible el código, y el cual hace que se comuniquen las clases más fácilmente sin malgastar código, haciendo que el código no vuelva a crear cosas o funciones que ya están en otra clase, si no lo que hace es llamar eso que ya se está creando en otra clase distinta.

Casos de uso: es un diagrama de software es cual describe la acción de cómo se comunica un proyecto para modelarlo más fácilmente, también es la descripción de una actividad de cómo se debería o se realiza actualmente.

MER: Modelo entidad relación, el cual es un diagrama que se utiliza para modelar la base de datos, tipos de datos y como se comunican las tablas que tiene, es muy necesario para la creación de base de datos del proyecto informático.

Iso9126: es un estándar internacional para evaluar la calidad de software de un proyecto, a base de unos parámetros establecidos.

Iso25000: es un estándar internacional para evaluar la calidad de software de un proyecto, pero la diferencia del 9126 es que el 25000 es más actualizado y viene mejorado, se podría decir que es la actualización del 9126.

Github: GitHub es un sistema de gestión de proyectos y control de versiones de código, así como una plataforma de red social diseñada para desarrolladores. ¿Pero para qué se usa GitHub? Bueno, en general, permite trabajar en colaboración con otras personas de todo el mundo, planificar proyectos y realizar un seguimiento del trabajo. (Hostinger tutoriales, 2019)

Jmetter: Es una herramienta de carga para llevar acabo simulaciones sobre cualquier recurso de Software. (Osmosis Latina, s.f.)

Heroku: Heroku es una plataforma en la nube que permite a las empresas construir, entregar, supervisar aplicaciones y alojarlas en la nube. (Romero, s.f.)

MySql: MySQL es un sistema de administración de bases de datos (Database Management System, DBMS) para bases de datos relacionales. (esepestudio, s.f.)

PostgreSQL: PostgreSQL es una de las opciones más interesantes en bases de datos relacionales open-source. (reicek, 2015)

# Referencias

Ecured. (s.f.). *Ecured*. Obtenido de Ecured: Ecured

Ecured. (s.f.). *Ecured*. Obtenido de Ecured: https://www.ecured.cu/SQA#:~:text=SQA%20es%20un%20set%20de,requerimientos%2C%20estándares%2C%20y%20procedimientos.

esepestudio. (s.f.). Obtenido de https://www.esepestudio.com/noticias/que-es-mysql#:~:text=Analítica%20digital-,¿Qué%20es%20MySQL%3F,llamados%20de%20bases%20de%20datos.

Hostinger tutoriales. (13 de Mayo de 2019). *Hostinger tutoriales*. Obtenido de Hostinger tutoriales: https://www.hostinger.co/tutoriales/que-es-github/

Osmosis Latina. (s.f.). *Osmosis Latina*. Obtenido de Osmosis Latina: https://www.osmosislatina.com/jmeter/basico.htmOsminosis Latin

Platzi. (s.f.).

reicek. (2015). *Platzi*. Obtenido de Platzi: https://platzi.com/blog/que-es-postgresql/

Romero, E. (s.f.). Obtenido de https://estebanromero.com/herramientas-emprender-desarrollar-proyectos/heroku-una-plataforma-para-la-creacion-de-aplicaciones/

Tesis y monograficos ideas originles. (18 de 05 de 2014). *Tesis y monograficos ideas originles*. Obtenido de Tesis y monograficos ideas originles: http://tesisymonograficos.blogspot.com/2014/05/apendice-y-anexo-diferencia.html