**UNIVERSIDAD RICARDO PALMA**

**FACULTAD DE INGENIERÍA**

**ESCUELA POSGRADO**

**“Proyecto Aplicación móvil de reserva de servicios de SPA”**

****

**PRESENTADO POR:**

1. ESPINOZA DELGADO JOSE
2. NOVOA TAFUR MANUEL
3. PALOMINO NARVAEZ CLAUDIA
4. PALOMINO QUICHCA ANTONIO
5. SAMANIEGO ZANABRIA ANA

**LIMA PERU**

**2017**

APLICACIÓN MOVIL PARA RESERVAS Y PAGOS DE SERVICIO DE SPA

Plan de SQA

Versión [1.0]

Historia de revisiones

|  |  |  |  |
| --- | --- | --- | --- |
| Fecha | Versión | Descripción | Autor |
| 22/04/2017 | 1.0 | Creación del Plan | G1 |

1. Propósito

Los establecimientos de belleza brindan servicios estética y masajes al público en general. Asimismo, cuentan con un área comercial que es la encargada de llevar el control manual de la información sobre diversos puntos de reservas que realizó el cliente.

EasyCode es el dueño de la idea de negocio, que consiste en implementar una aplicación móvil que permitirá realizar a los clientes reservas y pagos de servicios online en el spa de su preferencia. La aplicación móvil estará publicada en google play.

Este Plan solo cubre la parte del ciclo de vida correspondiente al desarrollo del software.

1. Gestión

La gestión del proyecto está a cargo del Jefe del Proyecto, sin embargo, será monitoreada tanto por este, como por el Responsable de SQA. Se intenta controlar que las actividades se ajusten al plan propuesto y minimizar posibles desviaciones.

* 1. Organización

La mayoría de las actividades realizadas durante el proyecto impactan, en mayor o menor medida, en la calidad del producto final. Las líneas de trabajo con un impacto más directo son:

• Concepto

• Especulación

• Exploración

• Revisión

• Cierre

El equipo de trabajo está estructurado de la siguiente forma:

|  |  |
| --- | --- |
| **Roles** | **Funciones** |
| Jefe del Proyecto | El Jefe del proyecto será responsable de la ejecución y entrega del proyecto.  Reportara directamente y cada fin de semana sobre el avance.  Define el equipo, su organización y gestión.  Aprueba los procesos. |
| Analista Programador | Participa en la automatización del proceso, desarrolla las aplicaciones y organiza los datos.  Reporta directamente al Gerente del Proyecto |
| Analista Funcional | Convierte la informalidad de las necesidades del cliente a especificaciones y requerimientos formales desde el punto de vista de ingeniería de software.  Elaborar el análisis de las funcionalidades para el proyecto, así como actualizar y mejorar las conceptualizaciones; es decir, debe trabajar de manera conjunta con el desarrollo funcional de las aplicaciones informáticas, asegurando su correcta explotación y su óptimo rendimiento.  Evaluar tanto la viabilidad técnica de los desarrollos de las aplicaciones que se han de ejecutar, y preparar y elaborar toda la documentación técnica y de usuario de cada aplicación. |
| Analista de Calidad | Es el responsable del orden y analisis de los documentos de gestión  Participa en la automatización del sistema en sus respectivos entregables asegurando la calidad de los mismos.  Reporta directamente al Gerente del Proyecto |

**ORGANIGRAMA DEL PROYECTO**

* 1. Actividades
     1. Ciclo de vida del software cubierto por el Plan

Como ya se mencionó el presente Plan comprende las etapas de: Concepto, Especulación, Exploración, Revisión y Cierre. Así mismo, se revisará la calidad de los siguientes productos:

Dentro del Concepto:

• Estimación del Backlog.

* Estimación del Sprint.

Pautas para la interfaz del usuario Dentro del área de Diseño:

• Desarrollo del Sprint

• Interpretación del diagrama Burdown.

Descripción de la arquitectura Implementación, Verificación e Implantación:

• Informe de verificación unitaria

• Plan de Verificación y validación

Plan de Implantación Dentro de la Gestión del Proyecto:

• Plan de Proyecto

• Gestión de Riesgos

• Plan de iteración

Gestión de la Configuración y Control de Cambios:

• Plan de Configuración de SCM

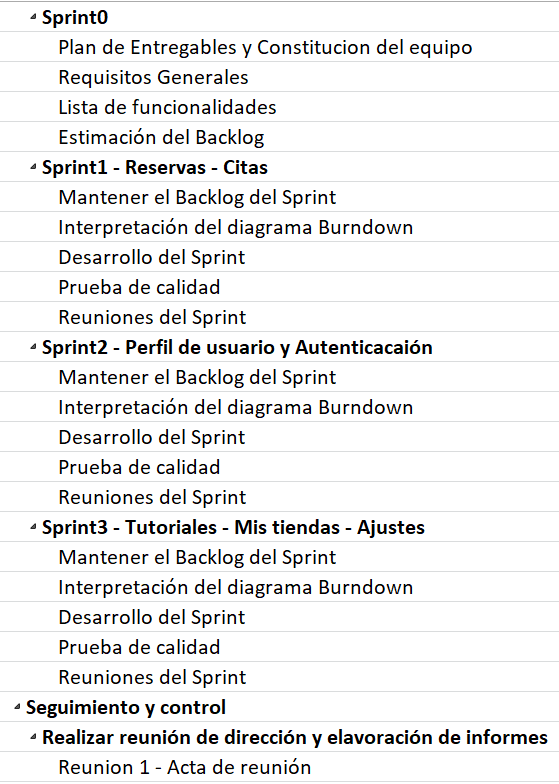
• Informe de la Línea Base del Proyecto Actividades de calidad a realizarse

Las tareas a ser llevadas a cabo deberán reflejar las evaluaciones a realizar, los estándares a seguir, los productos a revisar, los procedimientos a seguir en la elaboración de los distintos productos y los procedimientos para informar de los defectos detectados a sus responsables y realizar el seguimiento de los mismos hasta su corrección.

Las actividades que se realizarán son:

* Revisar cada producto
* Revisar el ajuste al proceso
* Realizar Revisión Técnica Formal (RTF)
* Asegurar que las desviaciones son documentadas.
  + 1. Relaciones entre las actividades de SQA y la planificación

Las pruebas se realizaran en cada Sprint y durante el seguimiento y control del desarrollo del software.



1. Documentación
   1. Documentación mínima requerida

La documentación mínima es la requerida para asegurar que la implementación logrará satisfacer los requerimientos.

* + 1. Plan del Proyecto

Es una de las primeras responsabilidades a las que se enfrenta un Project Manager. De esa planificación se derivará la organización de los recursos, la distribución de responsabilidades y la asignación de presupuesto.

* + 1. Documento de Negocio

Su función es la de demostrar la relación establecida entre las partes que intervienen, las fases de ese negocio y las obligaciones que surgen.

* + 1. Especificación de requerimientos del software

Deberá describir, de forma clara y precisa, cada uno de los requerimientos esenciales del software además de las interfaces externas.

El cliente deberá obtener como resultado del proyecto una especificación adecuada a sus necesidades en el área de alcance del proyecto, de acuerdo al compromiso inicial del trabajo y a los cambios que este haya sufrido a lo largo del proyecto, que cubra aquellos aspectos que se haya acordado detallar con el cliente.

Los requerimientos de calidad del producto a construir son considerados dentro de atributos específicos del software que tienen incidencia sobre la calidad en el uso’ y se detallan a continuación:

**Funcionalidad**

a. Adecuación a las necesidades

b. Precisión de los resultados

c. Interoperabilidad

d. Seguridad de los datos

**Confiabilidad**

a. Madurez

b. Tolerancia a faltas

**Usabilidad**

a. Comprensible

b. Aprendible

c. Operable

d. Atractivo

**Eficiencia**

a. Utilización de recursos

**Mantenibilidad**

a. Analizable

b. Modificable

c. Estable, no se producen efectos inesperados luego de modificaciones

d. Verificable

**Portabilidad**

a. Adaptable

b. Instalable

c. Co-existencia

Cada uno de estos atributos debe cumplir con las normas y regulaciones aplicables a cada uno.

* + 1. Descripción del diseño del software

Especifica como el software será construido para satisfacer los requerimientos. Deberá describir los componentes y subcomponentes del diseño del software, incluyendo interfaces internas. Este documento deberá ser elaborado primero como Preliminar y luego será gradualmente extendido hasta llegar a obtener el Detallado.

El cliente deberá obtener como resultado del proyecto el diseño de una aplicación móvil que cubra aquellos aspectos que se haya acordado con el cliente incorporar al diseño, en función de la importancia que estos presenten y de sus conexiones lógicas.

El diseño debe:

• Corresponder a los requerimientos a incorporar:

a. Todo elemento del diseño debe contribuir a algún requerimiento

b. La implementación de todo requerimiento a incorporar debe estar contemplada en por lo menos un elemento del diseño.

• Ser consistente con la calidad del producto

* + 1. Plan de Verificación & Validación

El Plan de V & V deberá identificar y describir los métodos a ser utilizados en:

• La verificación de que:

a. Los requerimientos descritos en el documento de requerimientos han sido aprobados por una autoridad apropiada. En este caso sería que cumplan con el acuerdo logrado entre el cliente y el equipo.

b. Los requerimientos descritos en el documento de requerimientos son implementados en el diseño expresado en el documento de diseño.

c. El diseño expresado en el documento de diseño esta implementado en código.

• Validar que el código, cuando es ejecutado, se adecua a los requerimientos expresados en el documento de requerimientos.

* + 1. Reportes de Verificación & Validación

Estos documentos deben especificar los resultados de la ejecución de los procesos descritos en el Plan de V & V.

* + 1. Documentación de usuario

La documentación de usuario debe especificar y describir los datos y entradas de control requeridos, así como la secuencia de entradas, opciones, limitaciones de programa y otros elementos necesarios para la ejecución exitosa del software.

Todos los errores deben ser identificados y las acciones correctivas descritas. Como resultado del proyecto el cliente obtendrá una documentación para el usuario de acuerdo a los requerimientos específicos del proyecto.

* + 1. Plan de Gestión de configuración

El Plan de gestión de configuración debe contener métodos para identificar componentes de software, control e implementación de cambios, y registro y reporte del estado de los cambios implementados.

Productos entregables:

|  |  |
| --- | --- |
| **Especificación de requerimientos de Software** | **Describe los requisitos del software tanto funcionales como no funcionales** |
| **Plan de Aseguramiento de calidad** | Describe los planes y roles que adoptara cada uno de los interesados en el desarrollo del software |
| **Plan de pruebas** | Describe los módulos a ser probados, así como las pruebas que se utilizaran , entradas y salidas esperadas para cada prueba |
| **Administración de la configuración** | Describe la nomenclatura utilizada en el proyecto así como la forma en que se determina la línea base |
| **Plan de desarrollo de SW** | Describe lo que se va a implementar, los calendarios, actividades y responsabilidades de los miembros del equipo de desarrollo |

1. Estándares de Documentación

* **Programación**

Buenas prácticas de programación en android

* **Interface grafica**

Patrón de diseño, Mobile UI Design Patterns.

1. Estándares, prácticas, convenciones y métricas

**5.1 ESTÁNDAR DE CALIDAD DEL PRODUCTO**

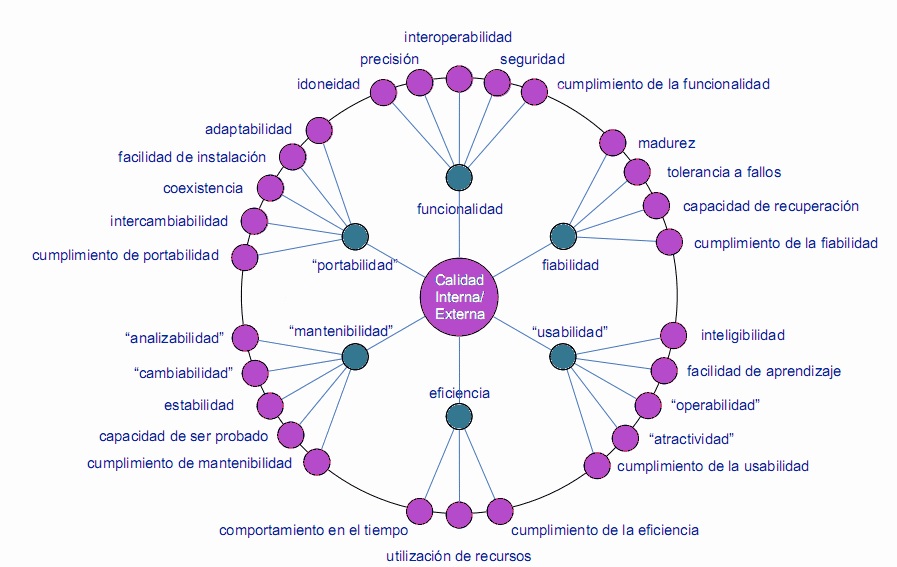
**5.1 NTP ISO/IEC 9126 MODELO DE CALIDAD DE PRODUCTO**

Es un estándar internacional para la evaluación del Software, fue originalmente desarrollado en 1991 para proporcionar un esquema para la evaluación de calidad del software.

La normativa define seis características de la aplicación, estas seis características son dividas en un número de sub- características, las cuales representan un modelo detallado para la evaluación de cualquier sistema informático.

**Características norma ISO 9126**

* Calidad Interna: Medible a partir de las características intrínsecas, como el código fuente.
* Calidad Externa: Medible en el comportamiento del producto, como en una prueba.
* Calidad en Uso: Medible durante la utilización efectiva por parte del usuario.



A continuación se describen las características y subcaracterísticas propias de este estándar que se encuentran dentro de las vistas interna y externa, las cuales usaremos para evaluar el software de CMI:

1. Funcionalidad: capacidad del software de proveer los servicios necesarios para cumplir con los requisitos funcionales.

**Subcaracterísticas:**

* 1. Idoneidad.- Hace referencia a que si el software desempeña las tareas para las cuales fue desarrollado.
  2. Exactitud.- Evalúa el resultado final que obtiene el software y si tiene consistencia a lo que se espera de él.
  3. Interoperabilidad.- Consiste en revisar si el sistema puede interactuar con otro sistema independiente.
  4. Seguridad.- Verifica si el sistema puede impedir el acceso a personal no autorizado.

1. Fiabilidad: capacidad del software de mantener las prestaciones requeridas del sistema, durante un tiempo establecido y bajo un conjunto de condiciones definidas.

**Subcaracterísticas:**

* 1. Madurez.- Se debe verificar las fallas del sistema y si muchas de estas han sido eliminadas durante el tiempo de pruebas o uso del sistema.
  2. Recuperabilidad.- Verificar si el software puede reasumir el funcionamiento y restaurar datos perdidos después de un fallo ocasional.
  3. Tolerancia a fallos.- Evalua si la aplicación desarrollada es capaz de manejar errores.

1. Usabilidad: esfuerzo requerido por el usuario para utilizar el producto satisfactoriamente.

**Subcaracterísticas:**

* 1. Aprendizaje.- Determina que tan fácil es para el usuario aprender a utilizar el sistema.
  2. Comprensión.- Evalúa que tan fácil es para el usuario comprender el funcionamiento del sistema
  3. Operatividad.- Determina si el usuario puede utilizar el sistema sin mucho esfuerzo.
  4. Atractividad.- Verifica que tan atractiva se ve la interfaz de la aplicación.

1. Eficiencia: relación entre las prestaciones del software y los requisitos necesarios para su utilización.

**Subcaracterísticas:**

* 1. Comportamiento en el tiempo.- Verifica la rapidez en que responde el sistema
  2. Comportamiento de recursos.- Determina si el sistema utiliza los recursos de manera eficiente

1. Mantenibilidad: esfuerzo necesario para adaptarse a las nuevas especificaciones y requisitos del software.

**Subcaracterísticas:**

* 1. Estabilidad.- Verifica si el sistema puede mantener su funcionamiento a pesar de realizar cambios.
  2. Facilidad de análisis.- Determina si la estructura de desarrollo es funcional con el objetivo de diagnosticar fácilmente las fallas.
  3. Facilidad de cambio.- Verifica si el sistema puede ser fácilmente modificado
  4. Facilidad de pruebas.- .- Evalúa si el sistema puede ser probado fácilmente

1. Portabilidad: capacidad del software ser transferido de un entorno a otro.

**Subcaracterísticas:**

* 1. Capacidad de instalación.- Verifica si el software se puede instalar fácilmente
  2. Capacidad de reemplazamiento.- Determina la facilidad con la que el software puede remplazar otro software similar.
  3. Adaptabilidad.- El software se puede trasladar a otros ambientes
  4. Co-Existencia.- El software puede funcionar con otros sistemas

Cada una de las características debe ser evaluada dentro del software basándonos en pruebas de funcionamiento, medición de rendimiento y pruebas con usuarios que harán uso del sistema.

**5.2 ESTÁNDAR DE CALIDAD DEL PROCESO**

**NTP ISO/IEC 12207**

Information Technology / Software Life Cycle Processes, es el estándar para los procesos de ciclo de vida del software de la organización ISO.

Los procesos se clasifican en tres tipos: procesos principales, procesos de soporte y procesos de la organización.

****

* **Procesos principales**

En este proceso se debe identificar la necesidad, preparar una solicitud y seleccionar un proveedor. Determinar los procedimientos y recursos para gestionar el proyecto. Contiene actividades de análisis y diseño para el producto software.

Asimismo, se debe modificar el producto software preservando su integridad. Incluye la migración y retirada del producto.

**Actividades:**

1. Adquisición.
2. Suministro.
3. Desarrollo.
4. Explotación.
5. Mantenimiento

* **Procesos de apoyo**

El propósito de este proceso es proveer mecanismos para la creación de procesos capaces de resolver problemas y tomar acciones correctivas para remover nuevos problemas detectados. Asimismo se debe obtener y persistir información, así como también asegurar la completitud y correctitud de los elementos que pertenecen a la configuración.

Se debe de proveer de mecanismos para asegurar que los productos y/o servicios cumplan con los estándares y requerimientos establecidos, y que el desarrollo de otros procesos se apegue los más posibles a lo planificado originalmente.

**Actividades:**

1. Documentación
2. Gestión de la configuración.
3. Aseguramiento de calidad.
4. Verificación. Validación.
5. Revisión conjunta.
6. Auditoría.
7. Resolución de problemas.

* **Procesos de gestión**

El propósito de este proceso es proveer actividades y tareas genéricas que pueden emplearse y ajustarse para gestionar otros procesos.

**Actividades:**

* 1. Inicio y Definición de Alcance
  2. Planificación
  3. Ejecución y Control
  4. Revisión y Evaluación
  5. Terminación

**5.3 ESTÁNDAR PARA MÉTRICAS DE CALIDAD**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **FACTOR DE**  **CALIDAD**  **RELEVANTE** | **OBJETIVO DE**  **CALIDAD** | **MÉTRICA A**  **USAR** | **FRECUENCIA Y**  **MOMENTO DE**  **MEDICIÓN** | **FRECUENCIA Y**  **MOMENTO DE**  **REPORTE** |
| Software | Auditar al acceso del usuario al sistema y a los datos | Auditoría de acceso | • Frecuencia, semanal  • Medición, lunes en la mañana | • Frecuencia, semanal  • Medición, lunes en la mañana |
| Software | Frecuencia del producto de software causa caídas en el ambiente de producción | Prevención de caídas | • Frecuencia, semanal  • Medición, lunes en la mañana | • Frecuencia, semanal  • Medición, lunes en la mañana |
| Software | Disponibilidad del sistema para el uso durante un periodo de tiempo específico | Disponibilidad | • Frecuencia, semanal  • Medición, lunes en la mañana | • Frecuencia, semanal  • Medición, lunes en la mañana |
| Software | Medir la proporción de las funciones es entendida después de leer la descripción del producto | Claridad de la descripción | • Frecuencia, semanal  • Medición, lunes en la mañana | • Frecuencia, semanal  • Medición, lunes en la mañana |
| Software | Tiempo que toma el usuario aprender a usar una función | Facilidad de aprender la función | • Frecuencia, semanal  • Medición, lunes en la mañana | • Frecuencia, semanal  • Medición, lunes en la mañana |

**5.4 CASOS DE PRUEBAS**

**PRUEBAS PERIODICAS**

Enfocados hacia la funcionalidad, compatibilidad, interoperabilidad, localización y usabilidad.

**Pruebas de funcionalidad:**

Están relacionadas con lo que hace la aplicación y si cumple los requisitos funcionales específicos.

**Pruebas de compatibilidad:**

Comprobar que la aplicación es compatible con todos los sistemas operativos móviles que se quiera y que por supuesto, funcione correctamente en todos ellos.

**Interoperabilidad:**

La aplicación intercambia información con otras aplicaciones o con el dispositivo móvil. Comprobar que los datos que se mandan y se reciben sean correctos.

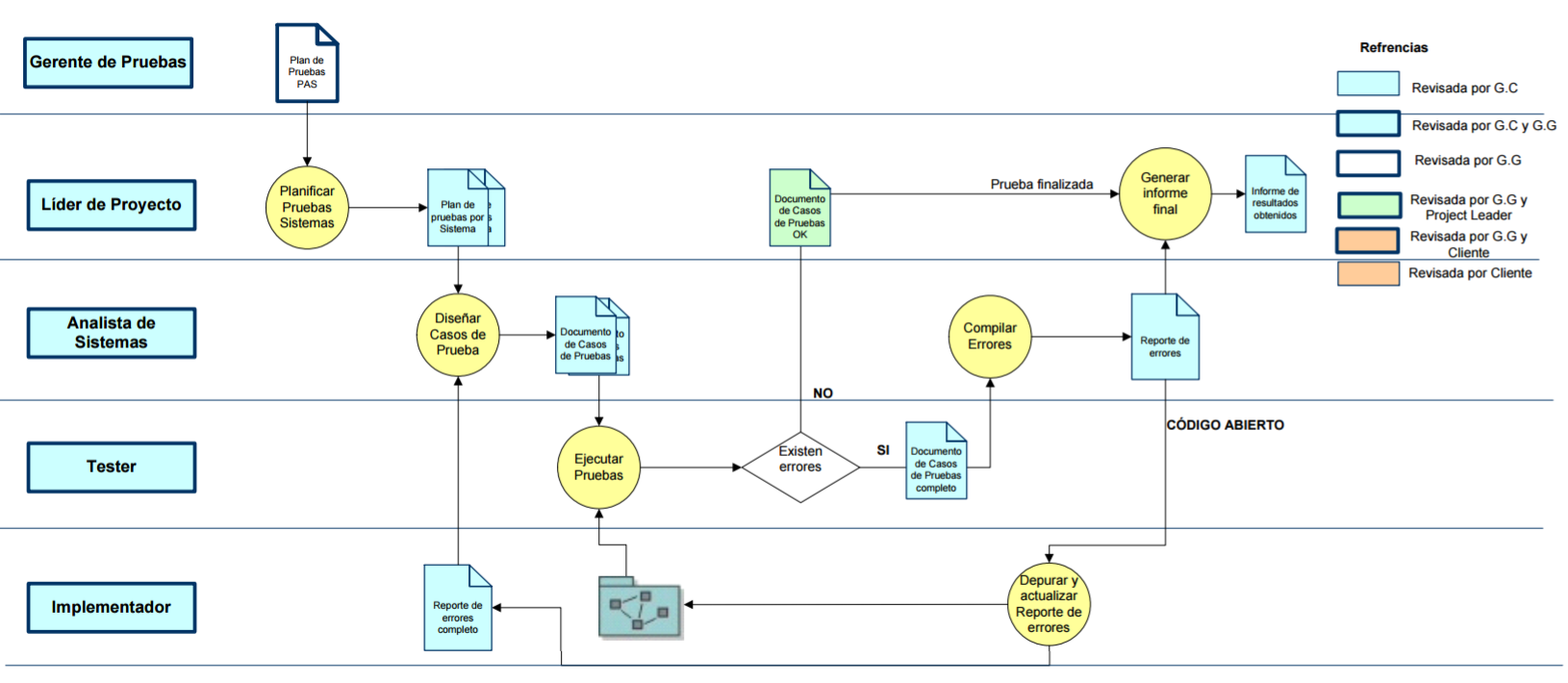
**Localización:**

Llevar control de etiquetado.

**Usabilidad:**

Comprobar como los usuarios finales pueden instalar, configurar y probar la aplicación en su dispositivo móvil.

**PROCESO DE PRUEBAS**



**5.5 MODELO DE CALIDAD APLICADO AL PROYECTO.**

**FUNCIONALIDAD:**

Se le asigna el 21 % a esta característica porque es importante que el producto de software cumpla con los requerimientos solicitados por los clientes, así mismo estas funciones debe ser adecuadas y muestre resultados exactos.

La información es sensible por lo que debe tener un grado de seguridad adecuada.

**FIABILIDAD:**

Se le asigna el 21 % a esta característica porque es necesario que la aplicación móvil cumpla con la prestación de servicios en situación cuando ocurra fallas del sistema y debe ser capaz de reestablecer el servicio y recuperar los datos.

**USABILIDAD:**

Se le asigna el 21% a esta característica porque es importante que el usuario sea capaz de entender, aprender y usar la aplicación móvil con facilidad.

**EFICIENCIA:**

Se le asigna el 14% a esta característica porque al ser una aplicación móvil no compleja con afectará la eficiencia del dispositivo en el cual será usado y no demandará muchos recursos.

**MANTENIBILIDAD:**

Se le asigna el 14% a esta característica debido a que los cambios en la aplicación móvil no son de alto impacto.

**PORTABILIDAD:**

Se le asigna el 7% a esta característica porque está diseñada solamente para ejecutarse en el sistema operativo Android y la instalación es sencilla.

1. Revisiones y auditorías
   1. Objetivo

Definición de las revisiones y auditorías técnicas y de gestión que se realizarán.

Especificación de cómo serán llevadas a cabo dichas revisiones y auditorías.

* 1. Requerimientos mínimos

Auditoria de Calidad:

* + Mejores prácticas.
  + Identificar anomalías.
  + Contribuciones de cada auditoria.

|  |  |  |
| --- | --- | --- |
| **CRITERIO DE EVALUACION** | **AUDITORIA A PROYECTOS EN PROCESO** | **AUDITORIA A PROYECTOS TERMINADOS** |
| Plazo | Corto Plazo | Largo Plazo |
| Orientación de auditoria | Progreso y desempeño del proyecto | Mejora del sistema de administración de proyectos |
| Ambiente de auditoria | Estresante , preocupante | Relajado , analítico |
| Enfoque de auditoria | Corregir desviaciones | Mejorar |
| Profundidad de auditoria | Superficial | Detallada e incluyente |
| Impacto de la Auditoria | Proyectos actuales | Proyectos futuros |

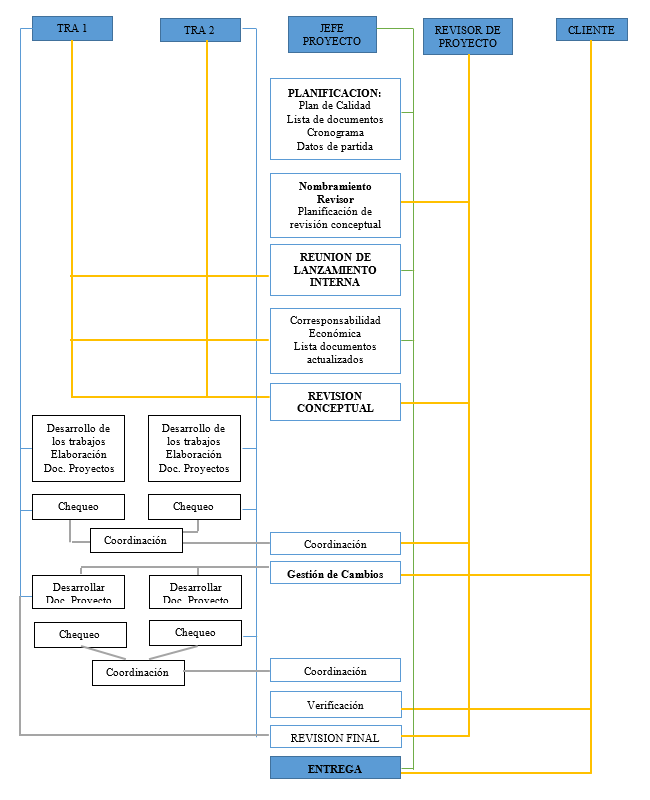
Tipos de hallazgos en auditoria:

* No conformidad mayor (NCM): Incumplimiento total o repetitivo de un requisito de calidad.
* No conformidad menor (NCm): Incumplimiento aislado o parcial de los requisitos del sistema de gestión de calidad que no pone en riesgo el cumplimiento de los requisitos del cliente o legales.
* Hallazgos que requieren corrección (RC): Omisiones o inconsistencias que requieren ser corregidas y que no ponen en riesgo el cumplimiento de los requisitos del cliente.
* Oportunidades de mejora (OM): Situaciones que se encuentran en cumplimiento. Recomendaciones.
* Hallazgos para su seguimiento (H): Situaciones que se encuentran en fase de implementación con un plan de trabajo.
  1. Revisión de requerimientos

Esta revisión se realiza para asegurar que se cumplió con los requerimientos especificados por el Cliente.

* 1. Revisión de diseño preliminar

Esta revisión se realiza para asegurar la consistencia y suficiencia técnica del diseño preliminar del software.

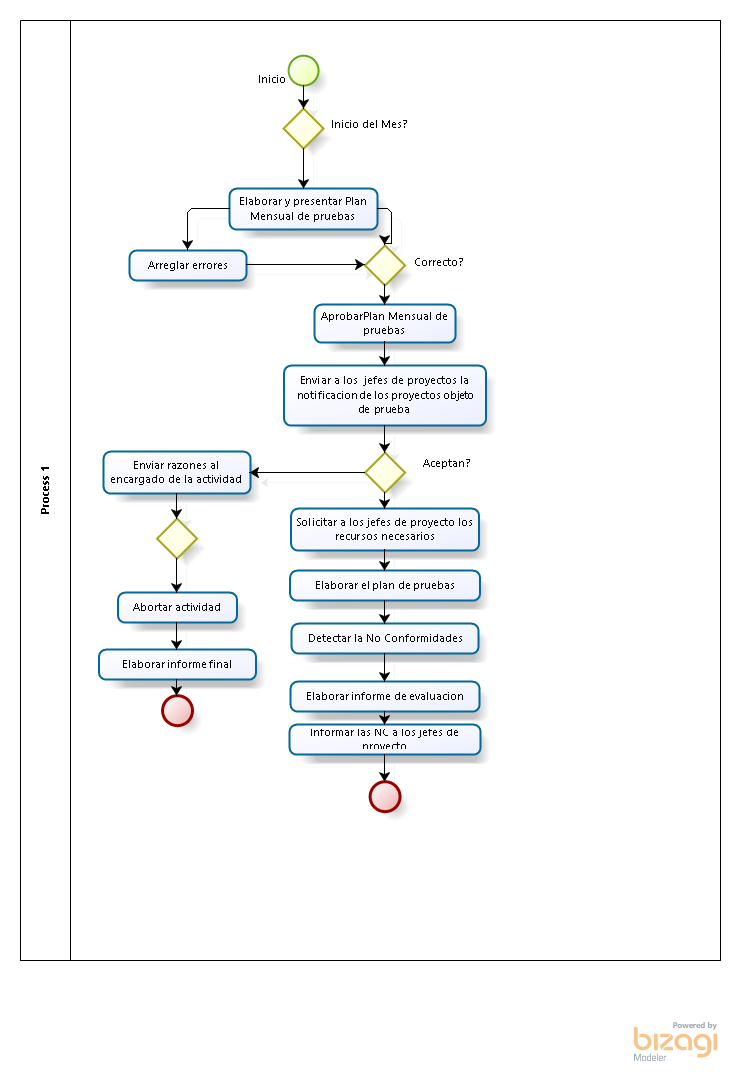


* 1. Revisión de diseño crítico

Análisis de procesos.

* + Actividades que no agregan valor.
  + Análisis causal
    1. Diagrama de Causa y efecto
    2. Diagrama de Pareto.
    3. Diagrama de Líneas.
    4. Listas de verificación.
  + Determinar mejoras necesarias.
  1. Revisión del Plan de Verificación & Validación

Esta revisión se realiza para asegurar la consistencia y completitud de los métodos especificados en el Plan de V & V.



* 1. Auditoría funcional

Esta auditoría se realiza previa a la liberación del software, para verificar que todos los requerimientos especificados en el documento de requerimientos fueron cumplidos.

* 1. Auditoría física

Esta revisión se realiza para verificar que el software y la documentación son consistentes y están aptos para la liberación.

* 1. Revisiones de gestión

Estas revisiones se realizan periódicamente para asegurar la ejecución de todas las actividades identificadas en este Plan. Deben realizarse por una persona ajena al grupo de trabajo (en caso de que sea posible).

* 1. Revisión del Plan de gestión de configuración

Esta revisión se realiza para asegurar la consistencia y completitud de los métodos especificados en el Plan de gestión de configuración.