Service Information:  
 a - Abstract  
 b - Table of Contents  
 c - Document Status Sheet  
 d - Document Change records made since last issue

1 Purpose  
2 Reference Documents  
3 Management  
4 Documentation  
5 Standards, practices, conventions and metrics  
5.1 Documentation standards  
5.2 Design standards  
5.3 Coding standards  
5.4 Commentary standards  
5.5 Testing standards and practices  
5.6 Selected software quality assurance metrics  
5.7 Statement of how compliance is to be monitored  
6 Review and audits2  
7 Test

8 Problem reporting and corrective action

9 Tools, techniques and methods

10 Code control  
11 Media control  
12 Supplier control  
13 Records collection, maintenance and retention  
14 Training  
15 Risk Management  
16 Outline of the rest of the project  
 Appendix A Glossary

[1. Propósito 2](#_Toc2101285175)

[2. Documentos referenciados 3](#_Toc1548522395)

[3. Gestión 3](#_Toc624739020)

[4. Documentación 6](#_Toc1210514297)

[5. Estándares, prácticas, convenciones y métricas 7](#_Toc1742191546)

[5.1 Estándares de documentación 8](#_Toc1449329821)

[5.2 Estándares de diseño 8](#_Toc1494739442)

[5.3 Estándares de código 8](#_Toc1528157744)

[5.4 Estándares de comentarios 8](#_Toc490096852)

[5.5 Estándares de testing y prácticas 8](#_Toc1735466928)

[5.6 Métricas de aseguramiento de calidad seleccionadas 8](#_Toc96992924)

[5.7 Declaración de cómo se debe monitorear el cumplimiento 8](#_Toc1639189216)

[6. Revisiones y auditorías 8](#_Toc599948012)

[7. Test 9](#_Toc547476594)

[8. Reportes de problemas y acción correctora 10](#_Toc1772401223)

[9. Herramientas, técnicas y métodos 11](#_Toc131761102)

[9.1 Servidor 12](#_Toc1565188716)

[9.2 Repositorio 12](#_Toc2051634671)

[9.3 Servidor web 12](#_Toc1654073757)

[9.4 Lenguaje 12](#_Toc1819645811)

[9.5 Testing 12](#_Toc915927226)

[10. Control de código 12](#_Toc634090993)

[10.1 Librería de desarrollo o dinámica 13](#_Toc1047097102)

[10.2 Librería de master o controlada 13](#_Toc1898323139)

[10.3 Librería de archivados o estática 13](#_Toc363399655)

[11. Control de multimedia 14](#_Toc651879827)

[12. Control de proveedores 16](#_Toc988666624)

[13. Colección de registros, mantenimiento and retención 18](#_Toc1699858338)

[14. Entrenamiento 20](#_Toc1297212317)

[15. Gestión de riesgos 22](#_Toc2019655772)

[16. Esquema del resto del proyecto 23](#_Toc2021461766)

# 1. Propósito

El SQAP se define como un conjunto de actividades planificadas y sistemáticas, cuyo primer objetivo es evaluar la calidad y la adherencia de los productos de software a los estándares, procesos y procedimientos para el proyecto Zeus.

# 2. Documentos referenciados

* Estándar de la Ingeniería de Software, ESA PSS-05-0 edición 2 febrero de 1991.
* Guía para la Gestión de Proyectos Software, ESA PSS-05-11 edición 1 revisión 1 marzo de 1995.

# 3. Gestión

Describir la organización del aseguramiento de la calidad y las responsabilidades asociadas a cada uno.

Debe definir los roles pero no las personas específicas.

Se recomienda la siguiente estructura:

## 3.1. Organización

Identificar los roles organizacionales que controlan y monitorean la calidad de software

Describir las relaciones entre los roles organizacionales

Escala de tiempo

Descripción generada automáticamente  
QUITAR NOMBRES Y FOTOS!!

Describir la interfaz con el usuario

CREO QUE SE PUEDE COPIAR DE ANTERIORES DOCUMENTOS

## 3.2. Tareas

Describir las tareas (seleccionadas del SQAP secciones 3.4 a 3.13)

Describir la secuencia de las tareas seleccionadas

3.2.1 Quality Assurance Coordination

This task deals with the constant coordination of all other SQA activities, reporting and review of project milestones and deliverables

3.2.2 Quality Assurance Process Definition and Monitoring

This task deals with the definition of a standards-compliant software engineering process and the continual activity of monitoring its correct application within the activities of the EMI project. Success criteria of this task are the availability of an agreed, documented and regularly updated process and the minimization of deviation in its application by the project members

3.2.3 Definition and collection of metrics and KPIs

This task deals with the definition and continual collection and reporting of software quality metrics according to the EMI Quality Model [R43]. It provides information to project bodies on status of software as an instrument for corrective actions. Success criteria are the regular production of reports and their use to identify improvement areas.

## 3.3. Responsabilidades

**JEFE DE PROYECTO**

* Se encargará de gestionar de manera global el proyecto, planificar y definir los recursos. Además, coordinará las labores de desarrollo e implementación de software.
* Responsable de que se cumplan los plazos de cada uno de los entregables y de que el proyecto se ajuste al presupuesto establecido. Además, se encargará del correcto cumplimiento de las metodologías de desarrollo utilizadas.
* Se responsabilizará de posibles ampliaciones del proyecto. Coordina y supervisa a todos los miembros del equipo y se encarga de gestionar relaciones con empresas externas y proveedores de servicios.
* Responsable de identificar y definir los posibles riesgos asociados al proyecto y definir quién es el responsable de mitigarlos.

**RESPONSABLE DE PRUEBAS**

* Define la estrategia de pruebas para validar el correcto funcionamiento del producto generado.
* Sera encargado de gestionar al equipo de pruebas
* Durante la toma de requisitos es el encargado de que se defina la funcionalidad correctamente desde un punto de vista de ingeniería de pruebas.
* Encargado de gestionar el mantenimiento de la aplicación.
* Supervisará que según avanza el desarrollo del producto, se realicen las pruebas correspondientes.

**RESPONSABLE DE CONFIGURACIÓN Y CALIDAD**

* Desarrollará el plan de gestión de configuración.
* Determinará la versión actual de cada uno de los componentes.
* Se encargará de la verificación y auditoría de la gestión de configuración.
* Realizará el seguimiento de los indicadores de calidad y corregir posibles desviaciones y se encargará de que se lleven a cabo todos los objetivos del plan de calidad.
* Realizar y revisar el DAFO, así como revisar el documento de matriz de grupos de interés.
* Será el encargado de recibir el feedback del cliente y generar partes de incidencias.

**ANALISTA**

* Será el encargado de definir y analizar los requerimientos de usuario.
* Comprobará que el producto final cumpla con todos los requisitos definidos en el documento de requisitos de usuario.
* Se encargará de redactar los manuales de usuario que serán entregados al cliente para facilitar la familiarización del usuario con el software.
* Redactará informes que evalúen el rendimiento del sistema.
* Creará planes de cambio.

**JEFE DE OBRA**

* Tomará las decisiones relacionadas con la adquisición de elementos hardware del sistema que se implementará.
* Mantendrá conversaciones con proveedores e inspeccionará el mercado para encontrar las mejores soluciones tecnológicas en relación calidad-precio.

# 4. Documentación

Los documentos que se adjuntarán en la presente entrega serán:

* SQAP. Documento en el que se definirán las actividades necesarias para el aseguramiento de la calidad en el proyecto Zeus.
* Informe de seguimiento. Documento en el que se detalla el estado actual del proyecto, los retrasos y los recursos utilizados.

# 5. Estándares, prácticas, convenciones y métricas

Identificar los estándares, prácticas, convenciones y métricas usados para especificar la calidad de software.

Explicar cómo SQA va a comprobar que se ha alcanzado la calidad requerida

VER SI HAY ALGO EN OTROS DOCUMENTOS

## 5.1 Estándares de documentación

Estándares para producir documentación

SE SUELE DEFINIR EN EL PUNTO 4 DE ESTE DOCUMENTO

## 5.2 Estándares de diseño

## 5.3 Estándares de código

## 5.4 Estándares de comentarios

## 5.5 Estándares de testing y prácticas

## 5.6 Métricas de aseguramiento de calidad seleccionadas

## 5.7 Declaración de cómo se debe monitorear el cumplimiento

# 6. Revisiones y auditorías

Describe las revisiones técnicas que se van a llevar a cabo y su proposito.

Describe cómo van a ser monitorizados los procedimientos puestos en el SVVP

# 7. Test

Describir como las actividades de testeo descritas en el SVVP van a ser monitorizadas

# 8. Reportes de problemas y acción correctora

REFERENCIAR / COGER DEL SCMP???????

Además, describir como va a ser monitorizada

Describir las métricas que se van a aplicar al proceso de reporte para estimar la calidad de software

# 9. Herramientas, técnicas y métodos

## 9.1 Servidor

Para el lanzamiento de la aplicación utilizaremos el servidor de Heroku (https://www.heroku.com/) donde subiremos la aplicación. Heroku es una plataforma como servicio (PaaS) de computación en la Nube que soporta distintos lenguajes de programación.

Heroku es propiedad de Salesforce.com. Heroku, es una de las primeras plataformas de computación en la nube, que fue desarrollada desde junio de 2007, con el objetivo de soportar solamente el lenguaje de programación Ruby, pero posteriormente se ha extendido el soporte a Java, Node.js, Scala, Python, PHP, Go y Clojure. La base del sistema operativo es Debian o, en la nueva plataforma, el sistema basado en Debian Ubuntu.

## 9.2 Repositorio

El repositiorio donde almacenaremos el código será GitHub (https://github.com/) donde llevaremos el control de versiones conveniente. GitHub es una forja (plataforma de desarrollo colaborativo) para alojar proyectos utilizando el sistema de control de versiones Git. Se utiliza principalmente para la creación de código fuente de programas de ordenador. El software que opera GitHub fue escrito en Ruby on Rails. Desde enero de 2010, GitHub opera bajo el nombre de GitHub, Inc. Anteriormente era conocida como Logical Awesome LLC. El código de los proyectos alojados en GitHub se almacena generalmente de forma pública.

## 9.3 Servidor web

Se utilizará Jetty (https://www.eclipse.org/jetty/). Jetty es un servidor HTTP 100% basado en Java y un contenedor de Servlets escrito en Java. Jetty se publica como un proyecto de software libre bajo la licencia Apache 2.0.

## 9.4 Lenguaje

El lenguaje que se utilizará será Java específicamente el framework Vaadin (https://vaadin.com/). Vaadin es una plataforma de desarrollo para Java. Vaadin incluye un conjunto de Componentes Web, un framework para aplicaciones web y un conjunto de herramientas que permite a los desarrolladores implementar interfaces gráficas de usuario usando exclusivamente el lenguaje de programación Java, TypeScript o una combinación de los dos.

## 9.5 Testing

JUnit será el sistema de testing que se utilizará (https://junit.org/junit5/). JUnit es un conjunto de clases (framework) que permite realizar la ejecución de clases Java de manera controlada, para poder evaluar si el funcionamiento de cada uno de los métodos de la clase se comporta como se espera. Es decir, en función de algún valor de entrada se evalúa el valor de retorno esperado; si la clase cumple con la especificación, entonces JUnit devolverá que el método de la clase pasó exitosamente la prueba; en caso de que el valor esperado sea diferente al que regresó el método durante la ejecución, JUnit devolverá un fallo en el método correspondiente.

# 10. Control de código

## 10.1 Librería de desarrollo o dinámica

Al tratarse de un equipo de desarrollo pequeño, todos los miembros del equipo tienen permitido cambiar cualquiera de los CIs en la librería de desarrollo. Esto significa que cualquier miembro puede crear, editar y eliminar ficheros de la librería de desarrollo.

Además, en caso de que cualquier miembro del equipo cometa algún error y, por ejemplo, elimine un fichero que no debía ser eliminado, se podrá recuperar el trabajo eliminado gracias al uso de Git. Así mismo, gracias a la gestión de conflictos de Git, varias personas pueden trabajar sobre el mismo fichero a la vez.

Los únicos que podrán modificar la estructura general de la librería de desarrollo serán el jefe de proyecto y el responsable de configuración.

## 10.2 Librería de master o controlada

Los ficheros que se encuentran en la librería master solo pueden ser modificados por el responsable de configuración o por el jefe de proyecto. Si cualquier otro miembro del equipo de desarrollo quiere modificar algún CI, deberá contactar con el responsable de configuración y este a su vez concertará una reunión de revisión para aprobar o rechazar los cambios propuestos.

En caso de que los cambios sean aprobados, cualquiera de los dos sujetos que pueden modificar esta librería pasarán la nueva versión del CI a la librería master y la última versión de esta misma será pasada a la librería de archivo.

## 10.3 Librería de archivados o estática

Los ficheros que se encuentran en la librería de archivado no pueden ser modificados, únicamente es podrán añadir ficheros nuevos. Las únicas personas que pueden realizar esta acción son el responsable de configuración y el jefe de proyecto.

# 11. Control de multimedia

Características del hardware sobre el que se va a construir y almacenar el proyecto:

* Windows 10 Pro
* Sistema de 64 bits
* RAM 8GB (7,88GB)
* ROM 16GB
* Procesador: Intel Core i7, 1.70GHz
* Disco duro: 1TB
* Entradas HDMI y USB3

# 12. Control de proveedores

Todas las herramientas descritas en el capítulo 6 han sido desarrolladas por entidades externas al proyecto.

|  |  |  |  |
| --- | --- | --- | --- |
| Herramienta | Descrito en la sección | Entidad desarrolladora | Enlace a su web |
| Heroku | 6.1 | Heroku | <https://www.heroku.com/> |
| GitHub | 6.2 | GitHub | <https://github.com/> |
| Jetty | 6.3 | Eclipse Foundation | <https://www.eclipse.org/jetty/> |
| Vaadin | 6.4 | Vaadin | [https://vaadin.com/](https://vaadin.com/%20)) |
| JUnit | 6.5 | JUnit | <https://junit.org/junit5/> |

Todas estas herramientas han sido verificadas previamente y cumplen con todas las exigencias del proyecto. El equipo de desarrollo confía en las entidades desarrolladores plenamente.

# 13. Colección de registros, mantenimiento and retención

Las reuniones, revisiones, tutoriales, auditorías y correspondencias se almacenarán en la librería de archivado. Estos documentos no pueden ser modificados, únicamente se podrán añadir ficheros nuevos. Las únicas personas que pueden realizar esta acción son el responsable de configuración y el jefe de proyecto.

# 14. Entrenamiento

N/A

# 15. Gestión de riesgos

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Riesgos posibles** | **Probabilidad** | **Impacto** |
| **1** | Subestimación tiempo de desarrollo | 1. Baja | Tolerable |
| **2** | Subestimación tasa de fallos | 1. Baja | Grave |
| **3** | Enfermedades e imprevistos relacionados. | 2. Media | Medio |
| **4** | Problemas de tiempo relacionados con actividades externas | 2. Media | Tolerable |
| **5** | Mala gestión de las actividades. | 2. Media | Tolerable |
| **6** | Problemas con el desarrollo de la solución. | 1. Baja | Catastrófico |
| **7** | Perdida de la información por problemas técnicos. | 1. Baja | Grave |
| **8** | Baja de integrantes del equipo | 2. Media | Grave |

**Plan de contigencia**

|  |  |  |
| --- | --- | --- |
| **ID** | **Riesgos posibles** | **Plan de contingencia** |
| **1** | Subestimación tiempo de desarrollo | Realizar una reunion en la que tratar estos fallos y reorganizer la planificación establecida en el Gantt |
| **2** | Subestimación tasa de fallos | Mantener una comunicación constante con reuniones presenciales para evitar los problemas de comunicación. |
| **3** | Enfermedades e imprevistos relacionados. | Organizar una reunión de emergencia para reasignar tareas. |
| **4** | Problemas de tiempo relacionados con actividades externas | Reorganizar las actividades para adaptarlos a la situación que acontezca. |
| **5** | Mala gestión de las actividades. | Reasignación de tareas. |
| **6** | Problemas con el desarrollo de la solución. | Invertir más recursos en Desarrollo y complementar con formación adicional |
| **7** | Perdida de la información por problemas técnicos. | Creación de copias de seguridad de los archivos, así como copias de estos para evitar su perdida. |
| **8** | Baja de integrantes del equipo | Establecer cláusulas en los contratos que obliguen a avisar con 1 mes de antelación cualquier abandon de la empresa. |

# 16. Esquema del resto del proyecto

N/A