
PLAN DE GARANTÍA DE CALIDAD



GESTIÓN DE PROYECTOS SOFTWARE Y METODOLOGÍAS DE DESARROLLO

Facultad de Informática

Universidad Complutense de Madrid

Jonathan Carrero Aranda
José Luis Díaz Minaya
José Francisco García Ruiz
Eloy González Acedo
Daniel Martín Barrios
Javier Mendoza Gómez
Pedro Sánchez Ramírez
David de los Santos Gil
Jesús Vélez Ayuso

Historial de versiones

Fecha	Versión
24/04/2016	LK-SQA-v1.0
10/05/2016	LK-SQA-v1.1
17/05/2016	LK-SQA-v1.2

Índice de Contenido

1. Propósito	2
2. Documentos de referencia.....	3
3. Gestión.....	4
3.1. Organización:	4
3.2. Actividades	4
3.2.1 Ciclo de vida del software cubierto por el Plan	4
3.2.2 Actividades de calidad a realizarse	5
3.2.3 Revisar cada producto	5
3.2.4 Revisar el ajuste al proceso	6
3.2.5 Realizar Revisión Técnica Formal (RTF)	6
3.2.6 Asegurar que las desviaciones son documentadas	6
3.2.7 Relaciones entre las actividades de SQA y la planificación	6
4. Documentación	8
4.1. Propósito.....	8
4.2. Requisitos mínimos de documentación	8
4.2.1 Descripción de requisitos software	8
4.2.2 Descripción de diseño software.....	8
4.2.3 Plan de verificación y validación.....	8
4.2.4 Informe de resultados de verificación y validación	8
4.2.5 Documentación de usuario	9
4.2.6 Plan de gestión de configuración software	9
4.3 Otra documentación.....	9
5. Estándares, prácticas, convenciones y métricas	10
5.1 Propósito	10
5.2 Contenido.....	11
6. Revisiones del software	12
6.1 Propósito:	12
6.2 Requisitos mínimos.....	12
6.3 Otras revisiones y auditorías	13
7. Pruebas	14
8. Informe de errores y acciones correctoras	15
8.1 Reporte del proceso de auditoría	15
8.1.1 Presentación del reporte del proceso de auditoría	15

9. Herramientas, técnicas y metodologías.....	16
10. Control de medios	17
10.1 Identificación de medios	17
10.2 Protección de medios.....	17
11. Control de proveedor	19
12. Colección de registros, mantenimiento y conservación	20
13. Formación.....	21
13.1 Funciones.....	21
14. Gestión del riesgo.....	22
15. Glosario.....	23
16. Procedimiento de cambio e historial del plan de SQA	24

Índice de Tablas

Tabla 1: Estructura de equipo4

Tabla 2: Planificación.....7

Tabla 3: Identificación de medios17

Índice de Ilustraciones

Ilustración 1: Actividades SQA	14
--------------------------------------	----

1. Propósito

El objetivo de este plan es establecer las pautas y actividades que deben desarrollarse para garantizar la calidad del producto a desarrollar. Para ello se indicará para cada actividad los atributos de calidad relevantes, los métodos de evaluación y los responsables. Garantizando el cumplimiento de los estándares y de las metodologías propuestas se obtendrá un producto de mayor calidad. Además, este plan brinda elementos de apoyo a la gestión del proyecto para realizar verificaciones sobre la adecuación al proceso y así detectar desvíos que puedan resultar en acciones correctivas en etapas tempranas.

Este plan abarca las partes del ciclo de vida relacionadas con: elaboración, construcción, evaluación y transición. No estando contempladas, por salirse del alcance del proyecto, las etapas relacionadas con el mantenimiento del producto, aunque se tomarán consideraciones acerca del futuro del producto.

2. Documentos de referencia

- Material didáctico de la asignatura Gestión de Proyectos Software.
- 730-2002-IEEE Standard for Software Quality Assurance Processes.

3. Gestión

La gestión de calidad es un trabajo que recae sobre los propios miembros del equipo que se centrará en asegurar la correcta implementación del proceso establecido y que los productos derivados del proceso se ajusten a los criterios de calidad. Se controlará que las actividades realizadas se adapten al plan de calidad.

3.1. 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:

- Requerimientos
- Análisis
- Diseño
- Implementación
- Verificación

El equipo de trabajo está estructurado de la siguiente forma:

Nombre	Roles
Jonathan Carrero	Analista, asistente de verificación
David de los Santos	Analista, asistente de verificación
Daniel Martín Barrios	Analista, asistente de verificación
José García	Analista, asistente de verificación
José Luis Díaz Minaya	Analista, asistente de verificación
Pedro Sánchez Ramírez	Analista, asistente de verificación
Eloy González Acedo	Analista, asistente de verificación
Jesús Vélez Ayuso	Analista, asistente de verificación
Javier Mendoza	Analista, asistente de verificación

Tabla 1: Estructura de equipo

3.2. Actividades

3.2.1 Ciclo de vida del software cubierto por el Plan

Como ya se mencionó el presente Plan comprende las etapas de: Requerimientos, Análisis, Diseño, Implementación y Verificación.

Dentro del área de Requerimientos y Análisis:

- Especificación de Requerimientos
- Modelos de Casos de uso
- Alcance del Sistema
- Pautas para la interfaz del usuario

Dentro del área de Diseño:

- Modelo de Dominio
- 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

3.2.2 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.

3.2.3 Revisar cada producto

En esta actividad se revisan los productos que se definieron como claves para verificar en el Plan de calidad. Se debe verificar que no queden correcciones sin resolver en los informes de revisión previos, si se encuentra alguna no resuelta, debe ser incluida en la siguiente revisión. Se revisan los productos contra los estándares, utilizando la checklist definida para el producto.

Se debe identificar, documentar y seguir la pista a las desviaciones encontradas y verificar que se hayan realizado las correcciones. Como salida se obtiene el Informe de revisión de SQA, este informe debe ser distribuido a los responsables del producto y se debe asegurar de que son conscientes de desviaciones o discrepancias encontradas.

3.2.4 Revisar el ajuste al proceso

En esta actividad se revisan los productos que se definieron como claves para verificar el cumplimiento de las actividades definidas en el proceso. Con el fin de asegurar la calidad en el producto final del desarrollo, se deben llevar a cabo revisiones sobre los productos durante todo el ciclo de vida del software.

Se debe recoger la información necesaria de cada producto, buscando hacia atrás los productos previos que deberían haberse generado, para poder establecer los criterios de revisión y evaluar si el producto cumple con las especificaciones.

Esta información se obtiene de los siguientes documentos: Plan del Proyecto, Plan de planificación, Plan de gestión de la configuración, Plan de estimación.

Antes de comenzar, se debe verificar en los informes de revisión previos que todas las desviaciones fueron corregidas, si no es así, las faltantes se incluyen para ser evaluadas. Como salida se obtiene el Informe de revisión de SQA correspondiente a la evaluación de ajuste al Proceso, este informe debe ser distribuido a los responsables de las actividades y se debe asegurar de que son conscientes de desviaciones o discrepancias encontradas.

3.2.5 Realizar Revisión Técnica Formal (RTF)

El objetivo de la RTF es descubrir errores en la función, la lógica o la implementación de cualquier producto del software, verificar que satisface sus especificaciones, que se ajusta a los estándares establecidos, señalando las posibles desviaciones detectadas. Es un proceso de revisión riguroso, su objetivo es llegar a detectar lo antes posible, los posibles defectos o desviaciones en los productos que se van generando a lo largo del desarrollo. Por esta característica se adopta esta práctica para productos que son de especial importancia.

En la reunión participan el responsable de SQA e integrantes del equipo de desarrollo.

Se debe convocar a la reunión formalmente a los involucrados, informar del material que ellos deben preparar por adelantado, llevar una lista de preguntas y dudas que surgen del estudio del producto a ser revisado.

La duración de la reunión no debe ser mayor a dos horas. Como salida se obtiene el Informe de RTF.

3.2.6 Asegurar que las desviaciones son documentadas

Las desviaciones encontradas en las actividades y en los productos deben ser documentadas y ser manejadas de acuerdo a un procedimiento establecido. Se debe chequear que los responsables de cada plan los modifiquen cada vez que sea necesario, basados en las desviaciones encontradas.

3.2.7 Relaciones entre las actividades de SQA y la planificación

En esta sección se incluye una lista con las actividades de calidad a realizarse durante el proyecto, especificando en qué semana del proyecto se realizan.

Actividad	Semana en la que se realiza
Estimación del plan de calidad	4, 5, y 6
Evaluar y ajustar el plan de SQA	7, 8 y 9
Evaluar la calidad de los entregables	Todas (1-12)
Revisar el ajuste del proceso	4, 5, 6, 7, 8, 9, 10 y 11
RTF	6, 7, 8, 9 y 10
Realizar el informe final de calidad	11 y 12

Tabla 2: Planificación

4. Documentación

4.1. Propósito

Identificación de la documentación relativa a desarrollo, Verificación & Validación, uso y mantenimiento del software. Establecer como los documentos van a ser revisados para chequear consistencia: se confirman criterio e identificación de las revisiones.

4.2. Requisitos mínimos de documentación

La documentación que asegura que la implementación del software satisface los requerimientos planteados y que aporta información para la gestión de calidad del proyecto es la siguiente:

- Especificación de Requisitos (SRS)
- Plan de Gestión de Configuración
- Plan del Proyecto
- Plan de Gestión de Riesgos
- Plan de Calidad (SQA)
- Revisiones Técnicas Formales (RTF'S)
- Documentación generada a raíz de las RTF'S (Actas de cambios realizados).

4.2.1 Descripción de requisitos software

Los requisitos de Linker se describen en el documento de Especificación de Requisitos que se ha realizado siguiendo el estándar IEEE 8301998.

4.2.2 Descripción de diseño software

No se ha realizado ningún documento específico del diseño de la aplicación. Esta información está recogida en el documento de Especificación de Requisitos. El estándar que utilizaremos en el diseño de la aplicación ha sido creado por el equipo Linker y se especifica más adelante en este mismo documento (apartado 5.2).

4.2.3 Plan de verificación y validación

Linker no dispone de un documento específico de verificación y validación. Debido a la dimensión académica de este proyecto no se llegará a desarrollar código por lo que no habrá pruebas de verificación y validación. Toda la información teórica asociada a esta temática se recoge en el documento de Gestión de Configuración.

4.2.4 Informe de resultados de verificación y validación

Como se indica en el apartado anterior, no se realizarán pruebas de verificación y validación porque no dispondremos de código real. Por este motivo no habrá tampoco informes.

4.2.5 Documentación de usuario

La documentación para el usuario constituye un elemento de consulta para toda aquella persona que va a usar el programa por primera vez o que trata de saber si el programa servirá a sus objetivos. Igualmente es útil para usuarios que ya realizan un manejo básico y quieren profundizar hacia un conocimiento avanzado.

En este proyecto no realizaremos dicha documentación porque la asignatura no lo requiere.

4.2.6 Plan de gestión de configuración software

Linker dispone de un documento Gestión de Configuración en el cual se puede consultar toda la información de este campo con detalle.

4.3 Otra documentación

La documentación adicional de nuestro proyecto incluye:

- Informe de personalidad MBTI y KIVIAT

5. Estándares, prácticas, convenciones y métricas

Se definirán estándares para la documentación, los cuales estarán contenidos en los siguientes documentos:

- Estándar de Documentación Técnica
- Estándar de Documentación de Usuario

Además, la documentación deberá cumplir principios básicos como ser:

- Ser accesible (entendible) para terceros que deseen realizar mantenimiento sobre el producto.
- Incluir fuentes, Modelos de Casos de Uso y Objetos.

Para la elaboración de los documentos se han elaborado plantillas, accesibles para todo el equipo, en el cual se da un marco de:

- Encabezados y Pies de página
- Tamaño y tipo de fuentes, ya sea para el texto, títulos o subtítulos.
- Datos mínimos que se deben incluir, como ser: fecha de elaboración, versión, responsable.

5.1 Propósito

En esta sección se identifican las normas, prácticas, convenciones, técnicas estadísticas, los requisitos de calidad y las métricas que se aplicarán. Las medidas de producto y de proceso se incluirán en las métricas utilizadas.

El proyecto Linker utiliza:

- Estándares: IEEE 10581998 (Plan de proyecto), IEEE 8301998 (Especificación de requisitos), IEEE 15402001 (Plan de gestión de riesgos), IEEE 7302002 (Plan de Garantía de Calidad), IEEE 8282012 (Plan de la Gestión de la Configuración).
- Convenciones: Plantilla oficial de la documentación del equipo Linker (Se explica con detalle en el punto 5.2 de este documento).
- Requisitos de calidad: No aplicados por indicación del profesor.
- Métricas:
 - Índice de productividad = tamaño / esfuerzo = líneas de código generado / horas trabajadas.
 - Tasa de defectos = defectos / tamaño = número de errores / líneas de código generadas.
 - Facilidad de mantenimiento: Es la facilidad con la que se puede corregir un programa si se encuentra un error, se puede adaptar si su entorno cambia, o mejorar si el cliente desea un cambio de requisitos. Esta actividad cuenta con más esfuerzo que cualquier otra actividad de ingeniería del software.
 - Facilidad de uso: La facilidad de uso es un intento de cuantificar lo amigable que puede ser el programa con el usuario. Se puede medir en función de cuatro características:

- Habilidad intelectual y/o física requerida para aprender el sistema.
- El tiempo requerido para llegar a ser moderadamente eficiente en el uso del sistema.
- Aumento neto en productividad, medida cuando alguien utiliza el sistema moderadamente y eficientemente.
- Valoración subjetiva de la disposición de primeros usuarios hacia el sistema, a veces obtenida mediante un cuestionario.
- Índice de fallo en documentación: número de errores detectados en un documento tras una revisión técnica formal por hoja.

5.2 Contenido

Esta sección incluye los aspectos técnicos y el diseño básico. Se cubren temas tales como documentación, variables y el módulo de nombres, la programación, inspección y pruebas.

- Estándares de documentación: En cuanto a los documentos del proyecto Linker, existe una plantilla de diseño oficial y de carácter obligatorio para todos los documentos. En dicha plantilla el formato establecido incluye:
 - La portada con el logo de la aplicación, el título, versión de documento y los nombres de los miembros del equipo.
 - La cabecera los datos del documento.
 - La fuente para cuerpo (Calibri 11) y para títulos de apartados (Calibri 14 en negrita).
 - Los índices, indexados para facilitar el acceso y con el número de página en el que está ese apartado.
- Estándares de diseño: El diseño de la web sigue un estándar creado por el equipo de Linker.
El diseño de la web se muestra en el documento de Especificación de Requisitos en el punto de interfaces externas.
- Estándares de codificación: Linker utilizará el estándar de codificación Java, en él se incluye que la documentación del código, los formatos de las cabeceras de dicho estándar, las sentencias del paquete, etc.

6. Revisiones del software

6.1 Propósito:

Definición de las revisiones y auditorías técnicas y de gestión que se realizarán.
Especificación de cómo serán llevadas a cabo dichas revisiones y auditorías.

6.2 Requisitos mínimos

Como mínimo deberán revisarse todas las entregas mensualmente, basado en los estándares definidos anteriormente. Estas revisiones serán realizadas por el/los responsables de SQA

- **Revisión de requisitos:**
Esta revisión se realiza para asegurar que se cumplió con los requerimientos especificados por el Cliente.
- **Revisión de diseño preliminar:**
Esta revisión se realiza para asegurar la consistencia y suficiencia técnica del diseño preliminar del software.
- **Revisión de diseño crítico:**
Esta revisión se realiza para asegurar la consistencia del diseño detallado con la especificación de requerimientos.
- **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.
- **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.
- **Auditoría física**
Esta revisión se realiza para verificar que el software y la documentación son consistentes y están aptos para la liberación.
- **Auditorías internas al proceso:**
Estas auditorías son para verificar la consistencia: del código versus el documento de diseño, especificaciones de interfaz, implementaciones de diseño versus requerimientos funcionales, requerimientos funcionales versus descripciones de testeo.
- **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 (un Profesor o una persona de un grupo distinto).

- **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.

- **Revisión Final:**

Esta revisión se realiza al concluir el proyecto para especificar las actividades de desarrollo implementadas durante el proyecto y para proveer recomendaciones.

6.3 Otras revisiones y auditorías

Se revisa la completitud, claridad, correctitud y aplicación de uso.

7. Pruebas

Las actividades de pruebas que se realizarán para el proyecto Linker son:

- Pruebas de integración
- Pruebas de aceptación
- Pruebas de Cajas Negras

El administrador del proyecto designará a una persona como el líder de pruebas, el cual realizará un plan de pruebas para el software Linker.

Todo personal que desarrolle algún CU del proyecto realizará las siguientes actividades de pruebas necesarias del software y estas son responsables de:

- Realizar las pruebas Unitarias
- Realizar las pruebas de acuerdo al plan de pruebas del software Linker
- Anotar los resultados de las pruebas
- Recomendar acciones correctivas si se encontraron defectos en los módulos probados.

Los módulos a probar deben de estar bajo la administración de la configuración

Actividades del SQA:

El personal de calidad se encargará de auditar las actividades descritas en esta sección, así como de verificar que el software y la documentación de pruebas sea el adecuado y estén completos, correctos y aprobados antes de utilizarlos.

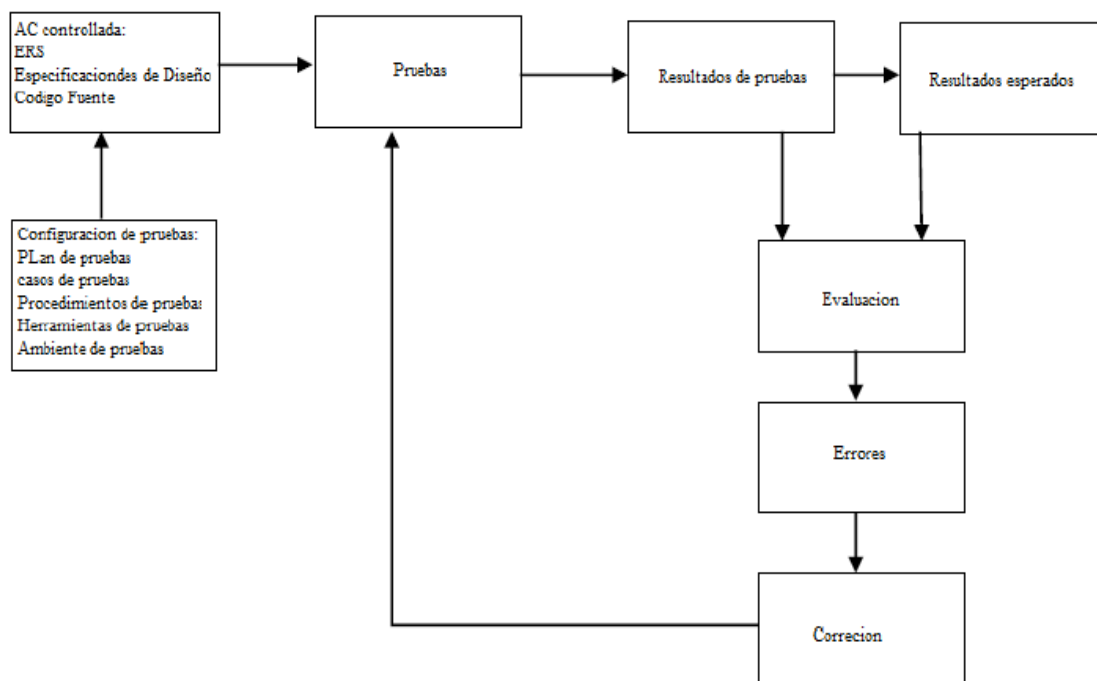


Ilustración 1: Actividades SQA

8. Informe de errores y acciones correctoras

En esta sección se describe el reporte y control del sistema utilizado por el personal de calidad para registrar y analizar las discrepancias encontradas, así como para monitorear la implementación de las acciones correctivas. Los formatos utilizados para realizar los reportes se describen más adelante en esta sección.

8.1 Reporte del proceso de auditoría

El personal de calidad reportará el resultado de las auditorías y las recomendaciones proporcionadas. Este reporte se usa para asegurarse que el proceso:

- Se está siguiendo de manera correcta y se está trabajando de forma efectiva
- Se está siguiendo, pero no se está trabajando de manera efectiva
- No se está siguiendo

8.1.1 Presentación del reporte del proceso de auditoría

El proceso de Reporte de Auditorías está dirigido hacia el Administrador del Proyecto el cual utilizará los reportes de las siguientes maneras:

- Para saber si los procesos de desarrollo son acatados y si son efectivos para el cumplimiento de las metas del proyecto. Cuando sea necesario el administrador del proyecto puede iniciar cambios a los procesos, mediante los procedimientos establecidos, para que los procesos queden estables.
- Para indicar el acuerdo, desacuerdo, o el aplazamiento de las recomendaciones hechas en el Proceso de Reporte de Auditoría. En caso de que el administrador del proyecto indica desacuerdo con las recomendaciones registradas en el proceso de reporte de auditoría, la disposición final de recomendaciones del informe se hace por el patrocinador del proyecto.

9. Herramientas, técnicas y metodologías

Herramientas:

Utilidades del sistema operativo, Debuggeadores, documentos de ayuda, checklist, , analizadores de estructuras, analizadores de código, auditorías de estándares, monitoreo de rendimiento, software de desarrollo, matrices de seguimiento de software, pruebas de generadores de casos.

- Como lenguajes de programación: PHP, HTML5, CSS3, J2EE, Struts, JavaScript, JQuery
- Herramientas de diagramas UML: IBM RSA.
- Herramientas de Casos de Uso: IBM RSA.
- Herramientas de bases de datos: MYSQL o phpmyAdmin.
- Herramienta de procesamiento de texto: Microsoft Word, Documentos Google.
- Herramientas de apoyo: Internet, Drive, Excel, Photoshop.
- Herramientas de desarrollo: Eclipse, Dreamweaver.
- Herramientas de planificación: Trello.
- Herramientas de comunicación: Whatapp.

Técnicas:

En las técnicas se incluirá la revisión de uso de estándares, inspecciones de software, rastreo de requerimientos, verificación y validación de diseño y requerimientos, mediciones y evaluaciones de fiabilidad, análisis de lógica de negocio.

- Estándares: Codificación de Lenguajes, UML, Diseño de BD Lógicas y Físicas
- Programación en Pares.
- Programación orientada a objetos y Programación orientada a eventos.

Metodologías:

Estas son un grupo de técnicas y herramientas. Estas metodologías se deben de documentar para completar la tarea o actividad y proporcionar una descripción del proceso que se va a usar.

- Paradigma de programación: Orientado a Objetos y Orientado a Eventos.

10. Control de medios

Esta sección determina los métodos para identificar el medio físico de cada producto software y para protegerlo de daños durante el desarrollo del proyecto.

10.1 Identificación de medios

Para cualquier documento o informe que se vaya generando a lo largo del desarrollo de Linker se utilizan los mismos medios, por ello no hablamos de cada documento sino de todos en general como “Documentos/Informes”.

Tipo de entregable	Medio	Documentación	Copia
Documentos/Informes	<ul style="list-style-type: none"> • Creado/editado mediante Google Docs • Almacenado y entregado mediante Google Drive. • Entregas en formato PDF. 	GDrive: http://www.google.com/policies/privacy/	Se generan copias dentro del GoogleDrive de las distintas versiones entregadas y revisadas
Código y ejecutables	<ul style="list-style-type: none"> • Desarrollado mediante eclipse, Dreamweaver y Mozilla Firefox. • Almacenamiento mediante GitHub 	Eclipse: http://www.eclipse.org/documentation/?sess=32267642b71e61a5209ee01bb85d8fdb GitHub: https://help.github.com/articles/github-privacy-policy/	El equipo de desarrollo e implementación tiene copias locales de código/ejecutables que va generando, todo se sube a la nube, utilizando branches y tags cuando es necesario.

Tabla 3: Identificación de medios

10.2 Protección de medios

En esta sección describiremos la política de seguridad/protección que el equipo de Linker a cabo para evitar el plagio o el sabotaje de nuestro proyecto durante su desarrollo e implementación posterior. En primer lugar, queremos destacar que, debido a que es un proyecto académico sin implementación real ni ánimo de lucro, la confidencialidad no es de los puntos más importantes en Linker. Asumimos que mediante el uso de Google Docs y Google Drive, tanto Google como la UCM tienen acceso a toda nuestra documentación e informes. En cualquier caso, tomamos algunas medidas para proteger nuestro proyecto:

- Solo hay un propietario de la carpeta compartida en Google Drive (Jonathan Carrero). Solo él puede añadir/eliminar miembros a la carpeta.
- Cualquier persona externa que tenga que tener acceso, por cualquier motivo a nuestro proyecto lo hará tan solo con permiso de lectura. (Tanto el profesor y los informes/entregas o nuestros compañeros de otros proyectos para las Revisiones Técnicas Formales).
- Por política de privacidad, restringimos el acceso a la información del nombre, apellidos o teléfono del usuario que está registrado en Linker, solo se publican los datos que el usuario desea mostrar.
- Uso de un github privado, para aseguramos que los únicos conocedores de nuestro código seamos nosotros.

No se utilizan medidas de cifrado para encriptar información.

11. Control de proveedor

Nuestra aplicación puede interactuar con software proveniente de otras redes sociales, potenciando la visibilidad del sitio y de su contenido.

Puesto que nuestro equipo se formó este año y ninguno de nuestros miembros ha trabajado o conoce proyectos relacionados con Linker, no existe la posibilidad de reutilizar código o funcionalidades ya desarrolladas que pudieran cumplir alguno de nuestros requisitos e incorporarse a nuestro trabajo.

12. Colección de registros, mantenimiento y conservación

La documentación que vamos a preservar será la listada a continuación:

- Especificación de requisitos: se conservará debidamente para facilitar la validación y verificación del producto.
- Plan de Gestión de Riesgos: se debe conservar para poder localizar con facilidad los riesgos contemplados y ser más efectivos a la hora de relacionar el riesgo con su prioridad y planes de gestión.
- Plan SQA: se debe conservar para poder verificar en todo momento que se están realizando las actividades de calidad reflejadas en el plan.
- Plan de Gestión de la Configuración: se debe conservar para evitar los máximos errores posibles.
- Plan de Proyecto: se conservarán registros del mismo y se tendrá que revisar que las referencias a otros documentos son válidas y correctas.
- Revisiones Técnicas Formales: se deben conservar para futuras revisiones y para el control de cambios realizados.
- Actas Semanales: se deben conservar para poder verificar en semanas posteriores acuerdos apalabrados en las reuniones realizadas.
- La documentación se guardará en formato digital. Se almacenarán copias tanto en la herramienta “Google drive” como en el ordenador del jefe de proyecto. También puede haber miembros del grupo que guarden información en sus ordenadores. Para mantener la documentación en un estado óptimo y actualizado cada cierto tiempo todo el grupo realizaremos revisiones de los documentos a punto de validar, además de los ya entregados si algún punto ha sido modificado. Las modificaciones de algún documento que todavía no se haya establecido como definitivo se debe comunicar al resto del grupo. El período de conservación de la documentación será fijado en 1 año desde la última versión realizada.

13. Formación

Creemos que con los conocimientos adquiridos en las asignaturas Ingeniería de Software, Gestión de Proyectos Software y Metodologías del Desarrollo, tenemos la capacidad suficiente para llevar a cabo un correcto plan SQA.

13.1 Funciones

Las funciones que deberá cumplir nuestro equipo de calidad, y que por tanto sus integrantes deberán recibir formación acorde a ello, serán las siguientes:

- Gerencial: Verifica la existencia de estándares y procedimientos claramente definidos, si no existen, debe involucrarse en su creación.
- Auditoría: Normaliza y audita un correcto proceso, cumpliendo los procedimientos establecidos. Verifica la entrega y mantenimiento de la documentación requerida.
- Control de calidad: Certifica el Software para que pueda ser implementado en un ambiente de producción mediante la ejecución de planes y casos de prueba.
- Mejora continua: Entrega reportes de métricas y recomendaciones para un mejor apego a los estándares y procedimientos.

14. Gestión del riesgo

Los riesgos identificados, la estrategia de mitigación, monitoreo y plan de contingencia a ser llevados a cabo, serán descritos en el Documento de Gestión de Riesgos, con lo cual se podrá hacer referencia a él.

15. Glosario

- Aseguramiento de la calidad del Software (SQA, Software Quality Assurance) Conjunto de actividades sistemáticas y planeadas para asegurar que los procesos y productos de software cumplen con los requerimientos, estándares y procedimientos
- Auditoría Evaluación independiente de los productos de trabajo y de un conjunto de procesos de software para asegurar la adherencia con las especificaciones, los estándares, procedimientos y otros acuerdos.
- Gestión de la configuración del software (SCM, Software Configuration Management) Conjunto de procesos destinados a asegurar la calidad de un producto software y mantener la integridad de los productos a través de todo el ciclo de vida del software, mediante un estricto control de los cambios realizados y de la disponibilidad constante de una versión estable de cada elemento de configuración para toda persona involucrada en el citado desarrollo.
- Revisión - Metodología definida, estructurada y disciplinada para la detección e identificación de defectos en los productos de trabajo durante el ciclo de vida del software.
- Prueba (Testing) Actividad que evalúa los atributos y la capacidad de un programa o sistema para determinar si se cumple con los resultados definidos.
- Métrica Medición del software en base a parámetros predeterminados, como puede ser el número de líneas de código de que consta o el volumen de documentación asociada.

16. Procedimiento de cambio e historial del plan de SQA

Los posibles cambios en el plan SQA, se realizarán las modificaciones pertinentes indicando en el historial de cambios el identificador de la variación, la causa del cambio, la persona o personas encargadas de realizar dicha modificación y la fecha en la que se realizó. Para garantizar el estado óptimo del plan SQA se realizará dos revisiones de los documentos, una cada mes y otra revisión final días antes de cada fecha de entrega prevista.

El historial de cambios deberá de seguir el siguiente formato:

Fecha de realización del cambio	Identificador del cambio	Causa del cambio	Personas encargadas
24/04/2016	LINKER-SQA-v1.0	Revisión interna	Eloy González
10/05/2016	LINKER-SQA-v1.1	Revisión del control de medios	Eloy González
17/05/2016	LINKER-SQA-v1.2	Revisión del puntos 5, 6, 12, 13	Pedro Sánchez