

## Identificación del problema y análisis de requerimientos

### Caso de Estudio : GreenSQA

<b>Cliente</b>	GreenSQA
<b>Usuario</b>	Empleados de GreenSQA
<b>Requerimientos funcionales</b>	<p>RF0: Creación de proyecto RF1: Creación de las etapas del proyecto. RF2: Culminación de una etapa del proyecto RF3: Registrar cápsulas de conocimiento RF4: Aprobación de las cápsulas RF5: Publicación de las cápsulas a la organización. RF6: Informar al usuario cuantas de las cápsulas registradas hay por cada tipo de cápsula. RF7: Informar al usuario un listado de lecciones aprendidas correspondientes a las cápsulas registradas en los proyectos para una etapa en particular RF8: Informar al usuario el nombre del proyecto con más cápsulas registradas RF9: Informar al usuario si un colaborador ha registrado cápsulas en algún proyecto. RF10: Informar al usuario las situaciones y lecciones aprendidas de las cápsulas aprobadas y publicadas, de acuerdo a una cadena de búsqueda.</p>
<b>Contexto del problema</b>	<p>GreenSQA desea retener el conocimiento de los empleados antes de que roten a otros empleadores. La fuga del conocimiento en empresas de TI es común y puede impactar negativamente la calidad del software. Para evitar la pérdida de conocimiento valioso, GreenSQA te pide implementar un proceso de captura de conocimiento mediante cápsulas generadas por los colaboradores de la empresa. Una cápsula de conocimiento es un texto que describe situaciones, elementos o datos importantes del proyecto. Al capturar el conocimiento de esta manera, GreenSQA espera asegurarse de que la información crítica del proyecto se mantenga disponible para futuras generaciones de empleados, lo que a su vez permitirá mejorar la calidad del software y la continuidad del negocio.</p>
<b>Requerimientos no funcionales</b>	<ul style="list-style-type: none"><li>● Usabilidad: el sistema debe ser fácil de usar y comprender para el personal de GreenSQA, con la capacidad de dictar con voz para facilitar el proceso de registro de cápsulas.</li><li>● Rendimiento: el sistema debe ser capaz de manejar grandes cantidades de cápsulas de conocimiento, búsquedas y consultas de manera eficiente<ul style="list-style-type: none"><li>○ La empresa espera que cada reporte sea generado en no más de 15 segundos..</li></ul></li></ul>

	<ul style="list-style-type: none"> <li>● Seguridad: el sistema debe asegurar la privacidad y seguridad de la información del personal y de la organización, como los datos de identificación y fechas de ingreso.</li> <li>● Disponibilidad: el sistema debe estar disponible para su uso en todo momento, con tiempos mínimos de inactividad.</li> <li>● Escalabilidad: el sistema debe ser escalable para manejar un mayor volumen de información y usuarios, en caso de que la empresa crezca.</li> <li>● Mantenibilidad: el sistema debe ser fácil de mantener, actualizar y extender en caso de cambios futuros en los requerimientos o tecnologías.</li> <li>● Fiabilidad: el sistema debe ser confiable y no perder información, por ejemplo, al agregar o cambiar etapas del proyecto o asociar personal a ellas.</li> <li>● Portabilidad: el sistema debe ser accesible desde diferentes dispositivos, navegadores y plataformas, como computadoras de escritorio, laptops, tabletas y teléfonos móviles. <ul style="list-style-type: none"> <li>○ Las operaciones que realiza el gerente de servicios requieren una aplicación web donde se pueda configurar el proyecto, el personal y gestionar la diferente información.</li> </ul> </li> </ul>
--	--

Identificador y nombre	RF0-Creación de proyecto		
Resumen	<p>Parte del requerimiento de gestión de proyectos consiste en la creación de proyectos. Esta actividad consiste en que, cada vez que Green SQA acepte un proyecto de un cliente, el software deberá de crear un apartado que permita al usuario registrar dicho proyecto. Para esto, se le pedirá al usuario la siguiente información: nombre del proyecto, nombre del cliente, fecha planeada para el inicio del proyecto y fecha planeada para la finalización del proyecto, el valor correspondiente al presupuesto del proyecto y los nombres y números celulares de los gerentes del proyecto tanto por parte de Green como por parte del cliente.</p>		
Entradas	Nombre entrada	Tipo de dato	Condición valores válidos
	projectName	String	-Mínimo 5 caracteres y Máximo 20 -no debe de contener símbolos -no debe contener números

	clientsName	String	-Mínimo 5 caracteres y Máximo 20 -no debe de contener símbolos -no debe contener números
	startDate	Calendar	dd-mm-yyyy
	endDate	Calendar	dd-mm-yyyy
	budget	Double	NA
	projectManagerName	String	-Mínimo 5 caracteres y Máximo 20 -no debe de contener símbolos -no debe contener números
	clientManagerName	String	-Mínimo 5 caracteres y Máximo 20 -no debe de contener símbolos -no debe contener números
	projectManagerCellPhone	String	NA
	clientManagerCellPhone	String	NA
Resultado o Postcondición	Si el número máximo de proyectos que se pueden guardar (10) es superado, entonces saldrá un mensaje que advierta que el número de proyectos límite se ha superado. Si esto no sucede, entonces el sistema debe registrar correctamente la información del proyecto.		
Salidas	<b>Nombre salida</b>	<b>Tipo de dato</b>	<b>Formato</b>
	Mensaje de confirmación	String	“La información del proyecto ha sido guardada exitosamente”
	Mensaje de advertencia	String	“El límite del número de proyectos ha sido alcanzado”

Identificador y nombre	<b>RF1-Creación de las etapas del proyecto</b>
Resumen	Parte del requerimiento de gestión de las etapas de proyecto consiste en la creación de estas. Esto consiste en que, automáticamente después de registrar un proyecto, el programa deberá de crear las 6 etapas: inicio, análisis, diseño, ejecución, cierre y seguimiento y control del proyecto. Se deben asignar fechas planeadas y reales a cada etapa, y se debe guardar la aprobación del cumplimiento de cada etapa.

	Es importante tener en cuenta que, para las fechas planeadas, se le pedirá al usuario la cantidad de meses que se lleva cada etapa. Mientras que para las fechas reales son guardadas por el sistema.		
Entradas	<b>Nombre entrada</b>	<b>Tipo de dato</b>	<b>Condición valores válidos</b>
	starPhaseDuration	int	Debe de ser un número entero positivo diferente a 0.
	analysisPhaseDuration	int	Debe de ser un número entero positivo diferente a 0.
	designPhaseDuration	int	Debe de ser un número entero positivo diferente a 0.
	executionPhaseDuration	int	Debe de ser un número entero positivo diferente a 0.
	closing_trackingPhaseDuration	int	Debe de ser un número entero positivo diferente a 0.
	projects_controlPhaseDuration	int	Debe de ser un número entero positivo diferente a 0.
Resultado o Postcondición	El sistema asigna a cada etapa las fechas registradas y guarda la cantidad de meses por etapa en un arreglo de cantidad de meses. Además, guarda el estado de la aprobación del cumplimiento de la etapa como un booleano y también, activa la etapa de inicio (asignándole un estado de activo) y deja las otras etapas como inactivas.		
Salidas	<b>Nombre salida</b>	<b>Tipo de dato</b>	<b>Formato</b>
	Mensaje de confirmación	String	“La información de las etapas ha sido guardada exitosamente”
	approvalOfPhase	boolean	false
	startPhaseStauts	boolean	true

Identificador y nombre	<b>RF2-Culminación de una etapa del proyecto</b>
Resumen	<p>Para cumplir con este requerimiento, si el usuario lo desea, seleccione desde el menú la opción “Culminar la etapa de un proyecto” para terminar con la etapa que se encuentre activa.</p> <p>Para esto, la aplicación deberá de preguntarle al usuario por el nombre del proyecto que busca.</p>

Entradas	Nombre entrada	Tipo de dato	Condición valores válidos
	projectsName	String	Deberá de coincidir con alguno de los nombres de los proyectos registrados
Resultado o Postcondición	Si el código no coincide con alguno de los nombres de los proyectos registrados, entonces el sistema lanzará un mensaje de error. De lo contrario, el sistema cambia el estado de la etapa, registra la aprobación de la misma y comienza con la siguiente. Además, guarda la fecha y ésta es asignada como la fecha real de finalización de la etapa que terminó y también como la fecha real de inicio de la siguiente etapa correspondiente. Para la última etapa, simplemente se asigna esa fecha como la de finalización.		
Salidas	Nombre salida	Tipo de dato	Formato
	Mensaje de Error	String	“No hay ningún proyecto con ese nombre-
	endDatePhase	Date	dd-mm-yyyy
	startDateOfNextPhase	Date	dd-mm-yyyy
	finishedPhaseStatus	boolean	false
	nextPhaseStatus	boolean	True
	approvalFinishedPhase	boolean	True

Identificador y nombre	RF3-Registrar cápsulas de conocimiento		
Resumen	<p>La aplicación debe permitir la creación de cápsulas de conocimiento en cada etapa del proyecto por parte de los colaboradores. Cada cápsula debe tener un identificador único, una descripción, un tipo de cápsula (técnico, gestión, dominio o experiencia), y debe estar relacionada con la persona que la registra, es decir que también debe de almacenar su nombre, el cargo y el aprendizaje o lección aprendida con dicha situación. Por otro lado, la aplicación debe permitir la inclusión de hashtags en el texto de la cápsula para facilitar su búsqueda y clasificación.</p> <p>La capsula será registrada en la fase del proyecto que se encuentre activa en ese momento.</p>		
Entradas	Nombre entrada	Tipo de dato	Condición valores válidos
	id	String	El id no se puede repetir.

	description	String	Debe contener al menos una palabra con dos “#” a su alrededor
	capsuleType	Int	opciones: 1. Tecnico 2. Gestion 3. Dominio 4. Experiencia
	collaboratorsName	String	NA
	collaboratorsPosition	String	NA
	learnedLesson	String	Debe contener al menos una palabra con dos “#” a su alrededor
Resultado o Postcondición	Si las entradas no corresponden con los valores válidos, el número máximo de capsulas que se pueden registrar ha llegado a su límite o no haya ninguna fase activa en el proyecto, entonces el programa deberá de enviar un mensaje de error avisando de esto. De lo contrario, el sistema debe de extraer las palabras claves señaladas por los “#” y deben ser almacenadas como una característica de la cápsula llamada “hashtag”. Por otro lado, las cápsulas registradas quedan en revisión y pendientes por ser aprobadas, esto se puede interpretar que el estado de la cápsula está inactivo.		
Salidas	<b>Nombre salida</b>	<b>Tipo de dato</b>	<b>Formato</b>
	Mensaje de error	String	NA
	approvalStatus	boolean	false
	hashtag	NA	NA

Identificador y nombre	<b>RF4-Aprobación de las cápsulas:</b>		
Resumen	La aplicación debe permitir al usuario elegir entre las cápsulas registradas, cuáles serán aprobadas para poder ser consultadas. Para esto, se le pedirá al usuario el nombre del proyecto donde está la capsula que desea aprobar y el identificado de esta		
Entradas	<b>Nombre entrada</b>	<b>Tipo de dato</b>	<b>Condición valores válidos</b>
	projectsName	String	Deberá de coincidir con alguno de los nombres de los proyectos registrados

	idCapsule	String	Deberá coincidir con alguno de las capsulas registradas
Resultado o Postcondición	Si el nombre del proyecto o el identificador de la capsula no son encontrados por el sistema, se mostrará un mensaje en pantalla avisando de esto. De lo contrario, se muestra un mensaje de confirmación, se sube al sistema la fecha de aprobación y se cambia el estado de la capsula.		
Salidas	<b>Nombre salida</b>	<b>Tipo de dato</b>	<b>Formato</b>
	Mensaje de Error	String	Ejemplo: "El identificador de la capsula no se ha encontrado"
	approvalDate	Date	dd-mm-yyyy
	approvalStatus	boolean	True
	Mensaje de confirmación	String	"Se ha registrado la cápsula con éxito"

Identificador y nombre	<b>RF5-Publicación de las cápsulas a la organización</b>		
Resumen	El sistema deberá permitirle al usuario publicar las capsulas aprobadas al intranet de la organización. Para esto se le pedirá al usuario el nombre del proyecto donde está la capsula que desea publicar y el identificador de esta, luego se generará un link en formato HTML.		
Entradas	<b>Nombre entrada</b>	<b>Tipo de dato</b>	<b>Condición valores válidos</b>
	projectsName	String	Deberá de coincidir con alguno de los nombres de los proyectos registrados
	idCapsule	String	Deberá coincidir con alguno de las capsulas aprobadas.
Resultado o Postcondición	Si el nombre del proyecto o el identificador de la capsula no son encontrados por el sistema, se mostrará un mensaje en pantalla avisando de esto. De lo contrario, se muestra un mensaje de confirmación con la url.		
Salidas	<b>Nombre salida</b>	<b>Tipo de dato</b>	<b>Formato</b>
	String	Ejemplo: "El identificador de la capsula no se ha encontrado"	Mensaje de Error

	Mensaje de confirmación	String	Ejemplo: "https://intranet.Green SQA.com/Tienda_de_Alimentos/Inicio/123/"
--	-------------------------	--------	--

Identificador y nombre	<b>RF6: Informar al usuario cuantas de las cápsulas registradas hay por cada tipo de cápsula.</b>		
Resumen	El usuario desde el menú de la aplicación podrá elegir la opción de ver el numero de capsulas registradas hasta el momento.		
Entradas	<b>Nombre entrada</b>	<b>Tipo de dato</b>	<b>Condición valores válidos</b>
	NA	NA	NA
Resultado o Postcondición	El sistema se encargará de contar el número de capsulas registradas que hay por tipo (hasta el momento solo están: técnico, gestión, dominio y experiencias) por las fases de todos los proyectos. Luego, imprimirá un mensaje con una lista de las capsulas registradas.		
Salidas	<b>Nombre salida</b>	<b>Tipo de dato</b>	<b>Formato</b>
	Mensaje	String	Ejemplo: "Numero de capsulas técnicas: 10 Numero de capsulas de gestión: 0 Numero de capsulas de dominio: 5 Numero de capsulas de experiencias: 7"

Identificador y nombre	<b>RF7: Informar al usuario un listado de lecciones aprendidas correspondientes a las cápsulas registradas en los proyectos para una etapa en particular</b>		
Resumen	El sistema deberá de mostrar un listado de las lecciones aprendidas de las capsulas registradas hasta el momento en una etapa en particular de los proyectos. Para esto, cuando el usuario escoja esta opción desde el menú, se le mostraran los tipos de fases que podrá seleccionar (hasta ahora solo están estas: inicio, análisis, diseño, ejecución, cierre y seguimiento y control del proyecto)		
Entradas	<b>Nombre entrada</b>	<b>Tipo de dato</b>	<b>Condición valores válidos</b>
	selectedOption	int	Debe de ser un numero entre 0 y 5



Resultado o Postcondición	<p>Si la opción seleccionada no corresponde con alguna de los valores válidos, entonces el sistema preguntara de nuevo la opción.</p> <p>Si aún no se ha registrado ningún proyecto, el sistema se encargará de enviar un mensaje avisando de esto. De lo contrario, el sistema recopilara las lecciones aprendidas de las capsulas registradas en la fase seleccionada para todos los proyectos, luego enviara un mensaje con las lecciones aprendidas y el nombre del proyecto donde están registradas.</p>		
Salidas	<b>Nombre salida</b>	<b>Tipo de dato</b>	<b>Formato</b>
	Mensaje de error	String	“ningún proyecto se ha registrado aun”
	Lista de lecciones aprendias	String	<p>Ejemplo:</p> <p>“Proyecto: Tienda de alimentos</p> <p>1. aprendí que las #papas# no son #frutas#</p> <p>2.las #peras# van en la sección de #frutas#”</p>

Identificador y nombre	<b>RF8: Informar al usuario el nombre del proyecto con más cápsulas registradas</b>		
Resumen	El sistema deberá de poder informar al usuario cual es el proyecto con más capsulas registradas, como una opción desde el menú.		
Entradas	<b>Nombre entrada</b>	<b>Tipo de dato</b>	<b>Condición valores válidos</b>
	NA	NA	NA
Resultado o Postcondición	Si aun no se ha registrado ni un proyecto, el sistema enviara un mensaje de error avisando de esto. De lo contrario, el sistema se encargara de contar cada capsula registrada en cada proyecto, después va a comparar estos resultados para determinar cuál es el proyecto y, por último, imprimirá un mensaje en pantalla con el nombre del proyecto y el numero de capsulas que tenga.		
Salidas	<b>Nombre salida</b>	<b>Tipo de dato</b>	<b>Formato</b>
	Mensaje de error	String	“ningún proyecto se ha registrado aun”
	Mensaje de confirmacion	String	<p>Ejemplo:</p> <p>“El proyecto con mas capsulas registradas es Tienda de alimentos con 27 capsulas registradas”</p>

Identificador y nombre	<b>RF9: Informar al usuario si un colaborador ha registrado cápsulas en algún proyecto.</b>		
Resumen	El sistema deberá de mostrar un listado de las lecciones aprendidas de las capsulas registradas hasta el momento en una etapa en particular de los proyectos. Para esto, cuando el usuario escoja esta opción desde el menú, se le pedirá ingresar el nombre del colaborador.		
Entradas	<b>Nombre entrada</b>	<b>Tipo de dato</b>	<b>Condición valores válidos</b>
	collaboratorsName	String	Debe de ser un nombre de un colaborador registrado
Resultado o Postcondición	El sistema se encargará de buscar el nombre del colaborador en las capsulas registradas de todos los proyectos. Si el nombre del colaborador no se encuentra, se asumirá que no ha escrito ninguna capsula. De lo contrario, imprimirá un mensaje con el nombre del colaborador y el nombre de alguna de las capsula que registro.		
Salidas	<b>Nombre salida</b>	<b>Tipo de dato</b>	<b>Formato</b>
	Mensaje de error	String	Ejemplo: "El colaborador Juan Camilo no ha escrito ni una capsula de conocimiento"
	Mensaje de confirmacion	String	Ejemplo: "El colaborador Juan Camilo ha registrado la capsula 123 en el proyecto Tienda de Alimentos"

Identificador y nombre	<b>RF10: Informar al usuario las situaciones y lecciones aprendidas de las cápsulas aprobadas y publicadas, de acuerdo con una cadena de búsqueda.</b>		
Resumen	El sistema deberá de mostrar un listado de las situaciones y lecciones aprendidas de las capsulas aprobadas y publicadas a partir de una cadena de búsqueda. Para esto, cuando el usuario escoja esta opción desde el menú, se le pedirá ingresar la cadena de búsqueda, que no es mas que una de las palabras claves de los hashtags.		
Entradas	<b>Nombre entrada</b>	<b>Tipo de dato</b>	<b>Condición valores válidos</b>
	searchString	String	Debe de corresponder con una de las palabras claves de los hashtags.
Resultado o Postcondición	Si aún no se ha publicado ninguna capsula, entonces se lanzará un mensaje de error avisando de esto. De lo contrario, el sistema buscara si alguno de los hashtags extraídos de las situaciones o lecciones		

	aprendidas de las capsulas aprobadas y publicadas, corresponden con la cadena de búsqueda. Si esto sucede, se imprimirá en pantalla un mensaje con el id de la capsula encontrada, junto a la descripción y lección aprendida. Si sucede lo contrario, que el sistema no encuentre la cadena de búsqueda, entonces imprimirá un mensaje avisando de esto		
Salidas	<b>Nombre salida</b>	<b>Tipo de dato</b>	<b>Formato</b>
	Mensaje de error	String	"Aun no se ha registrado ninguna capsula"
	Mensaje de error al no encontrar capsula	String	"No existe ninguna capsula con ese hashtag"
	Mensaje de confirmacion	String	<p>Ejemplo:</p> <p>"Id de la capsula: 123</p> <p>Descripción : El 12 de abril de 2004, inició el proyecto #Tienda de alimentos# y empezamos a registrar los #productos# disponibles de la tienda"</p> <p>Lección Aprendida: aprendí que las #papas# no son #frutas# "</p>