



Departamento de Informática
Universidad Técnica Federico Santa María



Entregable III

Proyecto: K-Search

Integrantes:

Nombres y Apellidos	Email	ROL USM
Alfredo Ignacio Silva Celpa	alfredo.silva.13@sansano.usm.cl	201373511-8
Juan Ignacio Escalona Padilla	juan.escalona.13@sansano.usm.cl	201373551-7
Francisco Ignacio Perez Castro	francisco.perezca.13@sansano.usm.cl	201373516-9

Índice:

Contexto del proyecto	03
<i>Objetivo</i>	03
<i>Contenido del Informe a Entregar</i>	03
Modelo de Calidad	04-05
Árbol de utilidad (actualización)	05-06
Pruebas de Software	06-07
Formulación de pruebas de software de requisitos no funcionales	07-08
Listado de Pruebas/Requisitos	08-11

Contexto del proyecto:

Objetivo del proyecto:

El objetivo principal del sistema es brindar apoyo en la búsqueda de contenidos de una asignatura mediante un buscador ad-hoc para posteriormente clasificarlos y presentarlos según los tipos de aprendizaje de cada estudiante.

Contenido del informe a entregar:

Modelo de Calidad: Se presenta un modelo de calidad para el software, en donde se detalla el porqué de ciertos atributos y sub-atributos que son importantes para los distintos stakeholders.

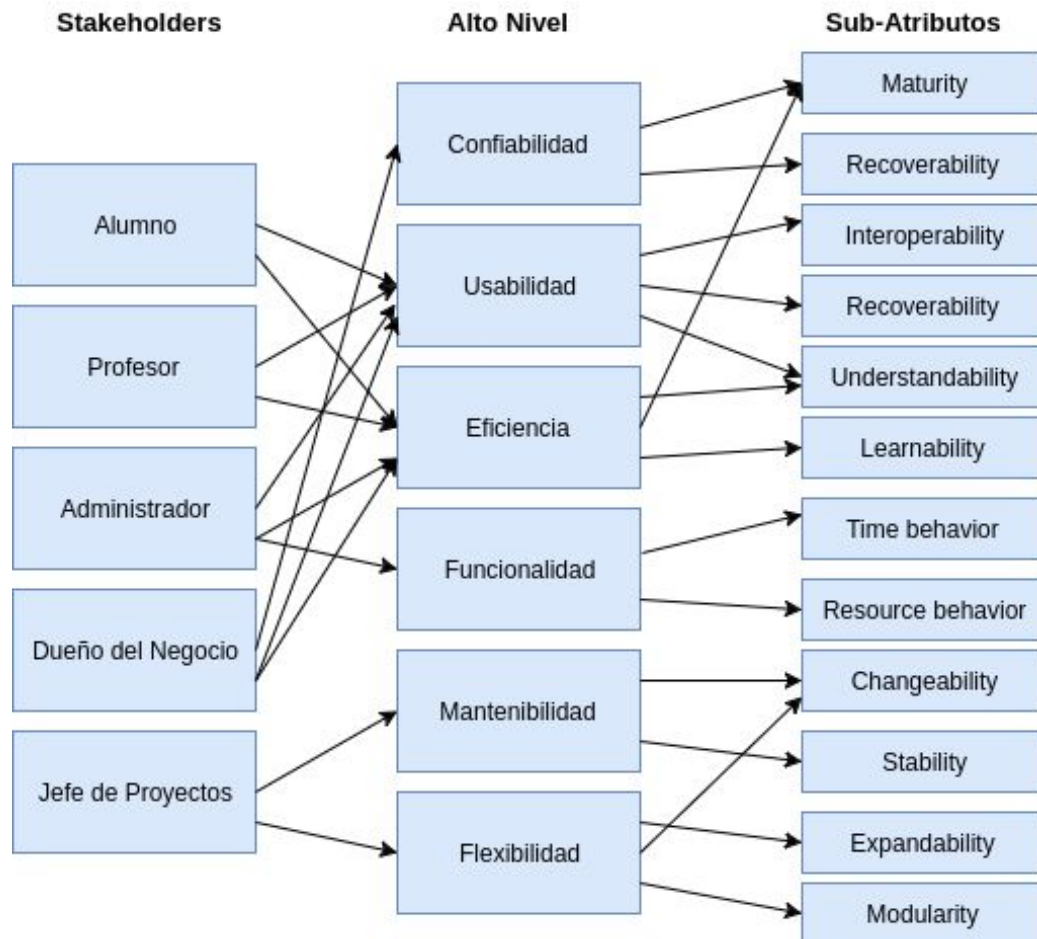
Árbol de utilidad (actualización): Actualización del árbol de utilidad, conforme a errores en el primero o a requerimientos nuevos descubiertos durante el transcurso del proyecto.

Pruebas de Software: Se presentan propuestas de pruebas funcionales y no funcionales de software, las cuales se formulan en base al testing realizado por el consultor QA.

Formulación de pruebas de software de requisitos no funcionales: No se generan formulaciones de requisitos no funcionales junto con el consultor QA, pero se proponen posibles pruebas de software válidas.

Listado de Pruebas/Requisitos: Se presenta un listado de pruebas y/o requisitos entre requisitos funcionales y extra funcionales, y los casos de prueba realizados por el consultor QA de forma de resumir lo realizado en el testing.

Modelo de Calidad



El modelo de Calidad respecto a nuestro proyecto identifica como stakeholders a los usuarios (Alumno y Profesor), Administrador, Dueño de negocio y Jefe de proyecto, donde cada uno presenta requerimientos específicos por al menos alguno de los atributos principales (columna central).

Por parte del usuario (Alumno y Profesor) a éste le interesa que en esencia el producto entregado tenga una buena usabilidad asociada a la facilidad de manejo y simplicidad de la interfaz gráfica de la plataforma y por otro lado le interesa que existan umbrales de eficiencia en cuanto al tiempo de búsqueda y de resolución de algoritmos utilizados para el funcionamiento de la página. Continuando, el Administrador demuestra énfasis en los dos atributos anteriormente presentados para los usuarios sumados al grado de funcionalidad que posee la plataforma en cuanto a administración se refiere.

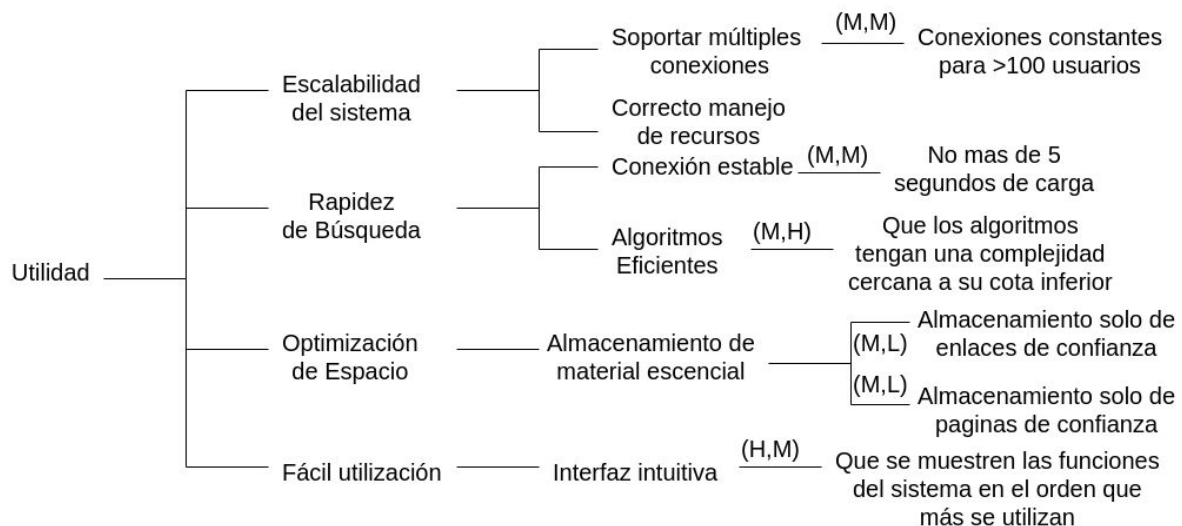
Con respecto al dueño del negocio se observa que su énfasis se encuentra en los mismos atributos que los usuarios además del atributo confiabilidad, ya que se

requiere contar con métodos de respaldo para subsanar algún imprevisto generalmente en hardware. Finalmente el Jefe de proyectos tiene mayor atención y fija sus requerimientos enfocados hacia los atributos de mantenibilidad y flexibilidad ya que son factores que afectan el desempeño del equipo de desarrollo y donde se debe colocar especial cuidado para la entrega de futuras funcionalidad y avances en la plataforma.

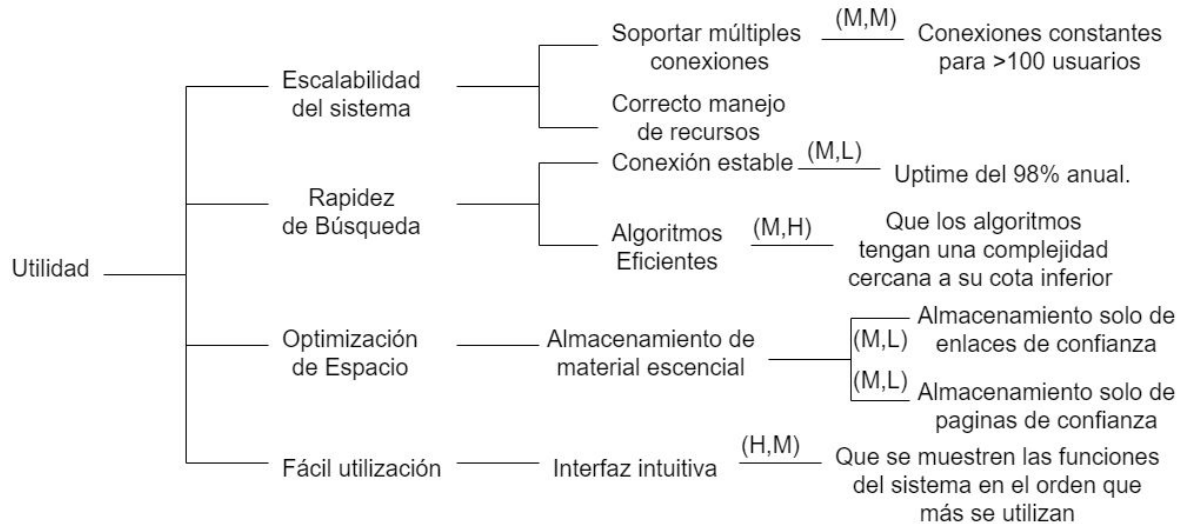
Árbol de utilidad (actualización)

Se realizó solo un cambio en el árbol de utilidad en la rama de conexión estable, la restricción pasó de ser “No más de 5 segundos de carga” a “Uptime del 98% anual”, de manera que fuera más pertinente a la estabilidad de la conexión en vez de a la rapidez/congestión de la conexión, además se bajo la dificultad de este M a L.

Antiguo árbol:



Nuevo árbol:



Pruebas de Software

El equipo de desarrollo se reunió con el consultor QA asignado para la revisión de diseño y pruebas generales del proyecto, por lo que se procederá con el análisis de los resultados obtenidos por el testing.

Los resultados obtenidos se ajustaban a las expectativas del equipo para las primeras dos pruebas, de votación y de registro de usuarios, no así con las pruebas de agregación de páginas de confianza, donde se podían añadir URL sin el formato especificado dado la falta de un filtro. De esta manera se desarrollará dicho filtro de páginas aceptadas como páginas de confianza.

En base al informe entregado por parte del QA y tal como se mencionó en la sección anterior se realizaron tanto pruebas con resultados satisfactorios como con resultados defectuosos. Las pruebas realizadas con resultados satisfactorios corresponden al sistema de votación por parte de los profesores y el registro de usuarios, ya que al ingresar una votación por parte del profesor ésta efectivamente se guarda y actualiza en caso de haber votado previamente; y para el caso de registro de usuarios se lleva a cabo sin problema al ingresar los campos requeridos por el formulario.

Por otro lado, las pruebas defectuosas o con fallas estaban ligadas al caso de uso “El profesor debe ser capaz de ingresar páginas de confianza” dado que, según el informe emitido por el QA asignado existe la posibilidad de agregar tanto una palabra vacía (“”) como un string que no corresponde a ninguna sitio web. Es por esto que se llegó a la conclusión que es necesario utilizar filtros para verificar la autenticidad de la página de confianza a ingresar (tanto en formato como su existencia).

Como equipo UNeed y dado los resultados expuestos anteriormente por parte del QA,

se propone desarrollar un filtro para las páginas de confianza para mitigar las fallas, el cual deberá permitir el ingreso sólo de sitios con un formato establecido de páginas. Cabe mencionar que actualmente existe un caso en que la página tengo un formato correcto pero que esta no exista, por lo cual se desarrolla un sistema de verificación de existencia de las páginas añadidas.

Luego de aplicar el filtro, junto con la verificación de existencia de los sitios web, se le entregó el proyecto nuevamente al consultor QA para revisión de la mitigación de errores desarrollada, pasando las pruebas sin problemas y arreglando los errores encontrados en el testing, lo que implica que se aprueba totalmente el proyecto por parte del QA para los casos de prueba realizados en esta iteración.

Como equipo, consideramos el trabajo del QA como primordial para la detección de fallas previas al entregable y con respecto a su desempeño, estamos bastante conformes dado el detalle y formalidad entregado para cada una de las pruebas realizadas, ya que se nos ha entregado de manera clara el feedback para tomar las medidas de mitigación necesarias. Con respecto a las pruebas realizadas, tal y como se mencionó anteriormente éstas han sido de valor ya que el proyecto al estar en fase de desarrollo no está exento de fallas y así se pudo corroborar, y a su vez permitió asegurar la consistencia del trabajo para las pruebas aprobadas satisfactoriamente.

Para finalizar la sección de Pruebas de Software, consideramos como equipo que en base al desempeño y tiempos propuestos por el QA éste se merece ser evaluado con nota 4, es decir, Buen trabajo con respecto a lo esperado por el equipo ya que realizó de buena forma las pruebas y fueron bien presentadas en el informe con sus respectivos pasos para replicar las fallas encontrados. No se le asigna puntaje máximo puesto que no se lograron formular pruebas satisfactorias para requisitos no funcionales.

Formulación de pruebas de software de requisitos no funcionales

Para esta sección, el grupo de desarrollo se reúne con el consultor QA acerca de las posibles formulaciones de los requisitos no funcionales y se llega a la conclusión de que el cliente no especifica ninguna necesidad con respecto a esto, por lo que no se generan formulaciones en conjunto.

Sin embargo, como equipo de desarrollo proponemos las siguientes formulaciones de requisitos no funcionales:

Pruebas de inyección de código a la base de datos: Aunque en el curso no se exigen requisitos con respecto a la seguridad, se piensa que construir software seguro contra un posible ataque a la base de datos puede ser un valor agregado deseable. Es por esto que se proponen hacer pruebas que se basen en intentar acceder a la base de datos directamente, que en un caso real puede provocar un uso malintencionado de la

información.

Seguridad de datos de usuario y formularios (encriptación de contraseñas y datos en general): Por los motivos mencionados anteriormente se propone realizar pruebas de seguridad con respecto a la posible interceptación de paquetes de información al ingresar datos importantes en los formularios, como las contraseñas, usuarios, etc.

Escalabilidad del sistema: Se proponen realizar pruebas de stress o de sobrecarga de usuarios al sistema para verificar si la capacidad máxima de usuarios es mayor a los esperados (más de 100 usuarios con conexión constante).

Rapidez de búsqueda: Se proponen realizar pruebas con respecto a la rapidez de búsqueda del buscador, ya que se requieren búsquedas eficientes para la extracción de contenidos. Esto se puede lograr sacando tiempos promedios para búsquedas ficticias.

Pruebas de capacidad de almacenamiento: Finalmente se proponen hacer pruebas para verificar si la capacidad máxima del sistema es la requerida, ya que se cree que la capacidad puede estar limitada por el tamaño del disco duro de la máquina que haga de host para el sistema.

Listado de Pruebas/Requisitos

ID	Nombre del Requerimiento	Roles asociados	Descripción
RF1	Votar Página	1) Profesor	<u>Profesor</u> : El Profesor puede asignar puntajes a las páginas de confianza para calificarlas.
RF2	Registrar Alumno	1) Alumno	<u>Alumno</u> : El Alumno puede registrar sus datos en el sistema.
RF3	Añadir Página	1) Profesor 2) Administrador	<u>Profesor</u> : El Profesor puede agregar páginas de confianza nuevas. <u>Administrador</u> : El Administrador puede agregar páginas de confianza nuevas.

Requerimientos técnicos (RT) que fueron utilizados en los casos de prueba:

1. RF1:

- a. RT1: Se debe estar ingresado como Usuario tipo Profesor.
- b. RT2: Debe existir al menos una página de confianza.

2. RF2:

- a. RT1: Correo del Alumno no debe estar en blanco.
- b. RT2: La contraseña no debe estar en blanco.
- c. RT3: Nombre de usuario no debe estar en blanco.
- d. RT4: La fecha nacimiento no debe estar en blanco.
- e. RT5: La institución no debe estar en blanco.
- f. RT6: Los datos anteriormente mencionados deben ser válidos.

3. RF3:

- a. RT1: Se debe estar ingresado como Usuario tipo Profesor o Administrador.
- b. RT2: La página debe tener un formato de URL (ej: www.***.***).

Caso de Prueba #	RF#	Caso de Prueba	Pasos del caso	Datos utilizados	Resultados obtenidos
CP1	RF1	Votar Página	1) Se ingresa a Opciones->Páginas de confianza. 2) Ingresar una cantidad de estrellas. 3) Se Ingresa a Opciones -> Páginas de confianza para verificar cambios.	Cantidad de estrellas=3, Página= "www.google.com"	La página " www.google.com " ahora tiene 3 estrellas.
CP2	RF2	Registrar Alumno	1) Se ingresa a la aplicación 2) Se ingresa el nombre de usuario 3) Se ingresa la contraseña 4) Se ingresa el mail 5) Se ingresa la fecha de nacimiento 6) Se comprueba en el inicio de sesión que el usuario puede ingresar con sus credenciales	Username= "erick.vale nzuela", Pass= "estanoesmi password", Mail= " erick.vale_nzuela.13@sansano.usm.cl ", Fecha= 07/09/1994	El Alumno queda registrado en el sistema

CP3	RF3	Añadir Páginas	1) Se ingresa a la Opciones->Páginas de confianza 2) Se hace click en la casilla "Añadir+" 3) Se escribe la URL. 4) Se esperan resultados.	ULR= ""	Es posible ingresar la página "", la cual no tiene formato de hipervínculo.
CP4	RF3	Añadir Paginas	1) Se ingresa a la Opciones->Páginas de confianza 2) Se hace click en la casilla "Añadir+" 3) Se escribe la URL. 4) Se esperan resultados.	URL= "jajaPágina falsa"	Es posible ingresar la página "jajaPágina falsa", la cual no tiene formato de hipervínculo.

Caso de Prueba #	RF#	TR#	Caso de Prueba	Pasos del caso	Datos utilizados	Resultados obtenidos
CP1	RF1	RT1	Votar Página	1) Se ingresa a Opciones->Páginas de confianza. 2) Ingresar una cantidad de estrellas. 3) Se Ingresa a Opciones -> Páginas de confianza para verificar cambios.	Cantidad de estrellas= 3, Página= "www.google.com"	La página " www.google.com " ahora tiene 3 estrellas.
CP2	RF2	RT1 RT2 RT3 RT4 RT5 RT6	Registrar Alumno	1) Se ingresa a la aplicación 2) Se ingresa el nombre de usuario 3) Se ingresa la contraseña 4) Se ingresa el mail 5) Se ingresa la fecha	Username = "erick.valenzuela", Pass= "estanoes mipassword", Mail= " erick.vale "	El Alumno queda registrado en el sistema

				de nacimiento 6) Se comprueba en el inicio de sesión que el usuario puede ingresar con sus credenciales	nzuela.13@sansano.usm.cl , Fecha= 07/09/1994	
CP3	RF3	RT2	Añadir Páginas	1) Se ingresa a la Opciones->Páginas de confianza 2) Se hace click en la casilla "Añadir+" 3) Se escribe la URL. 4) Se esperan resultados.	ULR= ""	Es posible ingresar la página "", la cual no tiene formato de hipervínculo.
CP4	RF3	RT2	Añadir Páginas	1) Se ingresa a la Opciones->Páginas de confianza 2) Se hace click en la casilla "Añadir+" 3) Se escribe la URL. 4) Se esperan resultados.	URL= "jajaPáginas afalsa"	Es posible ingresar la página "jajaPáginas afalsa", la cual no tiene formato de hipervínculo.

Debe tenerse en cuenta que los casos de prueba presentados en este entregable son todos los que el consultor QA le realizó al sistema, por lo que el equipo de desarrollo no se hace responsable si la cantidad de pruebas requerida no es alcanzada.