



**Departamento de Informática**  
Universidad Técnica Federico Santa María



## Entregable III

### Análisis y Diseño de Software / Fundamentos de Ingeniería de Software



Integrantes:

Nombre	Email	Teléfono
Sebastián Angla	<a href="mailto:sebastian.angla.13@sansano.usm.cl">sebastian.angla.13@sansano.usm.cl</a>	61189209
Tomás Gómez	<a href="mailto:tomas.gomez.13@sansano.usm.cl">tomas.gomez.13@sansano.usm.cl</a>	85009961
Gonzalo Pontigo	<a href="mailto:gonzalo.pontigo@alumnos.usm.cl">gonzalo.pontigo@alumnos.usm.cl</a>	85882040

## 1- Listado de requerimientos

### 1.1 Listado de requerimientos funcionales

Alumno		
ID	Requerimiento	Obligatoriedad
FR1	El sistema debe exigir loguearse al alumno	Obligatorio
FR2	El sistema debe requerir contestar la encuesta al alumno para poder ingresar al área de asignaturas	Obligatorio
FR3	El sistema debe proveer de una vista para seleccionar la asignatura que el alumno desea estudiar	Obligatorio
FR4	Una vez contestada la encuesta, el sistema debe categorizar al alumno según el perfil que obtuvo en la encuesta	Obligatorio
FR5	El sistema debe permitir al alumno ver el contenido de la asignatura que seleccionó previamente	Obligatorio
FR6	El sistema debe exigir al alumno evaluar la organización que tiene el perfil para poder continuar con el siguiente perfil	Obligatorio
FR7	Luego de que el alumno termine de estudiar por medio de un perfil, el sistema debe dirigir al alumno al siguiente perfil del contenido	Obligatorio
FR8	Después que el alumno evalúe la organización del perfil, debe poder proponer una nueva organización óptima para él	Deseable

Profesor		
ID	Requerimiento	Obligatoriedad
FR9	El sistema debe exigir loguearse al profesor	Obligatorio
FR10	El sistema debe proveer la opción de subir contenido al profesor	Obligatorio
FR11	El sistema debe proveer de una vista para seleccionar la asignatura que el profesor desea actualizar	Obligatorio
FR12	El sistema debe permitir que el profesor obtenga estadísticas de cada perfil de una asignatura	Obligatorio

### Entregable 3

Administrador		
ID	Requerimiento	Obligatoriedad
FR13	El sistema debe exigir loguearse al administrador	Obligatorio
FR14	El sistema debe proveer la opción de cambiar las contraseñas de los estudiantes y profesores	Obligatorio
FR15	El sistema debe proveer de una vista para ver las asignaturas y poder modificarlas, agregarlas o eliminarlas	Obligatorio

Técnico		
ID	Requerimiento	Obligatoriedad
FR16	El sistema debe contar con soporte técnico en caso de eventuales fallas	Obligatorio

### 1.2 Listado de requerimientos no funcionales

NonFunctionalRequirements		
ID	Requerimiento	Obligatoriedad
NFR1	El sistema debe ser desarrollado utilizando el modelo MVC	Opcional
NFR2	El coordinador del sitio no debe tener conocimientos computacionales	Opcional
NFR3	La conexión entre el cliente y el servidor debe utilizar SSL	Opcional
NFR4	El sistema de almacenamiento de archivos debe tener una capacidad de 10 GB	Opcional

## 2- Casos de uso

2.1) Se encuentra adjunta la descripción breve de los 9 casos de uso en el Diagrama Correspondiente. (Si no logra visualizarlos en el diagrama, igualmente están en la Carpeta Casos de uso).

2.2) A continuación se muestran 4 casos de uso extendidos como se indico:

<b>Caso de Uso:</b>	Registrar estudiante		
<b>Propósito:</b>	El estudiante puede registrarse en el sistema para poder utilizarlo.		
<b>Actores:</b>	Estudiante		
<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>El estudiante no debe estar registrado previamente</li> </ul>		
<b>Postcondiciones:</b>	<ul style="list-style-type: none"> <li>Registro agregado a la base de datos</li> </ul>		
<b>Escenario:</b>	<b>Pasos</b>	<b>Estudiante</b>	<b>Sistema</b>
<b>Principal:</b>	1	El estudiante presiona el botón registro en la página principal.	
	2		El Sistema lo redirige a un formulario de registro.
	3	El estudiante completa el formulario de registro con sus datos y presiona enviar registro.	
	4		El Sistema pregunta si confirma el registro.
	5	El estudiante confirma que desea registrarse.	
	6		El Sistema verifica los datos y reenvía al estudiante a la página del test.

### Entregable 3

<b>Alternativo:</b>	5.1	El estudiante decide no registrarse.	
	5.2		El Sistema reenvía al estudiante a la página principal.

<b>Caso de Uso:</b>	Ver Material		
Propósito:	El estudiante puede ver el material disponible.		
Actores:	Estudiante		
Precondiciones:	<ul style="list-style-type: none"> <li>El estudiante debe haber iniciado sesión.</li> </ul>		
Postcondiciones:			
Escenario:	Pasos	Estudiante	Sistema
<b>Principal:</b>	1		El sistema redirige al usuario a su página principal dependiendo de qué tipo de usuario es.
	2	El estudiante puede cerrar sesión.	
	3		El sistema lo dirige a la página de inicio.
<b>Alternativo:</b>	1.1.1	El estudiante selecciona que contenido desea ver.	

### Entregable 3

	1.1.2		El sistema le muestra el contenido que seleccionó.
	1.1.3	El estudiante puede cerrar sesión.	

<b>Caso de Uso:</b>	Realizar test		
Propósito:	El estudiante realiza test para ser registrado en el sistema		
Actores:	Estudiante		
Precondiciones:	<ul style="list-style-type: none"> <li>El estudiante debe haber completado formulario de registro</li> </ul>		
Postcondiciones:	<ul style="list-style-type: none"> <li>El sistema obtiene el tipo de estudiante y le asigna un perfil.</li> </ul>		
Escenario:	Pasos	Estudiante	Sistema
<b>Principal:</b>	1		El sistema indica instrucciones para realizar el test .
	2	El estudiante selecciona comenzar test.	
	3		El sistema muestra formulario para realizar test.
	4	El estudiante responde formulario de preguntas, respondiendo con valores de 1 a 4 cada respuesta de la pregunta sin repetir un número.	

### Entregable 3

	5	El estudiante presiona enviar test.	
	6		El Sistema envía y el test y calcula el tipo de perfil que corresponde con las respuestas del estudiante.

<b>Caso de Uso:</b>	Reparar errores		
Propósito:	El Técnico puede reparar errores de código y caída del sistema.		
Actores:	Técnico		
Precondiciones:	<ul style="list-style-type: none"> <li>El Técnico debe tener conocimiento de angular js, node js y sql.</li> <li>El Técnico debe haber leído o estar informado sobre la documentación del sistema.</li> </ul>		
Postcondiciones:	<ul style="list-style-type: none"> <li>El sistema queda sin errores y en actividad normal.</li> </ul>		
Escenario:	Pasos	Técnico	Sistema
<b>Principal:</b>	1	Ingresa a código del sistema y repara errores.	
	2		El sistema responde con aciertos o fallos en las pruebas.
	3	El Técnico encuentra y repara la falla.	
	4		El sistema se encuentra activo y funcionando a la normalidad.

### Entregable 3

	5	El Técnico guarda los cambios en el código.	
--	---	---	--

## 3- Diagramas

Los Diagramas respectivos a los ítems 2,3,4,5,6 y 7, se encuentran adjuntos como fue solicitado en el respectivo Proyecto del Software Visual Paradigm de nombre: "Entregable3.vpp".

## 8- Lecciones aprendidas

- 1) La importancia de definir bien los requerimientos con el cliente. Pues este va cambiando su parecer conforme al tiempo, por lo que es esencial establecer en una primera instancia bien los puntos a abarcar.
- 2) Mantener reuniones periódicas con los stakeholders para recibir feedback en cuanto al avance del proyecto. De esta manera prevemos cualquier disconformidad de parte del cliente en cuanto a un gran avance, así detenemos el "efecto domino" de cualquier mal entendido con nuestro cliente.
- 3) Estudiar a fondo las herramientas a utilizar, para adquirir una buena base antes de comenzar con el desarrollo y saber cuáles van a ser esenciales a la hora de crear nuestro software, en palabras mas sencillas, saber que herramientas van a cubrir los requerimientos del cliente.
- 4) Aprovechar las nuevas herramientas como Trello y Github en cuanto al trabajo en equipo ya que nos ayudaron a trabajar de una manera más eficiente, pues no se requería mantener reuniones presenciales durante el desarrollo del proyecto, por lo que la distancia ya no es un impedimento para trabajar en equipo.
- 5) Por último, fijar bien los tiempos de trabajo, para poder avanzar periódicamente sin atrasarse en las entregas correspondientes.