

# ICES-SGA

Administración de Proyectos



Ingeniería en Informática  
11/10/2016

Scandalo Pablo



## **Preámbulo**

El presente trabajo tiene como objetivo realizar las fases de iniciación y planificación para el proyecto ICES-SGA (Sistema de Gestión Académica) plasmando los conocimientos de la materia Administración de Proyectos. Se ha elegido abordar una problemática real el cual será caso de estudio para el Proyecto Final de la carrera Ingeniería en Informática.

## Índice

Introducción .....	5
Misión.....	5
Organigrama.....	6
Business Case .....	7
Problemática .....	7
Marco teórico.....	7
Alternativas .....	9
Solución propuesta .....	9
Objetivos .....	9
Beneficios globales.....	10
Factibilidad .....	10
Factibilidad financiera .....	10
Riesgos.....	11
Project Charter .....	12
Project Scope.....	14
Metodología .....	15
Partición estructurada de tareas (WBS).....	16
Calendario del proyecto (Schedule) .....	18
Plan de Recursos Humanos .....	20
Integrantes del Proyecto.....	20
Organigrama de RRHH .....	20
Sistema de recompensas.....	20
Matriz de asignación de responsabilidades (RACI) .....	21
Gestión de Costos.....	22
Plan de Riesgos.....	24
Gestión de las Comunicaciones.....	24
Canales de comunicación .....	24
Tecnología de las comunicaciones .....	24
Gestión de la Calidad.....	25
Monitoreo y Control.....	25
Conclusión .....	26

### **Anexo I**

Herramientas.....	26
Visual Studio.....	28

SQL Server .....	28
Team Foundation Version Control (TFVC).....	28
Nhibernate .....	29
Fluent NHibernate.....	29
Castle Windsor .....	29
Fluent Security.....	30
Sharp Architecture .....	30
Templify.....	31
Jquery .....	31
Bootstrap.....	31
Arquitectura .....	32
Patrones de desarrollo .....	32
Inversión de control .....	33
Inyección de dependencia.....	33
Repositorio .....	33
ASP.NET MVC .....	33
Command .....	34
Bibliografía .....	36

## Introducción

El Instituto Superior Particular Autorizado N° 9244 ICES, es una entidad educativa ubicada en Sunchales, provincia de Santa Fe. Desde sus inicios, la visión del instituto fue formar profesionales dentro de las siguientes áreas de conocimiento:

- Analista en Contabilidad y Gestión
- Analista en Sistemas

A lo largo de los años, se incorporaron nuevas carreras con el objetivo de fomentar la formación de nuevos perfiles:

- Técnico Superior en Higiene y Seguridad en el Trabajo
- Técnico Superior en Gestión de Cooperativas y Mutuales

Los procesos de integración educativa y cultural, posibilitaron consolidar en el año 1991 y 2010 convenios con la Universidad Católica de Santiago del Estero (UCSE), lo que permite finalizar a sus egresados sus carreras “No Universitarias”.

## Misión

### Educar en cooperación

Brindar una educación de alta calidad, para formar profesionales efectivos y solidarios.

## Visión

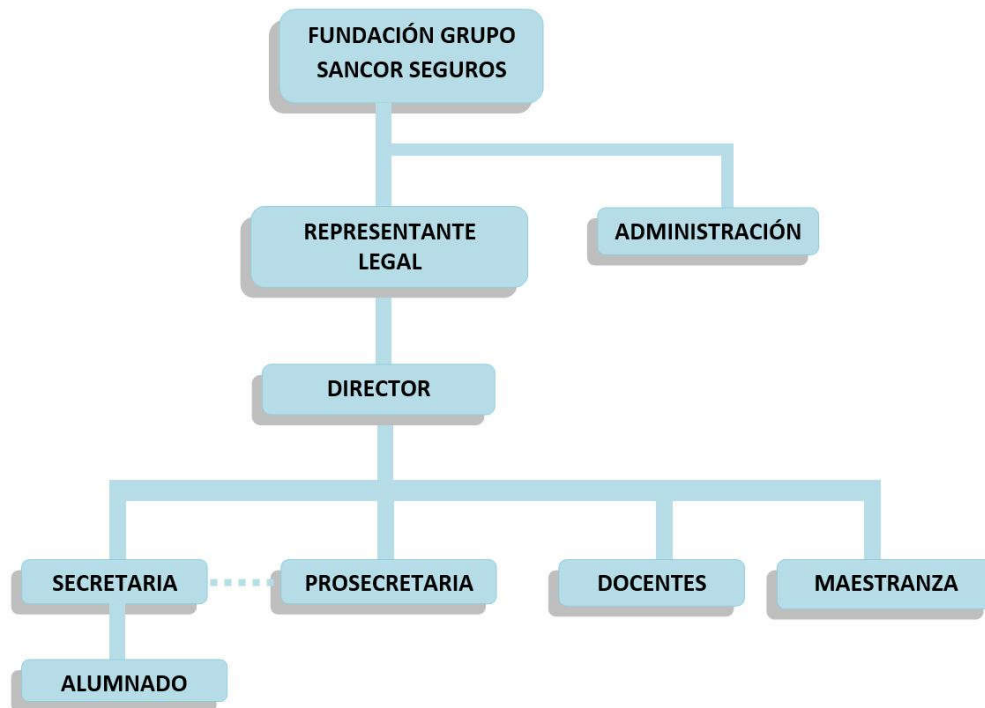
- Participar en el desarrollo regional, mediante la formación de personas íntegras, capacitadas profesionalmente e identificadas con la búsqueda del bienestar común a través del servicio.
- Contribuir al desarrollo regional, a través del desarrollo humano.

## Ejes Estratégicos

1. Recuperación del potencial de desarrollo.
2. Reconstrucción del patrimonio tecnológico-cultural.
3. Formación del ser cooperativo, apto para el desenvolvimiento en comunidad.

## Organigrama

Actualmente, ICES está bajo la órbita de la Fundación Grupo Sancor Seguros, que ha potenciado sus actividades en el marco de su política permanente de generar condiciones propicias para el desarrollo de los jóvenes y también para la promoción de los valores del cooperativismo.



## Business Case

### Detalles del Proyecto:

**Organización:** Instituto Superior Particular  
Autorizado Nº 9244 ICES



**Proyecto:** ICES-SGA (Sistema de Gestión Académica)

**Patrocinador:** Blanche Carlos  
**Director de proyecto:** Scandalo Pablo

**Fecha de Inicio:** 04/07/2016  
**Fecha de Finalización:** 25/11/2016

### Detalles del Documento:

**Estado:** Aprobado / Rechazado

**Versión:**  
1.0

**Aprobado por:**

**Fecha:** \_\_/\_\_/\_\_

**Fecha actualización:** \_\_/\_\_/\_\_

### Problemática

El instituto ICES actualmente cuenta con dos sistemas que provee el ministerio de educación y que utiliza para llevar a cabo su gestión académica. Uno de ellos es un sistema de escritorio que se implementó ya hace varios años, mientras que el recientemente incorporado en el año 2010 es totalmente Web. (Sigae WEB<sup>1</sup>)

Con la transición surgieron algunas problemáticas producto de dicho cambio. Al comenzar a usar el nuevo sistema, se detectó que algunas de las funcionalidades que proveía el anterior no estaban resueltas en su versión Web. Por ejemplo, algunas de ellas son la impresión de planillas de asistencia, regularidades y actas de condición final.

Desde entonces, para cumplir con estas necesidades, los actores vinculados deben realizar una carga por duplicado de sus alumnos, regularidades, inscripciones, lo cual conlleva un doble esfuerzo y por ende una demanda de tiempo importante.

Con el avance de la tecnología, y como consecuencia a estos requisitos, también se reconoció necesaria la posibilidad de que los alumnos cuenten con un sistema expuesto en la web para autogestión. Hoy en día, los procesos involucrados se realizan de forma personal en la secretaría de la institución y mediante diferentes medios de comunicación (envío de mails, conversaciones telefónicas).

### Marco teórico

La tecnología se inmiscuye en todos los ámbitos que hacen al hombre. Con su avance irrefrenable, vemos que la forma en que este se relaciona consigo mismo, con sus pares y con el resto de las cosas está trastocada. Se ve inmersa en un proceso de cambio que acrecentó el ritmo en los últimos cien años.

---

<sup>1</sup> <https://www.santafe.gov.ar/sigae-web/>

Los adultos han tenido que aprender a seguirle el ritmo a los avances. Los adolescentes y niños han vivido la mayor parte de su vida rodeados de la informática y la movilidad. Las instituciones; esos ámbitos donde el ser humano se desarrolla, ven la necesidad imperiosa de aggiornarse. De otro modo, dejan de representarlo. La persona no se ve identificada en ella, esto le genera cierto rechazo. Cito a continuación una reflexión que realizó Darío Sztajnszrajber en una entrevista con el diario La Nación<sup>2</sup>:

“Para los adultos del siglo XX la relación con las nuevas tecnologías suele ser una relación de exterioridad, mientras que para los jóvenes viven a las tecnologías como parte de su hábitat natural. Esto provoca consecuencias en las instituciones. La más notoria es la escuela, que sigue utilizando categorías del siglo pasado.”

Una de las transformaciones a las que se enfrentan las entidades educativas es tomar esa tecnología que rodea a los adolescentes en el mundo actual, y transformarla en una herramienta potenciadora en su desarrollo.

La informática es un ápice en el desarrollo de la técnica, que comenzó con el fin de simplificar la vida a las personas, aunque en su afán, cayó en el mecanicismo, el pragmatismo y el utilitarismo.

“Tenemos, pues, que la técnica es, por lo pronto, el esfuerzo por ahorrar el esfuerzo o, dicho de otra forma, es lo que hacemos para evitar por completo, o en parte, los quehaceres que la circunstancia primariamente nos impone”  
(Pág.35)<sup>3</sup>

Una de las grandes ventajas de la informática, para las organizaciones, sería la oficina sin papeles. Si bien el papel es el soporte fundamental de múltiples actividades, la implementación de un sistema informático bajo el compromiso con el medio ambiente, contribuyen a la reducción de manera significativa del consumo de papel. En contraparte, esto implica una serie de cambios y adaptaciones, no sólo tecnológicos, sino también culturales, administrativos y organizativos que se traducen en una mejora en la gestión y los procesos involucrados.

Así mismo, un sistema se presenta como un centralizador de labor en el cual se implementa una base de datos común. Más aún si tomamos la forma de una aplicación web, accesible desde cualquier lugar del planeta que posee conexión a internet. Un estudio que analiza el uso de Internet en países emergentes, revela que nuestro país se encuentra (en América Latina) en el tercer lugar en penetración del acceso a la red, con un 62% de la población. Dicha información fue presentada en un reporte elaborado por el centro de investigaciones Pew en marzo de 2015.

---

<sup>2</sup><http://www.lanacion.com.ar/1766411-dario-sztajnszrajber-la-tecnologia-nos-transforma-todo-el-tiempo-en-nosotros-mismos>

<sup>3</sup> Meditación de la Técnica - Ortega y Gasset



Desde el punto del marketing una página web, y la posibilidad de que el instituto, academia o universidad exponga servicios a través de internet, aprovechando así todas las posibilidades que este nos ofrece, se marca como una ventaja competitiva, un elemento diferenciador. Se crea la imagen de una institución que toma un rol proactivo ante las posibilidades, que antes de esperar que los cambios se hagan inevitables, toma la novedad, ve la posibilidad y la transforma en acción.

### Alternativas

- Implementación de algún sistema enlatado:
  - Ventajas
    - Ya están testados
    - Se implementan en el corto plazo.
  - Desventajas
    - Demasiada variedad
    - Hechos genéricamente
    - Tiempos de customización, adaptación y capacitación
    - Implican un costo para la institución.
    - La institución queda "esclava" de un sistema enlatado, del cual no tiene el código fuente, por lo tanto no es dueño de modificarlo o adaptarlo a sus necesidades.
- Desarrollar un software a medida:
  - Ventajas
    - El software se ajusta a la operatoria de la institución.
    - Al ser propietarios del sistema, existe la posibilidad de hacer cambios a futuro.
    - La capacitación es simple, ya que los involucrados participan activamente del proyecto.
  - Desventajas
    - Implica un tiempo de desarrollo e implementación más prolongado.

### Solución propuesta

Desarrollar un sistema Web utilizando el framework .NET, bajo la arquitectura MVC. Que permita cubrir cuestiones académicas del instituto. El enfoque Web da una versatilidad y una potencialidad muy amplia; permitiendo en un futuro próximo y según la administración de la institución necesite, publicar el sitio Web de cara a internet, agregando una gran facilidad a los alumnos.

Además se buscará llevar a cabo muchas de las buenas prácticas que se encuentran en el mercado, permitiendo un sistema bien estructurado, con buena mantenibilidad. Un aspecto muy útil, pensando en las potencialidades del sistema y el futuro crecimiento.

### Objetivos

#### Particulares

- Desarrollar un Sistema de Autogestión para Alumnos
- Contribuir a la solución de una problemática real.

## Generales

- Facilitar la administración del nivel terciario del instituto.
- Complementar el sistema provisto por el ministerio.
- Lograr un desarrollo que sirva como base para nuevas funcionalidades:
  - Sistema de Autogestión para Alumnado.
  - Administración de aranceles y estado contable.

## Beneficios globales

- Mejora la imagen de la institución
- Simplificar y organizar los trámites y procesos académicos.
- Apoyar a las secretarías de la institución en los procesos académicos.

## Factibilidad

### Factibilidad técnica

Desde el punto de vista de la factibilidad técnica hay una curva de aprendizaje de algunas herramientas utilizadas en el desarrollo del sistema. Más allá de eso, los involucrados ya poseen conocimientos en el desarrollo de aplicaciones de igual magnitud.

### Factibilidad financiera

Se ha optado por utilizar herramientas gratuitas para el desarrollo de software. Dentro del entorno .NET, Microsoft ofrece herramientas de desarrollo totalmente gratuitas: "Visual Studio Community", las cuales son versiones básicas para estudiantes y programación amateur. Estas ediciones utilizadas son iguales al entorno de desarrollo comercial, pero sin características avanzadas de integración, las cuales no son relevantes para este caso puntual. Así mismo, utilizaremos una versión gratuita del reconocido motor de base de datos de Microsoft (SQL Server Express).

El proyecto no incluye costos de publicación y registros de nombres de dominio (NIC). Ambos aspectos quedan como responsabilidad del cliente y no están dentro de esta planificación o alcance. Además, no será necesario incluir adquisiciones para su realización.

Los costos del proyecto están determinados por el pago a los involucrados. Desarrolladores, analista y tester recibirán una remuneración de acuerdo a las horas de trabajo, mientras que el director de proyecto recibirá un pago estipulado de \$40000.

Tomando como referencia los valores del colegio de profesionales de ciencias informáticas de Entre Ríos<sup>4</sup>, se utilizará como precio por hora: \$300 para todos los casos.

Se estima durante la ejecución del mismo un total aproximado de 550 horas de trabajo

	Horas Requeridas	Precio Hora
Desarrolladores	2 a 4 horas diarias	\$300
Tester	4 horas semanales	\$300
Analista	40 horas en total	\$300

---

<sup>4</sup> [http://www.coprocier.org.ar/seccion\\_2.html](http://www.coprocier.org.ar/seccion_2.html)

Aproximación presupuesto:

550 horas * \$300 (precio hora)	\$165.000,00
Sueldo Director de Proyecto	\$40.000,00
<b>TOTAL</b>	<b>\$205.000,00</b>

## Riesgos

El riesgo detectado en el proyecto es la postergación de la fecha de finalización. Teniendo en cuenta que los involucrados no se dedicarán full-time a la realización de este proyecto, debemos anticipar un posible incumplimiento de algunos plazos definidos en los entregables. Se debe tener en cuenta que cada uno de ellos posee actividades tanto laborales y personales fuera del proyecto. A raíz de esto, el tiempo disponible se ve acotado a la predisposición de cada colaborador e involucrado en sus momentos libres. De todas formas se estima que en condiciones normales o favorables el plazo de finalización será respetado.

## Project Charter

### Información del Proyecto:

**Organización:** Instituto Superior Particular  
Autorizado Nº 9244 ICES



**Proyecto:** ICES-SGA (Sistema de Gestión Académica)

**Fecha de Inicio:** 04/07/2016

**Fecha de Finalización:** 25/11/2016

**Cliente:** Fundación Grupo Sancor Seguros

**Fecha:** \_\_/\_\_/\_\_

### Justificación del Proyecto:

La realización de este proyecto tiene origen en el trabajo final de grado obligatorio para la finalización de la carrera Ingeniería en Informática de la Universidad Católica de Santiago del Estero –UCSE. Como caso de estudio, resulta significativo tomar una problemática real del instituto ICES de la ciudad de Sunchales y reunir los esfuerzos necesarios para dar curso a una solución que permita plasmar los conocimientos y la formación adquirida en esta etapa educativa.

### Objetivos del Proyecto:

Desarrollar un sistema Web utilizando el framework .NET, bajo la arquitectura MVC.

Además se buscará llevar a cabo muchas de las buenas prácticas que se encuentran en el mercado, permitiendo un sistema bien estructurado, con buena mantenibilidad. Un aspecto muy útil, pensando en las potencialidades del sistema y el futuro crecimiento.

### Roles y responsabilidades

Rol	Persona asignada	Responsabilidades
Patrocinador	Blanche Carlos	Proveer los medios necesarios para la realización del proyecto.
Director de proyecto	Scandalo Pablo	Afrontar los desafíos del proyecto, resolver inconvenientes en su transcurso y entregar a tiempo un sistema de calidad que cumpla con las expectativas del cliente.
Analista	Piovano Pablo	Realizar el análisis de requerimiento, modelado, casos de uso y especificación de los procesos del sistema.
Desarrollador	Lazzaroni Jose Bertolini Lucas Giuliano Agustín	Desarrollar la interfaz gráfica y codificar los módulos de procesos y acceso a datos del sistema. Crear la base de datos de la aplicación.
Tester	Mallía Lucas	Definir y llevar a cabo casos de prueba para detectar potenciales vulnerabilidades y bugs en la aplicación.
Usuarios claves	Assone Andrea Riberi Laura	Colaborar en el análisis de los procesos de la organización. Participar en las reuniones

		del ciclo de vida del proyecto para analizar el entregable resultante y especificar solicitudes de cambio.
--	--	--

**Interesados:**

Los principales interesados del proyecto son el director, secretarias, docentes y alumnado del instituto. A sí mismo, debemos destacar a Fundación Grupo Sancor Seguros.

## Project Scope

### Información del Proyecto:

**Organización:** Instituto Superior Particular  
Autorizado N° 9244 ICES



**Proyecto:** ICES-SGA (Sistema de Gestión Académica)

**Fecha de Inicio:** 04/07/2016

**Fecha de Finalización:** 25/11/2016

**Cliente:** Fundación Grupo Sancor Seguros

**Fecha de actualización:** 26/06/2016

**Preparado por:** Scandalo Pablo (Director de proyecto)

### Breve descripción del proyecto:

El proyecto se enfoca en el desarrollo de un sistema Web de gestión académica el cual provea mecanismo de automatización para registrar y administrar las actividades del instituto. Se busca simplificar y organizar los trámites y procesos académicos.

### Alcance del producto:

#### **Funcionalidades**

- Gestionar calificaciones
- Autogestión (inscripciones, consultas)
- Generar e imprimir planilla de asistencias
- Generar e imprimir planilla de regularidades
- Generar e imprimir acta de condición final

#### **Funcionalidades implícitas**

- Gestión docentes
- Gestión alumnos
- Gestión administrativos
- Gestión de cátedras
- Gestión de carreras
- Gestión de planes de estudios
- Gestión de correlatividades

#### **El entregable final excluye las siguientes funcionalidades**

- Aula virtual
- Espacio para los docentes para subir archivos
- Chat y sistema de mensajería
- Contabilidad y pago de aranceles

#### **El proyecto no incluye**

- Servicio de Hosting y publicación del entregable.
- Registros de nombres de dominio (NIC)

Para la publicación del sitio en la Web, el cliente está dispuesto a contratar el servicio de hosting a la empresa local “Integrar – Soluciones Informáticas”. Integrar es una empresa del grupo cooperativo SanCor, creada en 1990 con el fin de proveer servicios informáticos a SanCor Coop. Unidas Ltda., a entidades del Grupo Cooperativo y a otras empresas. Fundación Grupo Sancor Seguros ya contrata algunos servicios al a empresa de referencia.

#### **Puntos críticos:**

Se debe contar a lo largo de todo el proyecto con buena predisposición de los usuarios para colaborar en el análisis y recopilación de los requerimientos. Además se les pedirá asistir a las reuniones periódicas de control de avances del proyecto. Esto es sumamente importante para que no haya desviaciones trascendentales en el desarrollo del software, las cuales pueden hacer fracasar el proyecto si no son tenidas en cuenta en el tiempo adecuado.

El entregable final del proyecto traerá cambios significativos en los procesos administrativos de la organización. Por este motivo es importante capacitar y concientizar a los usuarios y al personal sobre las ventajas y resultados obtenidos de su implementación.

#### **Requisitos de aprobación:**

- Software funcional.
- Se requerirá que el software cumpla con el documento OWASP Top 10 para prevenir los 10 riesgos más relevantes en la seguridad de aplicaciones Web.
- Documentación del proyecto.

## Metodología

Ciclo de vida: se utilizará como metodología el proceso de desarrollo “Iterativo e incremental”.

El proyecto se dividirá en fases correspondientes a los entregables más significativos para los interesados y se ejecutarán de forma secuencial.

Además se hará énfasis en la continua participación de los interesados, lo que conlleva a una metodología orientada al cambio.

Al finalizar cada una de las fases, será necesario hacer una evaluación del estado actual del desarrollo y analizar las funcionalidades del producto obtenidas en el incremento de dicha iteración. El entregable resultante será aprobado o aceptado por el cliente proporcionando retroalimentación y garantizando que el producto refleja sus necesidades actuales.

Esta metodología tiene ciertas ventajas:

- El cliente pueda obtener los beneficios del proyecto de forma incremental. (En cada iteración el equipo evoluciona el producto).
- El cliente obtiene entregables importantes y funcional a partir de las primeras iteraciones
- Se puede gestionar las expectativas del cliente y se pueden tomar decisiones de cambio en cada iteración.
- Se minimiza el número de errores que se producen en el desarrollo y se aumenta la calidad.
- Las probabilidades de que el proyecto fracase son bajas.

## Partición estructurada de tareas (WBS)

### 1) Planificación:

- a. Business Case - Análisis de factibilidad
- b. Aprobación del Proyecto
  - i. Asignar al director de proyecto y los miembros del equipo.
  - ii. Creación del Acta de Constitución del proyecto.
  - iii. Definición de objetivos y alcance.
- c. Etapa de relevamiento
  - i. Relevar información esencial del sistema.
  - ii. Definir y desarrollar mockups de los diferentes módulos.

### 2) Ejecución:

- a. Etapa de iniciación
  - i. Creación del repositorio de código fuente.
  - ii. Definir la arquitectura de la aplicación.
  - iii. Realización del modelo de datos - Modelado de Clases
  - iv. Integrar las librerías a utilizar dentro de la solución.
- b. Codificación del sistema:
  - i. Perfiles de Usuario
    - 1. Módulo Alumnos
      - a. Integración/Importación desde SIGAE
      - b. Alta/Baja/Modificación y Consultas
    - 2. Módulo Docentes
      - a. Alta/Baja/Modificación y Consultas
    - 3. Administrativos
      - a. Alta/Baja/Modificación y Consultas
  - ii. Módulo Seguridad
    - 1. Aislar vista de cada perfil (Generar áreas)
    - 2. Administrador de cuentas de usuario
      - a. Login/Logout/Recuperar Cuenta
      - b. Administrar perfiles
  - iii. Módulo Planes de estudio
    - 1. Carreras
      - a. Alta/Baja/Modificación y Consultas
    - 2. Cátedras
      - a. Alta/Baja/Modificación y Consultas
    - 3. Gestión planes de estudio
      - a. Alta/Baja/Modificación y Consultas
  - iv. Módulo Ciclo lectivo:
    - 1. Alta/Baja/Modificación y Consultas
    - 2. Inscripciones
      - a. Alta/Baja/Modificación y Consultas
    - 3. Calificaciones
      - a. Alta/Baja/Modificación y Consultas



b. Filtros

- i. Administrativo
- ii. Docente
- iii. Alumno

v. Reportes

- 1. Elegir herramienta
- 2. Reporte Asistencias
- 3. Reporte Regularidades
- 4. Reporte Acta final

c. Testing, Testing Seguridad (OWASP Top 10). Corrección de bugs

3) Cierre del proyecto

- a. Deploy de la solución
- b. Documentación del proyecto

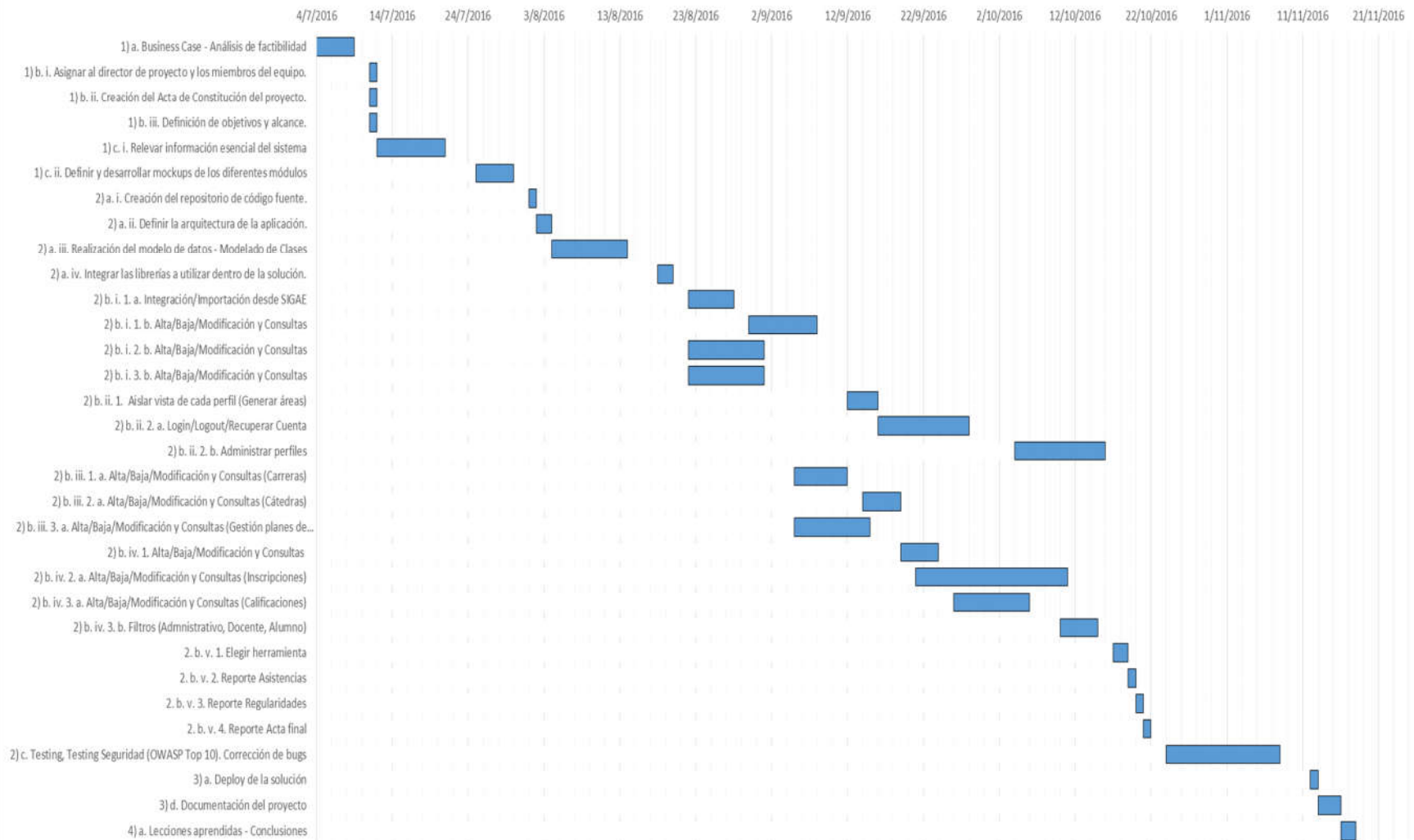
4) Evaluación del proyecto

- a. Lecciones aprendidas – Conclusiones

## Calendario del proyecto (Schedule)

Etap	Subetapa/ Módulo	Fecha Inicio	Dur. (días)	Fecha Fin	Actividad	
Planificación	--	4/7/2016	5	8/7/2016	1) a. Business Case - Análisis de factibilidad	
	Aprobación del Proyecto	11/7/2016	1	11/7/2016	1) b. i. Asignar al director de proyecto y los miembros del equipo.	
		11/7/2016	1	11/7/2016	1) b. ii. Creación del Acta de Constitución del proyecto.	
		11/7/2016	1	11/7/2016	1) b. iii. Definición de objetivos y alcance.	
	Relevamiento	12/7/2016	9	22/7/2016	1) c. i. Relevar información esencial del sistema	
		25/7/2016	5	29/7/2016	1) c. ii. Definir y desarrollar mockups de los diferentes módulos	
Ejecución	Etap	1/8/2016	1	1/8/2016	2) a. i. Creación del repositorio de código fuente.	
		2/8/2016	2	3/8/2016	2) a. ii. Definir la arquitectura de la aplicación.	
		4/8/2016	10	17/8/2016	2) a. iii. Realización del modelo de datos - Modelado de Clases	
		18/8/2016	2	19/8/2016	2) a. iv. Integrar las librerías a utilizar dentro de la solución.	
	Codificación del sistema	Módulo Alumnos	22/8/2016	6	29/8/2016	2) b. i. 1. a. Integración/Importación desde SIGAE
			30/8/2016	9	9/9/2016	2) b. i. 1. b. Alta/Baja/Modificación y Consultas
		Módulo Docentes	22/8/2016	10	2/9/2016	2) b. i. 2. a. Alta/Baja/Modificación y Consultas
		Módulo Admin.	22/8/2016	10	2/9/2016	2) b. i. 3. a. Alta/Baja/Modificación y Consultas
		Módulo Seguridad	12/9/2016	4	15/9/2016	2) b. ii. 1. Aislar vista de cada perfil (Generar áreas)
			16/9/2016	12	3/10/2016	2) b. ii. 2. a. Login/Logout/Recuperar Cuenta
			4/10/2016	12	19/10/2016	2) b. ii. 2. b. Administrar perfiles
		Módulo Planes de estudio	5/9/2016	7	13/9/2016	2) b. iii. 1. a. Alta/Baja/Modificación y Consultas (Carreras)
			14/9/2016	5	20/9/2016	2) b. iii. 2. a. Alta/Baja/Modificación y Consultas (Cátedras)
			5/9/2016	10	16/9/2016	2) b. iii. 3. a. Alta/Baja/Modificación y Consultas (Gestión planes de estudio)
		Módulo Ciclo lectivo	19/9/2016	5	23/9/2016	2) b. iv. 1. Alta/Baja/Modificación y Consultas
			21/9/2016	20	18/10/2016	2) b. iv. 2. a. Alta/Baja/Modificación y Consultas (Inscripciones)
			26/9/2016	10	7/10/2016	2) b. iv. 3. a. Alta/Baja/Modificación y Consultas (Calificaciones)
			10/10/2016	5	14/10/2016	2) b. iv. 3. b. Filtros (Admnistrativo, Docente, Alumno)
		Reportes	17/10/2016	2	18/10/2016	2. b. v. 1. Elegir herramienta
			19/10/2016	1	19/10/2016	2. b. v. 2. Reporte Asistencias
			20/10/2016	1	20/10/2016	2. b. v. 3. Reporte Regularidades
			21/10/2016	1	21/10/2016	2. b. v. 4. Reporte Acta final
	Testing		24/10/2016	15	11/11/2016	2) c. Testing, Testing Seguridad (OWASP Top 10). Corrección de bugs
Cierre del proyecto		12/11/2016	1	12/11/2016	3) a. Deploy de la solución	
		13/11/2016	3	15/11/2016	3) d. Documentación del proyecto	
Evaluación del proyecto		16/11/2016	2	17/11/2016	4) a. Lecciones aprendidas - Conclusiones	

# ICES-SGA



## Plan de Recursos Humanos

### Integrantes del Proyecto

**Patrocinador:** Blanche Carlos

**Director de proyecto:** Scandalo Pablo

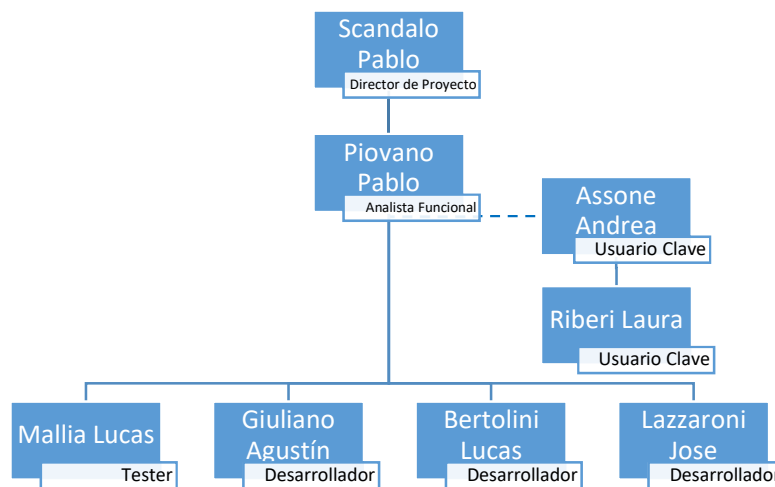
**Analista:** Piovano Pablo

**Desarrollador:** Lazzaroni Jose  
Bertolini Lucas  
Giuliano Agustín

**Tester:** Mallía Lucas

**Usuarios claves:** Assone Andrea  
Riberi Laura

### Organigrama de RRHH



Al ser un equipo pequeño, es ligeramente jerárquico y democrático.

### Sistema de recompensas

- Viernes libre si los tiempos van bien.
- Gran cena de cierre en proyecto exitoso.

## Matriz de asignación de responsabilidades (RACI)

Rol			Descripción
<b>R</b>	Responsable	Responsable	Este rol corresponde a quien efectivamente realiza la tarea. Lo más habitual es que exista sólo un encargado (R) por cada tarea; si existe más de uno, entonces el trabajo debería ser subdividido a un nivel más bajo, usando para ello las matrices RASCI.
<b>A</b>	Accountable	Quien rinde cuentas	Este rol se responsabiliza de que la tarea se realice y es el que debe rendir cuentas sobre su ejecución. Sólo puede existir una persona que deba rendir cuentas (A) de que la tarea sea ejecutada por su responsable (R).
<b>C</b>	Consulted	Consultado	Este rol posee alguna información o capacidad necesaria para realizar la tarea.
<b>I</b>	Informed	Informado	Este rol debe ser informado sobre el avance y los resultados de la ejecución de la tarea. A diferencia del consultado (C), la comunicación es unidireccional.

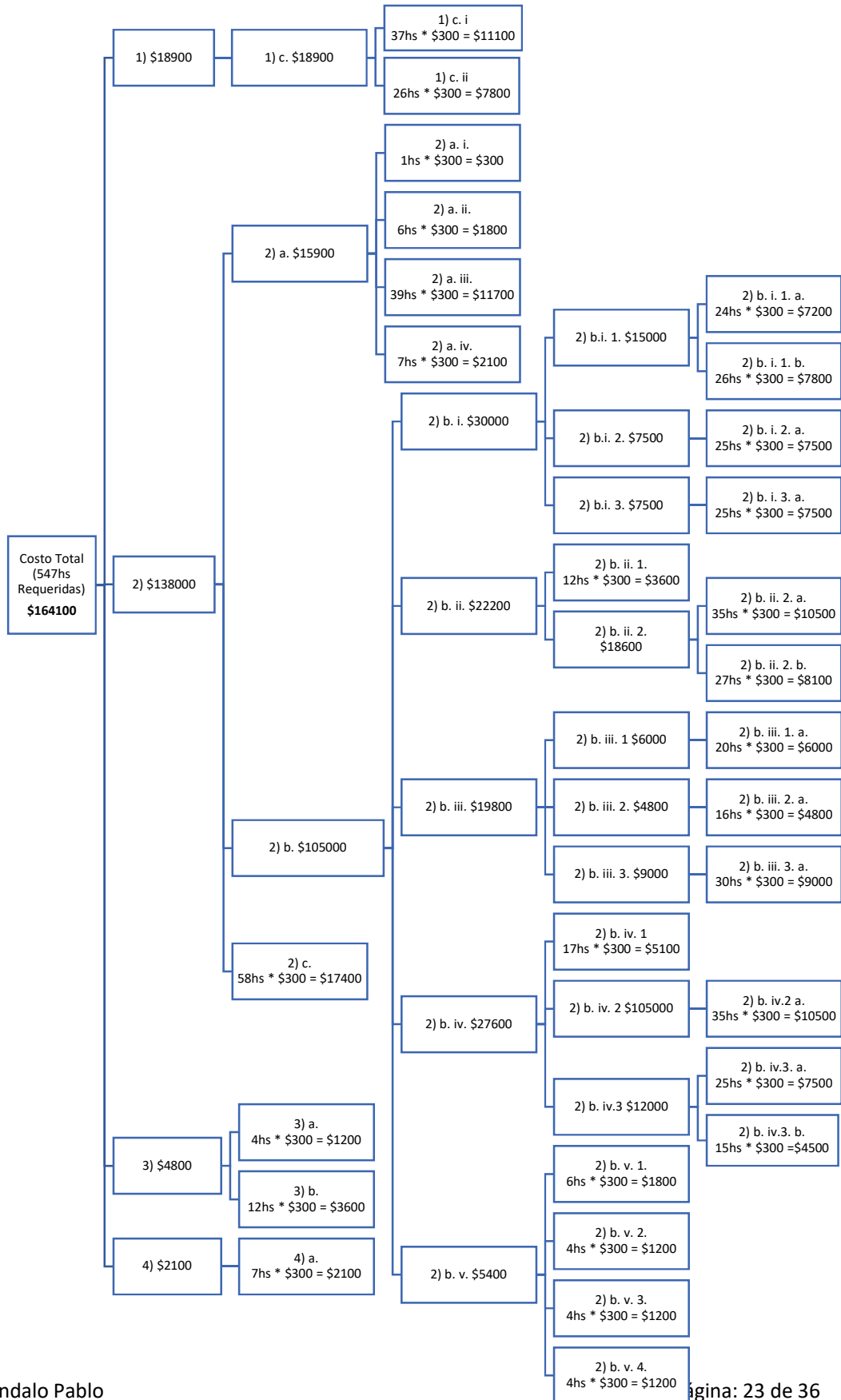
Responsabilidad		Patrocinador	Dir. Proy	Analista	Desarrollador			Tester	Usuarios Claves	
		Blanche Carlos	Scandalo Pablo	Piovano Pablo	Lazzaroni Jose	Bertolini Lucas	Giuliano Agustín	Mallía Lucas	Assone Andrea	Riberi Laura
Planificación	Realización Busines Case	C	R						C	C
	Aprobación del Proyecto	R	I							
	Relevamiento		A	R					C	C
Ejecución	Etapas de Iniciación		A		R	C	C			
	Codificación del sistema	Módulo Alumnos	A		R					
		Módulo Docentes	A			R				
		Módulo Admin.	A				R			
		Módulo Seguridad	A		R					
		Módulo Planes de estudio	A			R	R			
		Módulo Ciclo lectivo	A			R				
		Reportes	A				R			
	Testing		A					R		
	Cierre del proyecto	I	A		R					
	Evaluación del proyecto	I	R	C	C	C	C	C	C	C

## Gestión de Costos

Para la estimación de costos se utilizará como herramienta “Estimación ascendente” tomando como base la partición estructurada de tareas. Se ha optado por esta metodología ya que, si bien no se tiene en un alto nivel de detalle de cada actividad puntual, podemos obtener una estimación más precisa que si haríamos “Estimación análoga”.

Se implementará en primera instancia estimación mediante PERT para aproximarnos a la cantidad de horas requerida de cada actividad/componente de nuestro WBS.

Actividad	Costo estimado en Horas			
	Costo mínimo	Costo más probable	Costo máximo	Costo estimado
1) c. i. Relevar información esencial del sistema	27	36	50	<b>37</b>
1) c. ii. Definir y desarrollar mockups de los diferentes módulos	25	25	30	<b>26</b>
2) a. i. Creación del repositorio de código fuente.	1	1	2	<b>1</b>
2) a. ii. Definir la arquitectura de la aplicación.	4	6	8	<b>6</b>
2) a. iii. Realización del modelo de datos - Modelado de Clases	30	40	45	<b>39</b>
2) a. iv. Integrar las librerías a utilizar dentro de la solución.	6	7	8	<b>7</b>
2) b. i. 1. a. Integración/Importación desde SIGAE	18	24	30	<b>24</b>
2) b. i. 1. b. Alta/Baja/Modificación y Consultas	20	27	30	<b>26</b>
2) b. i. 2. b. Alta/Baja/Modificación y Consultas	18	26	28	<b>25</b>
2) b. i. 3. b. Alta/Baja/Modificación y Consultas	20	25	30	<b>25</b>
2) b. ii. 1. Aislar vista de cada perfil (Generar áreas)	8	12	15	<b>12</b>
2) b. ii. 2. a. Login/Logout/Recuperar Cuenta	30	35	42	<b>35</b>
2) b. ii. 2. b. Administrar perfiles	24	26	32	<b>27</b>
2) b. iii. 1. a. Alta/Baja/Modificación y Consultas (Carreras)	14	20	24	<b>20</b>
2) b. iii. 2. a. Alta/Baja/Modificación y Consultas (Cátedras)	12	16	20	<b>16</b>
2) b. iii. 3. a. Alta/Baja/Modificación y Consultas (Gestión planes de estudio)	25	30	35	<b>30</b>
2) b. iv. 1. Alta/Baja/Modificación y Consultas	10	18	22	<b>17</b>
2) b. iv. 2. a. Alta/Baja/Modificación y Consultas (Inscripciones)	30	35	40	<b>35</b>
2) b. iv. 3. a. Alta/Baja/Modificación y Consultas (Calificaciones)	20	25	28	<b>25</b>
2) b. iv. 3. b. Filtros (Administrativo, Docente, Alumno)	10	15	18	<b>15</b>
2. b. v. 1. Elegir herramienta	5	6	7	<b>6</b>
2. b. v. 2. Reporte Asistencias	3	4	4	<b>4</b>
2. b. v. 3. Reporte Regularidades	3	4	4	<b>4</b>
2. b. v. 4. Reporte Acta final	3	4	4	<b>4</b>
2) c. Testing, Testing Seguridad (OWASP Top 10). Corrección de bugs	45	60	60	<b>58</b>
3) a. Deploy de la solución	3	4	4	<b>4</b>
3) d. Documentación del proyecto	10	12	12	<b>12</b>
4) a. Lecciones aprendidas - Conclusiones	6	7	8	<b>7</b>
<b>TOTAL</b>				<b>547 hs</b>



Se estipulará un monto adicional para contingencias equivalente a 50hs de trabajo, lo cual representa un total de \$15000 de reserva.

A su vez, desde 05/09/2016 hasta a la fecha 21/10/2016 se estiman 4 horas adicionales por semana destinadas al testing de los diferentes módulos de la aplicación. Esto es el equivalente a 7 semanas de trabajo, lo cual da una suma total de 28hs (7 semanas \* 4hs semanales)

Aproximación final de costos:

Horas Desarrolladores	547 horas * \$300 (precio hora)	\$164.100,00
Horas adicionales Testing	28 horas * \$300 (precio hora)	\$8.400,00
Sueldo Director de Proyecto		\$40.000,00
Reserva para contingencias		\$15.000,00
<b>TOTAL</b>		<b>\$227.500,00</b>

## Plan de Riesgos

Como se mencionó anteriormente en el Business Case, el principal riesgo identificado es la postergación de la fecha de finalización del proyecto. Los tiempos que poseen los involucrados se ven acotados por las responsabilidades que poseen diariamente.

Se requiere una actitud proactiva de cada uno de los miembros del equipo de proyecto para evitar demoras ante cualquier problema o eventualidad, una fluida comunicación entre los miembros y un alto grado de solidaridad.

El impacto de dicho riesgo se acrecienta a medida que el plazo de finalización del proyecto se posterga. Se compromete la implementación a tiempo del sistema ICES – SGA para su utilización en el ciclo lectivo del año 2017.

Se utilizará como estrategia *mitigar el riesgo*, haciendo una monitorización y control exhaustivo del proyecto. Como punto estratégico, en las reuniones de estado del proyecto se identificarán y analizarán los aspectos considerables a este riesgo puntual.

## Gestión de las Comunicaciones

### Canales de comunicación

El equipo de proyecto está compuesto por 9 interesados:

*(Patrocinador, Director de Proyecto, Analista funcional, 3 desarrolladores, 1 tester, 2 usuarios claves).*

Por lo tanto se detectan 36 canales de comunicación  $[(9 * 8) / 2 = 36]$

### Tecnología de las comunicaciones

Se ha optado por la utilización de 3 tecnologías para distribuir la información dentro del equipo del proyecto.

- **Entrevistas** que coordinará el analista funcional con los usuarios claves para el análisis de los requerimientos del sistema a desarrollar.



- **Reuniones cada 15 días**, para evaluar en conjunto el estado de avance del proyecto.
  - Se realizarán en ICES, previo acuerdo con todos los interesados
  - La duración establecida es de una hora aproximadamente.
- **Correo electrónico** para coordinar reuniones o consultar dudas puntuales que no requieran una respuesta urgente.
- **Mensajería y videoconferencia** mediante el uso de Skype. Para cuestiones más urgentes o que requieran compartir pantallas (documentos/código fuente) en puntos geográficamente distantes.

## Gestión de la Calidad

Con objetivo de prevenir errores y/o defectos en el entregable final del proyecto y con el fin de tener un cliente satisfecho, se trabajará con un enfoque proactivo para la gestión de calidad.

Se utilizará como metodología el proceso de desarrollo “Iterativo e incremental”, la cual permite generar software de calidad buscando en forma permanente mejora continua.

El área de testing destinada al proyecto participará para garantizar la calidad del entregable final. Se busca mitigar cualquier costo de falla antes de que el producto llegue al cliente.

Como particularidades no se contará con auditorias puesto que no se tiene a involucrados para dicha actividad y la dimensión del proyecto no lo justifica. Del mismo modo, se descartara el uso de alguna norma de calidad preexistente.

El comité de control de cambios, en el cual participarán el patrocinador y director de proyecto, será el responsable de aprobar y/o rechazar las solicitudes de cambio.

## Monitoreo y Control

El monitoreo y control del Proyecto se llevara a cabo a lo largo de toda el ciclo de vida del proyecto, y estará a cargo del director de proyecto (Scandalo Pablo).

Para evaluar el desempeño del proyecto durante su ejecución se utilizara la Gestión del valor ganado, lo cual permitirá controlar la gestión integrada del alcance, la agenda y los costos. La intención es detectar a tiempo aquellas partes del proyecto que requieran una atención especial e identificar acciones preventivas o correctivas para resolver cualquier problema de desempeño que tenga el equipo.

Cada 15 días, y posteriori a la reunión de avance, el director de proyecto entregará un informe a los interesados con la situación actual del proyecto. El informe no

Ante la necesidad de adquirir algún cambio, el comité de cambios evaluará el impacto del mismo e informará la decisión final a todos los interesados.

## Conclusión

A lo largo de este trabajo se ha trabajado con la “Guía de los fundamentos para la dirección de proyectos (Guía del PMBOK®)” aplicando conocimientos, habilidades, herramientas y técnicas a las actividades del proyecto. Si bien por el momento solo se han abordado las fases de iniciación y planificación, el presente trabajo dará continuidad al resto de las fases de la dirección de proyectos con el fin de cumplir con los requerimientos del Sistema de Gestión Académica ICES-SGA.

Es importante destacar que las diferentes estimaciones realizadas no ha resultado una tarea sencilla debido a la falta de experiencia en este tipo de labor. Así mismo, es importante enfatizar la oportunidad de aplicar los conocimientos de la materia “Administración de Proyectos” a un caso de estudio real que dará soporte al Proyecto Final de la carrera “Ingeniería en informática”.

La administración de proyectos sin dudas asegura la materialización de los resultados esperados.

## **Anexo I**

## Herramientas

### Visual Studio

Entorno de desarrollo creado por Microsoft Inc.

El lenguaje de programación elegido es C#, el cual pertenece a la plataforma de lenguajes de .Net. En conjunto con Visual Studio, brinda soluciones simples para la realización de un proyecto web.



### SQL Server

Motor de base de datos relacionales perteneciente a Microsoft Inc. Se acopla muy bien con el framework de desarrollo y el IDE.



### Team Foundation Version Control (TFVC)

Es un sistema de control de versiones centralizado ofrecido por Microsoft. Forma parte de Team Foundation Server, que además incluye, reportes, manejo de requerimientos, manejo de proyectos (para desarrollo ágil y waterfall teams), builds automatizados, administración de laboratorios, testeo.

Dentro del ecosistema Microsoft, .net framework y Visual Studio, TFS se integra muy bien. Existe "Team Explorer Everywhere" para utilizarlo fuera de estas plataformas, aunque no está tan maduro.

Entre las alternativas que actualmente Microsoft ofrece y frente a las necesidades que el proyecto implica, se optó por el uso de TFS desde [www.visualstudio.com](http://www.visualstudio.com) en Visual Studio Online. Como particularidad, esta versión es totalmente gratuita hasta 5 desarrolladores dentro del equipo.

### Nhibernate

NHibernate es una herramienta madura de mapeo objeto-relacional (ORM) y de código abierto que deriva de Hibernate (Java) pero orientado a la plataforma .NET. Al utilizar esta herramienta, podemos elegir libremente el motor de base de datos a utilizar con solo cambiar una línea de código en nuestra configuración. Entre ellas: MySQL, PostgreSQL, Oracle, MS SQL Server. Como referencia, se desarrolla activamente y es utilizado en miles de proyectos exitosos.



### Fluent NHibernate

Fluent NHibernate es una alternativa a la tradicionalmente conocida forma de mapeo por archivos XML que ofrece NHibernate estándar. Como particularidad, los mapeos se establecen mediante lenguaje C# lo que permite una fácil refactorización, mejora la legibilidad y produce código más conciso.

Asimismo posee otras herramientas que aportan ciertas ventajas:

- Mapeo automático mediante convenciones previamente definidas.
- Testing para el mapeo de entidades.

### Castle Windsor

Castle Windsor es un framework maduro de código abierto para la plataforma .NET. Se trata puntualmente de un “Contenedor” que usa el principio “Inversión de control” para manejar clases. Es decir, su creación, destrucción, ciclo de vida, configuración y dependencias. De esta manera las clases no necesitan obtener y configurar las clases sobre las cuales dependen.

Esto reduce drásticamente el acoplamiento en un sistema y, como consecuencia, facilita el reúso y la testabilidad.

Muchas veces se piensa en esto como “inyección de dependencia”. Si bien esto es una consecuencia y una característica de los contenedores de inversión de control, no es el objetivo principal.

### Fluent Security

Fluent Security es una librería que permite configurar fácilmente la seguridad de una aplicación ASP.NET MVC.

Entre sus principales ventajas podemos nombrar:

- La autorización puede setearse a nivel de acciones individuales o a nivel de controlador de la aplicación.
- La seguridad se especifica en un lugar centralizado de manera legible y mantenible, por lo que no es necesario decorar los controladores con atributos que saturan el código fuente.
- Posee mecanismos para especificar reglas por defecto para evitar olvidos o autorizaciones no deseadas.
- Permite extender y crear nuestras propias reglas de autorización.
- La configuración de seguridad se puede testear para verificar su correcto funcionamiento



### Sharp Architecture

Fundamento de arquitectura que permite crear rápidamente aplicaciones mantenibles haciendo provecho del marco ASP.NET MVC con NHibernate.

Mediante esta arquitectura, el desarrollador disminuye el código necesario para tareas comunes y repetitivas incrementando la calidad del producto final. Se caracteriza por desligar al equipo de trabajo a detalles de infraestructura permitiendo centrar sus esfuerzos sólo a la problemática que necesitan resolver. De ésta forma, el equipo pasa la mayor parte de su tiempo agregando valor para el cliente, centrándose en la lógica de negocio.

Sharp Architecture se basa en principios descritos en el libro “Design Patterns: Elements of Reusable Object-Oriented Software (ISBN 0-201-63361-2)”<sup>5</sup>

---

<sup>5</sup> Design Patterns: Elements of Reusable Object-Oriented Software (ISBN 0-201-63361-2), a menudo apodado GoF o Gang-Of-Four (la banda de los cuatro, debido a sus cuatro autores), es un

## Templify

Es un proyecto Open Source realizado por “endjin” que permite generar una solución pre-armada y configurada. Tiene ciertas ventajas porque ya involucra muchas buenas prácticas y componentes previamente mencionados. Templify propone modificar la estructura de las soluciones que ofrece Microsoft para dar soporte a muchas de las ventajas hasta aquí explicadas.

## Jquery

Jquery es una biblioteca JavaScript que permite generar código que se ejecuta del lado cliente. Mediante su uso logramos simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX a páginas Web.

Es software libre y de código abierto, posee un doble licenciamiento bajo la Licencia MIT y la Licencia Pública General de GNU v2, permitiendo su uso en proyectos libres y privados.

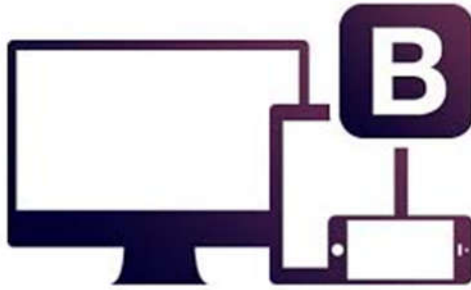


## Bootstrap

Hoja de estilos creada por Twitter y liberada para el uso común. Entre las principales ventajas está su amplio uso y su diseño responsivo multiplataforma. . Contiene plantillas de diseño con tipografía, formularios, botones, cuadros, menús de navegación y otros elementos de diseño basado en HTML y CSS, así como, extensiones de JavaScript opcionales adicionales.

---

manual sobre ingeniería del software que describe soluciones a problemas habituales en el diseño de software. Los autores de este libro son Erich Gamma, Richard Helm, Ralph Johnson y John Vlissides. El manual se divide en dos partes: los primeros dos capítulos, que exploran las ventajas y dificultades de la programación orientada a objetos, y el resto, que describen un conjunto de patrones de diseño software.



## Arquitectura

Para llevar a cabo un software mantenible se creará una solución en capas, algunas predefinidas en la arquitectura propuestas por Sharp Architecture. Podemos listar las siguientes:

- Dominio
- Infraestructura
- Seguridad
- Tareas
- Web

## Patrones de desarrollo

Así mismo el uso de patrones ayuda ampliamente a un desarrollo de calidad.

“Los patrones de diseño son el esqueleto de las soluciones a problemas comunes en el desarrollo de software.”

Brindan una solución ya probada y documentada a problemas de desarrollo de software que están sujetos a contextos similares.

Existen varios patrones de diseño popularmente conocidos, los cuales se clasifican como se muestra a continuación:

- Patrones Creacionales: Inicialización y configuración de objetos.
- Patrones Estructurales: Separan la interfaz de la implementación. Se ocupan de cómo las clases y objetos se agrupan, para formar estructuras más grandes.
- Patrones de Comportamiento: Más que describir objetos o clases, describen la comunicación entre ellos.

En el sistema se implementarán los siguientes patrones:



## Inversión de control

Inversión de Control es un principio usado por frameworks como una forma de permitir que los desarrolladores extiendan el framework o creen aplicaciones usándolo. La idea básica es que el framework conoce los objetos del programador, y hace invocaciones sobre ellos.

Esto es lo opuesto de usar una API, donde el código del desarrollador hace invocaciones al código de la API. Por lo tanto, los frameworks invierten el control: no es el código del desarrollador el que está a cargo, sino que el framework hace algunas invocaciones basado en ciertos estímulos.

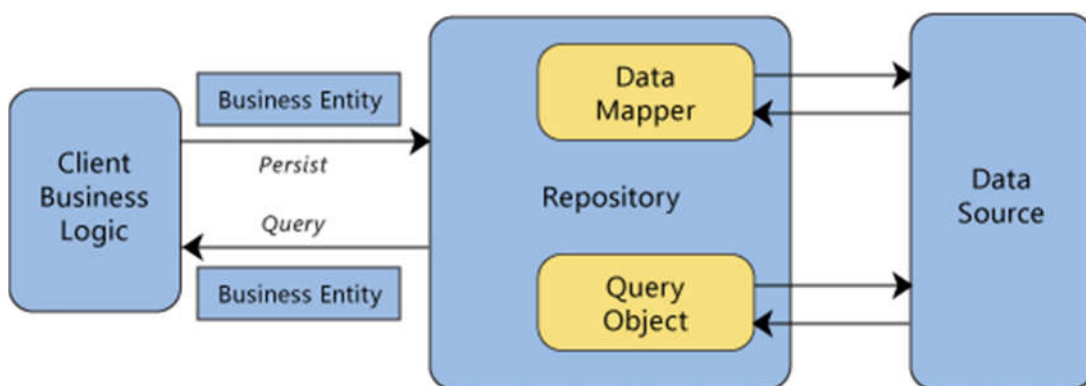
## Inyección de dependencia

Es un patrón de diseño orientado a objetos que consiste en inyectar comportamientos a componentes. Esto no es más que extraer responsabilidades a un componente para delegarlas en otro, estableciendo un mecanismo a través del cual el nuevo componente pueda ser cambiado en tiempo de ejecución.

Como nombramos anteriormente, esto es una consecuencia y una característica de los contenedores de inversión de control.

## Repositorio

El patrón repositorio está íntimamente relacionado con el acceso a datos y nos permite tener una abstracción de la implementación de acceso a datos en nuestras aplicaciones, de modo que nuestra lógica de negocio no conozca ni esté acoplada a la fuente de datos. En pocas palabras esto quiere decir que el repositorio actúa como un intermediario entre nuestra lógica de negocio y nuestra lógica de acceso a datos para que se centralice en un solo punto, y de esta forma se logre evitar redundancia de código.

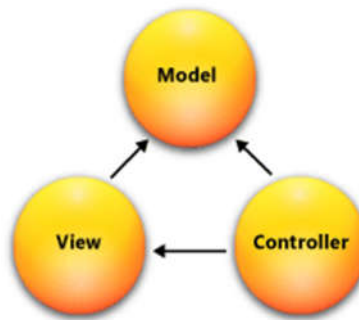


## ASP.NET MVC

El patrón de arquitectura Modelo-Vista-Controlador también conocido como por sus siglas MVC propone la construcción de tres componentes o capas de software. Con ellas se busca separar los

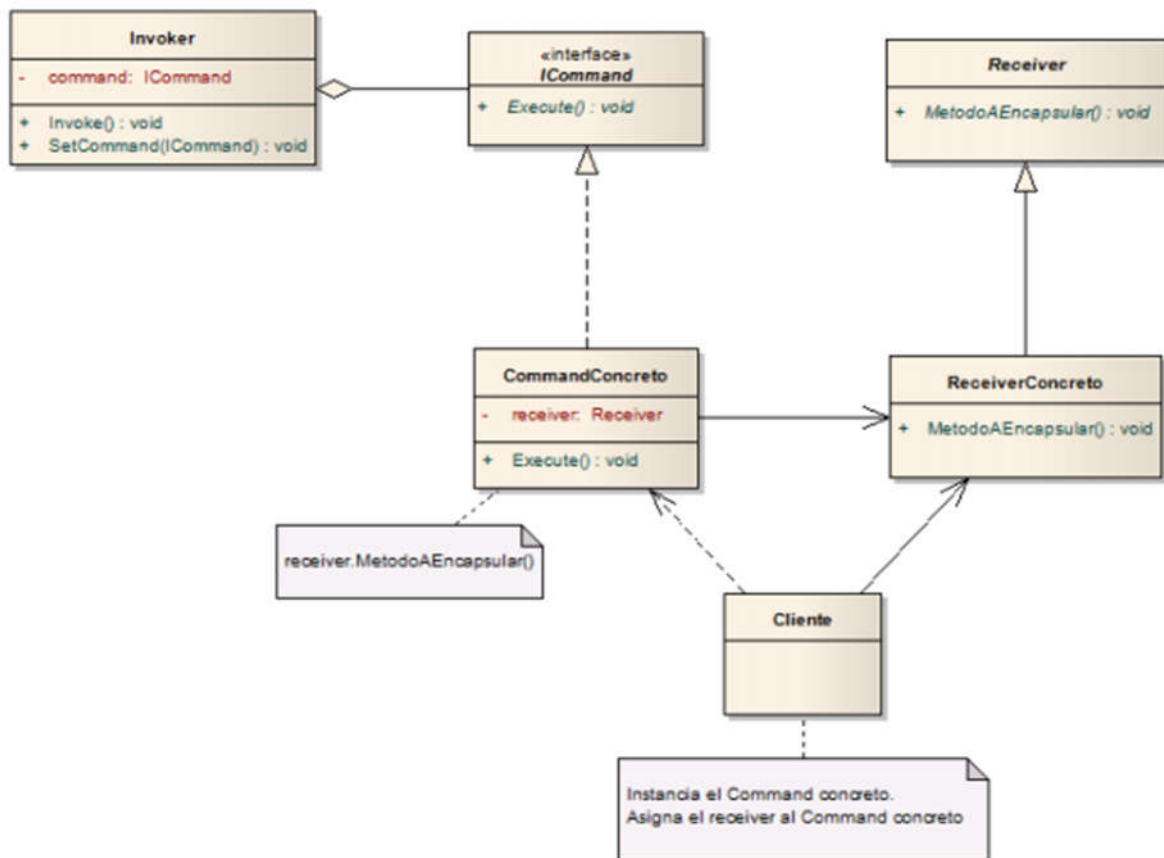
datos y la lógica de negocio de una aplicación con la interfaz de usuario y el módulo encargado de gestionar los eventos y las comunicaciones.

Esto permite diferenciar los conceptos facilitando el desarrollo y el mantenimiento de aplicaciones con la posibilidad de reutilizar módulos o bloques de código.



### Command

Hace referencia a un patrón de comportamiento. El patrón command tiene como objetivo encapsular la invocación de un método de modo que puedan parametrizarse otros objetos con distintas peticiones o colas de peticiones y proporcionar soporte para realizar operaciones que puedan deshacerse.



Si nos centramos en el diagrama de clases del diseño del patrón, podremos identificar los siguientes actores:

- **ICommand**: interfaz que expone el método genérico (`execute()`).
- **CommandGenerico**: implementa `ICommand` y posee una referencia al objeto cuyo método `execute()` tendrá que encapsular. Este objeto recibe el nombre de `Receiver`.
- **Receiver**: como acabamos de decir, es el objeto que implementa la funcionalidad real. Alguno de sus métodos será encapsulado por `ICommand.execute()`.
- **Invoker**: clase encargada de invocar el método `ICommand.execute()`. Posee una referencia (o varias) a `ICommand`, y su método `SetCommand` le permite cambiar su funcionalidad en tiempo de ejecución. Ofrecerá también un método que invoque el método `ICommand.Execute()` que, a su vez, invocará `Receiver.MetodoAEncapsular()`.

## Bibliografía

- GUÍA DE LOS FUNDAMENTOS PARA LA DIRECCIÓN DE PROYECTOS (Guía del PMBOK®) Quinta Edición
- <http://www.ices.com.ar>
- <http://www.integrarsi.com.ar>
- <https://nic.ar>
- <http://hipertextual.com/2014/12/educacion-y-tecnologia-no-son-enemigas>
- <http://la-pasion-inutil.blogspot.com.ar/2009/06/jose-ortega-y-gasset-meditacion-de-la.html>
- <http://www.pewglobal.org/2015/03/19/internet-seen-as-positive-influence-on-education-but-negative-influence-on-morality-in-emerging-and-developing-nations/>
- <http://www.oficinasinpapeles.es/la-oficina-sin-papeles>
- <http://sharp-architecture.readthedocs.org/en/latest/#>
- <https://danielggarcia.wordpress.com/2014/04/28/patrones-de-comportamiento-ii-patron-command/>
- <https://jquery.com>
- <https://github.com/castleproject/Windsor/blob/master/docs/README.md>
- <http://nhibernate.info/>
- <http://www.fluentnhibernate.org/https://github.com/castleproject/Windsor/blob/master/docs/ioc.md>
- <https://github.com/castleproject/Windsor/blob/master/docs/ioc.md>
- <http://www.fluentsecurity.net>
- <https://blog.mariusschulz.com/2011/12/05/securing-an-aspnet-mvc-application-using-fluentsecurity>
- <http://getbootstrap.com>
- <https://endjin.com/opensource>