

# Escuela Colombiana de Ingeniería

## Arquitecturas de Software – ARSW

### Proyecto de curso

### 2020 - 2



Definir, diseñar e implementar una aplicación Web, la cual debe cumplir con las características expuestas a lo largo de este documento. El proyecto se desarrollará a lo largo del semestre, para ello se hará uso de la metodología Scrum a través de la implementación de 3 Sprints de trabajo. Se espera que el estudiante aplique los conocimientos aprendidos en el curso actual así como en los cursos previos.

### Condiciones

#### Sobre el grupo de trabajo:

El grupo de trabajo debe estar conformado por máximo 3 estudiantes, los grupos no podrán disolverse ni modificarse a lo largo del semestre.

- Si un integrante cancela la materia, el grupo se verá reducido, pero el alcance del proyecto y los criterios de calificación no serán alterados.
- Si un grupo de trabajo presenta problemas y desea disolverse, cada integrante trabajará por separado. Una vez más el alcance del proyecto y los criterios de calificación no serán alterados.

#### Sobre la propuesta del proyecto:

La propuesta del proyecto es 100% responsabilidad del equipo de trabajo, los requisitos funcionales de la aplicación web deben contener mínimo las siguientes características:

- Sistema de colaboración en tiempo real. Para hacerse a una idea de algunas plataformas de colaboración en tiempo real, se puede revisar los siguientes artículos
  - <http://edtechreview.in/news/825-20-options-for-real-time-collaboration-tools>
  - <http://www.creativebloq.com/design/online-collaboration-tools-912855>
- Orientado a múltiples usuarios concurrentes. Esto implica que múltiples usuarios podrán interactuar con la plataforma al mismo tiempo en tareas completamente diferentes.

Aunque la idea es que la propuesta sea un producto original, en caso de ser similar a otros existentes se debe hacer referencia a los mismos en el documento de la propuesta, y se debe indicar claramente cuál es el **valor agregado** que se está proponiendo. Por ejemplo, se pueden tomar modelos de aplicación (o juegos) normalmente pensados para una sola persona, y plantear un uso alternativo (e interesante) en el que se involucren varias personas simultáneamente.

Muchas aplicaciones pueden considerarse de colaboración, pero se debe tener especial cuidado en el requisito de que sea **en tiempo real**. Esto significa que dicha colaboración debe poderse hacer de forma asíncrona y concurrente (por ejemplo, un foro permite colaboración, pero **NO** funciona en tiempo real).

Independientemente de que la aplicación propuesta sea enfocada a la colaboración en tiempo real, se deben considerar historias de usuario No colaborativas -pero imprescindibles-, tales como registros de usuarios, consultas de estadísticas, reportes, etc.

## Primer Sprint (Tercio 1)

### Objetivo

Definir el proyecto.

### Entregable

La propuesta del proyecto se documentará dentro de la wiki de un repositorio de GitHub que ustedes deben crear, y, donde posteriormente se harán los aportes de código en la sección de implementación: La definición debe contener las siguientes secciones:

- Título (nombre del producto)
- Integrantes
- Resumen
- Descripción (antecedentes, problema que se resuelve, etc.). Incluir diagramas o imágenes de apoyo.
- Historias de usuario.

Las historias de usuario deben cumplir con los principios INVEST: <http://agileforall.com/new-to-agile-invest-in-good-user-stories/> . Las mismas deben incluir (además del quien, como y para qué), criterios de aceptación concretos. Ejemplos:

- [http://nomad8.com/acceptance\\_criteria/](http://nomad8.com/acceptance_criteria/)
- <https://trello.com/b/T6lpA2gG>

### Criterios de evaluación

- Originalidad.
- Completitud (se considerará incompleto aquello que no permita una clara comprensión de lo que se propone hacer).
- Claridad (redacción) y ortografía.
- Calidad de las historias de usuario (criterios INVEST).
- Desempeño en la presentación final (Habilidades blandas).

## Segundo Sprint (Tercio 2)

### Objetivo

Avance funcional.

### Entregable

Enlace del repositorio de GitHub con el avance funcional esperado según la definición del proyecto.

### Criterios de evaluación

- Completitud del alcance funcional. (Desarrollo de servicios REST, Frontend y almacenamiento en base de datos).
- Completitud funcional de los requerimientos de colaboración en tiempo real.
- Integración continua.
- Desempeño en la presentación final (Habilidades blandas).

## Tercer Sprint (Tercio 3)

### Objetivo

Finalización de la funcionalidad principal y evaluación de atributos no funcionales.

### Entregable

Enlace del repositorio de GitHub con el avance funcional esperado según la definición del proyecto.

### Criterios de evaluación

- Completitud del alcance funcional. (Desarrollo de servicios REST, Frontend y almacenamiento en base de datos).
- Completitud funcional de los requerimientos de colaboración en tiempo real.
- Integración continua.
- Experimento de uso de la aplicación por parte de todo el curso.
- Evidencia del cumplimiento de los atributos no funcionales.
- Despliegue en proveedores de Cloud Computing (Heroku, AWS, Azure)
- Desempeño en la presentación final (Habilidades blandas).