



## PROYECTO SEGUNDO PARCIAL

### OBJETIVO:

Reforzar los conocimientos adquiridos sobre el proceso de gestión de la configuración del software, mediante el desarrollo colaborativo de una aplicación, aplicando buenas prácticas de control de versiones y control de cambios.

### ACTIVIDADES A DESARROLLAR:

#### 1. DESARROLLO DEL PROYECTO

Cada grupo simulará el desarrollo de una aplicación (planteada en clase) siguiendo buenas prácticas de control de versiones y control de cambio:

- El proyecto consiste en desarrollar aplicación, utilizando herramientas de control de versión y de cambio en el lenguaje de su preferencia.
- Es obligatorio el uso de un sistema de control de versiones (**Git**), en el que se almacene tanto la documentación como el código fuente. Cada integrante del equipo debe mantener una copia local del repositorio y participar activamente en su desarrollo.
- El trabajo debe evidenciar la colaboración entre los distintos miembros del equipo, mediante el **uso de ramas, commits, pull requests, issues** y demás buenas prácticas en entornos colaborativos.
- Se debe implementar un flujo de trabajo definido para el control de versiones. Se sugiere el uso del flujo **Gitflow**, aunque se puede optar por otro según el criterio del equipo.
- Integrar el proyecto en las siguientes herramientas: **Git y GitHub** para el control de versiones y **Jira Service Management** para la gestión del control de cambios.
- Simular un proceso completo de **control de cambios**, mediante la generación y asignación de peticiones de cambio entre los miembros del equipo. El equipo debe conformar un **comité de control de cambio**, integrado por 1 o 2 miembros, dependiendo del tamaño del grupo.





# UNIVERSIDAD TÉCNICA DE AMBATO

## FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL



### CARRERA DE SOFTWARE MANEJO Y CONFIGURACIÓN DE SOFTWARE

- Generar reportes en donde se verifique el trabajo realizado conforme se hayan realizado las peticiones de cambio para simular el proceso de auditoria en gestión de la configuración (Dashboard en Jira).

#### 2. INFORME (2 PUNTOS)

- El informe debe seguir el **formato de deberes** que se encuentra en la plataforma.

#### 3. PRESENTACIÓN DEL PROYECTO (2 PUNTOS)

- Exposición oral grupal de **20 a 30 minutos**.
- Debe incluir:
  - Explicación y evidencias de trabajo del equipo
  - Demostración del flujo de trabajo en Git
  - Flujo de trabajo de control de cambios
  - Explicación de la aplicación desarrollada

---

#### CASO PROPUESTO:

Aplicación para la gestión de cursos/eventos académicos en la Facultad de Ingeniería en Sistemas, Electrónica e Industrial.

#### REQUERIMIENTOS DEL SISTEMA:

Se deberá tomar en cuenta los siguientes **requerimientos, los que se explique en clase** y los que se vayan planteando en las revisiones periódicas en clase:

- La aplicación web/escritorio se utilizará para la administración de la información (CRUD de cursos/eventos, organizadores, participantes, etc.) en donde deben existir varios roles (Ej. usuarios administradores, usuarios que utilizarán el servicio, etc.); y para la inscripción de eventos/cursos. (EXPLICACIÓN EN CLASE)





# UNIVERSIDAD TÉCNICA DE AMBATO

## FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL



### CARRERA DE SOFTWARE MANEJO Y CONFIGURACIÓN DE SOFTWARE

Las **reglas del negocio** que se debe tomar en cuenta al desarrollar la aplicación son las siguientes:

- Existen diferentes tipos de eventos académicos organizados por la facultad. (Ej. Cursos, Congresos, Webinars, Conferencias, Socializaciones, etc.)
- Cada evento académico puede ser asignado a una carrera en específico o a todas las carreras. También existen eventos a público en general. De la misma manera puede ser gratis o pagado.
- Analizar los atributos necesarios que son importantes en un evento académico y dependiendo del tipo de evento. Por ejemplo, un curso tiene: número de horas, nota de aprobación, área o categoría, quien organiza, etc. Y los demás eventos no tienen una nota de aprobación, pero si un registro de asistencia.
- Los eventos académicos tienen un registro de asistencia y para el caso de cursos, adicional al registro de asistencia, se tiene un registro de notas (solo nota final).
- Cada evento académico puede requerir requerimientos para su inscripción, estos requerimientos deben ser ingresados al momento de la inscripción o verificar directamente del perfil de usuario.
- El sistema debe generar varios reportes: orden de pago, reporte de asistencia y notas por evento, certificado de aprobación o asistencia, etc.

El **proceso** que debe seguir el sistema es el siguiente:

- Mostrar información de los eventos y permitir búsqueda avanzada de los mismos, por ejemplo: por nombre, por categoría, por carrera, por número de horas, por tipo de evento, etc.
- Inscripción al curso seleccionado (verificar requerimientos y que la información con el perfil de usuario este completa).
- Generar orden de pago dependiendo de la forma de pago. Al menos implementar pago mediante deposito o transferencia.
- Si seleccionó la forma de pago mediante deposito o transferencia debe permitir subir una fotografía con el comprobante el cual debe ser aprobado por el usuario responsable (admin). De otra manera se finalizará el proceso de inscripción y se debe generar un comprobante de inscripción.





**UNIVERSIDAD TÉCNICA DE AMBATO**  
**FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA**  
**E INDUSTRIAL**



**CARRERA DE SOFTWARE**  
**MANEJO Y CONFIGURACIÓN DE SOFTWARE**

- Cuando finalice el evento se debe generar un certificado de aprobación o asistencia y el reporte final en donde consta los estudiantes con sus notas y asistencia.

