



# Pauta evaluación de los proyectos.

© desafiolatam

# Información Importante

---

- El proyecto final debe ser individual, a menos que el profesor evalúe que el tamaño y la dificultad del proyecto amerite una excepción.
- Fallar en presentar el día de avances conlleva a la pérdida de los puntos asociado. En caso de no poder asistir a una de las revisiones coordinar con el profesor de antemano para fijar una nueva fecha de revisión.
- Es requisito obligatorio presentar proyecto final para aprobar Bootcamp.
- Es requisito obligatorio para la aprobación el paso a producción del sistema para la presentación final. Esto será evaluado durante las revisiones.
- Para lograr aprobar toda revisión debe tener al menos el **55%** del puntaje total (de las 4 Revisiones y Pauta Final)
- Desafío Latam y el profesor se reservan el derecho a modificar la pauta de evaluación en cualquier momento.

## Pauta evaluación:

## Implementación de las historias de usuario correspondientes al proyecto:

---

Corresponde a la revisión del tablero de Trello. Se evalúa la cantidad de historias de usuario implementadas en función a las que fueron definidas en un comienzo.

- **No se cumple** (0 punto)
- **Se cumple de manera parcial** (7 puntos)
- **Se cumple en su totalidad** (15 puntos)

## Deployment Final (total 5 puntos)

---

- Paso a producción en Heroku, la aplicación debe estar completamente funcional con las características presentadas en producción. (2 puntos)
- Tener integrado un Dominio o subdominio. Es válido tener la redirección del dominio hacia Heroku (2 puntos).
- Rollbar u otro sistema que reporte automáticamente errores (1 punto)

## Integración de servicios de terceros (total 5 puntos)

---

Integrar cualquiera de los siguientes servicios

- GoogleMaps.
- PayPal.
- Amazon S3.
- Mailing (diferente al servicio de Devise).

- Login por Facebook.
- Consumo de otro servicio REST.

La integración de uno de los servicios mencionados entrega 2 puntos, la integración de dos o más entrega 5 puntos.

## Autenticación (total 5 puntos)

---

- Tener un sistema de login con Devise, manejar correctamente los strong params y las redirecciones después de ingreso y registro y salida. (5 puntos)

## Integración de un panel de control (total 5 puntos)

---

- Puedes ser con ActiveAdmin o creado manualmente, pero debe permitir ingresar y eliminar datos. (3 puntos)
- Entrega de información agrupada con gráficos, este puede ser para el administrador o para alguna sección del cliente (2 puntos)

## Modelos, migraciones y seeds (total 5 puntos)

---

- El proyecto debe tener todas las migraciones, no puede incluir cambios directos a la base de datos. (3 puntos)
- El proyecto debe incluir un archivo seed y la documentación necesaria para que otra persona pueda descargarlo y correrlo sin tener que ingresar

datos manualmente. (2 puntos)

## Front (total 9 puntos)

---

- Diseño responsive (3 puntos)
- Utilización correcta del viewport, añadir favicon, descriptions y título a cada página. (1 punto)
- Utilización correcta de al menos un plugin sin desactivar turbolinks, como por ejemplo: (2 puntos)
  - Lightbox
  - Backstretch
  - googlecharts
- Integración del front-end (3 puntos)
  - Puede ser:
    - Integración completa de una plantilla.
    - Crear un diseño finalizado propio.

## Ajax (total 4 puntos)

---

- Utilizar Ajax (con UJS) en al menos 1 formulario asociado a algún requerimiento (2 puntos).
- Utilizar al menos 1 llamado ajax (sin utilizar UJS) asociado a algún requerimiento (2 puntos).

## Buenas prácticas y manejo correcto de modelo MVC (6 puntos)

---

- Identación correcta del código (pueden ser tabs o espacios, pero debe ser consistente a lo largo del proyecto) (1 punto).
- Utilización correcta de rutas REST (2 puntos).
- No realizar queries dentro de las vistas (2 puntos).

## Manejo de GIT (3 puntos)

---

- Manejo de al menos dos ramas, una para desarrollo y otra maestra (1 punto).
- Todas las ramas utilizadas deben estar subidas a Github o bitbucket (1 punto).
- Integración de las nuevas carecterísticas en una rama adicional (1 punto).

## Autorización y permisos (2 puntos)

---

- Tener un sistema de permiso basado en roles con CanCanCan u otro sistema.

## Deployment parcial (1 punto)

---

- Paso a producción de todas las tareas marcadas como finalizadas.

## Validación en los modelos (total 4 puntos)

---

- Validación de unicidad. 1 punto
- Validación de presencia. 1 punto
- Validación de inclusión. 1 punto
- Validación numérica. 1 punto

## Seguimiento de avance en el proyecto

---

Cada semana se hará una revisión de los avances realizados, para esto se revisará el avance de las tarjetas en Trello, cada revisión que haya cumplido con los avances tiene asociado un punto.

- Revisión 1: Revisión de adelantos de proyectos (15 puntos).
- Revisión 2: Revisión de adelantos de proyectos (15 puntos).
- Revisión 3: Revisión de adelantos de proyectos (15 puntos).
- Revisión 4: Revisión de adelantos de proyectos (15 puntos).

Un buen avance consiste en marcar las tarjetas del estado **to do** a **doing** y desde **doing** a **done**, la revisión consiste en ver el avance de las tarjetas, que correspondan con las realizadas en código y archivar aquellas en done.

Cada semana se debe implementar **nuevas** historias de usuario y en base a ellas hacer cada revisión.

# Primera revisión de proyectos

---

## Historias de usuario en Trello: (total 5 puntos)

Corresponde a la revisión del tablero de Trello. El tablero debe poseer una estructura *Agile* y la totalidad de las historias de usuario deben estar presentes.

- **No se cumple** (0 punto)
- **Se cumple de manera parcial** (2 puntos)
- **Se cumple en su totalidad** (5 puntos)

## Modelo de datos (total 10 puntos)

Corresponde al modelo físico de la base de datos del proyecto para ser implementado en PostgreSQL.

- **No se cumple** (0 punto)
- **Se cumple de manera parcial** (5 puntos)
- **Se cumple en su totalidad** (10 puntos)

# Segunda revisión de proyectos

---

## Historias de usuario en Trello:

Corresponde a la revisión de los avances en la implementación de historias de usuario en tablero de Trello. Las historias acordadas en la primera revisión deben estar implementadas en el proyecto.



- **No se cumple** (0 punto)
- **Se cumple de manera parcial** (2 puntos)
- **Se cumple en su totalidad** (5 puntos)

## Buenas prácticas y manejo correcto de modelo MVC (6 puntos)

- Identación correcta del código (pueden ser tabs o espacios, pero debe ser consistente a lo largo del proyecto) (2 puntos).
- Utilización correcta de rutas REST (2 puntos).
- No realizar queries dentro de las vistas (2 puntos).

## Manejo de GIT (4 puntos)

- Manejo de al menos dos ramas, una para desarrollo y otra maestra (1 punto).
- Integración de las nuevas carecterísticas en una rama adicional (2 puntos).
- Todas las ramas utilizadas deben estar subidas a Github o bitbucket (1 punto).

# Tercera revisión de proyectos

---

## Historias de usuario en Trello:

Corresponde a la revisión de los avances en la implementación de historias de usuario en tablero de Trello. Las historias acordadas en la primera revisión deben estar implementadas en el proyecto.

- **No se cumple** (0 punto)

- **Se cumple de manera parcial** (2 puntos)
- **Se cumple en su totalidad** (5 puntos)

## Buenas prácticas y manejo correcto de modelo MVC (6 puntos)

- Identación correcta del código (pueden ser tabs o espacios, pero debe ser consistente a lo largo del proyecto) (2 puntos).
- Utilización correcta de rutas REST (2 puntos).
- No realizar queries dentro de las vistas (2 puntos).

## Manejo de GIT (4 puntos)

- Manejo de al menos dos ramas, una para desarrollo y otra maestra (1 punto).
- Integración de las nuevas carecterísticas en una rama adicional (2 puntos).
- Todas las ramas utilizadas deben estar subidas a Github o bitbucket (1 punto).

# Cuarta y última revisión de proyectos

---

## Historias de usuario en Trello:

Corresponde a la revisión de los avances en la implementación de historias de usuario en tablero de Trello. Las historias acordadas en la primera revisión deben estar implementadas en el proyecto.

- **No se cumple** (0 punto)
- **Se cumple de manera parcial** (2 puntos)

- **Se cumple en su totalidad** (5 puntos)

## Buenas prácticas y manejo correcto de modelo MVC (6 puntos)

- Identación correcta del código (pueden ser tabs o espacios, pero debe ser consistente a lo largo del proyecto) (2 puntos).
- Utilización correcta de rutas REST (2 puntos).
- No realizar queries dentro de las vistas (2 puntos).

## Manejo de GIT (4 puntos)

- Manejo de al menos dos ramas, una para desarrollo y otra maestra (1 punto).
- Integración de las nuevas carecterísticas en una rama adicional (2 puntos).
- Todas las ramas utilizadas deben estar subidas a Github o bitbucket (1 punto).

