

Consejos para la realización de esta evaluación:

1. Leer la prueba completa antes de comenzar.
- Copia de código fuente será castigada con nota 0.
 - Concentrarse en incluir la mayor cantidad de ítems funcionales posibles, comenzar por los más fáciles.
 - **Se puede utilizar:** Google, StackOverflow, Blogs, material previo del curso, etc...

Prueba 1 de Rails

El objetivo de esta actividad es evaluar la implementación del patron de diseño MVC, el manejo de arquitectura REST, la creación de formularios y la validación de modelos.

En esta evaluación desarrollarás una **ToDo List app** que permita ingresar, leer, editar y eliminar ToDos. Cada ToDo debe poseer además la opción de ser marcado como completado.

Buena suerte !!

Parte I: CRUD

- Crear un nuevo proyecto en Rails 5.1 llamado **todo_app**.
- Crear el controller *todos* (se debe crear vacío, los métodos se irán agregando manualmente a medida que avanzamos)
- Crear el model *Todo* con los campos *description* (string) y *completed* (boolean).
- Revisar y correr la migración.
- Agregar el **CDN de jQuery** al *layout*.
- Agregar el **CDN de Bootstrap** al *layout*.

Index:

- Crear la ruta **'/todos'** que apunte al método **todos#index** con el verbo **get**.
 - Hint: Recordar las rutas deben obedecer a la arquitectura REST (prefixs y verbs).
- En el controller *todos* crear el método **index**. En este método se debe declarar una variable de instancia **@todos** que almacenará la colección de *ToDo* existentes.
- Crear la vista **index.html.erb**.
 - En la vista se debe listar cada *ToDo* utilizando la clase **well** de Bootstrap.

New:

- Crear la ruta **'todos/new'** con el verbo **get**.
- En el controller *todos* crear el método **new**. En este método se debe declarar una variable de instancia **@todo** que almacenará una nueva instancia de *ToDo*.
- Crear la vista **new.html.erb**.
 - La vista debe contener el formulario para la creación de un nuevo **ToDo**.
 - El formulario se debe generar utilizando **form_with** asociado a **@todo**. (No olvidar asignar el atributo *local:true*).
- En la vista *Index*: agregar un link a **Crear ToDo**.

Create:

- Crear la ruta **'/todos'** que apunte al método **todos#create** con el verbo **post**.
- En el controller *todos* crear el método **create**. Este método debe crear un nuevo *ToDo* a partir de los parámetros recibidos del formulario bajo el scope de *ToDo*

(params[:todo]).

- Filtrar los parámetros recibidos bajo el scope de *ToDo*s en un método **privado** llamado **todo_params** para poder utilizar asignación masiva.
- El método **create** después de almacenar el **todo** debe redireccionar al *index* de *ToDo*s.

Show:

- Crear la ruta **'/todos/:id'** que apunte al método **todos#show** con el verbo **get** y el prefijo **todo**.
- En el controller *todos* crear el método **show**. Este método debe buscar un *ToDo* en base al parámetro *'id'*. (**Todo.find**) y asignarlo a la variable **@todo**
- Crear la vista **show.html.erb**.
 - La vista debe mostrar la descripción del *ToDo* y si se encuentra completado o no.
- En la vista *Index*: agregar un link a **Show** a cada *ToDo* listado.

Edit:

- Crear la ruta **'/todos/:id/edit'** que apunte al método **todos#edit** con el verbo **get**.
- En el controller *todos* crear el método **edit**. Este método debe buscar un *ToDo* en base al parámetro *'id'*. (**Todo.find**) y asignarlo a la variable **@todo**.
- Crear la vista **edit.html.erb**.
 - La vista debe contener el formulario para la edición del **ToDo**.
 - El formulario se debe generar utilizando **form_with** asociado a **@todo**. (No olvidar asignar el atributo *local:true* y corroborar que el formulario se haya creado con el método **patch**).

- En la vista *Index*: agregar un link a **Edit** a cada *ToDo* listado.

Update:

- Crear la ruta `'/todos/:id'` que apunte al método `todos#update` con el verbo **patch**.
- En el controller *todos* crear el método **update**. Este método debe buscar un *ToDo* en base al parámetro *'id'*. (**Todo.find**) y asignarlo a la variable **@todo**.
- Filtrar los parámetros recibidos bajo el scope de *ToDo*s en el mismo método privado **todo_params** para poder utilizar asignación masiva.
- El método **update** después de actualizar el **todo** debe redireccionar al *index* de *ToDo*s.

Destroy:

- Crear la ruta `'/todos/:id'` que apunte al método `todos#destroy` con el verbo **delete**.
- En el controller *todos* crear el método **destroy**. Este método debe buscar un *ToDo* en base al parámetro *'id'*. (**Todo.find**)
- Eliminar el *ToDo* y luego redireccionar al *index* de *ToDo*s.
- En la vista *Index*: agregar un link a **Destroy** (con el método *delete*) a cada *ToDo* listado.

Marcar como completado:

- Crear la ruta `'/todos/:id/complete'` que apunte al método `todos#complete` con el verbo **get**.

- En el controller *todos* crear el método **complete**. Este método debe buscar un *ToDo* en base al parámetro '*id*' (**Todo.find**) y asignarlo a la variable **@todo**. Finalmente modificar el atributo **completed = true**, guardar el registro y redirigir al *Index*.
- En la vista *index*, agregar un botón a cada *ToDo* que permita marcar como completado utilizando la ruta correspondiente.

Parte II: Landing nuevo y refactoring de vistas parciales.

- Crear la ruta **/todos/list** con el verbo **get**.
- Crear un método **list** en controller. Este método debe crear una variable de instancia **@todos** que almacene la colección de *ToDo*s existentes.
- Crear una vista **list.html.erb**.
 - Esta vista debe -utilizando la grilla de bootstrap- separar la pantalla en dos secciones de 6 columnas cada una.
 - A la izquierda se deben listar los *ToDo*s **NO** completados.
 - A la derecha se deben listar los *ToDo*s **completados**.
- Crear una **vista parcial** que contenga un **navbar fixed** de Bootstrap.
 - El navbar debe contener un link a *Index*, un link a *List* y un link al formulario de creación de nuevo *ToDo*.
 - Cargar la vista parcial en el *layout*.
- Refactorizar los formularios utilizados en las vistas de creación y edición de un *ToDo*. Se debe **crear una vista parcial que contenga el formulario**, la cual debe ser implementada en ambas vistas.

Parte III: Validaciones

- Se debe validar que la descripción debe estar presente al crear un nuevo *ToDo*.