



Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ciencia de la Computación
IIC2143 - Ingeniería de Software 2025-1

Tarea Introducción a **Rails**

Objetivos

- Aprender a utilizar el *framework* **Ruby on Rails**.
- Entender e implementar el concepto de *API*.

Fecha de Entrega

28 de Marzo del 2025, 23:59

Books API

En esta tarea usted tendrá que crear una API para registrar, leer, actualizar, y borrar libros, autores y reseñas. Para simplificar el modelo relacional de la tarea se harán algunos supuestos. En primer lugar, un libro solo puede estar escrito por una autora o autor. En segundo lugar, una reseña solo puede hablar sobre un libro, es decir, no hará referencias a otros libros.

Tips:

- Realizar la tarea con tiempo.
- Manejar los DELETE que afectan a los autores mediante eliminado en cascada.
- Cuando se indica que cierto modelo debe tener cierto atributo, se espera que su solución verifique que esto ocurra y no permita la presencia de vacíos o nulos.
- El código base de la tarea incluirá los tests que permiten ver cuantos de los objetivos a logrado, si existen problemas en la generación de controladores y modelos por ya contar con los tests, pueden utilizar "-skip-test-framework" para no sobrescribir los tests.
- Para ejecutar la batería de tests con la que podrán testear sus tareas deberán utilizar el comando "rails test" en su consola.

Modelos mínimos

En esta sección se detallarán los modelos mínimos que debe manejar la *API*, junto a sus atributos. Todo lo señalado a continuación, deberá ser implementado utilizando los **mismos nombres**, ya que de esta forma los atributos en los JSON de respuesta tendrán los nombres que se pide para cada endpoint. En caso de no utilizar los nombres mencionados, la corrección fallará, ya que utilizamos automatización de pruebas para revisar los modelos.

Model Author

- **Atributos:**
 - name de tipo string
 - average_rating de tipo float
 - main_genre de tipo string
 - biography de tipo string
- **Asociaciones:** Un autor puede haber escrito varios libros.
- **Consideraciones:**
 - Para simplificar los sistemas de búsqueda, los nombres de los autores tendrán un _ para simbolizar los espacios.
 - El atributo average_rating sera un float que ira del 1.0 al 10.0.
 - Es importante recalcar que un autor debe tener un nombre, biografía y calificación promedio. Puede no tener un género principal.

Model Book

- **Atributos:**
 - title de tipo string
 - description de tipo string
 - genre de tipo string
 - average_rating de tipo float
 - number_of_reviews de tipo integer
 - author_id de tipo references
- **Asociaciones:** Un libro pertenece exactamente a solo un autor.
- **Consideraciones:**
 - Es importante recalcar que un libro debe tener un titulo, descripcion, género, calificación promedio, número de reseñas e id de autor.

Model Review

- **Atributos:**

- rating de tipo float
- comment de tipo string
- review_author de tipo string
- book_id de tipo references

- **Asociaciones:** Cada reseña puede hacer referencia a un solo libro.

- **Consideraciones:**

- Es importante recalcar que una reseña debe tener un puntaje, comentario, nombre del autor de la reseña e id del libro al que hace referencia.

Rutas solicitadas

Las rutas solicitadas se dividen en 3 partes, donde cada una contiene los endpoints respectivos. Cada parte se evaluará usando tests que verifiquen el funcionamiento correcto de los distintos endpoints. El puntaje final será calculado en base a la proporción de pruebas aprobadas en cada parte.

Parte 1 (Easy): 2 pts

1. GET /authors

Debe retornar todos los autores que se encuentren en la base de datos.

Ejemplo response body:

```
1 [
2   {
3     "id": 2,
4     "name": "George Orwell",
5     "average_rating": 7.6,
6     "main_genre": "dystopic ficiton",
7     "biography": "Eric Arthur Blair was an English novelist, poet, essayist,
8       journalist and critic who wrote under the pen name of George Orwell.
9       His work is characterised by lucid prose, social criticism, opposition
10      to all totalitarianism (both authoritarian communism and fascism),
11      and support of democratic socialism",
12     "created_at": "2025-02-24T18:41:53.427Z",
13     "updated_at": "2025-02-24T18:41:53.427Z"
14   },
15   {
16     "id": 3,
17     "name": "JK Rowling",
18     "average_rating": 5.9,
19     "main_genre": "fantasy",
20     "biography": "Joanne Rowling, known by her pen name J. K. Rowling, is a
21       British author and philanthropist. She is the author of Harry Potter,
22       a seven-volume fantasy novel series published from 1997 to 2007. ",
23     "created_at": "2025-02-24T18:43:39.430Z",
24     "updated_at": "2025-02-24T18:43:39.430Z"
25   }
26 ]
```

2. POST /authors

Deberá crear un nuevo autor en base a los atributos enviados.

Ejemplo request body:

```
1 {
2   "author": {
3     "name": "George Orwell",
4     "average_rating": 7.6,
5     "main_genre": "dystopic ficiton",
```

```

6     "biography": "Eric Arthur Blair was an English novelist, poet, essayist,
       journalist and critic who wrote under the pen name of George Orwell.
       His work is characterised by lucid prose, social criticism, opposition
       to all totalitarianism (both authoritarian communism and fascism),
       and support of democratic socialism"
7   }
8 }

```

Ejemplo response body:

```

1 {
2   "id": 2,
3   "name": "George_Orwell",
4   "average_rating": 7.6,
5   "main_genre": "dystopic ficiton",
6   "biography": "Eric Arthur Blair was an English novelist, poet, essayist,
       journalist and critic who wrote under the pen name of George Orwell. His
       work is characterised by lucid prose, social criticism, opposition to all
       totalitarianism (both authoritarian communism and fascism), and support of
       democratic socialism",
7   "created_at": "2025-02-24T18:41:53.427Z",
8   "updated_at": "2025-02-24T18:41:53.427Z"
9 }

```

3. GET /authors/:id

Debe retornar al autor cuya id se entrega en la url.

Ejemplo url: <http://localhost:3000/authors/5>

Ejemplo response body:

```

1 {
2   "id": 2,
3   "name": "George_Orwell",
4   "average_rating": 7.6,
5   "main_genre": "dystopic ficiton",
6   "biography": "Eric Arthur Blair (25 June 1903      21 January 1950) was an
       English novelist, poet, essayist, journalist and critic who wrote under
       the pen name of George Orwell. His work is characterised by lucid prose,
       social criticism, opposition to all totalitarianism (both authoritarian
       communism and fascism), and support of democratic socialism",
7   "created_at": "2025-02-24T18:41:53.427Z",
8   "updated_at": "2025-02-24T18:41:53.427Z"
9 }

```

4. DELETE /authors/:id

Deberá eliminar el autor cuya id coincida con la entregada en la url.

Ejemplo url: <http://localhost:3000/authors/5>

Ejemplo response body:

```
1 {}
```

5. **DELETE** /authors

Deberá eliminar todos los autores.

NOTA: Este será el primer endpoint que se ejecutara en todos los tests.

Ejemplo url: <http://localhost:3000/authors>

Ejemplo response body:

```
1 []
```

Parte 2 (Medium): 3 ptos

1. **POST** /books/:author_id

Deberá crear un nuevo libro cuyo autor sea el correspondiente al id entregado :author_id.

Ejemplo url: <http://localhost:3000/books/2>

Ejemplo request body:

```
1 {
2   "book": {
3     "title": "1984",
4     "description": "totalitarian society ruled by Big Brother. It follows
      Winston Smith, a man who secretly despises the oppressive regime and
      seeks truth and freedom. As he rebels against constant surveillance
      and propaganda, he discovers the terrifying extent of government
      control. The novel explores themes of censorship, manipulation, and
      the dangers of absolute power.",
5     "genre": "dystopic ficiton",
6     "average_rating": 9.5,
7     "number_of_reviews": 2
8   }
9 }
```

Ejemplo response body:

```
1 {
2   "id": 1,
3   "title": "1984",
4   "description": "totalitarian society ruled by Big Brother. It follows Winston
      Smith, a man who secretly despises the oppressive regime and seeks truth
      and freedom. As he rebels against constant surveillance and propaganda, he
      discovers the terrifying extent of government control. The novel explores
      themes of censorship, manipulation, and the dangers of absolute power.",
5   "genre": "dystopic ficiton",
6   "average_rating": 9.5,
7   "number_of_reviews": 2,
8   "author_id": 2,
9   "created_at": "2025-02-24T19:01:01.650Z",
10  "updated_at": "2025-02-24T19:01:01.650Z"
11 }
```

2. **DELETE** /books

Deberá eliminar todos los libros.

NOTA: Este será el segundo endpoint que se ejecutara en todos los tests de tipo medium y hard.

Ejemplo url: <http://localhost:3000/books>

Ejemplo response body:

```
1 []
```

3. **GET** /books/:book_id

Deberá entregar el libro de id book_id.

Ejemplo url: <http://localhost:3000/books/1>

Ejemplo response body:

```
1 {
2   "id": 1,
3   "title": "1984",
4   "description": "totalitarian society ruled by Big Brother. It follows Winston
      Smith, a man who secretly despises the oppressive regime and seeks truth
      and freedom. As he rebels against constant surveillance and propaganda, he
      discovers the terrifying extent of government control. The novel explores
      themes of censorship, manipulation, and the dangers of absolute power.",
5   "genre": "dystopic ficiton",
6   "average_rating": 9.5,
7   "number_of_reviews": 2,
8   "author_id": 1,
9   "created_at": "2025-02-24T19:45:07.027Z",
10  "updated_at": "2025-02-24T19:45:07.027Z"
11 }
```

4. **GET** /books/:book_id/reviews

Deberá entregar las reseñas del libro solicitado.

Ejemplo url: <http://localhost:3000/books/1/reviews>

Ejemplo response body:

```
1 [
2   {
3     "id": 1,
4     "rating": 9.7,
5     "comment": "Amazing, great to re-read every once in a while",
6     "review_author": "Felipe M",
7     "book_id": 1,
8     "created_at": "2025-02-24T19:57:01.660Z",
9     "updated_at": "2025-02-24T19:57:01.660Z"
10  },
11  {
12    "id": 2,
13    "rating": 9.0,
14    "comment": "Great book!",
15    "review_author": "JTV",
16    "book_id": 1,
17    "created_at": "2025-02-24T19:57:04.828Z",
18    "updated_at": "2025-02-24T19:57:04.828Z"
19  }
20 ]
```


5. `POST /reviews`

Deberá crear una nueva reseña, notar que esta debe estar asociado a el libro indicado en el cuerpo de la request.

Ejemplo request body:

```
1 {
2     "review": {
3         "review_author": "Felipe M",
4         "comment": "Amazing, great to re-read every once in a while",
5         "rating": 9.7,
6         "book_id": 1
7     }
8 }
```

Ejemplo response body:

```
1 {
2     "id": 1,
3     "rating": 9.7,
4     "comment": "Amazing, great to re-read every once in a while",
5     "review_author": "Felipe M",
6     "book_id": 1,
7     "created_at": "2025-02-24T19:57:01.660Z",
8     "updated_at": "2025-02-24T19:57:01.660Z"
9 }
```

6. `PATCH /reviews/:review_id`

Deberá editar la reseña en base de los nuevos atributos entregados en el body.

Ejemplo url: `localhost:3000/reviews/4`

Ejemplo request body:

```
1 {
2     "review": {
3         "rating": 10.0
4     }
5 }
```

Ejemplo response body:

```
1 {
2     "rating": 10.0,
3     "book_id": 1,
4     "id": 1,
5     "comment": "Amazing, great to re-read every once in a while",
6     "review_author": "Felipe M",
7     "created_at": "2025-02-24T19:57:01.660Z",
8     "updated_at": "2025-02-24T20:00:07.391Z"
9 }
```

7. **GET** /reviews/:book

Deberá retornar todos las reseñas del libro cuyo nombre es el :book solicitado

Ejemplo url: localhost:3000/reviews/1984

Notar que 1984 es el libro de id 1 en este ejemplo, por lo que se retornan aquellas reseñas que hacen referencia al book 1

Ejemplo response body:

```
1  [
2    {
3      "id": 1,
4      "rating": 9.7,
5      "comment": "Amazing, great to re-read every once in a while",
6      "review_author": "Felipe M",
7      "book_id": 1,
8      "created_at": "2025-02-24T20:14:07.314Z",
9      "updated_at": "2025-02-24T20:14:07.314Z"
10   },
11   {
12     "id": 2,
13     "rating": 9.0,
14     "comment": "Great book!",
15     "review_author": "JTV",
16     "book_id": 1,
17     "created_at": "2025-02-24T20:14:07.371Z",
18     "updated_at": "2025-02-24T20:14:07.371Z"
19   }
20 ]
```

Parte 3 (Hard): 1 pto

1. **GET** /authors/:author_id/books

Deberá entregar todos los libros escritos por el autor solicitado.

```
1  [
2    {
3      "id": 2,
4      "title": "Harry Potter 1",
5      "description": "the book follows Harry Potter, a young up and coming
        wizard that must fight evils that have been waiting for him since his
        birth while balancing his relationship with his family's history and
        new friends as he is introduced into the magic world and the school of
        Hogwarts",
6      "genre": "fantasy",
7      "average_rating": 7.0,
8      "number_of_reviews": 0,
9      "author_id": 2,
10     "created_at": "2025-02-24T20:21:41.770Z",
11     "updated_at": "2025-02-24T20:21:41.770Z"
12   },
13   {
14     "id": 3,
15     "title": "Harry Potter 2",
16     "description": "the book continues the adventures of Harry Potter",
17     "genre": "fantasy",
18     "average_rating": 8.0,
19     "number_of_reviews": 2,
20     "author_id": 2,
21     "created_at": "2025-02-24T20:21:41.851Z",
22     "updated_at": "2025-02-24T20:21:41.851Z"
23   }
24 ]
```

2. **GET** /books/top/:quantity

Deberá entregar los mejores quantity libros considerando sus calificaciones.

En caso de empate, es decir, en caso de que dos juegos tengan la misma calificación promedio, será mejor el que tenga mayor número de id.

Nota: :quantity será un numero natural.

Ejemplo url: localhost:3000/books/top/2

Ejemplo response body

```
1  [
2    {
3      "id": 1,
4      "title": "1984",
```

```

5      "description": "totalitarian society ruled by Big Brother. It follows
      Winston Smith, a man who secretly despises the oppressive regime and
      seeks truth and freedom. As he rebels against constant surveillance
      and propaganda, he discovers the terrifying extent of government
      control. The novel explores themes of censorship, manipulation, and
      the dangers of absolute power.",
6      "genre": "dystopic ficiton",
7      "average_rating": 9.5,
8      "number_of_reviews": 2,
9      "author_id": 1,
10     "created_at": "2025-02-24T20:24:45.290Z",
11     "updated_at": "2025-02-24T20:24:45.290Z"
12 },
13 {
14     "id": 3,
15     "title": "Harry Potter 2",
16     "description": "the book continues the adventures of Harry Potter",
17     "genre": "fantasy",
18     "average_rating": 8.0,
19     "number_of_reviews": 2,
20     "author_id": 2,
21     "created_at": "2025-02-24T20:24:45.447Z",
22     "updated_at": "2025-02-24T20:24:45.447Z"
23 }
24 ]

```

3. GET /reviews/:id/book_author

Para la reseña del id entregado en :id, entregar el autor del libro.

Ejemplo url: http://localhost:3000/reviews/2/book_author

Ejemplo response body:

```

1 {
2   "id": 1,
3   "name": "George Orwell",
4   "average_rating": 7.6,
5   "main_genre": "dystopic ficiton",
6   "biography": "Eric Arthur Blair was an English novelist, poet, essayist,
7   journalist and critic who wrote under the pen name of George Orwell. His
8   work is characterised by lucid prose, social criticism, opposition to all
9   totalitarianism (both authoritarian communism and fascism), and support of
10  democratic socialism",
11  "created_at": "2025-02-24T20:43:16.257Z",
12  "updated_at": "2025-02-24T20:43:16.257Z"
13 }

```

4. DELETE /books/low

Deberá entregar y luego eliminar a el libro con la menor calificación.

En caso de empate, es decir, si existen dos libros con la misma calificacion promedio, será peor el que tenga menor id.

Ejemplo response body: Igual a Parte 2, Ruta 3

5. **PATCH** /books/rating_update/:book_id

Deberá actualizar el valor de average_rating que tenga el libro cuyo id sea :book_id tal que su nuevo valor sea el promedio de las calificaciones recibidas por ese libro en reseñas presentes en la base de datos.

Ejemplo url: books/rating_update/4

Ejemplo response body, suponiendo que el promedio de la calificación en las reviews recibidas es un :9.5

```
1 {
2   "average_rating": 9.5,
3   "author_id": 1,
4   "id": 1,
5   "title": "1984",
6   "description": "totalitarian society ruled by Big Brother. It follows Winston
   Smith, a man who secretly despises the oppressive regime and seeks truth
   and freedom. As he rebels against constant surveillance and propaganda, he
   discovers the terrifying extent of government control. The novel explores
   themes of censorship, manipulation, and the dangers of absolute power.",
7   "genre": "dystopic ficiton",
8   "number_of_reviews": 2,
9   "created_at": "2025-02-24T23:52:02.069Z",
10  "updated_at": "2025-02-24T23:52:03.522Z"
11 }
```

6. **PATCH** /books/reviews_num_update/:book_id

Deberá actualizar el valor de number_of_reviews que tenga el libro cuyo id sea :book_id tal que su nuevo valor sea el número de calificaciones recibidas por ese libro en reseñas presentes en la base de datos.

Ejemplo url: books/reviews_num_update/1

Ejemplo response body, suponiendo que el libro de id 1 ha recibido 2 reseñas:

```
1 {
2   "number_of_reviews": 2,
3   "author_id": 1,
4   "id": 1,
5   "title": "1984",
6   "description": "totalitarian society ruled by Big Brother. It follows Winston
   Smith, a man who secretly despises the oppressive regime and seeks truth
   and freedom. As he rebels against constant surveillance and propaganda, he
   discovers the terrifying extent of government control. The novel explores
   themes of censorship, manipulation, and the dangers of absolute power.",
7   "genre": "dystopic ficiton",
8   "average_rating": 9.5,
9   "created_at": "2025-02-24T23:32:08.925Z",
10  "updated_at": "2025-02-24T23:32:10.155Z"
11 }
```

7. **PATCH** /authors/rating_update/:author_id

Deberá actualizar el valor de average_rating que tenga el autor cuyo id sea :author_id tal que su nuevo valor sea el promedio de las calificaciones promedio de todos sus libros.

Ejemplo url: authors/rating_update/2

Ejemplo response body, suponiendo que el promedio de la calificación en los libros del autor es un 7.5:

```
1 {
2   "average_rating": 7.5,
3   "id": 2,
4   "name": "JK_Rowling",
5   "main_genre": "fantasy",
6   "biography": "Joanne Rowling (born 31 July 1965), known by her pen name J. K.
   Rowling, is a British author and philanthropist. She is the author of
   Harry Potter, a seven-volume fantasy novel series published from 1997 to 2
   007. ",
7   "created_at": "2025-02-24T23:52:01.786Z",
8   "updated_at": "2025-02-24T23:52:31.939Z"
9 }
```

8. **GET** /reviews/:review_id/reviewed_books

Deberá devolver todos los libros a los que el usuario autor de la review con id :review_id les ha dejado una reseña

Ejemplo url: reviews/1/reviewed_books

Ejemplo response body, :

```
1 [
2   {
3     "id": 1,
4     "title": "1984",
5     "description": "totalitarian society ruled by Big Brother. It follows
      Winston Smith, a man who secretly despises the oppressive regime and
      seeks truth and freedom. As he rebels against constant surveillance
      and propaganda, he discovers the terrifying extent of government
      control. The novel explores themes of censorship, manipulation, and
      the dangers of absolute power.",
6     "genre": "dystopic ficiton",
7     "average_rating": 9.5,
8     "number_of_reviews": 2,
9     "author_id": 1,
10    "created_at": "2025-02-24T23:52:02.069Z",
11    "updated_at": "2025-02-24T23:52:03.522Z"
12  },
13  {
14    "id": 2,
15    "title": "Harry Potter 1",
16    "description": "the book follows Harry Potter, a young up and coming
      wizard that must fight evils that have been waiting for him since his
```

```
        birth while balancing his relationship with his family's history and
        new friends as he is introduced into the magic world and the school of
        Hogwarts",
17     "genre": "fantasy",
18     "average_rating": 7.0,
19     "number_of_reviews": 0,
20     "author_id": 2,
21     "created_at": "2025-02-24T23:52:02.165Z",
22     "updated_at": "2025-02-24T23:52:02.165Z"
23 }
24 ]
```

Evaluación

Para evaluar que su API funciona correctamente, se realizarán consultas sobre su API donde se verificará que este entregue la respuesta y estado correctos. Cada test tendrá puntaje y dificultad asociado y este indicará si Pasa (1) o No Pasa (0), por su parte. El puntaje total corresponderá a la suma de todos los tests. Se les facilitará una batería de tests con la que podrán testear sus tareas utilizando el comando "rails test" en su consola.

Entrega

La entrega de esta tarea deberá realizarse en un repositorio de GitHub que se les facilitara, es importante que su código se encuentre en la rama *main*, de lo contrario la tarea no se revisara.

Es importante que el código de su entrega corresponda a las carpetas que conforman su proyecto de RoR sin estar contenidas dentro de otra carpeta del tipo BooksAPI. El no cumplir con este aspecto supondrá un descuento de 5 décimas a la nota obtenida por el alumno. De manera similar, la entrega de código ejecutable que presente errores o aspectos harcodeados en su archivo .env, usen una versión de ruby diferente a la del curso o cualquier otro motivo que dificulte la corrección de su tarea, serán penalizados con 5 décimas a su nota final por cada motivo de descuento

Es responsabilidad del alumno preocuparse de aceptar la invitación a su repositorio respectivo. No se aceptaran entregas fuera del repositorio entregado ni fuera de plazo.

Ruta de ejemplo

<https://github.com/IIC2143/2025-1-Tarea-UsuarioDeGithub>

Política de integridad académica

Los alumnos de la Escuela de Ingeniería de la Pontificia Universidad Católica de Chile deben mantener un comportamiento acorde a la Declaración de Principios de la Universidad. En particular, se espera que mantengan altos estándares de honestidad académica. Cualquier acto deshonesto o fraude académico está prohibido; los alumnos que incurran en este tipo de acciones se exponen a un Procedimiento Sumario. Es responsabilidad de cada alumno conocer y respetar el documento sobre Integridad Académica publicado por la Dirección de Docencia de la Escuela de Ingeniería en el SIDING. Específicamente, para los cursos del Departamento de Ciencia de la Computación, rige obligatoriamente la siguiente política de integridad académica. Todo trabajo presentado por un alumno para los efectos de la evaluación de un curso debe ser hecho individualmente por el alumno, sin apoyo en material de terceros. Por "trabajo" se entiende en general las interrogaciones escritas, las tareas de programación u otras, los trabajos de laboratorio, los proyectos, el examen, entre otros. Si un alumno copia un trabajo, obtendrá nota final 1.1 en el curso y se solicitará a la Dirección de Pregrado de la Escuela de Ingeniería que no le permita retirar el curso de la carga académica semestral. Por "copia" se entiende incluir en el trabajo presentado como propio partes hechas por otra persona. Obviamente, está permitido usar material disponible públicamente, por ejemplo, libros o contenidos tomados de Internet, siempre y cuando se incluya la referencia correspondiente. Lo anterior se entiende como complemento al Reglamento del Alumno de la Pontificia Universidad Católica de Chile. Por ello, es posible pedir a la Universidad la aplicación de sanciones adicionales especificadas en dicho reglamento.