
Trabajo Final Integrador

Taller de Tecnologías de Producción de Software - Ruby

Cursada 2023



En este Trabajo Final Integrador, buscaremos que apliques los conocimientos adquiridos en la materia para resolver un problema real, basado en un caso hipotético. El objetivo de la implementación que deberás hacer es tener un sistema que permita resolver las necesidades expuestas en este documento.

Contexto

Estás a cargo del desarrollo de una aplicación de generación y gestión de links cortos, al estilo de Bitly, T.ly, o TinyURL. El objetivo de esta aplicación web es brindar links cortos para URLs arbitrariamente largas, que puedan ser utilizados en redes sociales o cualquier comunicación online para compartir páginas web de manera sencilla.

En este documento se describirán los puntos principales y la funcionalidad mínima que la aplicación debe proveer, a la cual podrás agregar soporte para más acciones o interacciones, siempre respetando lo especificado aquí como base.

Requisitos técnicos

El desarrollo debe realizarse utilizando el siguiente stack tecnológico:

- Una versión reciente de Ruby (2.7 o superior).
- La versión estable más reciente del framework Ruby on Rails (7.1.2 al momento de escribir este documento).
- Una base de datos SQLite para dar soporte de persistencia.

Adicionalmente, se te pide que no reinventes la rueda y cuando sea posible utilices gemas bien establecidas que provean funcionalidad habitual en una aplicación web, como puede ser el caso de la autenticación.

Alcance

Deberás desarrollar una aplicación web para crear y gestionar links cortos llamada `chq.to` (que se lee “Chiquito”), que brindará las siguientes funcionalidades:

- Autenticación y registro (ver apartado **Gestión de usuarios**).
- Creación y manejo de links cortos (ver apartado **Gestión de links**).
- Acceso a los links cortos (ver apartado **Acceso a los links**).
- Reportes de las estadísticas de los links (ver apartado **Reportes sobre los links**).

Gestión de usuarios

`chq.to` deberá permitir que los usuarios se registren para poder gestionar sus links y acceder a los reportes. Para esto, la aplicación deberá brindar una interfaz de registro de usuarios que les permita crear su propio usuario, y luego mecanismos de autenticación y autorización que permitan que los usuarios se identifiquen y accedan a sus links. Cada link pertenece únicamente al usuario que lo creó y es únicamente ese usuario quien podrá modificar o borrar el link y visualizar la información relacionada a éste (estadísticas de acceso).

Cada usuario deberá tener, como mínimo, los siguientes atributos:

- Nombre de usuario (debe ser único)
- Correo electrónico (debe ser único)
- Contraseña

Los usuarios podrán borrarse, siendo únicamente el propio usuario quien podrá dar de baja (y borrar) su cuenta de la aplicación.

Gestión de links

Cada usuario autenticado tendrá acceso a un tablero de control que le mostrará sus links cortos y le permitirá:

- Crear un link nuevo (automáticamente asociará al usuario como dueño del link).
- Modificar un link existente.
- Borrar un link existente.
- Visualizar las estadísticas de acceso a un link existente.

Cada link corto tiene un código único o *slug* que lo identifica, el cual es generado y asignado automáticamente por el sistema al momento de crear el link. El algoritmo usado para generar la secuencia de caracteres única para cada link queda a tu criterio, así como también las cadenas que se generen, pero siempre deberán respetarse caracteres que sean *seguros* para ser utilizados en el *path* de una URL según el protocolo HTTP. Lo mismo ocurre con el patrón de URL que se utilice para generar las URL públicas de los links, el cual deberá ser diseñado apuntando a ser lo más corto posible. De aquí en más llamaremos “URL única” a la URL pública que se utilizará para compartir un link corto. Por ejemplo, si un link tiene el *slug* `dAFn2aB` y el patrón de URL a utilizar es `/l/:slug`, la *URL única* de ese link será `https://chq.to/l/dAFn2aB`.

Además de esta información, cada link deberá tener como mínimo un nombre opcional, y las validaciones pertinentes de cada campo.

Nota: los links cortos pueden ser gestionados únicamente por el usuario que los creó (su dueño).

Tipos de links

Existen diferentes tipos de links cortos:

- Links regulares: son los links más básicos, pueden accederse públicamente y simplemente redirigen al visitante a la dirección de destino (previo registro del acceso para las estadísticas del link).
- Links temporales: funcionan igual que los links cortos regulares, pero tienen además una fecha y hora de expiración. Pasada esa fecha y hora, el link seguirá existiendo en el sistema pero no funcionará más - es decir que cualquier acceso a su URL única retornará un código de respuesta HTTP 404 `Not Found`.
- Links privados: son links que, cuando un visitante que intenta accederlos mediante su URL única, le solicitan una clave (establecida por el usuario dueño del link) para acceder a la dirección de destino del link. Sólo si la clave que el visitante ingresa coincide con la que se estableció para el link, el visitante podrá acceder a la dirección de destino. En caso contrario, se le volverá a solicitar la clave sin permitir acceder a la dirección de destino.
- Links efímeros: son links que solo pueden accederse una vez. Luego de esto, el link ya no funcionará más y cualquier acceso a su URL única retornará un código de respuesta HTTP 403 `Forbidden`.

Acceso a los links

Cualquier acceso a la URL única de un link, si se cumplen las condiciones necesarias acorde a lo especificado en el apartado **Tipos de links** para el tipo del link al que se está intentando acceder, generará una redirección HTTP (con un código de respuesta apropiado) cuyo destino será la URL de destino del link. Previamente a responder con la redirección, se deberá contabilizar el acceso al link junto con la información necesaria del acceso para poder almacenar lo que se requiera para poder brindar al usuario los reportes detallados en el apartado **Reportes sobre los links**. En caso que no se cumpla con las condiciones necesarias para poder acceder al link según su tipo, el intento de acceso no se contabilizará para las estadísticas.

En el caso de los links privados, la aplicación deberá presentar una página solicitando la clave del link para poder accederlo.

Reportes sobre los links

Cada usuario podrá visualizar reportes detallados o resumizados de las estadísticas de acceso a los links de los que sea dueño. Como mínimo, se deberán implementar estos reportes:

- Detalle de los accesos: deberá presentar un listado detallado de los accesos que tuvo el link. Para cada fila en el reporte, se deberá mostrar la fecha y hora, dirección IP desde la que se accedió. El reporte deberá permitir buscar por dirección IP o rango de fechas.
- Cantidad de accesos por día: muestra la cantidad de accesos por cada día que tuvo el link.

En caso de así desearlo, podrás extender este listado de reportes con otros que consideres pertinentes y para los cuales podrás también almacenar más información sobre cada acceso. Por ejemplo, algunas posibilidades: cantidad de accesos por dispositivo o User Agent, cantidad de accesos únicos (es decir, contando una única vez por sesión del navegador los accesos), etc.

Consideraciones generales

- Se espera que todos los modelos tengan las validaciones adecuadas para asegurar la integridad de los datos.
- Se deberá incluir un conjunto de datos de prueba para poder evaluar el sistema. Estos datos deberán generarse mediante el script `db/seeds.rb` del proyecto Ruby on Rails.
- No se solicitará la implementación de un diseño visual para la interfaz de usuario. Se espera que la interfaz de usuario sea cómoda, entendible, y lo más intuitiva posible, pero no se solicitará para la entrega ningún tipo de librería ni interfaz en particular.
- Se deberá documentar en el archivo `README.md` del raíz del proyecto cualquier decisión de diseño importante que se haya tomado, así como también se deberán dejar escritos los requisitos técnicos y los pasos (comandos) para hacer funcionar la aplicación.

Modalidad de la entrega

El trabajo deberá realizarse de manera individual. Los trabajos entregados se compararán para encontrar posibles similitudes en su estructura que pudieran provenir de copias entre distintas entregas.

Esta entrega, además, es de carácter obligatorio, y su aprobación (en cualquiera de sus instancias) es condición necesaria para aprobar la cursada de la materia.

Tu trabajo tiene que estar versionado en un repositorio Git de GitHub, cuya URL deberás entregarnos mediante una tarea dedicada que publicaremos en el curso de la plataforma de Cátedras (Moodle). En caso que el repositorio sea privado, vamos a necesitar que nos brindes acceso al mismo para poder

descargarlo y evaluarlo. Si esta es tu decisión, por favor consultá antes de la fecha límite de la entrega con el Jefe de Trabajos Prácticos para saber a qué usuarios de GitHub deberás brindarles acceso a tu repositorio.

Las fechas límite de entrega del trabajo fueron publicadas en el curso de la plataforma de Cátedras. No se admitirán entregas fuera de término.