

Dewie

Cameron Lopez

December 13, 2011

Contents

1	Dewie	1
2	Purpose	2
3	Goal	2
4	Description	2
4.1	Client	2
4.2	Backend	2
4.2.1	Database	3
5	Discussion	3
5.1	Challenges	3
5.2	Strengths	3
5.3	Weaknesses	3
6	Future	3
7	Bugs	4

1 Dewie

The place for your Books Github: <https://github.com/cjoelrun/Books>
Heroku: <http://dewie.herokuapp.com/>

2 Purpose

The purpose of my project, Dewie, is to offer readers a way to track which books they want to read, are reading, and have read.

3 Goal

Dewie's goal is to allow readers to add books easily to reading lists. Functionality consists basically of searching for books, bookmarking books, managing bookmarks, and discovering new books other users have bookmarked on the site.

4 Description

4.1 Client

The face of Dewie is simple. Interfaces are meant to appeal to readers who love the simplicity of text on the page of a book. The homepage displays the user's current reading list to a logged in user. The search page allows the user to search for a book. Information is displayed along with resulting books. After choosing a book it shows up in your bookmarks. Your bookmarks page allows you to manage bookmarks, delete or move to another shelf. The books other users have added are on the books page which can be bookmarked with the click of a button.

4.2 Backend

Behind Dewie are three major components. Users, books, and bookmarks. How they're modeled describes how the site handles bookmarking well: Users and books have a many-to-many relationship with each other through bookmarks. Users are handled through devise, a RubyGem. They can have many books through bookmarks. Books are scaffolded to contain all the information given by the Google Book API query, carried out by the google_books RubyGem. Books can have many Users through Bookmarks. Bookmarks are the center functionality of the site. They track Books that Users wish to track.

4.2.1 Database

The User's database is generated through Devise. The Book database contains all the fields given by the Google Book API. The Bookmarks database contains the shelf the book is stored on and respective User and Book.

5 Discussion

5.1 Challenges

A challenge involved in creating Dewie was integrating the foreign data from Google to the application. I believe the largest challenge was simply learning the ways of Rails and expressing my application in such abstract ways. The magic of Rails still eludes me in certain aspects, but I hope to conquer them as I continue to develop Dewie.

5.2 Strengths

Features worth talking about are things like the ability to search any book indexed by Google. It offers a very intuitive way of entering books into the site, rather than filling all information in by hand. This information provides attractive thing such as photo covers and indepth information about each book.

5.3 Weaknesses

A weakness of the site now is that the api is still limited in uses because a key can't be published publicly to github so search will occasionally fail as too many requests are made. Also the intuitive nature of adding books is a multi step process. A book must be searched for, added, then navigated to, then finally added as a bookmark.

6 Future

I plan to add a private api key so that api querying can be carried out more practically. Also I intend to making searching and bookmarking books more intuitive.

7 Bugs

Bugs existing are occasional failures by searches because of API key complications, and routing errors because certain functionality has be removed from scaffolds but not unrouted.