Final Project Proposal: Online Bookstore

WEBD 3102 (J2EE)

Chris Arbuthnott, w0276812

Nov 24th, 2015

Project Description

Arbuthnott’s Books provides an online selection of books available to order for purchase at the local store during business hours. The website includes books available in-store, and some that can be ordered in from a storage facility. The website allows customers to order books via a shopping cart, and provides estimates as to arrival times and cost of the books.

The Administrative site can update the staff about which orders are pending, what payments specific customers are due to make, and allows them to update the book inventory as new shipments arrive.

Features

* A public website to browse book titles.
* Books may be browsed by author.
* A dedicated page for each book, detailing price, location and availability of the book. Users who are logged in could place an order for a book from this page.
* Administration page for employees to update online items. They can also use the site to keep track of which orders are due to arrive in the near future.
* A dedicated Error page will catch exceptions and display user-friendly messages.
* User profiles can be modified, or created, and include a profile picture.

Project Limitations

* Persisting data will be managed via logs and text-files rather than a database.
* There is no online payment available
* Customer interaction is not automated – staff and customers must get in contact themselves regarding payment and/or delivery of books from the local storefront.

Technology/Tools used in this Project

* Eclipse (Mars) IDE
* Tomcat Apache Server
* J2EE Web tools
* MVC Architecture

Features, Updated

The project has been mostly completed as projected, however one or two minor additions or subtractions are of note:

* User profiles exist at different levels (admin/basic), and can be edited by the user. However, the functionality for creating new users from the interface is not in place. This was omitted to limit the scope of this large project.
* A cart has been added, so users may add several books, then choose whether to clear their carts or make their order.
* Book inventories are updated when orders are made. Error handling is in place for when inventories change between when a user adds items to cart and make an order.
* Users can view their existing orders.
* Books can be browsed by author.
* Admin users can modify or delete orders, and modify or add new items to the inventory.

Software Components

My project contains several model classes, .jsp views, filters and servlets.

Model

* **User** class represents a user, including login details, and admin/normal status
* **Book** class represents an inventory item, including data like title and author, but also available stocks at the store and in remote storage.
* **OrderItem** class represents a purchase of one inventory item (possibly more than 1 copy) by one user. It contains references to the user and book, and calculates a delivery date on creation.
* **Order** class represents a collection of order items made at once by a single user. An order is needed for a user to order more than one book title at once. The Cart is saved as an Order.

Views (.jsp)

* admin.jsp (Admin hompage)
* bookDetail.jsp (Details of one book)
* bookInventory.jsp (Updatable form for existing inventory item)
* bookList.jsp (List of books)
* cart.jsp (user’s cart)
* error.jsp (error page)
* front.jsp (Homepage of site)
* login.jsp (Login form)
* newBook.jsp (Form to create new inventory item)
* order.jsp (Details of an existing order)
* profile.jsp (User profile details)
* profileEdit.jsp (Form to edit profile)

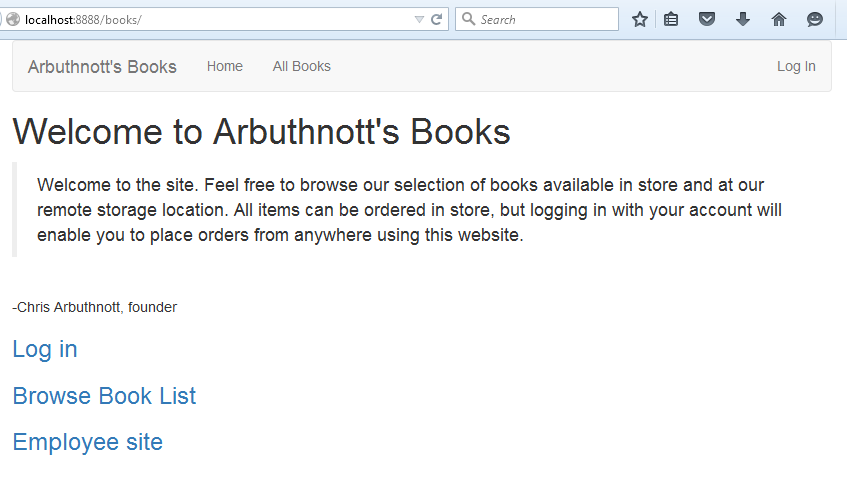
Filters

* **AdminFilter** A filter to make sure only admin users visit admin pages
* **DataFilter** A filter responsible for making sure text file data has been loaded into context variables. This filter also packages changed data back into text files for storage
* **LoggedInFilter** A filter to make sure users are logged in to visit certain pages (cart, profile, etc)
* **OwnOrderFilter** A filter that makes sure non-admin users can’t view orders of other users.

Controllers (Servlets)

* **AdminServlet** A servlet to handle admin request, such as viewing the admin home page, accessing inventory items and order details, and requests to update inventory or order info.
* **CartServlet** A servlet to handle cart views, and actions such as clearing the cart or purchasing the items in the cart
* **FrontServlet** A minimal servlet for some all-user views, such as home page, or book detail.
* **ImageServlet** A servlet to handle upload of images.
* **ListServlet** A servlet to handle display of book lists, by all books or by author
* **LoginServlet** A servlet to display the login and handle login requests, registering a successful login in the session variables.
* **NewBookServlet** A servlet to display a form to create new Inventory Items, and handle creation requests.
* **OrderServlet** A servlet to show users their own orders. Admin views of orders, and order updates are handled by the admin servlet
* **ProfileServlet** A servlet to display the user profile, or the form to edit it, and handle update requests.

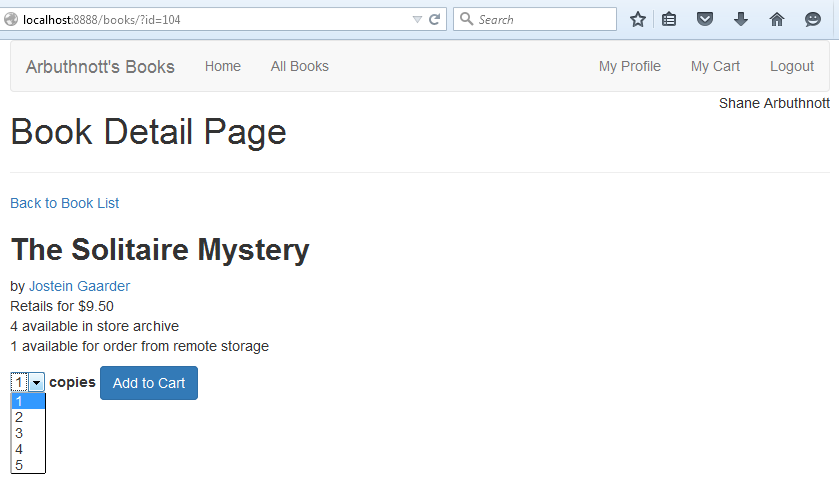
Screen Captures:



/books Home Page for site

Nav-bar present on all pages.

Login not required for home page

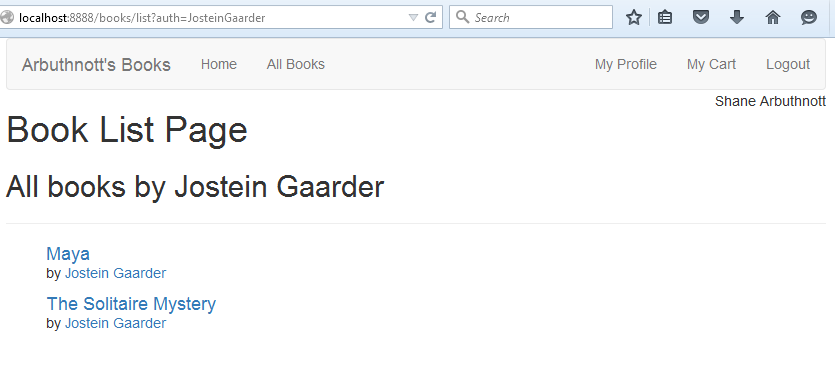


/books/list?aut=JosteinGaarder

Book list page

Login not required.

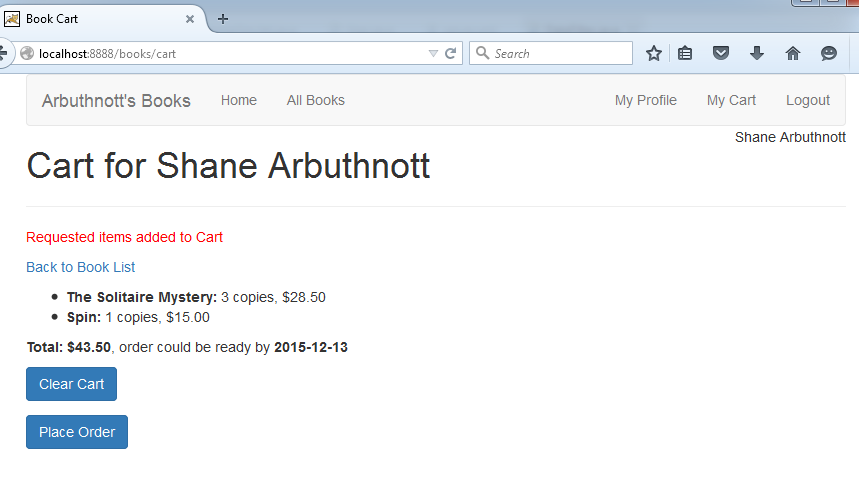
All books or by author

/books?id=101

Book detail page

If logged in, option to add to cart.

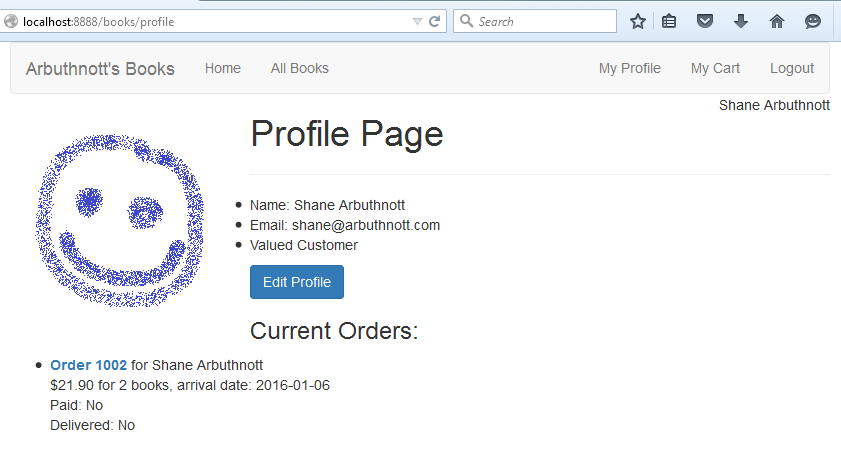
Copies select element matched to current inventory

/books/cart

Cart, available only if logged in

Completed actions displayed in red

Option to clear cart, or place order



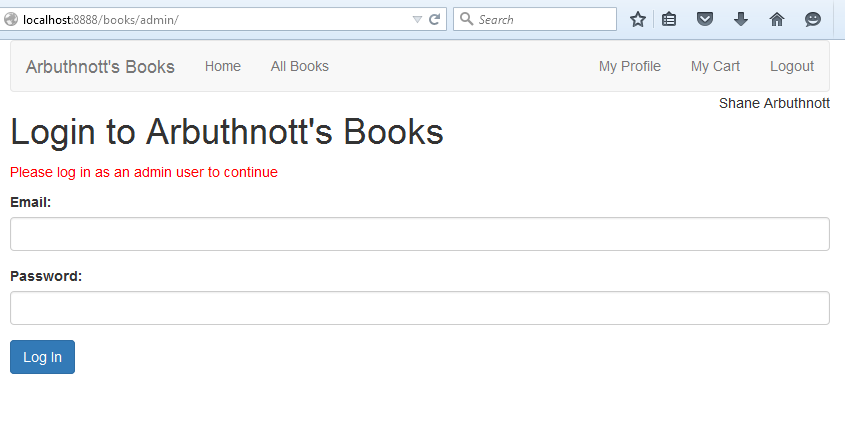
/books/profile

Available to logged in users

Profile picture

Option to edit

List of current orders

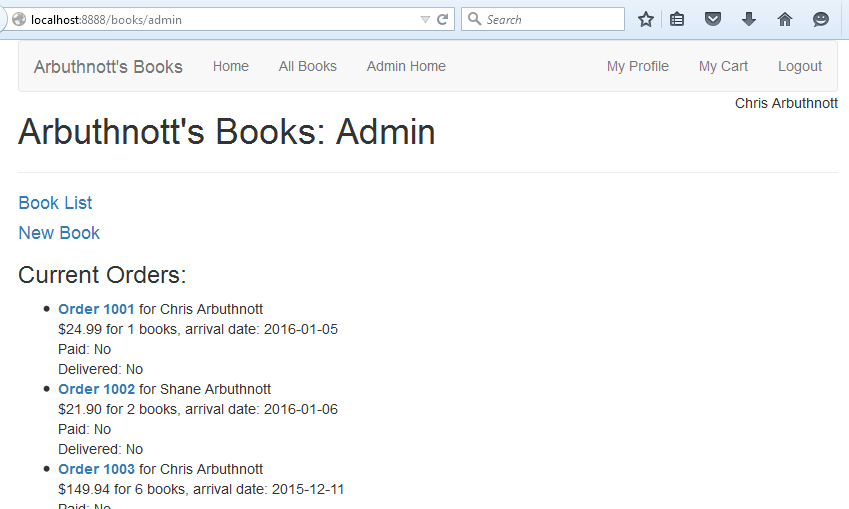


/books/login

Login screen

Diverted here with message if attempt to access user-only or admin-only areas.

Message displayed in red

/books/admin

Admin home page

Must be logged in as admin user

Option to add new book to inventory

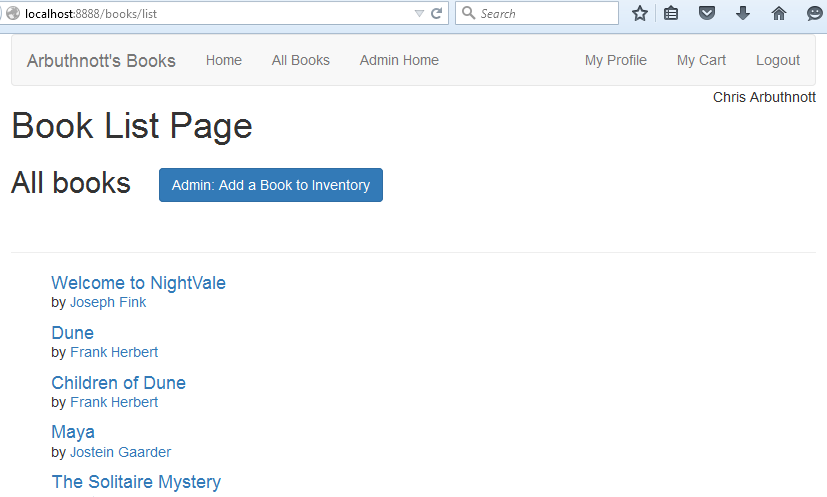
List of all current orders

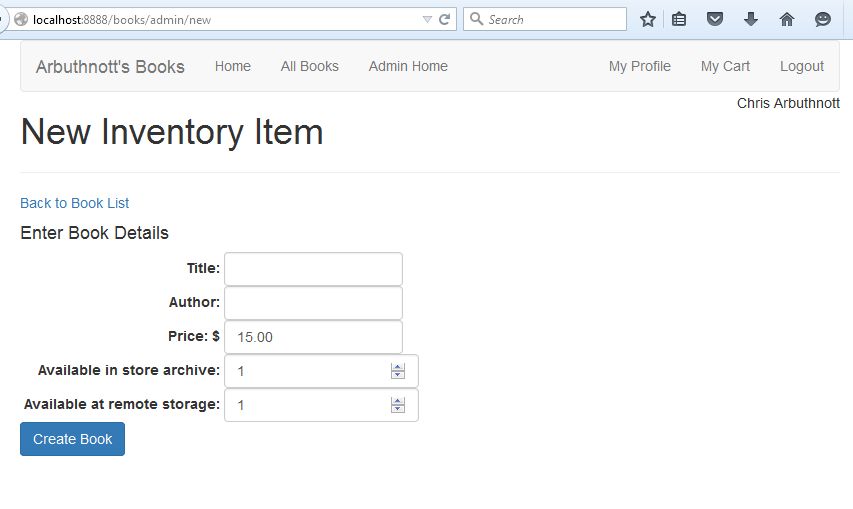
/books/list (admin view)

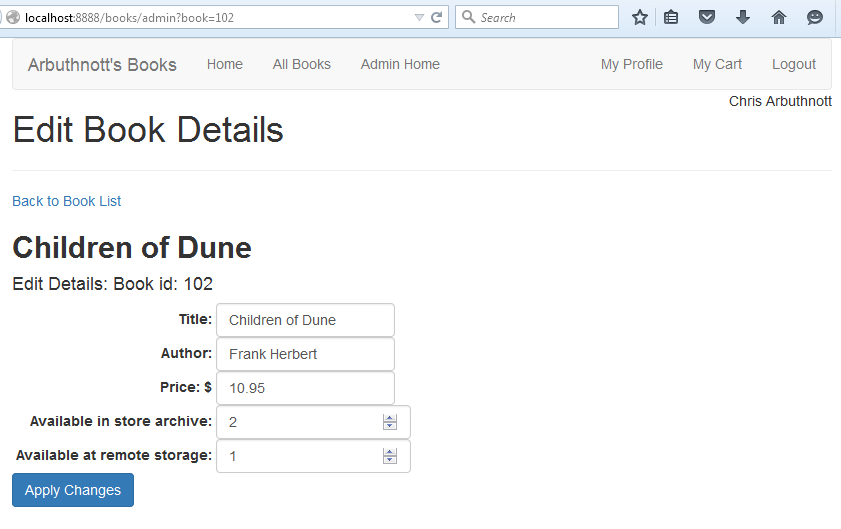
If admin user is logged in, option to add a book to inventory is present.

/books/admin/new

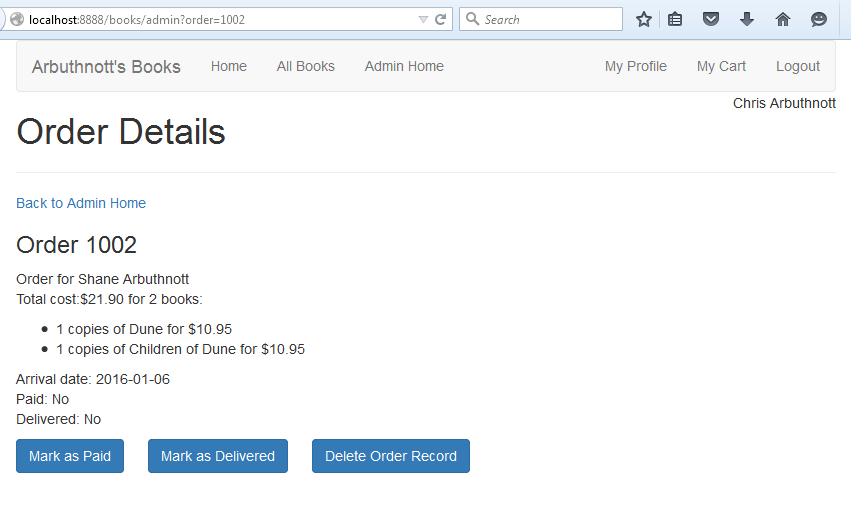
Page for admin user to add a book to inventory





/books/admin?book=101

Page for admin user to edit a book in the inventory



/books/admin?order=1002

Page for admin user to inspect any order

Option to mark as paid, mark as Delivered, or remove from list.