1 How to Run/Verify Project

1.1 Introduction

I'm embarrassed that my application is so plain and simple compared to others. You might recognize it from MSSE 680. It's the same thing, but I have rebuilt it from scratch for this course with things I learned in the last course. I started with an empty project and have spent no time at all on polishing up the presentation layer look and feel, because I've just been trying to get various things working in the application plumbing. In addition, some elements aren't fully working as intended and some are not yet implemented but very close. In particular, the "Browse Items Alphabetically" is close but not ready. I'm catching up a little more every week. Please feel free to review the source code and offer and feedback, advice, suggestions, etc.

1.2 Git Repository

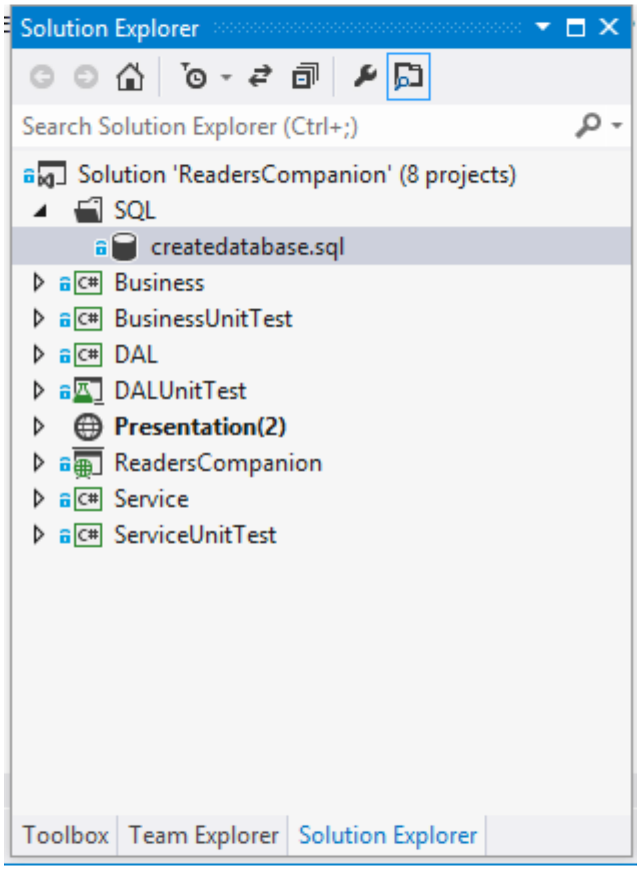
My Git master repository is located at: https://github.com/paulworthington/prw_msse682

The Week 5 branch is located at: https://github.com/paulworthington/prw_msse682/tree/week5

The solution file is ReadersCompanion.sln.

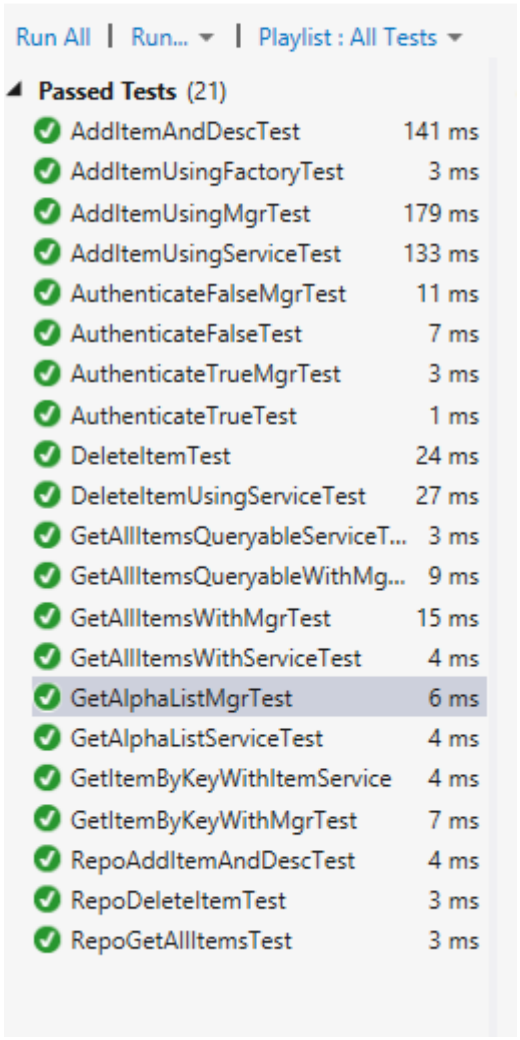
1.3 SQL Server

Open the createdatabase.sql script located in the ReadersCompanion > SQL folder in Visual Studio. In File Explorer, this script is located at the same level as the DAL, Service, etc., folders. When you run this script, it will create the readerDB01 database with my tables and the required membership tables.



1.4 Unit Tests

There are 21 unit tests you can run via Test – Run – All Tests. The tests that start with "Authenticate" were written against Service and Business classes for a Login I'd created prior to the Week 5 membership work, but I might repurpose them for this week's work. Otherwise, the names should be self-explanatory.



Run All | Run... | Playlist : All Tests

Passed Tests (21)

```
AddItemAndDescTest              141 ms
AddItemUsingFactoryTest           3 ms
AddItemUsingMgrTest             179 ms
AddItemUsingServiceTest         133 ms
AuthenticateFalseMgrTest         11 ms
AuthenticateFalseTest             7 ms
AuthenticateTrueMgrTest           3 ms
AuthenticateTrueTest              1 ms
DeleteItemTest                   24 ms
DeleteItemUsingServiceTest       27 ms
GetAllItemsQueryableServiceT...   3 ms
GetAllItemsQueryableWithMg...     9 ms
GetAllItemsWithMgrTest           15 ms
GetAllItemsWithServiceTest        4 ms
GetAlphaListMgrTest               6 ms
GetAlphaListServiceTest           4 ms
GetItemByKeyWithItemService       4 ms
GetItemByKeyWithMgrTest           7 ms
RepoAddItemAndDescTest            4 ms
RepoDeleteItemTest                3 ms
RepoGetAllItemsTest               3 ms
```

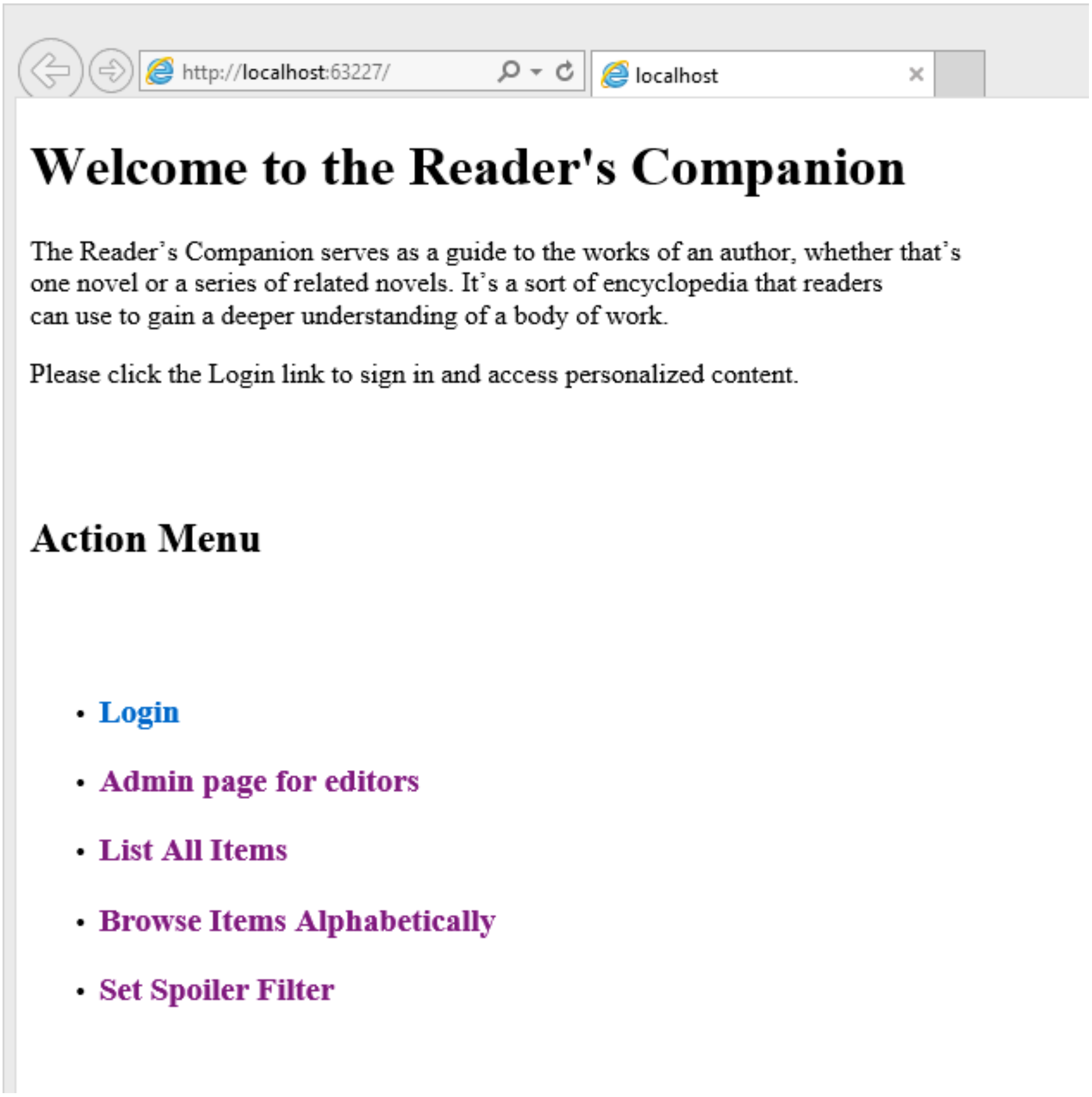
1.5 Launch the site

Open the solution in Visual Studio, press F5 to launch Default.aspx.

Default

This is the application's main start page. Ultimately, this would have links for various ways for users to interact with the database of items. My plan is that an editor would see different navigation links than a regular user. But I'm putting the "admin page for editors" link there for demonstrating this week's membership and Login changes.

If a user is not logged in, there's a "please login" message.



Welcome to the Reader's Companion

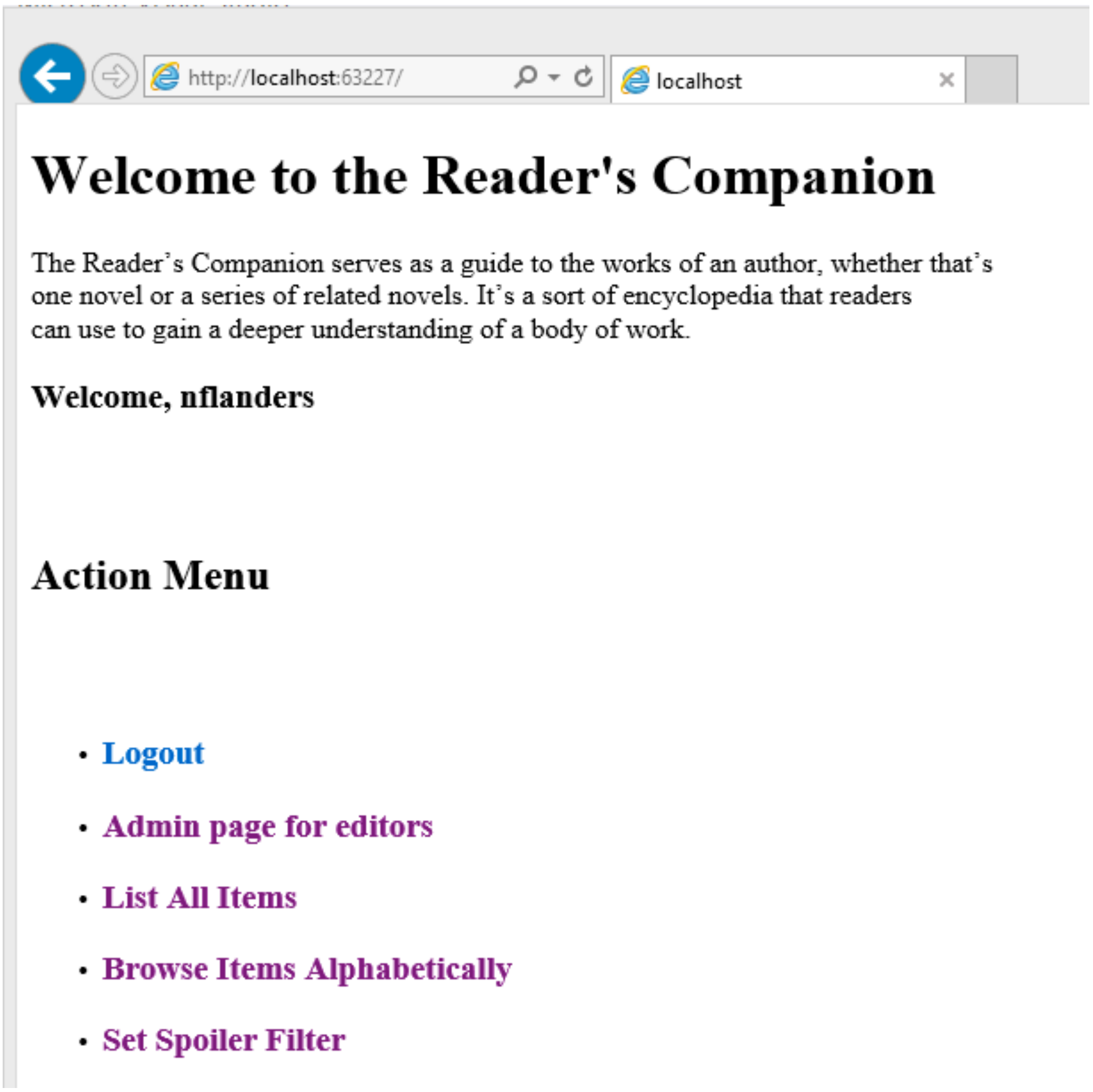
The Reader's Companion serves as a guide to the works of an author, whether that's one novel or a series of related novels. It's a sort of encyclopedia that readers can use to gain a deeper understanding of a body of work.

Please click the Login link to sign in and access personalized content.

Action Menu

- Login
- Admin page for editors
- List All Items
- Browse Items Alphabetically
- Set Spoiler Filter

After logging in, there is a message welcoming the user by username. Also, the Login link changes to Logout.



Welcome to the Reader's Companion

The Reader's Companion serves as a guide to the works of an author, whether that's one novel or a series of related novels. It's a sort of encyclopedia that readers can use to gain a deeper understanding of a body of work.

Welcome, nflanders

Action Menu

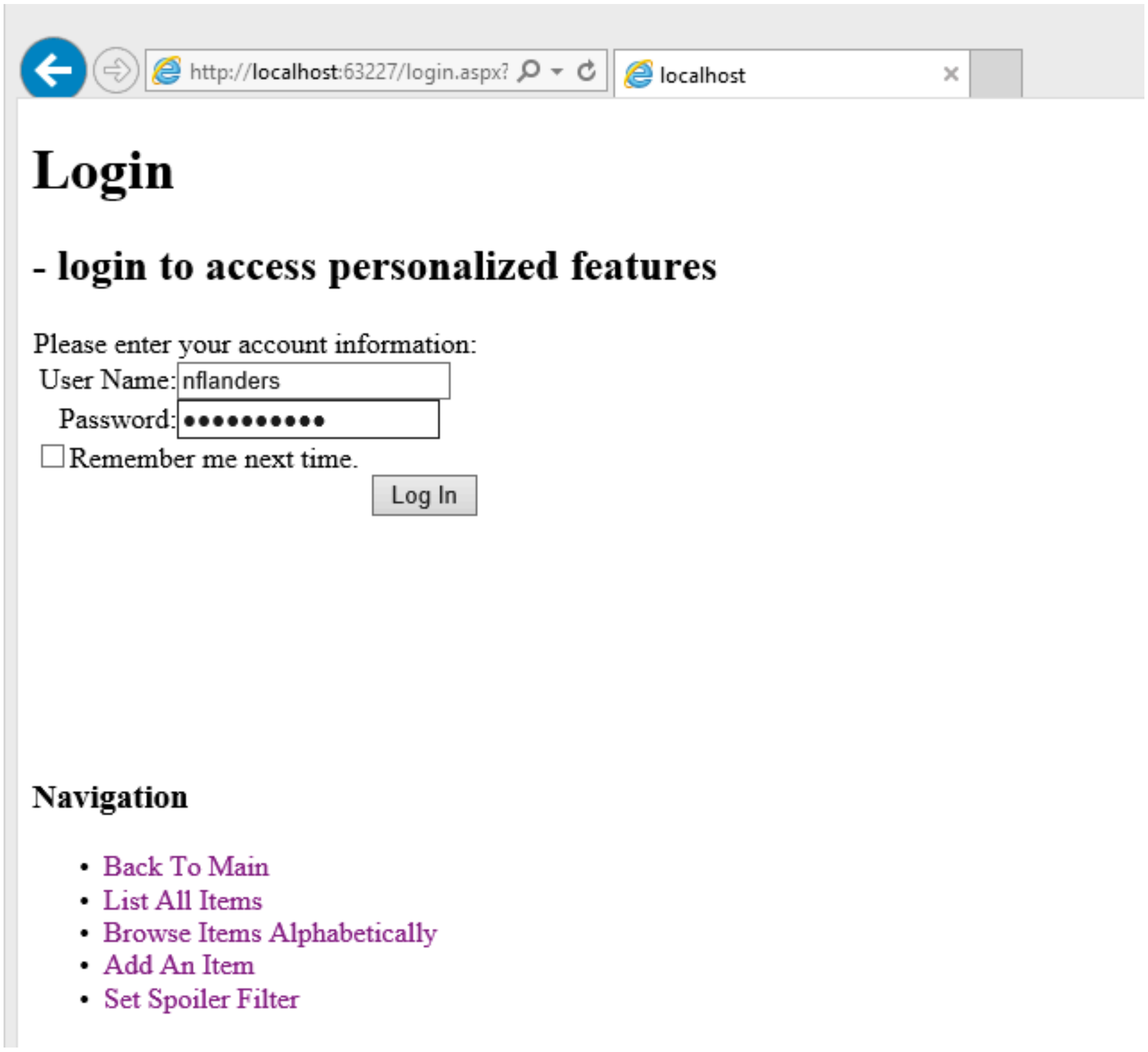
- Logout
- Admin page for editors
- List All Items
- Browse Items Alphabetically
- Set Spoiler Filter

Login

This is a simple login page that uses the Login control to collect a name and password, then authenticate the user.

Test:

1. Leave either or both fields blank and click Login.
2. Enter name: hsimpson and pass: 12345678++ for a user with the User role.
3. Enter name: nflanders and pass: abcdefgh++ for a user with the Editor role.
4. Enter either the correct name and incorrect password, or vice versa, for incorrect login.



Login

- login to access personalized features

Please enter your account information:
User Name: nflanders
Password: ●●●●●●●●●●●
☐ Remember me next time.
Log In

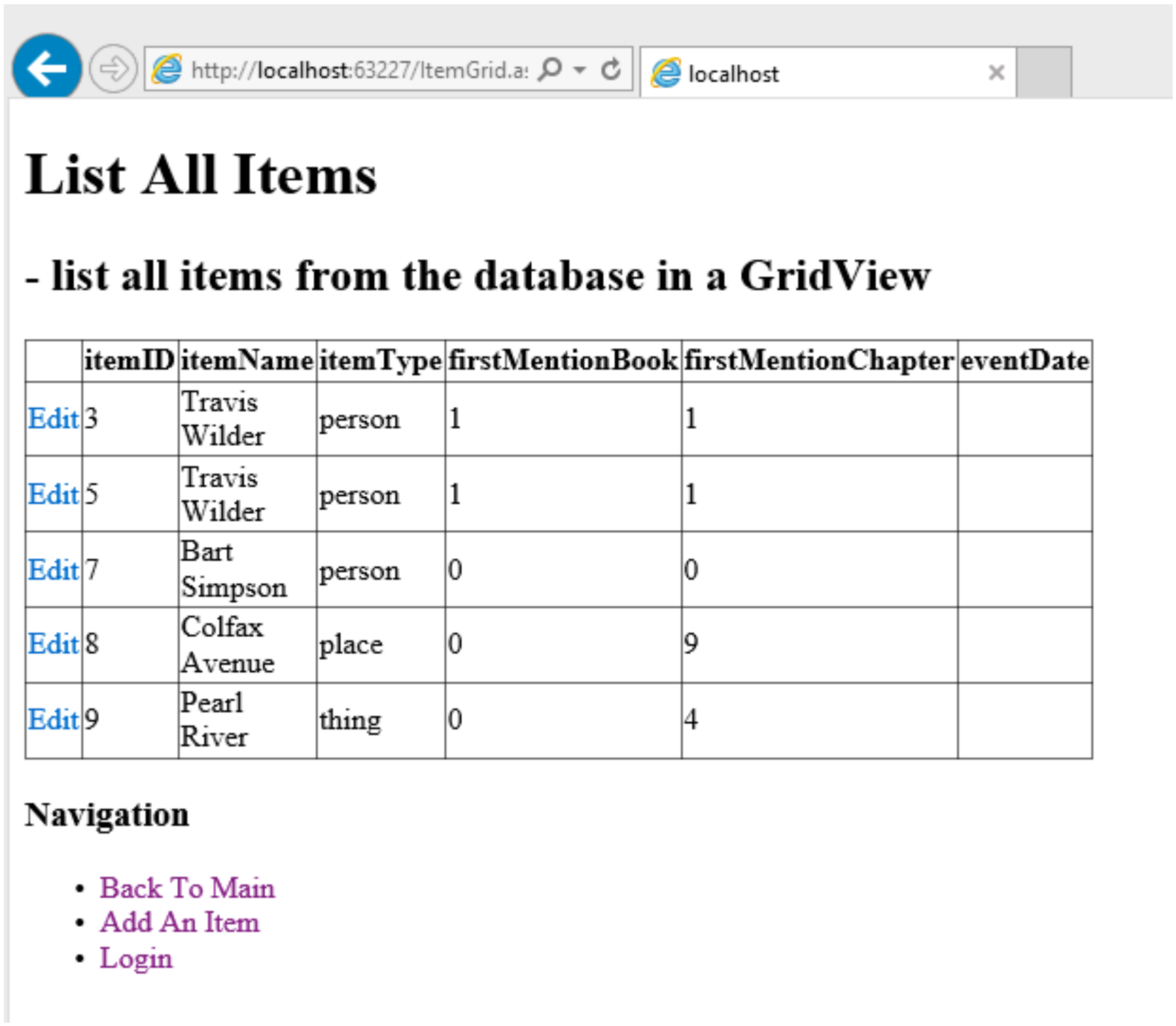
Navigation

- Back To Main
- List All Items
- Browse Items Alphabetically
- Add An Item
- Set Spoiler Filter

ItemGrid

Click on List All Items to open the ItemGrid form. Click Edit next to any row, change some data, then click Update.

For right now, this page is a very basic GridView of the Item table. I didn't spend any time on this page this week, because I was working on adding the membership and Login pieces. I chose to use an Entity data source for the GridView.



List All Items

- list all items from the database in a GridView

	itemID	itemName	itemType	firstMentionBook	firstMentionChapter	eventDate
Edit	3	Travis Wilder	person	1	1	
Edit	5	Travis Wilder	person	1	1	
Edit	7	Bart Simpson	person	0	0	
Edit	8	Colfax Avenue	place	0	9	
Edit	9	Pearl River	thing	0	4	

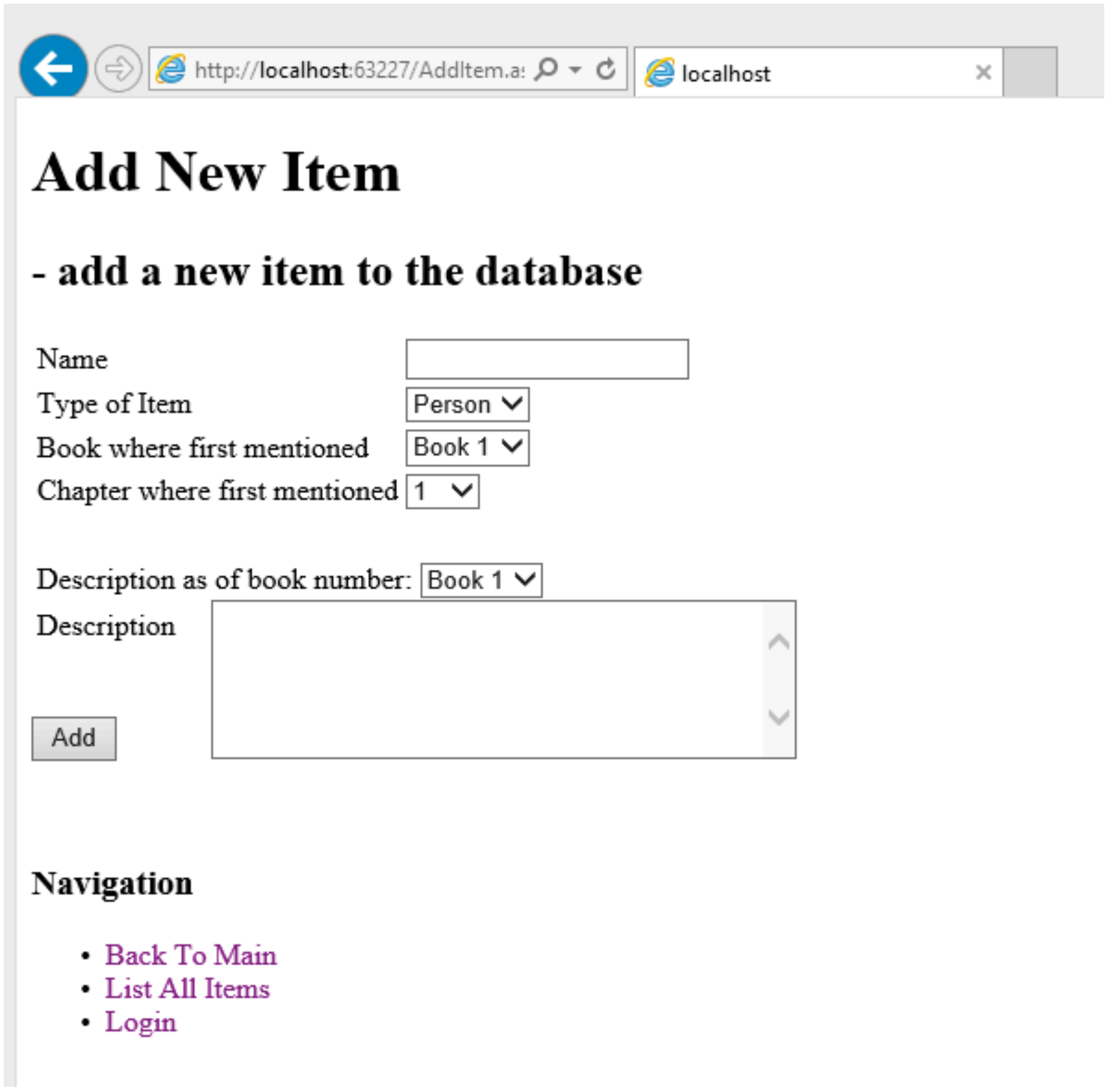
Navigation

- Back To Main
- Add An Item
- Login

AddItem

Click on Add An Item to open the AddItem form. Enter info into the form, then click Add. You will get a confirmation dialog. After that, you can click List All Items again, scroll to the bottom, and see that your added item is there in the database.

You should always add an item description whenever you add a new item. This keeps the database tidy and more manageable. I have a lot of code enhancements in mind for this form, but for now it collects information from the user, adds the item to the Item table, then retrieves the item's ID, and then adds the item description to the ItemDescription table. If you test this, please enter a unique item that hasn't already been added.



Add New Item

- add a new item to the database

Name	
Type of Item	Person
Book where first mentioned	Book 1
Chapter where first mentioned	1

Description as of book number: Book 1

Description

Navigation

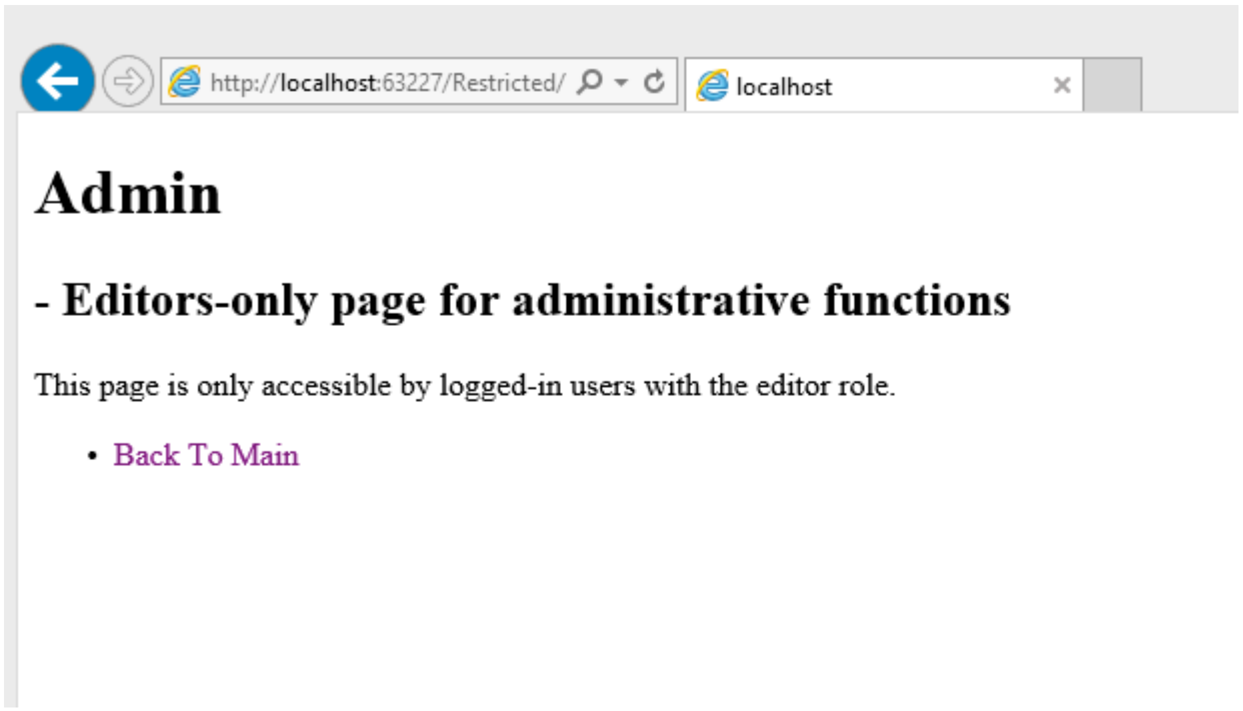
- Back To Main
- List All Items
- Login

Admin

This form is for demonstration purposes. It's located in the Restricted folder, and only a user with the Editor role can access it.

Test:

1. From the Default page, and while not logged in, click on the "admin page for editors" link. You will be directed to the Login form, demonstrating that casual anonymous users cannot access the page.
2. Login as user hsimpson – Enter name: hsimpson and pass: 12345678++ for a user with the User role. Now click on the "admin page for editors" link. You will be directed to the Login form, demonstrating that users who are not editors cannot access the page.
3. Login as user nflanders – Enter name: nflanders and pass: abcdefgh++ for a user with the Editor role. Now click on the "admin page for editors" link. You will be directed to the Admin form, demonstrating that users who are editors can access the Admin form in the Restricted folder.



Admin

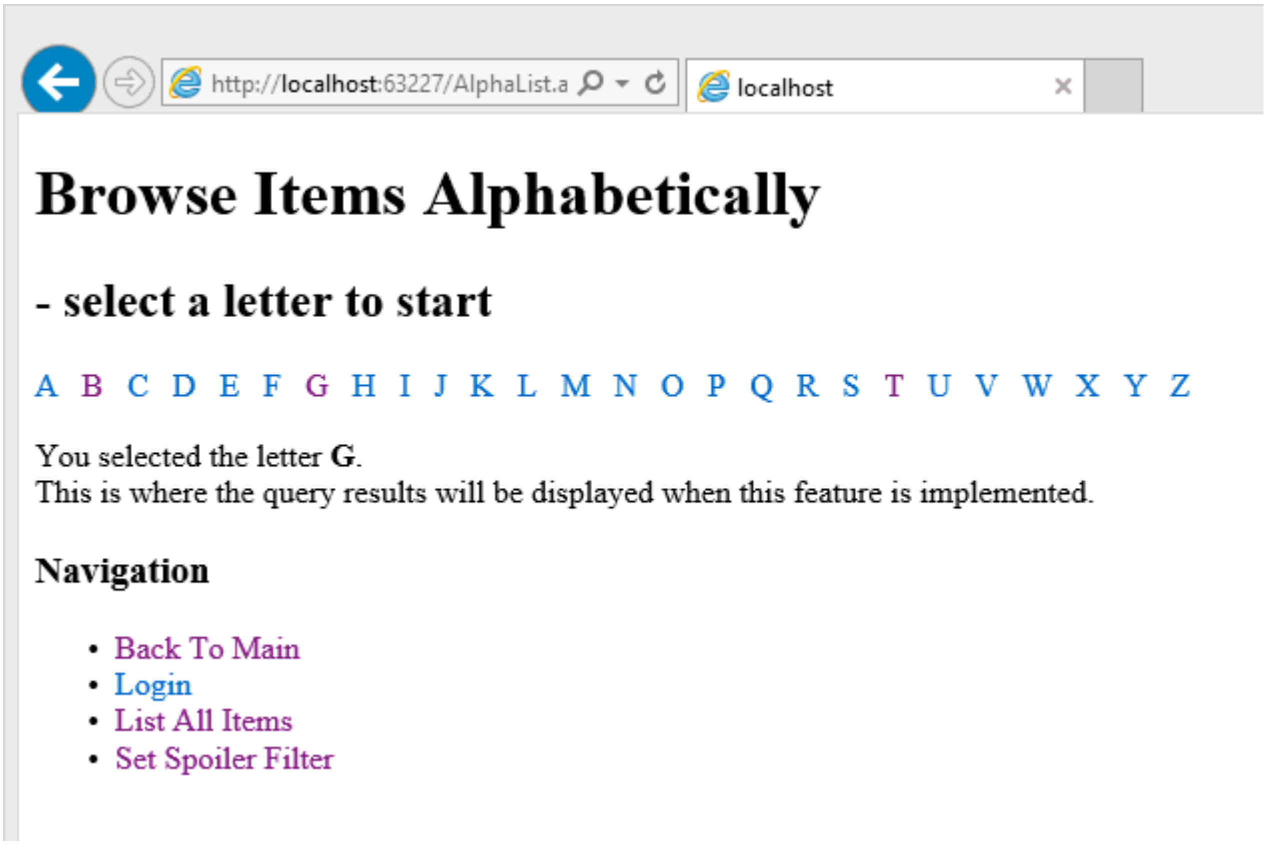
- Editors-only page for administrative functions

This page is only accessible by logged-in users with the editor role.

- Back To Main

AlphaList

To Browse Items Alphabetically, I present the user with the letters of the alphabet. Clicking one puts a query string on the URL that I use in the data query. I think there might be a better way to do this, but I haven't figured it out yet. For now the form only confirms which letter you clicked.



Browse Items Alphabetically

- select a letter to start

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

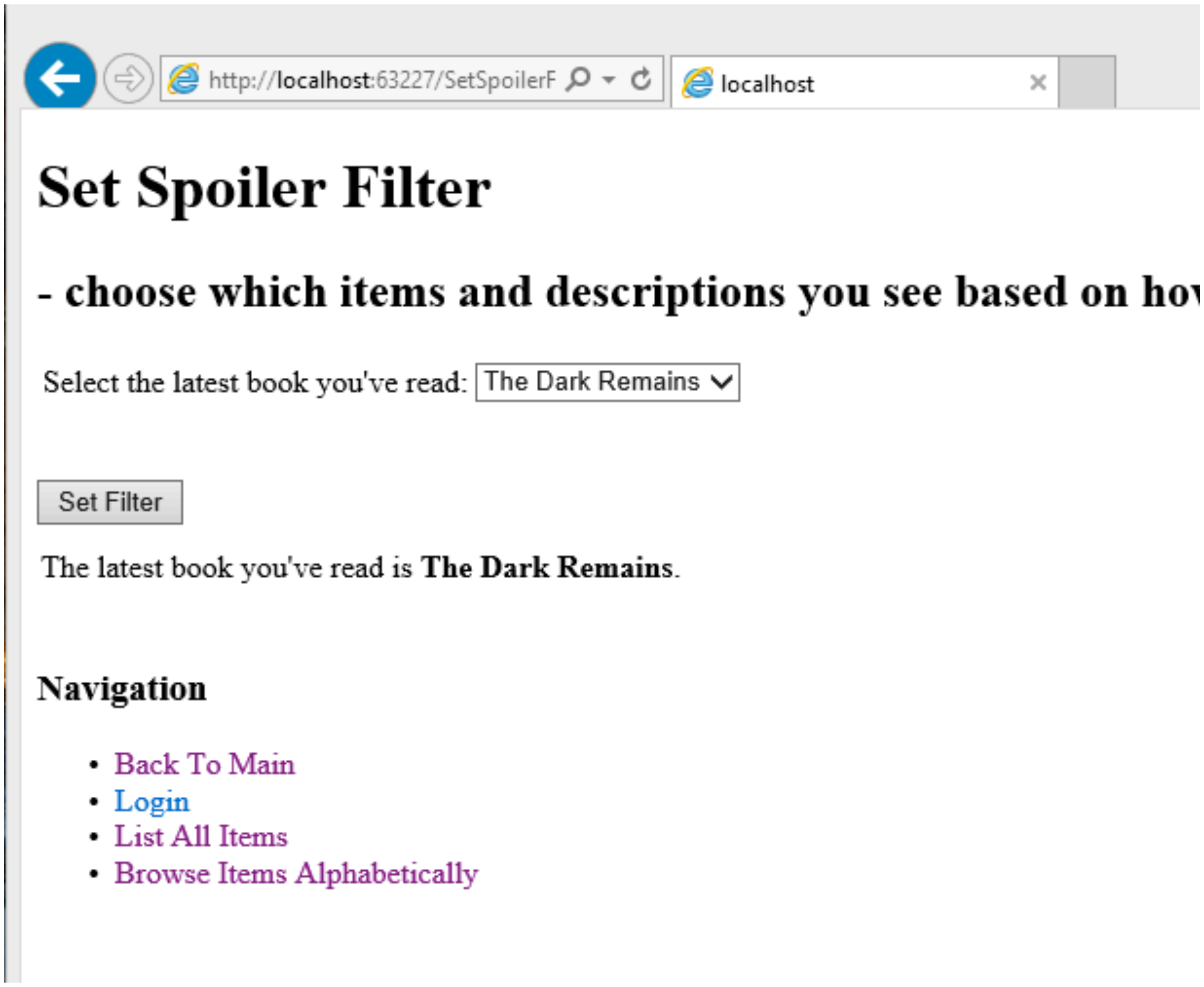
You selected the letter **G**.
This is where the query results will be displayed when this feature is implemented.

Navigation

- Back To Main
- Login
- List All Items
- Set Spoiler Filter

SpoilerFilter

The Set Spoiler Filter form will let the user control which items and item descriptions they see in the rest of the application. I have in mind how I will programmatically handle that, but haven't gotten around to implementing it yet. For now, this form is used to demonstrate usage of the Session object to preserve application state from page to page.


http://localhost:63227/SetSpoilerF

localhost

Set Spoiler Filter

- choose which items and descriptions you see based on ho

Select the latest book you've read: The Dark Remains

The latest book you've read is **The Dark Remains**.

Navigation

- Back To Main
- Login
- List All Items
- Browse Items Alphabetically