

# DATA2410 Portofolio 2 - Web Shop

Created by: s305057.

I have made a web shop using:

- Python3 with Flask for backend.
- MySQL for the database.
- MDBootstrap and jQuery's Ajax/Javascript for frontend.
- Google Sign in for login authentication.

I am working by myself, and not in a group, so there are a few features here and there that are missing as I didn't get time for those, but I will get more into what I mean by that later when describing the webshop in a later paragraph.

This is a webshop where you can buy books in a store called An Odd Bookstore.

I have mostly been testing in Chrome and Edge so that is where I believe they will work best. I did some testing in Safari too while making sure my program runs on other computers as well, and it didn't look as good there, so I recommend sticking to Chrome/Edge.

I will start off explaining how to start it so that you can get it up and running first so that explaining the site makes more sense as you can see it and use it as you go.

## How to start the server.

I assume you have Docker installed so I will start there. If not, download docker desktop app as it will start off the docker engine which is needed here.

First you need to create a network for it to run on, do that by doing the command:

```
docker network create isekai
```

Then you would need a couple images, get those by doing:

```
docker pull python
```

```
docker pull mysql
```

Now you will need 2 terminal windows as I didn't get Docker-Compose to work.

In both windows you need to move into the folder that is attached in this portfolio called "DATA2410\_Porto2". You should be in the folder so that when you use the ls (or dir command if on windows) you will see the folders /frontend, /backend, /sql.

In 1 window you will run the command:

```
docker run -it --rm --name mysql1 --network isekai -e  
MYSQL_ROOT_PASSWORD=pass123 -v  
$(pwd)/sql:/docker-entrypoint-initdb.d mysql
```

Now wait for this to finish. Its last line should be *"MySQL Community Server - GPL"*

In a different terminal window first you need to build an image from the Dockerfile.

Do the command:

```
docker build -t python_backend ./backend/
```

Then run that image with the command:

```
docker run -it --rm --name pyback --network isekai -p  
5000:5000 -v $(pwd)/backend/server.py:/var/fullstack/server.py  
-v $(pwd)/frontend:/var/fullstack/frontend/ -t python_backend  
python /var/fullstack/server.py
```

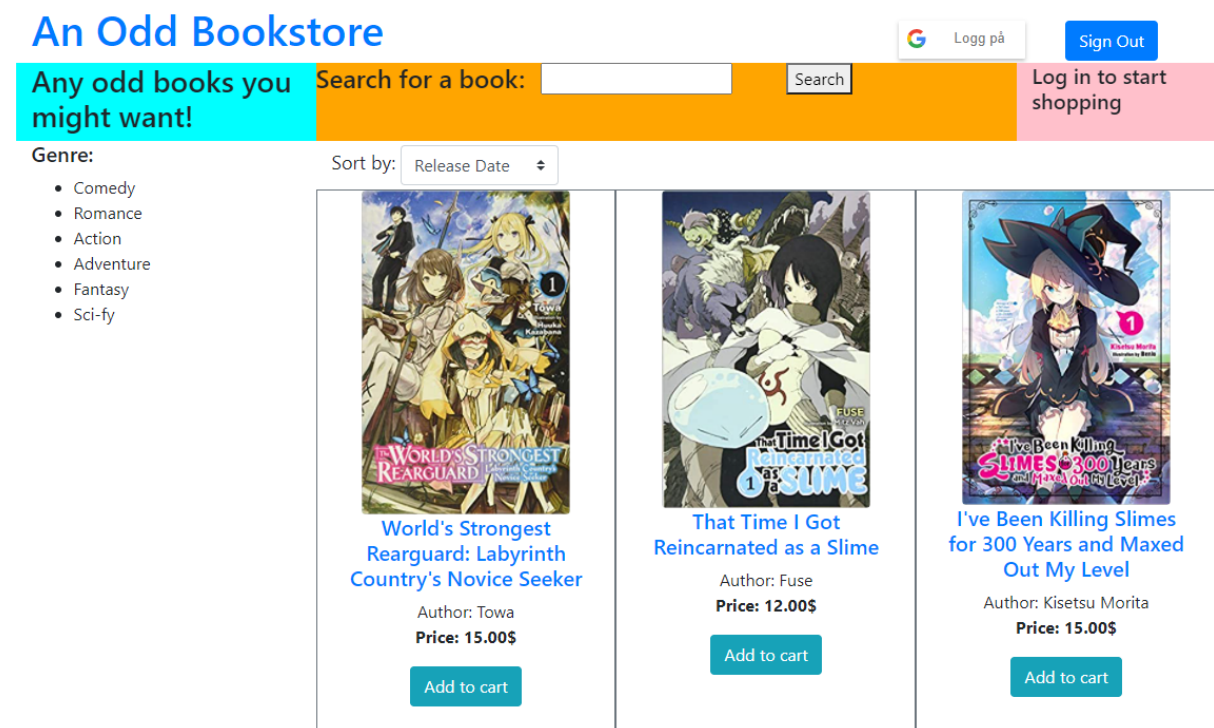
This command is ready when it says "Running on <IP address from docker> (Press CTRL+C to quit)". You can now go to <http://localhost:5000/> to check out the website.

To close the server you can do 'ctrl+c' in the Python Backend to close it. While for the MySQL window you need to 'ctrl+p' followed by 'ctrl+q', but you would then have to stop the container after to fully stop it (and it will then remove itself after that).

## How it works.

Now that you have it up and running I can explain how it all comes together. The design of the webpage did not really come together as I had to focus more on the backend stuff and messing around with many things that didn't even end up working in the end. The colors are more just to define parts of the site to easier work with so please don't mind the random coloring.

This is how the front page looks:



*(you can of course scroll to see more items I just added the first row that is there)*

Here you can see an image of the book cover for the books in the store, who they are written by, and how much they cost. When you are opening the page it will be sorted by "Release Date", as when looking for books you would usually be most interested in the newest releases, but you can also sort for title or cost. You can also search in the bar on the top for a title of a book if you are looking for something specific. You can for example search for "Slime" and you will get 2 hits. You can also use the list on the left to filter for a specific genre. Here is a minor "bug" in that you don't get a little hand when hovering. It makes sense with how I have coded it, but I never got around to fix that. However it does work, you can click on them regardless of it feeling right or not. If you try to click a "Add to cart" while not being signed in you will get an alert saying you need to sign in. I know this might lose me customers and

it is more reasonable to have signing during purchase, but it made more sense in the context of this exercise that I get to send around data to the backend instead of saving a ton of stuff in Sessions to make it easier for the user.

You can click on any book (image or title) and it will send you to a page with more info about it. And they look like this:

## An Odd Bookstore

Logg på
 [Sign Out](#)

Any odd books you might want!

Search for a book:  [Search](#)

Log in to start shopping

Genre:

- Comedy
- Romance
- Action
- Adventure
- Fantasy
- Sci-fy

**World's Strongest Rearguard: Labyrinth Country's Novice Seeker**

*Light Novel*  
 Author: Towa | *Adventure, Fantasy*

Corporate slave Arihito Atobe's death in a freak bus accident marks the beginning of his new life as a kind of adventurer called a Seeker. Reborn into a fantasy world, he settles into a previously unknown job class called "rearguard," capable of providing his (all-female) party with critical attack, defense, and recovery support. And it comes with an added bonus: Simply being at the back of the party line increases his companions' fondness for him! Freed from the shackles of corporate life, Arihito is eager to start fresh as a newly minted Seeker!

Price: 15.00\$

[Add to cart](#)

[Click here for sample pages.](#)

**Reviews:**

Profile Picture.

I think it is pretty good 4/5

Fake reviews for every book here, I can make up a lot of bullshit about these books so I feel like I am qualified for doing this nonsense, so lets get at it!

Here some missing features start appearing. You can use the Search and Genre filter ONLY on the front page. They just end up always being there so that it is a bit easier to design all the pages with similar layout. You can however always click the banner on top to get to the front page. You can on this page see a slightly bigger image of the frontpage. A description of the book. This is fairly standard practice for bookstores that the text that is in the description is the same as what is written on the back of the cover of the book. I discussed with the lecturer in this course that it should be fine using this as it is not used for commercial purposes nor to entertain a large audience and therefore goes under "fair use". I will however link to all the Amazon pages for these books where I have been collecting this info from at the bottom of this document. Another feature missing that I wanted to add was a "Reviews" part, but I never got around to making that. It should be fairly easy by adding a `<textarea>` at the bottom of this page where the placeholder review is right now and send that text with the user-email which is what I am saving to do most of these calls. And have a `<div class="row" id="reviews">` over that textarea to print all

reviews in a format like the placeholder. This will however all add another table in the database as I would need a connection between Books and Users, which would look something like “Reviews: \*book\_id, \*user\_id, review\_text, score”. If you are logged in you can use the “Add to cart” button, but if you are not you are still getting an alert about needing to log in.

To be able to do more than just browsing items like what has been done up until now you need to log in. Do that by clicking the “Logg på” or “Log in” depending on your language settings and you will be prompted by a standard Google Sign-in window where it asks for use of basic info. When you sign in you will get a “My Orders” button on top of the page. Since you are a new customer here it will not have anything in it other than a line saying “You have no past orders” for now. You will now also get a shopping cart. In the pink square top right it no longer says you have to log in, but how many items are in your cart, as well as a checkout button. This line on top will stick to the page and follow you as you scroll down the site so you can always keep track of how many things you have in your cart. Now pressing the “Add to cart” button on any page whether that is the front page or a product page it will increase the number of items in your cart.

Clicking this checkout button brings you to a page where you can see everything in your cart.

## An Odd Bookstore

My Orders

Signed In

Sign Out

Any odd books you might want!

Search for a book: 

Search


Items in cart: 2

Checkout

Genre:

- Comedy
- Romance
- Action
- Adventure
- Fantasy
- Sci-fy

Current Items in Cart:



**That Time I Got Reincarnated as a Slime**


Author: Fuse

Light Novel

Remove from cart

\$12.00

---



**I've Been Killing Slimes for 300 Years and Maxed Out My Level**

Author: Kisetu Morita

Light Novel

Remove from cart


\$15.00

---

Total Price: \$27.00

Delivery Address:

Credit Card Nr.:

This one obviously does nothing. 

Confirm Purchase

You can here see the total price of everything you have in your cart (and you also have a 1 in 1000 chance of getting everything for free!). You can click on the “Remove from cart” button if there is an item you regret putting in there. It will not necessarily remove that specific instance of that item in case you want to buy 3 copies of the same book, but it will remove 1 instance of that book. You can now insert the address you want your books delivered to and press “Confirm Purchase” to archive this order. The credit card field does nothing, but I felt like it was needed there for aesthetics, you don’t need to enter anything here. I imagine that you could have card number and the expiration date there and then send that to some third party site for money transactions, but that is not a part of this course. When you click “Confirm Purchase” you will get an alert thanking you for your purchase, and the cart will be emptied, and archived over to the “My Orders” page.

My Orders looks like this:

## An Odd Bookstore

[My Orders](#)
[Signed In](#)
[Sign Out](#)

Any odd books you might want!

Search for a book:  [Search](#)

Items in cart: 0 [Checkout](#)

**Genre:**

- Comedy
- Romance
- Action
- Adventure
- Fantasy
- Sci-fy

**Your past orders:**

Order Number: 1

Delivery Address: Dummy Road 5

	<p><b>That Time I Got Reincarnated as a Slime</b></p> <p>Author: Fuse Light Novel</p>	\$12.00
	<p><b>I've Been Killing Slimes for 300 Years and Maxed Out My Level</b></p> <p>Author: Kisesu Morita Light Novel</p>	\$15.00

Total Price: \$27.00

Order Number: 2

Delivery Address: Dummy Road 5

	<p><b>A Sister's All You Need.</b></p> <p>Author: Yomi Hirasaka Light Novel</p>	\$12.90
	<p><b>The Asterisk War</b></p> <p>Author: Yuu Miyazaki Light Novel</p>	\$15.00

It will print out all the orders you have done (I added here another one to what already shown). The orders have their Order Number, though this should maybe have something more interesting than an incremental number starting at 1, but it is the correct order ID. You can see all the books you bought in that order. What address it was sent to, and the total price of that purchase (it does not show on the last one as there is another book after that one in that order).

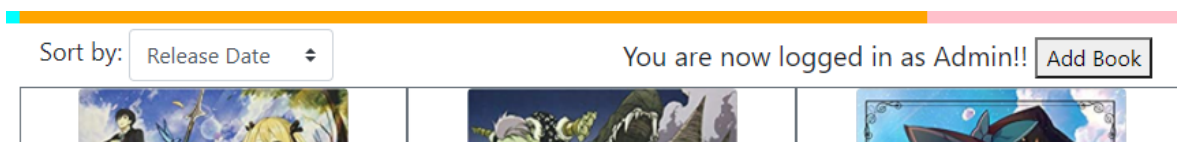
That is all from a user standpoint. You can now click “Sign Out” in the top right corner and log in as an admin using this user:

Username: [odds.dummy@gmail.com](mailto:odds.dummy@gmail.com)

Password: MySuperHardPassword!1

(If you can’t get this user to work then go into database.sql and in line 63 add your own user there with the last number being 100 as that is the access level of admins)

You are now in Admin Mode. This is clear as it says on the top of the list of books:



And all books now have an “Edit” button under the “Add to cart” button.

Let’s start with the “Edit” button. There can always be mistakes when entering a lot of text, or maybe there are sales, which leads to there being a need for edits.

*Go to the next page. I don’t want text and images on separate pages.*



The edit page looks like this:

## An Odd Bookstore

[My Orders](#)
[Signed in](#)
[Sign Out](#)


Any odd books you might want!

Search for a book:  [Search](#)

Items in cart: 0 [Checkout](#)

Genre:

- Comedy
- Romance
- Action
- Adventure
- Fantasy
- Sci-fy



Upload Image:

[Choose File](#) No file chosen

[Upload Image](#)

*You can not upload an image when making a new item as it does not have an ID yet. After creating it go into its edit page to add image.*

Title:

World's Strongest Rearguard: Labyrinth Country's Novice Seeker


Format:

Light Novel

Volume:

1

Release Date:

14-Feb-2020 

Author:

Towa

Genre:

Adventure, Fantasy

Synopsis:

Corporate slave Arihito Atobe's death in a freak bus accident marks the beginning of his new life as a kind of adventurer called a Seeker. Reborn into a fantasy world, he settles into a previously unknown job class called "rearguard," capable of providing his (all-female) party with critical attack, defense, and recovery support. And it comes with an added bonus: Simply being at the back of the party line increases his companions' fondness for him! Freed from the shackles of corporate life, Arihito is eager to start fresh as a newly minted Seeker!

Price:

15.00

[Confirm](#)

You can change any field and then press “Confirm” on the bottom to update the book info. The release date is an easy calendar date picker, and Volume and Price can only be numbers. You can also choose a file and click “Upload Image” to update the book’s image. This is useful for when you want to add a book that has yet to be released, but you want out for pre-orders, but you still don’t have the official art yet usually when a book is announced and you have to stick with a default image for a while.

Now for “Add Book”. It is basically the same page, but you don’t have any text entered, and the image is the default image. Here there is a bug however. I could not figure out how to send both a JSON object with the text info as well as a FormData object with the image file to the backend at the same time. Which is why “Confirm” handles the text on the page and “Upload Image” handles the image file. This results in that you can’t add an image when making a new book right away because the

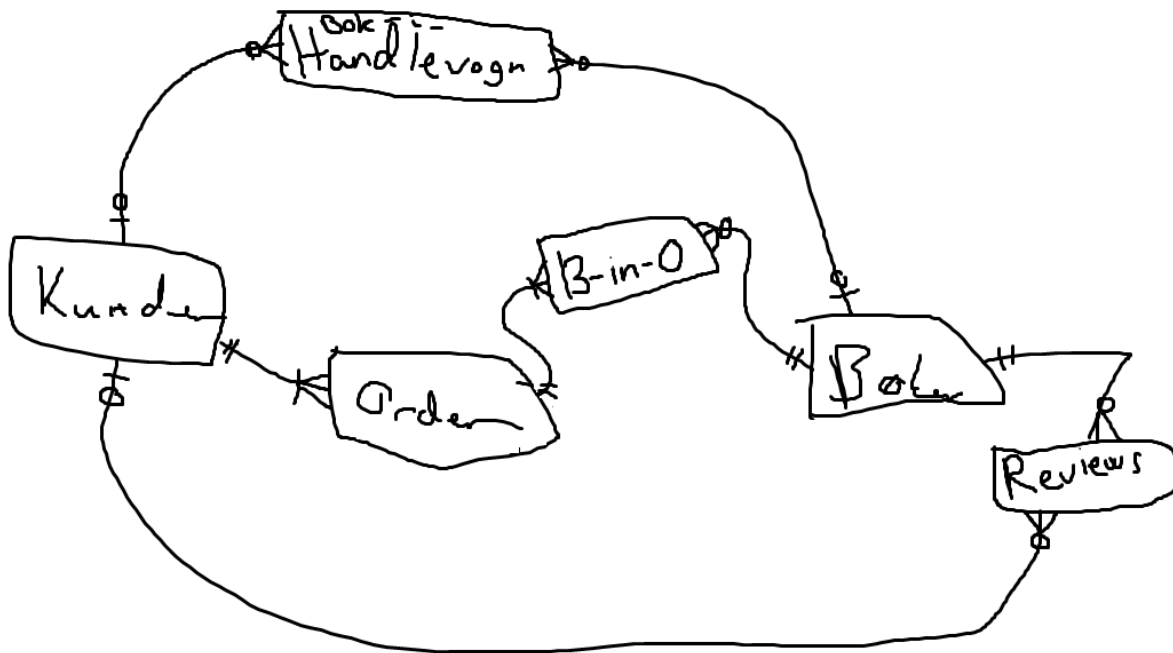


book doesn't have an ID yet since there is no info about the book in the database at this point, and no way for JavaScript to have any idea what the ID of the book should be when loading in the page (though I could ofc "guess" it by sending `SELECT MAX book_id + 1`, but that would not work if multiple people where to add books at the same time). So after you have added all the text info about the book you click "Confirm" then GO BACK TO THE FRONT PAGE, and then find the book there (you can ofc search in the search bar) and press "Edit". Now you can add an image to this book because it was now able to load the `book_id` so it knows where to send the image. There is a warning on the web page reminding you.

And that is it. That is all the functionality that I was able to make on my own and this took quite some time so there are a few things I didn't have time to get working which I want to touch on in the next paragraph. I have Format and Volume that are not used. I intended to have both Light Novels and Mangas on the site, but I didn't have time to add more books into the store and this would then be 2 separate categories in the left menu where it now says "Genre" it would say "Light Novel" and then in a list below say "Mangas" with another set of genres, and clicking those would then also sort on formats as these are 2 very different formats (one is a relatively short novel (average 200 pages) and the other is a black and whitecomic book, but Amazon frequently messes this up. And the Volumes was supposed to be a way to have a "drop down" or just a link to a list of products like the front page with all the volumes of that same series. One more "mistake" I made was that I didn't make a Genre table in the database and a "`books_genres`" and have books look up there. This made it so I can't "fetch" what genres I have in the store. If I had done that I could have made the "Genre" list far more accurate than it is now as it is just hardcoded what is there (though the links are fairly dynamic and can easily be added more of them), but I didn't have time to fix this.

## ER-Diagram sketch for the database

My Database layout is something like this:



But this is just taken from an early sketch, and as you know the Reviews never made it in.

And below is the list of items I took from Amazon.

### References:

Sword Art Online:

[https://www.amazon.com/gp/product/0316371246?pf\\_rd\\_r=VQP26ZZB16PJHQR30S1J&pf\\_rd\\_p=6fc81c8c-2a38-41c6-a68a-f78c79e7253f&pd\\_rd\\_r=ae5facf2-d854-44f4-9fe5-7f4b662167a1&pd\\_rd\\_w=osOYA&pd\\_rd\\_wg=zda4y&ref=pd\\_gw\\_unk](https://www.amazon.com/gp/product/0316371246?pf_rd_r=VQP26ZZB16PJHQR30S1J&pf_rd_p=6fc81c8c-2a38-41c6-a68a-f78c79e7253f&pd_rd_r=ae5facf2-d854-44f4-9fe5-7f4b662167a1&pd_rd_w=osOYA&pd_rd_wg=zda4y&ref=pd_gw_unk)

That time I got Reincarnated as a Slime

[https://www.amazon.com/That-Reincarnated-Slime-light-novel/dp/0316414204/ref=sr\\_1\\_2?crid=1THDLOBQT9YLK&dchild=1&keywords=that+time+i+got+reincarnated+as+a+slime&qid=1621804788&s=books&prefix=that+time+l%2Cstripbooks-intl-ship%2C227&sr=1-2](https://www.amazon.com/That-Reincarnated-Slime-light-novel/dp/0316414204/ref=sr_1_2?crid=1THDLOBQT9YLK&dchild=1&keywords=that+time+i+got+reincarnated+as+a+slime&qid=1621804788&s=books&prefix=that+time+l%2Cstripbooks-intl-ship%2C227&sr=1-2)

I've Been Killing Slimes for 300 Years and Maxed Out My Level, Vol.1

[https://www.amazon.com/Killing-Slimes-Years-Maxed-Level/dp/0316448273/ref=sr\\_1\\_3?dchild=1&keywords=i+have+been+killing+slimes+for+300&qid=1621804818&s=books&sr=1-3](https://www.amazon.com/Killing-Slimes-Years-Maxed-Level/dp/0316448273/ref=sr_1_3?dchild=1&keywords=i+have+been+killing+slimes+for+300&qid=1621804818&s=books&sr=1-3)

Asterisk War

[https://www.amazon.com/Asterisk-war-Encounter-Fiery-Princess/dp/0316315273/ref=sr\\_1\\_2?dchild=1&keywords=asterisk+war&qid=1621804856&s=books&sr=1-2](https://www.amazon.com/Asterisk-war-Encounter-Fiery-Princess/dp/0316315273/ref=sr_1_2?dchild=1&keywords=asterisk+war&qid=1621804856&s=books&sr=1-2)

The World's Strongest Rearguard: Labyrinth Country's Novice Seeker,

[https://www.amazon.com/Worlds-Strongest-Rearguard-Labyrinth-Countrys/dp/1975331540/ref=sr\\_1\\_4?dchild=1&keywords=world+strongest+rearguard&qid=1621804875&s=books&sr=1-4](https://www.amazon.com/Worlds-Strongest-Rearguard-Labyrinth-Countrys/dp/1975331540/ref=sr_1_4?dchild=1&keywords=world+strongest+rearguard&qid=1621804875&s=books&sr=1-4)

A Sister's All You Need.

[https://www.amazon.com/Sisters-Need-Vol-light-novel/dp/1975326423/ref=sr\\_1\\_9?dchild=1&keywords=a+sisters+all+you+need&qid=1621804917&s=books&sr=1-9](https://www.amazon.com/Sisters-Need-Vol-light-novel/dp/1975326423/ref=sr_1_9?dchild=1&keywords=a+sisters+all+you+need&qid=1621804917&s=books&sr=1-9)