

DBs & Dragons



***Elias Pinno
Ivan Carvalho
Rick Feng
Sean Roarty***

12.06.2020

1.1 Mission Statement

DBs & Dragons - A one-stop-shop for digital game streaming. We enable customers to play the latest-and-greatest PC games - without the need for an expensive and bulky gaming PC - by leveraging the power of the cloud. We believe gamers should be able to play whatever they want, whenever they want, on any device they want.

1.2 Purpose of Document

This document showcases the feature-set and design of DBs & Dragons. Additionally, we provide a walkthrough of both administrator and user experience and functionality.

1.3 Executive Summary

DBs & Dragons is an online digital game streaming marketplace where rather than purchasing a digital code for a game, customers pay for access to our cloud infrastructure. The game is run on our servers on-demand, and a low-latency video stream is rendered and served to the user through their web-browser.

2. System Features

Final Overall System Features	Category	Possible Points	Points	Comments / Code Files
Documentation (10 total points)				
Executive Summary (up to 1 page)	Core	2	2	
System Features (2 pages)	Core	2	2	
Walkthrough with screenshots (up to 10 pages)	Core	6	6	
Hosting (10 total points)				
Running on cloud server	Advanced	10	10	http://dragonstore.hopto.org/ (Deployed automatically by Github Actions, see .github/workflows/main.yml)
Main/Home Page (up to 14 total points)				
Search for a product by name	Core	1	1	listprod.js / listprod.handlebars
Browse products by category	Core	1	1	listprod.js / listprod.handlebars
List products (by search/browse)	Core	1	1	listprod.js / listprod.handlebars
List products with image	Core	1	1	listprod.js / listprod.handlebars
Page header with menu	Basic	1	1	banner.handlebars / main.handlebars
Page header shows current logged in user	Basic	1	1	banner.handlebars
Dynamic products on page based on sales	Basic	2	2	listprod.js/ listprod.handlebars
Improved UI (user-friendly, images, etc.)	Basic	2	2	
Product recommendation based on user	Advanced	4		
Shopping Cart (up to 8 total points)				

Add to shopping cart	Core	1	1	addcart.js / cart_manager.js
View shopping cart	Core	1	1	showcart.js / showcart.handlebars
Update quantity (with data validation)	Basic	1	1	updatecart.js / cart_manager.js
Remove item from shopping cart	Basic	1	1	removecart.js / cart_manager.js
Improved formatting/UI (e.g. in header bar)	Basic	1	1	banner.handlebars / style.css
Cart stored in database between sessions	Basic	3	3	cart_manager.js

Checkout (up to 9 total points)

Checkout with customer id	Core	1	1	order.js / checkout.js
Checkout with payment/shipment info	Basic	2	2	validShipment.js / validPayment.js
Checkout with data validation	Basic	2	2	validShipment.js / validPayment.js
Multiple shipments per order supported	Advanced	2		Note: Requires DB change.
Calculate taxes/shipping (by items/state)	Advanced	2		Note: Requires DB change.

Product Detail Page (up to 2 total points)

Product detail page and item description	Core	1	1	product.js / product.handlebars
Product detail has an image from database	Core	1	1	product.js / displayImage.js

User Accounts/Login (up to 12 total points)

Create user account page	Basic	2	2	createuser.js / createuser folder in views
Create account with data validation	Basic	2	2	createuser.js / validation.js
Edit user account info (address, password)	Basic	2		
Login/logout	Core	1	1	validateLogin.js
Register user/Forgot password email	Advanced	4		
Page listing all orders for user	Core	1	1	customer.js / customer/orders.handlebars

Product Reviews (up to 5 total points)

Ability to enter a review on a product	Basic	2	2	review.js / product.handlebars
Display product review on product detail page	Basic	1	1	product.js / product.handlebars
Restrict to one review per user on item purchased	Advanced	2	2	review.js / Only_One_Review constraint in data.ddl

Warehouses/Inventory (up to 5 total points)

Display item inventory by store/warehouse	Basic	2		
Edit item inventory by store/warehouse	Advanced	3		

Administrator Portal (up to 19 total points)				
Secured by login	Core	1	1	admin.js
List all customers	Core	1	1	admin.js / admin/index.handlebars
List report showing total sales/orders	Core	1	1	admin.js / admin/index.handlebars
Report with a graph	Advanced	3		
Add new product	Basic	2	2	admin.js / manageinsert.handlebars
Update/delete product	Basic	2	2	admin.js / manageupdate.handlebars / managedelete.handlebars
Change order status/ship order	Basic	1		
Upload a photo to file system for product	Advanced	2		
OR: Upload a photo to database for product	Advanced	4		
Database restore with SQL script	Basic	2		
Add/update warehouse, customer	Basic	2		
Database System/General				
Implement some validation using triggers	Advanced	2		
Use AJAX for some pages	Advanced	4		
User interface and navigation/usability		Up to 5		
<i>Others (your suggestions)</i>		Up to 5		
Total: (out of 50 with maximum of 10 bonus marks. i.e. 60 points out of 50 is max.)			61	

3. Walkthrough

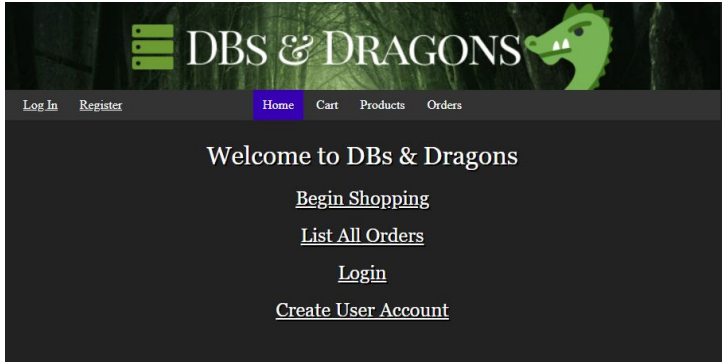

Please allow us to walk you through DBs & Dragons. To access our site, please use the info provided below.

ACCOUNT TYPE	USERNAME	PASSWORD
Administrator	bobby	bobby
User	arnold	test
Cloud-hosted Site	http://dragonstore.hopto.org/	

NOTE: If for any reason you wish to restore the database to its defaults, simply click this link:
<http://dragonstore.hopto.org/loaddata>

IF RUNNING LOCALLY, APPENDING '/loaddata' TO THE END OF THE LOCALHOST URL IS REQUIRED TO INITIALIZE THE DATABASE!

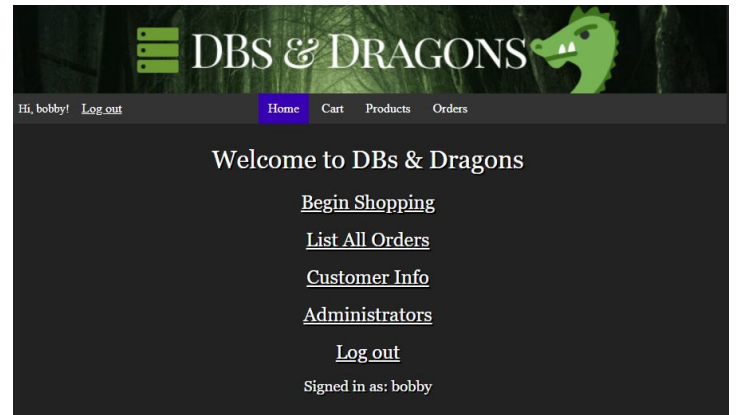
Ex: <http://127.0.0.1/loaddata>

<p>Welcome to the DBs& Dragons home page!</p> <p>Our site is built using NodeJS, and a dynamic HTML templating system called Handlebars which allows us to use to extend JavaScript variables and logic into our DOM. From here, we can go anywhere on the site. Please click "Log In" either in the header or on the homepage itself.</p>	
<p>Here, we authenticate the user through validateLogin.js and prompt the user to re-enter their information if there's an authentication error. We'll log on as an administrator, and explore the functionality.</p>	

Here is our administrator home-page. From here, we can:

- View all of the orders in our system.
- View sales reports.
- Delete, insert products in our database.

Let's click "List All Orders" and have a look.



In the back-end, listorder.js queries our database for all customer orders and passes that information to listorder.handlebars where it is rendered. Generally speaking, the front-end of our website is built upon these js/handlebar pairs.

Let's click "Home" in the header menu, then the "Administrators" button on the home-page.

Order Id	Order Date	Customer Id	Customer Name	Total Amount
1	Tuesday, October 15, 2019, 10:25:55 AM	1	Arnold Anderson	\$91.70
		Product Id	Quantity	Price
		1	1	\$18.00
		5	2	\$21.35
		10	1	\$31.00
2	Wednesday, October 16, 2019, 06:00:00 PM	2	Bobby Brown	\$106.75
		Product Id	Quantity	Price
		5	5	\$21.35
3	Tuesday, October 15, 2019, 03:30:22 AM	3	Candace Cole	\$140.00
		Product Id	Quantity	Price
		6	2	\$25.00
		7	3	\$30.00
4	Thursday, October 17, 2019, 05:45:11 AM	2	Bobby Brown	\$327.85
		Product Id	Quantity	Price
		3	4	\$10.00
		8	3	\$40.00
		13	3	\$23.25
		28	2	\$21.05
		29	4	\$14.00
5	Tuesday, October 15, 2019, 10:25:55 AM	5	Elizabeth Elliott	\$277.40
		Product Id	Quantity	Price
		5	4	\$21.35
		19	2	\$81.00
		20	3	\$10.00

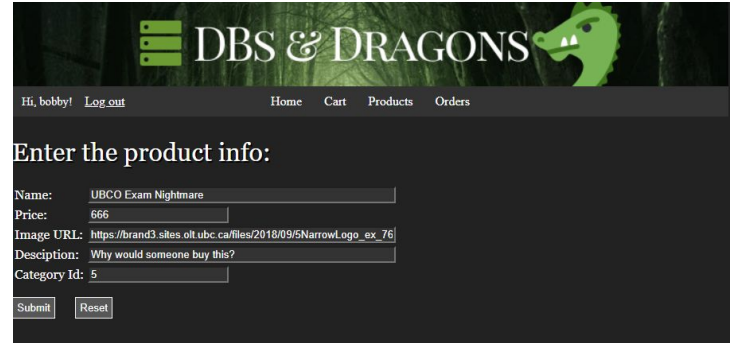
Here is the main administrator page, We're given a summary of sales data categorized by date and customer-name (which is processed in admin.js). From this page, we can also insert, delete, and update products in our database. Let's click "Insert" and add a new game.

Order Date	Total Order Amount
Thursday, October 17, 2019	\$327.85
Wednesday, October 16, 2019	\$106.75
Tuesday, October 15, 2019	\$509.10

Customers	Number of orders
Arnold Anderson	1
Bobby Brown	2
Candace Cole	1
Elizabeth Elliott	1

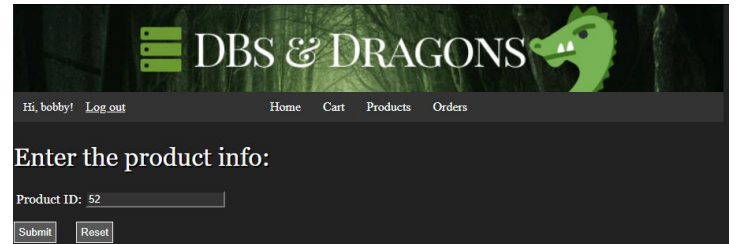
Total order so far	Revenue
5	\$943.70

Here, we can add “UBCO Exam Nightmare” to our list of products, and admin.js will perform the logic to insert it into our database. Similarly, we can also update an existing product, or delete it altogether. Let’s go back to the administrator page and remove this bad game.



The screenshot shows the top navigation bar with the logo and a dragon head. Below it, a user is logged in as 'bobby'. The main heading is 'Enter the product info:'. The form contains the following fields: Name (UBCO Exam Nightmare), Price (666), Image URL (https://brand3.sites.olt.ubc.ca/files/2018/09/5Narrow1.ogo_ex_76), Description (Why would someone buy this?), and Category Id (5). There are 'Submit' and 'Reset' buttons at the bottom.

Our database has an auto-incrementing productid which serves as our primary key for any products being sold. Here, we simply enter the id for our new and unpopular game and correct our mistake. If the product exists, it will be removed.



The screenshot shows the same navigation bar and heading as the previous screenshot. The form now has a single field for 'Product ID: 52' and 'Submit' and 'Reset' buttons.

Next, we’ll have a look at the user experience. Let’s log out of our admin account, and click “Create User Account” on the home page.

Here, we can fill in all of our details, and createuser.js will validate our information on the back-end. If a user enters invalid information, an error will be printed indicating which fields are invalid.



The screenshot shows the user registration form. The navigation bar includes 'Log In' and 'Register' links. The form fields are: Userid (JonDoe), Email (jondoe@fake.net), First Name (Jon), Last Name (Doe), Phone Number (optional) (555-555-5555), Address (12345 Fake St), Country (Canada), Province / State (British Columbia), City (Vancouver), Postal Code (V2X 0N1), Password (masked with asterisks), and Confirm Password (masked with asterisks). There are 'Submit' and 'Reset' buttons at the bottom.

Let’s create a Jon Doe, log in with our new credentials, and click the “Products” tab so we can start buying some games!

This is our “List Products” page, which is served up by listprod.js. The games are ordered by the number of copies sold. Let’s click on “Hades” and see if it’s something we’re interested in.

Hi, jondoe! [Log out](#) [Home](#) [Cart](#) [Products](#) [Orders](#)

Search products:

All

All Products

	Product Image	Product Name	Category	Price	Number sold
<input type="button" value="Add"/>		Cyberpunk 2077	Action RPG	\$79.99	0
<input type="button" value="Add"/>		NBA 2K21	Sports	\$79.99	0
<input type="button" value="Add"/>		Hades	Roguelike	\$28.99	0
<input type="button" value="Add"/>		Yakuza: Like a Dragon	RPG	\$79.99	0

Our product page (served up by product.js) shows a promotional image of the game, a short description prepared by the developers, and reviews by other customers who have previously purchased it.

Bobby seemed to like the game, although he gave it one star. Let’s click “Add to Cart” and buy it.

Hi, jondoe! [Log out](#) [Home](#) [Cart](#) [Products](#) [Orders](#)

Hades

Price: \$28.99
ProductID: 3

Defy the god of the dead as you hack and slash out of the Underworld in this rogue-like dungeon crawler from the creators of Bastion, Transistor, and Pyre.

Rating	Date	Comment	Reviewer
1	December 7, 2020, 05:39 AM	THIS GAME IS GREAT I LOVE VIDEO GAMES SO SO MUCH!	Bobby

Comment:

The shopping cart, served up by showcart.js, shows us the games we’ve added to our cart. Here, we can update the quantity (via updatecart.js), and we can check out to complete our purchase.

Let’s check out and buy our game.

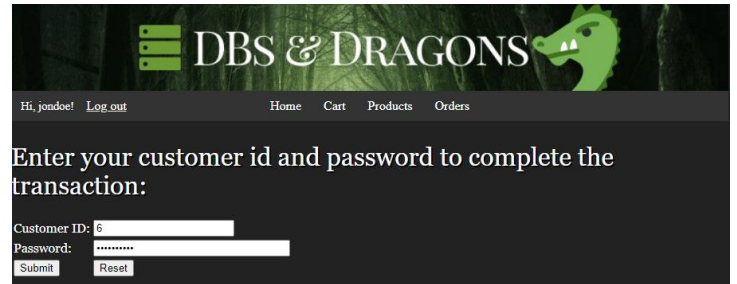
Hi, jondoe! [Log out](#) [Home](#) [Cart](#) [Products](#) [Orders](#)

Your Shopping Cart

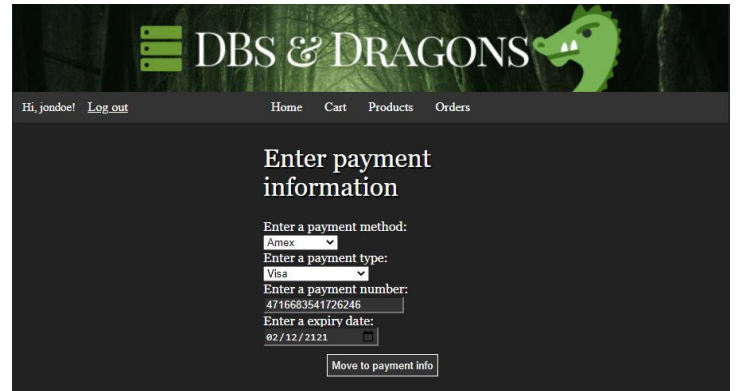
Product id	Product Name	Quantity	Price	Subtotal	
3	Hades	<input type="text" value="5"/>	\$28.99	\$144.95	<input type="button" value="Remove"/> <input type="button" value="Update Quantity"/>
				Order Total	\$144.95

Here, we verify our identity with our customer ID and password.

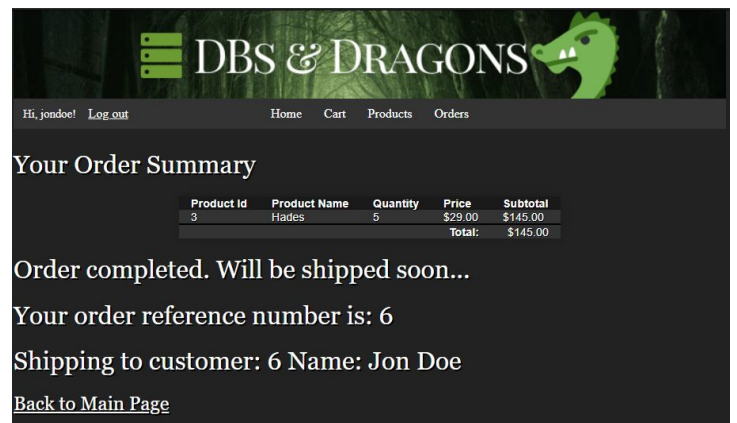
NOTE: This is not the same as the user id. Customer ID is an auto-incrementing numerical value. In the case of Jon Doe, that value is 6.



After filling in our shipping information, we provide a payment option that is validated in the back-end.



Success! Our information was validated, and we bought the game. Let's pretend we played it, and we want to leave a review. We can go back to the product page, scroll to the bottom, and let other customers know what we thought.



Product Id	Product Name	Quantity	Price	Subtotal
3	Hades	5	\$29.00	\$145.00
			Total:	\$145.00

After entering our review into the text input and submitting, we can see our newly-minted opinion. We also left a 1-star review. Huh.

Hi, jondoe! [Log out](#) [Home](#) [Cart](#) [Products](#) [Orders](#)

Hades

Price: \$28.99
ProductID: 3

9/10 MON
10/10 DICK BECKERS
10/10 THE SIXTHAXIS
10/10 WCGFTech
9.7/10 NOTRE PLAYING

[Add to Cart](#) [Continue Shopping](#)

Defy the god of the dead as you hack and slash out of the Underworld in this rogue-like dungeon crawler from the creators of Bastion, Transistor, and Pyre.

Rating	Date	Comment	Reviewer
1	December 7, 2020, 05:39 AM	THIS GAME IS GREAT I LOVE VIDEO GAMES SO SO MUCH!	Bobby
1	December 7, 2020, 06:06 AM	BOBBY IS SO RIGHT THIS GAME IS WORTHY OF ALL CAPS EASILY!!!	Jon

Comment: 1 star