Project Title: Italian Restaurant Website

Name for the Website: Mamamia

Description:
Mamamia is an Italian restaurant site that shows menu items in both lists and grid view. It has the following features:

1. Signup and Login feature with Validation checks.
2. List available items.
3. Pagination feature with 9 items showing on each page.
4. The site has the ability to search items and filter on categories, both search and filter are integrated together and have pagination as well.
5. A user can add a menu item to the shopping cart, update or delete an item in the cart, and do checkout by entering the address information.
6. Show history of purchases.
7. Admin CRUD operations with pagination and ability to delete and restore items.

Database Design:
MongoDB is used to store this website's data. There are 4 collections:

1. **Users**
   For users, only email id, password, and isAdmin fields are stored in the collection.
   Data Model: Example
   {
   "_id":{"$oid":"60863e073eab48dcc0595cce"},
   "email":"admin1@mamamia.com",
   "password":"$2b$12$K9amtF8KmIOcCdwc4uKn9uSWXO11iJelR8M1tQGjM7PKjg38TSdjm",
   "isAdmin":true
   }

2. **Menu**
   Name, category, description, price, image, isAvailable
   Data Model: Example
   {
   "_id":{"$oid":"6086052101a2f61338e4d38b"},
   "name":"Calamari Fritté",
   "category":"Appetizer",
   "description":" Fried squid with Marinara Sauce",
   "Price":"14.99",
   "Image":"6086052101a2f61338e4d38b.webp",
   "isAvailable":false
   }

3. **Cart**
   For each user an object is created in the carts collection with the following fields:
   Email, items, subtotal, tax, total

Data Model: Example

{
"_id":{"$oid":"608dcb08dda08b5362e59f88"},
"email":"anshul.pardhi@gmail.com",
"Items":[
        {
        "name":"Calamari Fritté",
        "category":"Appetizer",
        "description":" Fried squid with Marinara Sauce",
        "Price":"14.99",
        "Image":"6086052101a2f61338e4d38b.webp",
        "isAvailable":true,
        "Quantity":"2",
        "Id":"6086052101a2f61338e4d38b"
        }
        ,{
        "name":"Mozzarella Marinara",
        "category":"Appetizer",
        "description":" Breaded Mozzarella & Marinara Sauce",
        "Price":"8.99",
        "Image":"6086056d01a2f61338e4d38c.webp",
        "isAvailable":true,
        "Quantity":"1",
        "Id":"6086056d01a2f61338e4d38c"
        }],
"Total":"44.82",
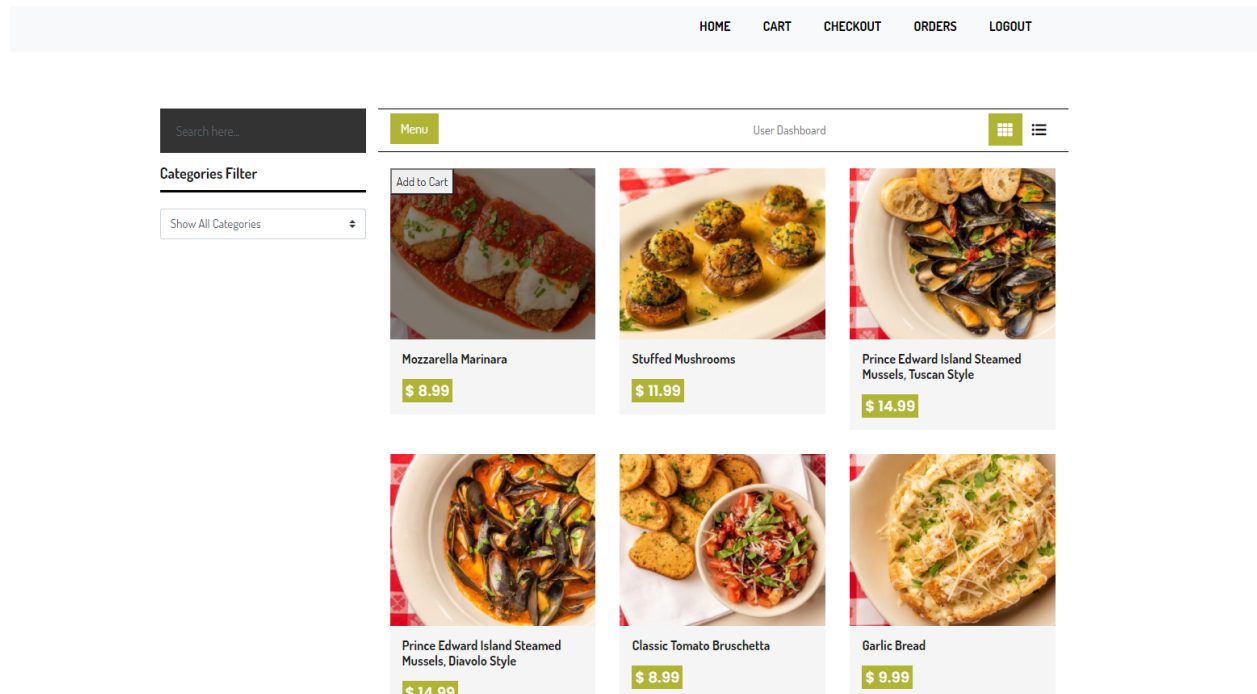"Subtotal":"38.97",
"Tax":"5.85"
}

4. **Orders**
   Email, an array of orders, each orders object contains: cart, address, date
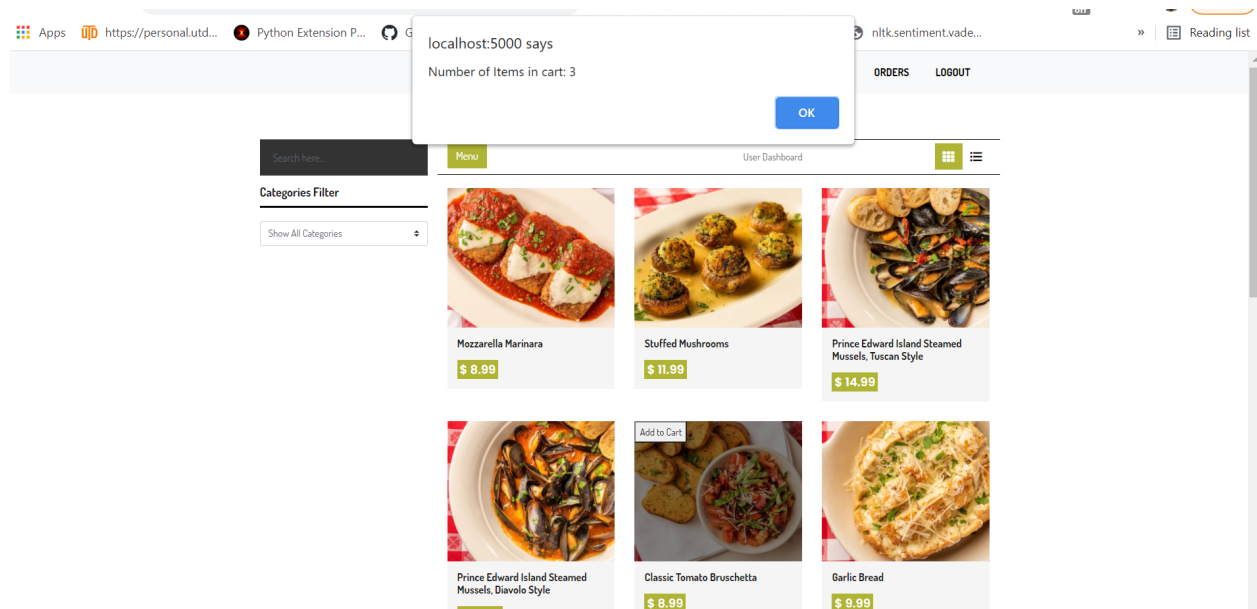   Data Model: Example

{
"_id":{"$oid":"608dd27c34f5771c33c9ba3f"},
"email":"user1@user.com",
"Orders":[
        {
        "Cart":
        {"_id":"608dd27c34f5771c33c9ba3e","email":"user1@user.com",
        "Items":[
        {"category":"Salad","description":"Grated Parmesan & Garlic
Croutons","id":"608607f301a2f61338e4d395","image":"608607f301a2f61338e4d395.web
p","isAvailable":true,"name":"Caesar Salad","price":"5.99","quantity":"2"},

{"category":"Steak","description":"Provolone & Marinara Sauce with Spaghetti Marinara","id":"60871c59cd20ef1a2d58aa46","image":"60871c59cd20ef1a2d58aa46.webp","isAvailable":true,"name":"Veal Parmesan","price":"28.99","quantity":"1"}
        ],
"Subtotal":"40.97",
"Tax":"6.15",
"Total":"47.12"
},
"Address":{
"address_1":"7575 Frankford Rd.",
"address_2":"Apt 3325",
"city":"Dallas",
"state":"Texas"
,"country":"United States",
"Zip":"75252"
},
"date":"05/01/21"
}
        ]}

Languages/frameworks used for implementation:
Languages: HTML, CSS, Javascript, Python
Framework: Flask

Screenshots:

Sign In Functionality

User Dashboard

Add to Cart functionality:

Cart feature:

S.No.	Product Name	Price	Quantity	Total	Remove
1	Prince Edward Island Steamed Mussels, Tuscan Style	14.99	1	14.99	✕
2	Classic Tomato Bruschetta	8.99	1	8.99	✕
3	Stuffed Mushrooms	11.99	2	23.98	✕

UPDATE CART

Order summary

Sub Total	$47.96
Tax	$7.19
Grand Total	**$55.15**

Checkout

Checkout feature:

Shipping address

Address *

7575 Frankford Rd.

Address 2

Apt 3325

City *

Dallas

Country *

United States

State *

Texas

Zip *

75252

Your order

Sub Total	$47.96
Tax	$7.19
Grand Total	**$55.15**

Place Order

Users Collection has hashed passwords:

_id: ObjectId("60863e073eab48dcc0595cce")
email: "admin1@mamamia.com"
password: "$2b$12$K9amtF8KmIOcCdwc4uKn9uSWXO11iJelR8M1tQGjM7PKjg38TSdjm"
isAdmin: true

_id: ObjectId("6086436f6eb8f4b3e9e470e7")
email: "anshul.pardhi@gmail.com"
password: "$2b$12$TiHia0/QN6kNsHpdTsWWFOi/AB28CF3vsFsrQ./UImeAFvVR.cCUC"
isAdmin: false

_id: ObjectId("608dd27c34f5771c33c9ba3d")
email: "user1@user.com"
password: "$2b$12$9B4rnzX3Dq2vEwl9OBKwd.7AZBotzHUSmPT9.jNXk97T1Fli5OMwW"
isAdmin: false

View Order Details:
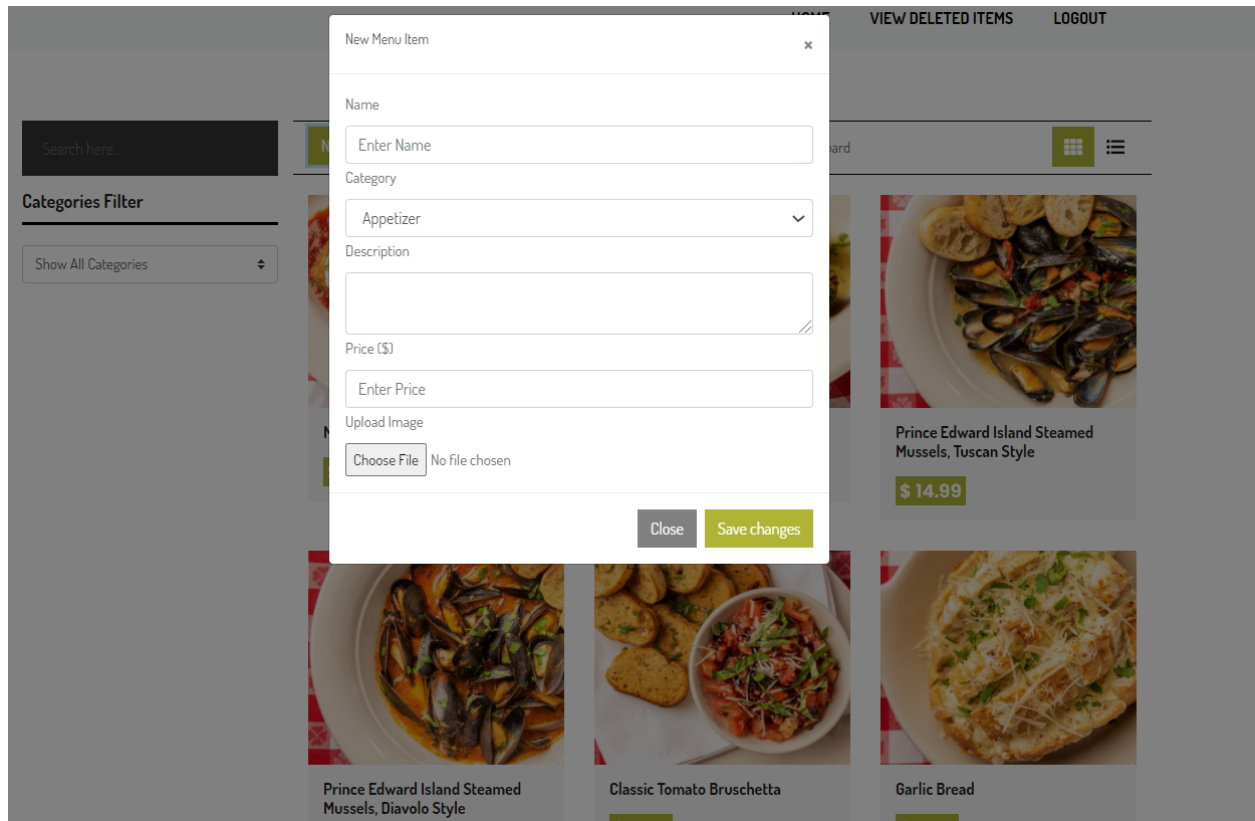
Admin Operations:

Create :

Update:

Delete/Restore Item:

Team Members:
Gunjan Agicha (gxa190010)
Anshul Pardhi (arp180012)
Kushagra Dar (kxd180025)

Work division:

1. **Database Design:**
   Kushagra: Created the MongoDB cluster on MongoDB Atlas. Also created cart and order collection.
   Anshul: Created menu collection and added 62 items in the menu with all the fields.
   Gunjan: Created users collection.
   All three were involved in discussing what collections are needed for the website and the initial design.

2. **Server-side and client-side scripting:**
   - **Signup and login feature**
     Kushagra: Created login and signup HTML pages and added validation checks to it. Created the Flask API and AJAX call for it.
     Gunjan: Contributed to password hashing before storing it in the database.
     Anshul: Found out the templates that could serve as the base to our site.
   - **List available items:**
     Anshul created the AJAX get request and handled the user dashboard and admin dashboard list and grid view functionality.
   - **Pagination:** Anshul added pagination to the user dashboard and admin dashboard.
   - **Search and filter:**
     Gunjan: created API for search and filter and changed the UI in the user dashboard and admin dashboard pages to show the categories and search field. Also requested the API via AJAX call for both search and filter and linked the response to the list and grid view in userdashboard.js and admindashboard.js
     Anshul: Helped in search and filter API, and added the pagination functionality to the results from search and filter API.
   - **Add to the shopping cart, checkout and orders**:
     Kushagra: Created the cart, checkout, and orders pages. Created the API for it and did an AJAX call to get the response and reflect on these pages.
   - **Admin CRUD Operations:**
     Anshul created the CRUD API's for the admin on the server side and consumed it on the client side using AJAX and JS.