

**Vanier College**  
**Faculty of Science and Technology**

**Application Development (Desktop)**  
**420-942-VA (sect. 01222)**

**Deliverable 3**  
**Optical Store Management System (OCMS)**  
**2023-11-13**

**Team Members:**  
**Shunzi Yao (2395028)**  
**Ana Pechonkin (0972547)**

# Table of Contents

<b>Table of Contents.....</b>	<b>2</b>
<b>Project Scope.....</b>	<b>3</b>
<b>Project Functionalities.....</b>	<b>4</b>
Functionalities for Mid-Term Evaluation:.....	4
Functionalities Added After Mid-Term Evaluation:.....	4
All Current Functionalities:.....	4
<b>Project Setup.....</b>	<b>22</b>
<b>Work Project in Future.....</b>	<b>24</b>

# Project Scope

## I. Primary users:

- **Store Managers:** To oversee store operations, employee performance, inventory levels, and financial transactions across multiple locations.
- **Optometrists:** To utilize the system for managing appointments, accessing patient records, and coordinating with the sales team for prescriptions and eyewear recommendations.
- **Sales Staff:** To access real-time inventory data, assist customers in making informed choices, and process sales transactions.

## II. Secondary Users:

- **IT and Support Staff:** Responsible for the maintenance of the system, ensuring its smooth operation, and providing technical support.
- **Marketing Team:** Utilizes customer data and purchase trends for creating targeted marketing campaigns and promotional offers.
- **Administrative Staff:** Engages with OCMS for various administrative tasks like payroll processing, employee records management, and scheduling.

## III. Beneficiaries:

- **Store Chains:** Optical store chains will benefit from enhanced operational efficiency, better inventory management, and improved customer satisfaction.
- **Optical Industry:** As a whole, the industry benefits from the implementation of advanced technology solutions like OCMS, setting new standards for service and management efficiency.

## IV. Ideal Clients for OCMS:

- **Large Optical Retail Chains:** Such as Luxottica Group (operating brands like LensCrafters, Pearle Vision), National Vision, Inc., and Vision Express. These large entities can leverage OCMS to harmonize operations across their vast network of stores.
- **Regional Optical Stores:** Chains with a regional footprint can utilize OCMS to compete effectively with larger players by enhancing their operational efficiency and customer engagement strategies.
- **Hospital-Based Optical Outlets:** Hospitals with dedicated eye care departments can integrate OCMS to streamline the management of their optical services.
- **Independent Optometrists and Small Chains:** Although smaller in scale, these businesses can benefit significantly from the customer management and inventory features of OCMS, enabling them to offer a level of service that rivals larger chains.

# Project Functionalities

## Functionalities for Mid-Term Evaluation:

1. Login Window
2. Dashboard for each user (Admin, Sales, Doctor)
3. Customer Info Page
4. Staff Info Page
5. Customer Detail Page
6. Employee Detail Page

## Functionalities Added After Mid-Term Evaluation:

1. Appointments Page
2. Appointment Detail Page
3. Inventory Page
4. Inventory Detail Page
5. Inventory with REST API
6. Invoice Page
7. Order Page
8. RestAPI\_OCMS
9. Unit Tests on RestAPI\_OCMS
10. Styling

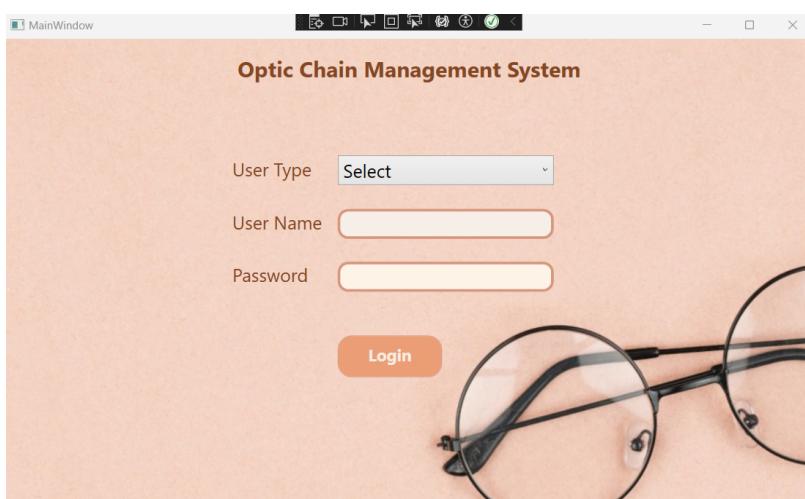
## All Current Functionalities:

### 1. Login Window

The application initiates with a login window.

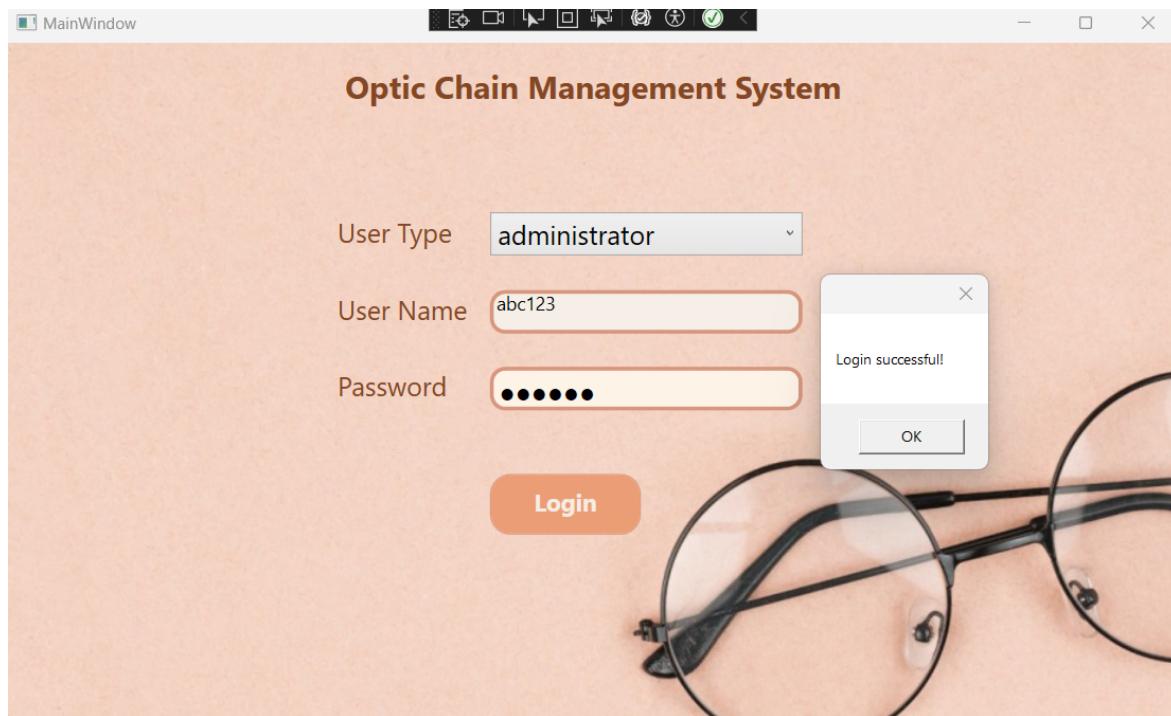
Three different user types can log in: Admin, Sales, and Doctor.

Users need to choose the user type from dropdown, enter username and password, then the system authenticates them against the data in our database.

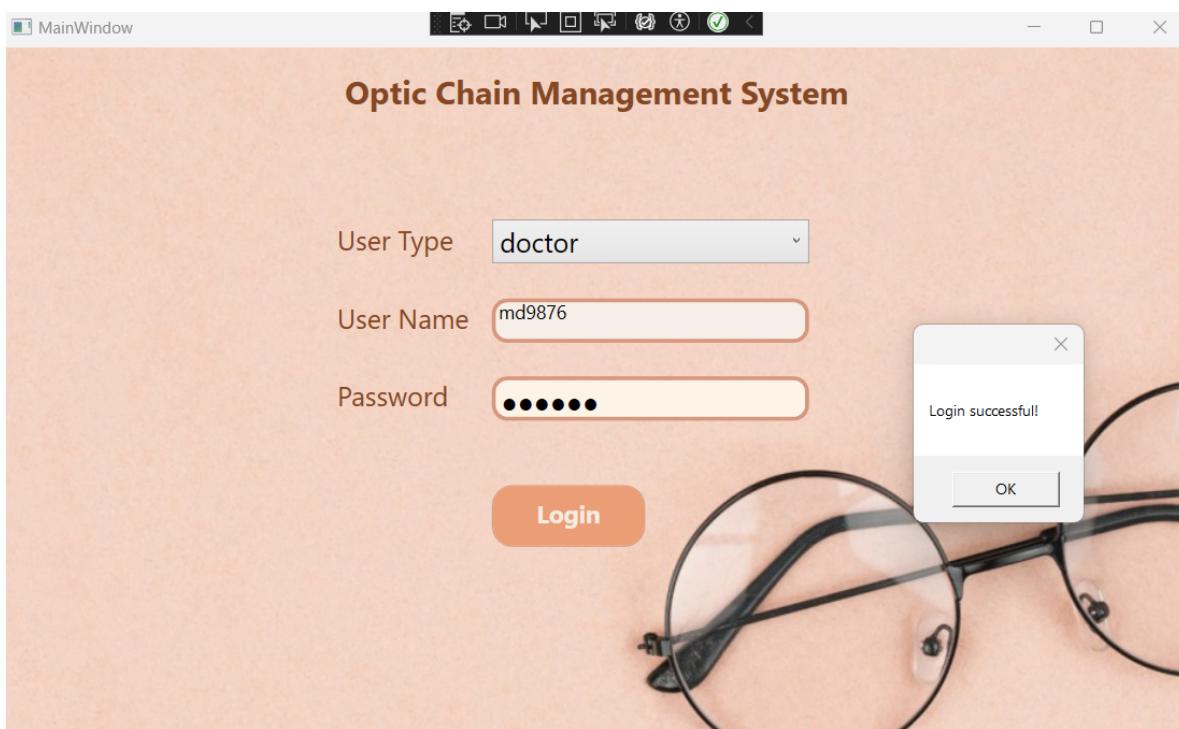


**Login Successful:** If the information is correct a message that the login was successful will pop up.

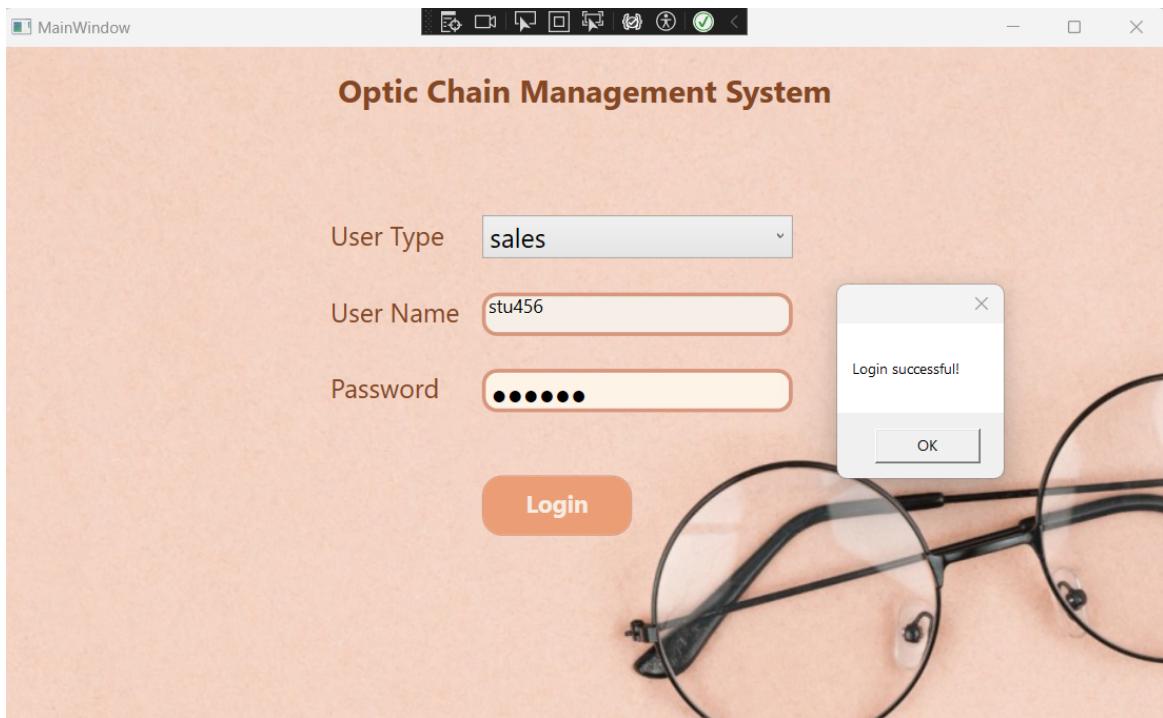
**Administrator Login:**



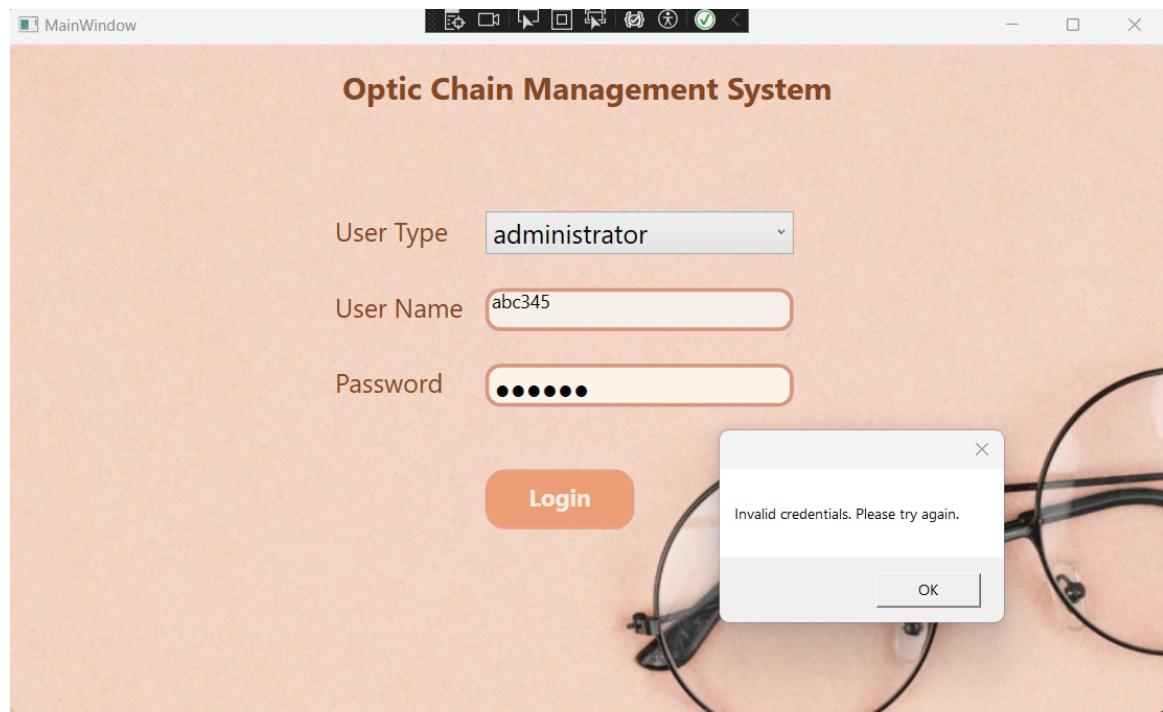
**Doctor Login:**



## Sales Staff Login:



**Login not successful:** If the information is not good then the login will not pass and the user will see a message that the login failed and to please try again.



## 2. Dashboard:

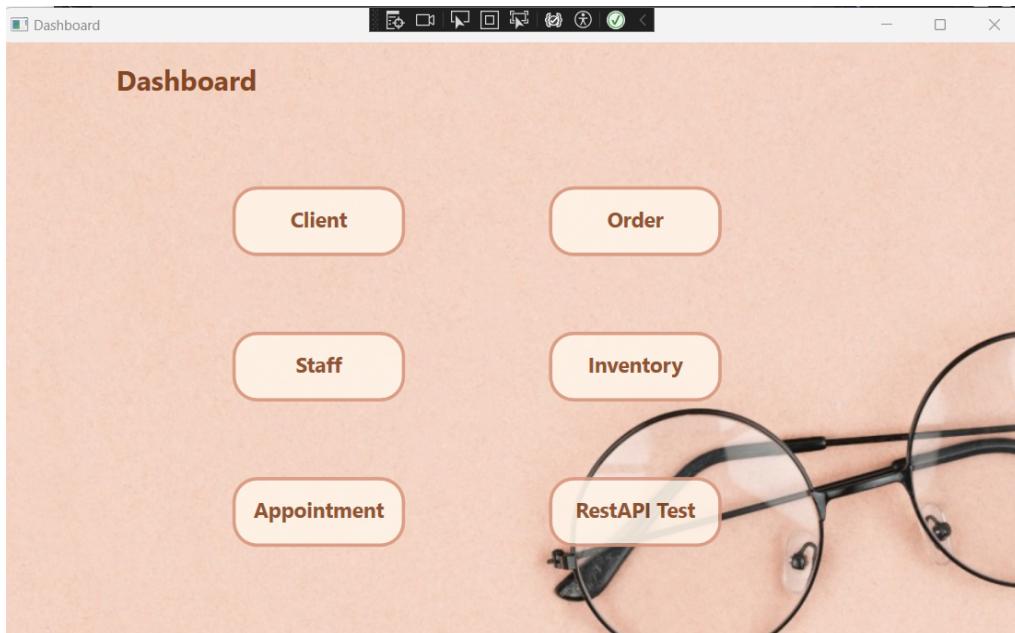
After a successful login, users are directed to the dashboard. The dashboard contains 6 buttons: Client, Staff, Appointment, Order, Inventory and REST API Test.

Different user types have different access.

**Admin:** Has access to all functionalities on the dashboard.

**Sales:** Cannot access the Staff section.

**Doctor:** Only has access to the Client and Appointment sections.



## 3. Customers Info page

- Database Integration: Customer information is fetched and displayed from the database in real time.
- Search Functionality: Users can search for details using name, phone number, or ID.
- Addition of New Entries: A 'New' button redirects to a detail page where users can add a new customer along with their detailed information.
- Double clicking on a customer: will open up the customer detail page with all of their information loaded.

The screenshot shows a Windows application window titled "CustomerInfoPage". The main title bar has standard window controls. Below the title bar, there is a header with the word "Customers" in bold, a search input field, and a "Search" button. To the right of the search area is a "New" button. The main content area is a table with columns: First Name, Last Name, DOB, Phone, Email, and Prescription. The table contains 12 rows of customer data. The background of the window features a photograph of a person's face wearing glasses.

First Name	Last Name	DOB	Phone	Email	Prescription
Joker	Jokings	11/12/2023	564-123-1234	joke@gmail.com	R:-1.25, L:-1.25
Jessie	John	11/15/2023	514-645-7652	jessie@outlook.com	
Antoinette	Lefevre	7/10/2023	514-678-2345	anto@gmail.com	R:-2.25, L:-2.75
Jennifer	Jones	11/12/2023	777-111-3456	jen@gmail.com	R:-1.25, L:-0.75
Lena	Chandra	3/20/2023	999-123-6543	lena@hotmail.com	R:-5.00, L:-3.75
Sasha	Williams	11/11/1990	438-222-5555	sashaw@hotmail.com	R: +1.25, L: +1.25
Mohammed	Ali		514-777-1234	mohammed.ali@gmail.com	
Fatima	Chen	8/20/1985	438-888-5555		
Juan	Garcia	11/2/1978	450-222-9999	juangarcia@hotmail.com	
John	Johnson	2/1/1971	438-333-4444	johnjohnson@hotmail.com	R: -0.25 -1.00 X100, L: -0.50
Jake	White		514-111-7777	jake.white@yahoo.com	

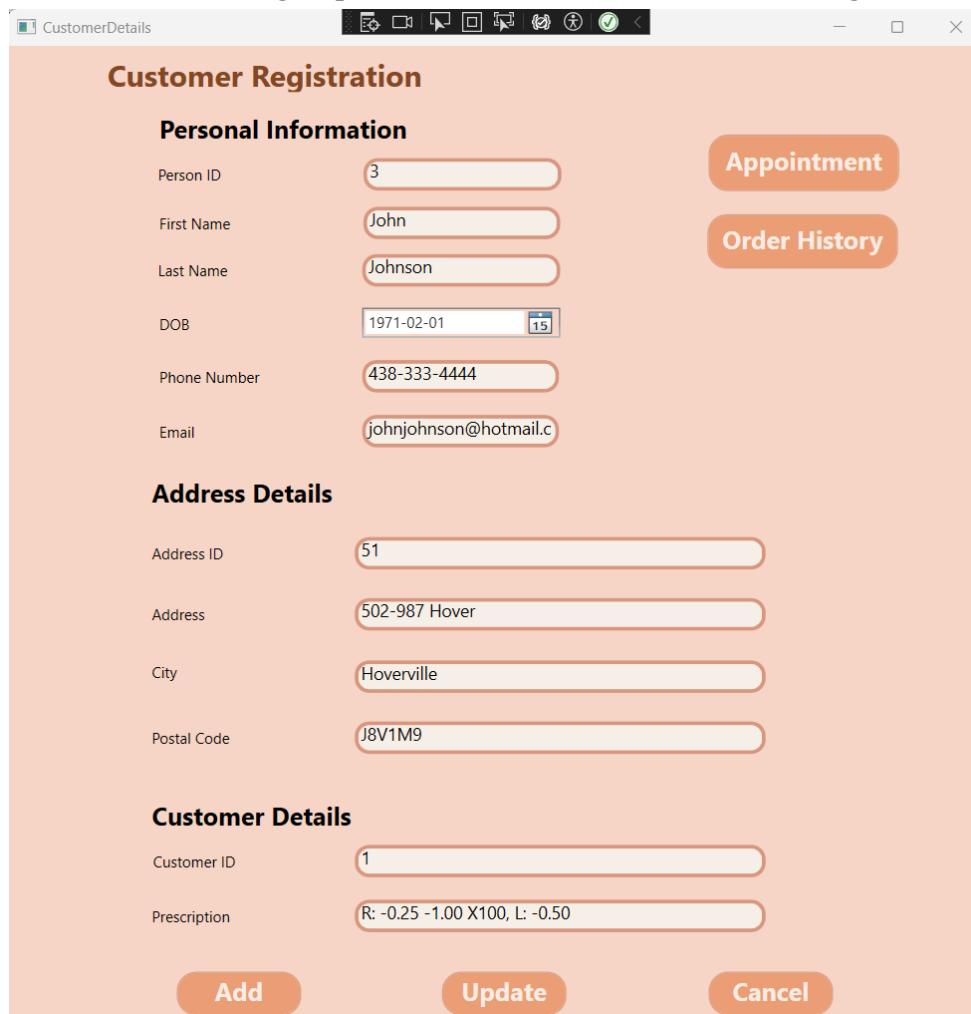
#### 4. Customer Detail Page

Users can add and update customer information, this information will be saved in the database. The appointment button will bring the user to the appointment page, where they can see all appointments or search for a specific one. The order history button will bring to the order list where the user can search through all client purchases.

#### When Clicking on New Button on the Customer Info Page:

The screenshot shows a Windows application window titled "CustomerDetails". The main title bar has standard window controls. Below the title bar, there is a header with the word "Customer Registration" in bold. The form is divided into several sections: "Personal Information" (Person ID, First Name, Last Name, DOB, Phone Number, Email), "Address Details" (Address ID, Address, City, Postal Code), and "Customer Details" (Customer ID, Prescription). At the bottom of the form are three buttons: "Add", "Update", and "Cancel".

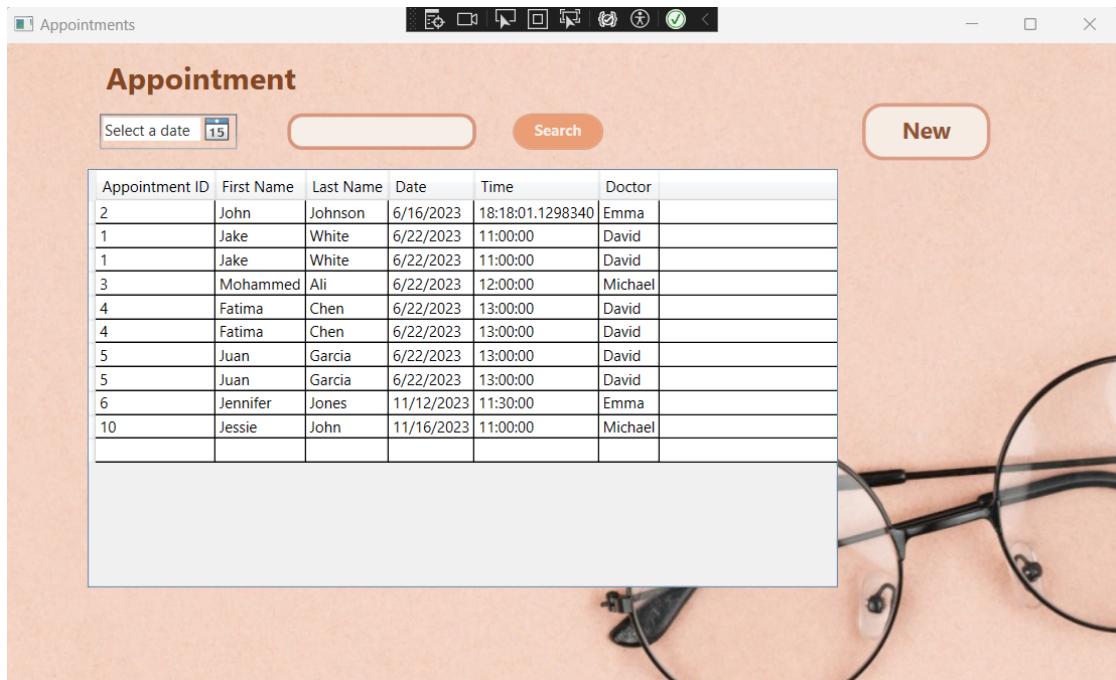
### When Double Clicking a Specific Customer from Customer Info Page:



### 5. Appointment Info Page

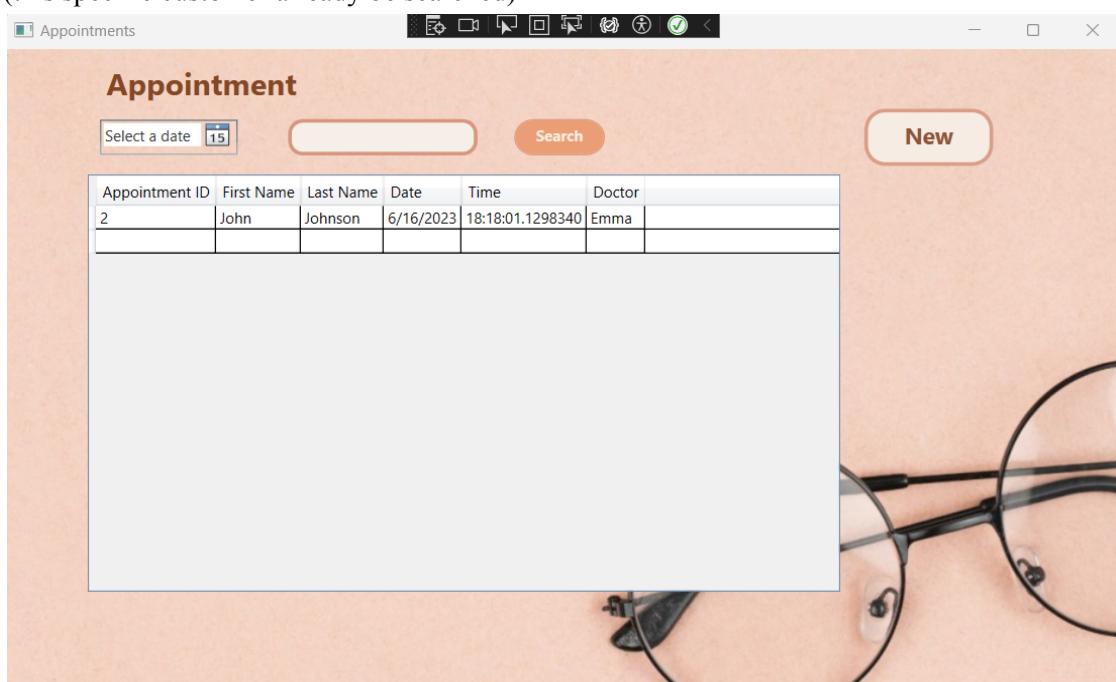
- Database Integration: Appointment information is fetched and displayed from the database in real time.
- Search Functionality: Users can search for details using doctor's name, customer's name or appointment date.
- Addition of New Entries: A 'New' button redirects to a detail page where users can add a new appointment for the specific customer. This button is only enabled when user verifies if customer exists from the customer info page and if not creates a new file on the customer details page. Once on the customer details page, the user needs to click on the appointment button. This will associate the booked appointment to a specific customer in the database.

**When Appointment Button is Clicked from the Dashboard:**



**When Appointment Button is Clicked from Customer Detail Page:**

(this specific customer already be searched)



## 6. Appointment Detail Page

- Double clicking on a specific appointment from the appointment info page will open up the appointment detail page with all of the information pulled from the database and loaded on the page.
- If user needs to create a new appointment for the customer, they can click on the New button and make a new appointment by filling out the form and clicking the Add button-Appointment was created successfully message will pop-up.

### When Double Clicking on Specific Appointment to Get the Details:

The screenshot shows a Windows-style application window titled "AppointmentDetails". The main title bar has several icons. The window itself is titled "Appointment Details". Inside, there are eight data entry fields with the following values:

- Customer ID: 1
- Appointment ID: 2
- First Name: John
- Last Name: Johnson
- Date: 2023-06-16
- Time: 18:18:01.1298340
- Eye Exam Fee: 85.00
- Store: (dropdown menu)
- Doctor: Emma Taylor (dropdown menu)

At the bottom are three buttons: "Add" (orange), "Delete" (orange), and "Cancel" (white).

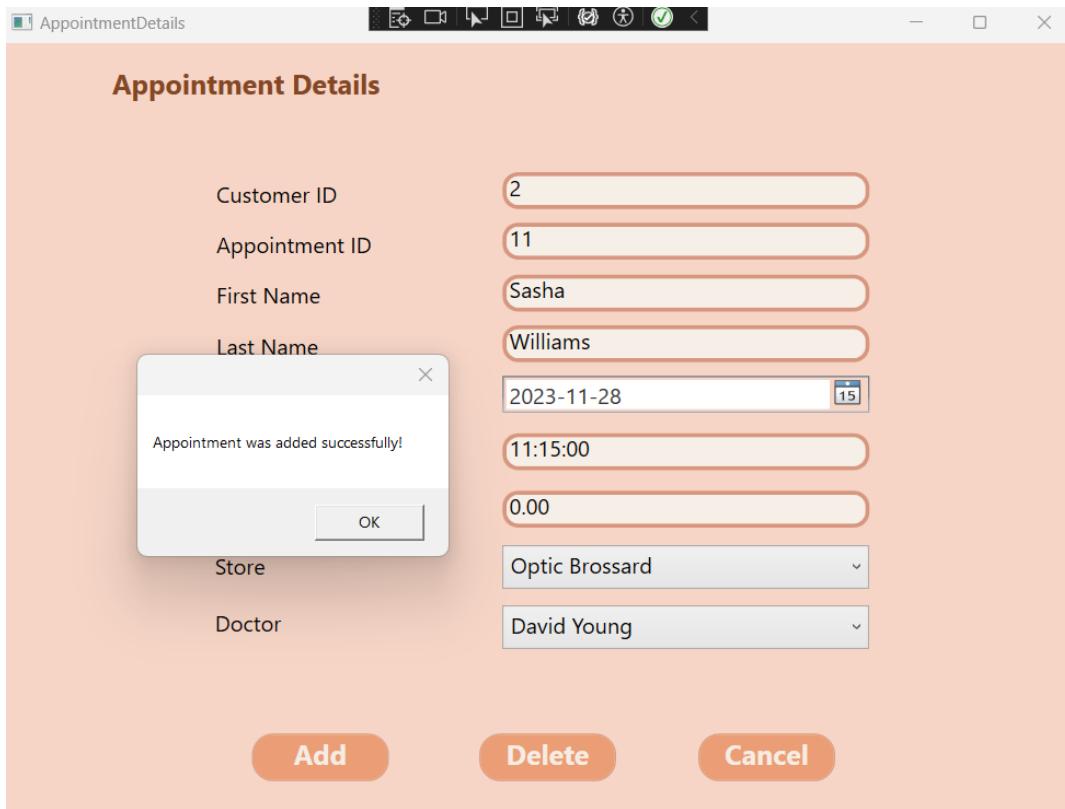
### When Clicking on the New Button to Create a New Appointment for the Customer:

The screenshot shows the same "AppointmentDetails" window, but the fields are mostly empty or show placeholder text:

- Customer ID: 1
- Appointment ID: (empty)
- First Name: John
- Last Name: Johnson
- Date: Select a date
- Time: (empty)
- Eye Exam Fee: (empty)
- Store: (dropdown menu)
- Doctor: (dropdown menu)

At the bottom are three buttons: "Add" (orange), "Delete" (orange), and "Cancel" (white).

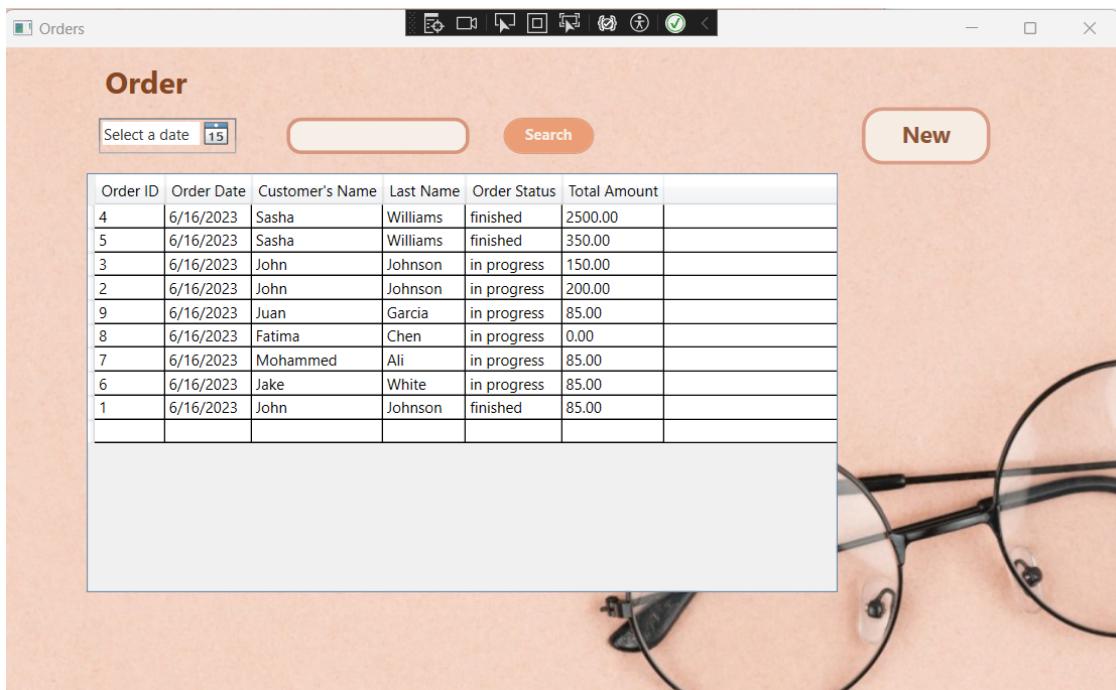
### When Appointment is Added Successfully:



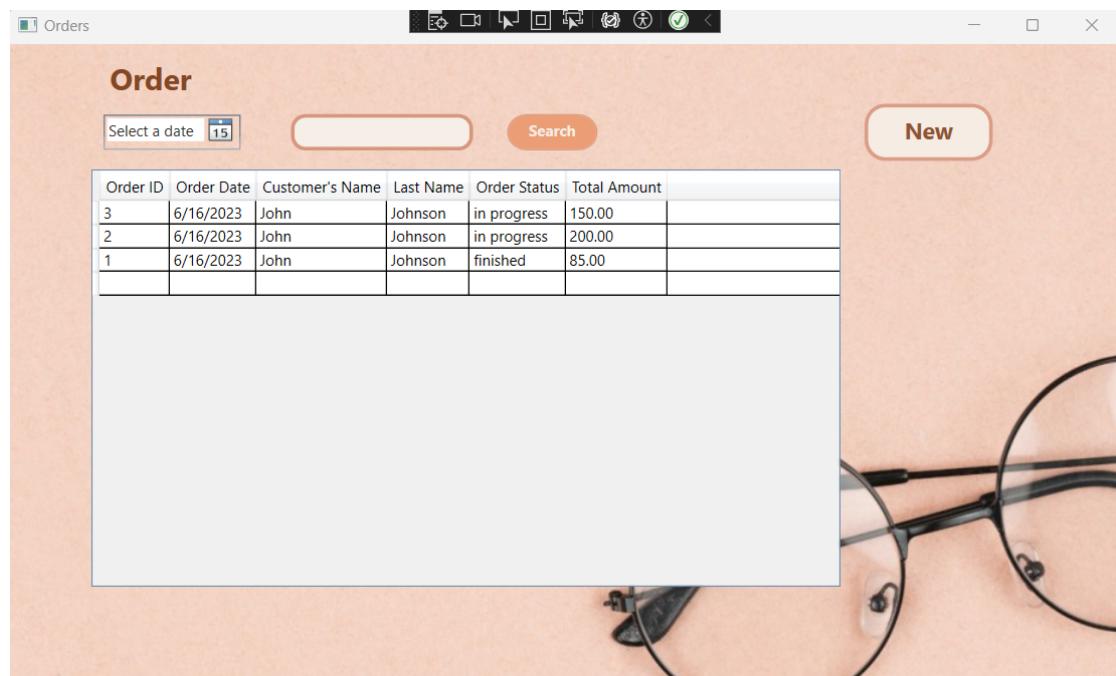
### 7. Order Info Page

- Database Integration: Order information is fetched and displayed from the database in real time.
- Search Functionality: Users can search for details using customer's name or order date.
- Addition of New Entries: A 'New' button redirects to a detail page where users can add a new order.
- Two ways to get to the order page:
  - From Dashboard
  - From Customer Detail Page
- User can double click on a specific order to see the invoice details.

**When Clicking on Appointment from Dashboard:**



**When Clicking on Order History from Customer Details Page:**  
(this specific customer already be searched)



## 8. Invoice Page

- Double clicking on a specific order from the appointment info page will open up the appointment detail page with all of the information pulled from the database and loaded on the page.
- Functionality we are still troubleshooting
  - If user needs to create a new order for the customer, they can click on the New button and make a new order by filling out the form and clicking the Add button-Invoice was created successfully message will pop-up.
- Future Functionalities
  - Send to Email Button: will send invoice to customers email
  - Print Button: will print invoice

**When Doubling Clicking on a Specific Order to see Details-Invoice for a Frame:**

The screenshot shows a Windows application window titled "OrderDetails". The main title bar has a close button (X) and a minimize button (-). The window title is "Invoice Form". Inside, there's a section titled "Invoice Details" with the following fields:

Order ID	3
Date	2023-06-16
Status	in progress
Store	Optic Brossard
Staff	Francois Leduc
Eye Exam	<input type="checkbox"/> Yes
Date of Appointment	Select a date
Name	John Johnson
Frame	Oakley Aviator, Colour: Silver, Size: M
Lens	
Quantity	1
Total Amount	150.00

At the bottom, there are four buttons: "Send to Email", "Print", "Submit", and "Cancel". A "Clear" button is also located near the Quantity field.

**When Doubling Clicking on a Specific Order to see Details-Invoice for an eye exam:**

The screenshot shows the 'Invoice Form' window with the title 'OrderDetails'. The form is titled 'Invoice Details' and contains the following fields:

Order ID	1
Date	2023-06-16
Status	finished
Store	(dropdown menu)
Staff	Francois Leduc
Eye Exam	<input type="checkbox"/> Yes 2
Date of Appointment	2023-06-16
Name	John Johnson
Frame	(dropdown menu)
Lens	(dropdown menu)
Quantity	1
Total Amount	85.00

Buttons at the bottom include 'Send to Email', 'Print', 'Submit', and 'Cancel'. A 'Clear' button is located on the right side of the form.

**When Clicking on New Button to Create New Invoice:**

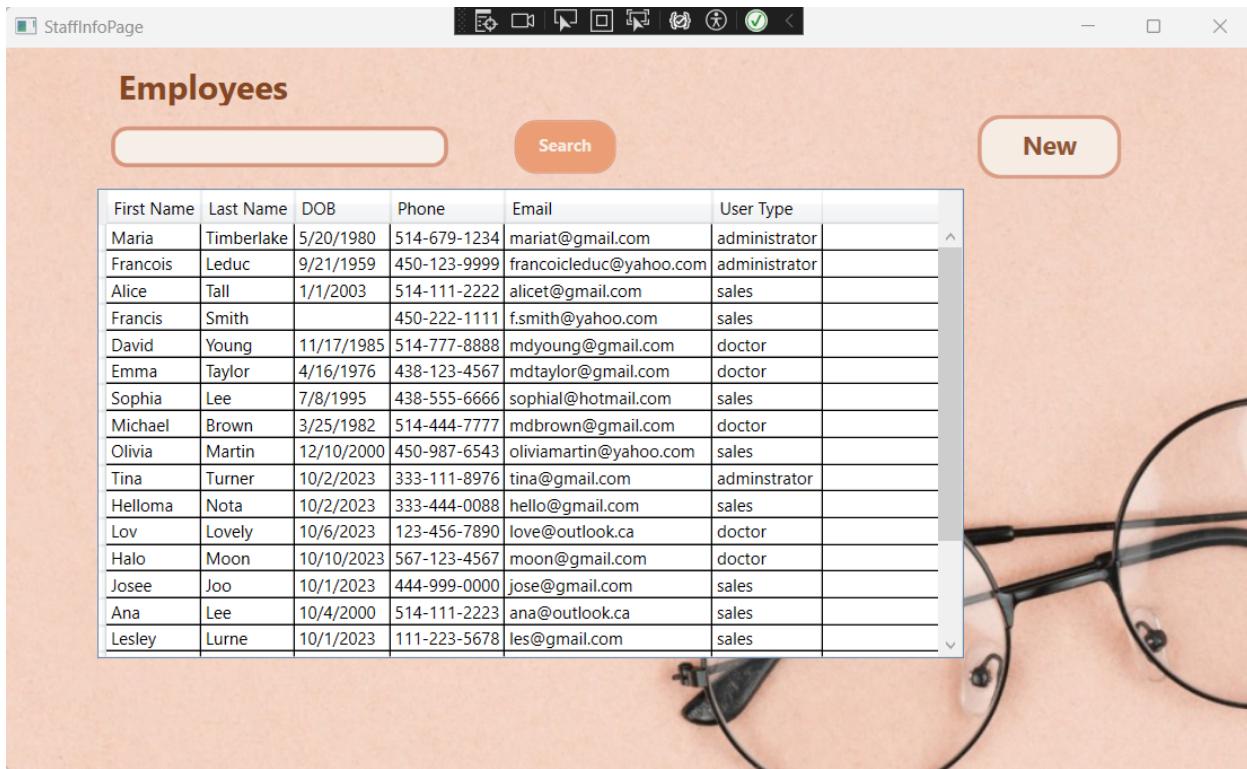
The screenshot shows the 'Invoice Form' window with the title 'OrderDetails'. The form is titled 'Invoice Details' and contains the following fields:

Order ID	(empty)
Date	Select a date
Status	(empty)
Store	(dropdown menu)
Staff	(dropdown menu)
Eye Exam	<input type="checkbox"/> Yes
Date of Appointment	Select a date
Name	(empty)
Frame	(dropdown menu)
Lens	(dropdown menu)
Quantity	(empty)
Total Amount	(empty)

Buttons at the bottom include 'Send to Email', 'Print', 'Submit', and 'Cancel'. A 'Clear' button is located on the right side of the form.

## 9. Staff Info Page:

- Database Integration: Employee information is fetched and displayed from the database in real time.
- Search Functionality: Users can search for details using employee's name, phone number or email.
- Addition of New Entries: A 'New' button redirects to a detail page where users can add a new employee.
- User can double click on a specific employee to see the employee details.



## 10. Employee Detail Page

- Double clicking on a specific employee from the employee info page will open up the employee detail page with all of the information pulled from the database and loaded on the page.
- User can add new employee by clicking on New Button on the staff info page. After filling out the employee details form, the user needs to click the add button to save the new employee in the database.
- User can update employee information by clicking the update button.
- Functionality we are still troubleshooting
  - At the moment when updating employee information, the person table and address table update the information in the database properly. But the staff table creates a new staff\_id with the updated information. We are working on fixing this problem.

**When Double Clicking on Employee in the Staff Info Page:**

The screenshot shows the "Employee Registration" window with the title bar "EmployeeDetails". The window contains two main sections: "Personal Information" and "Employee Profile".

**Personal Information:**

- Person ID: 9
- First Name: Francois
- Last Name: Leduc
- DOB: 1959-09-21
- Phone Number: 450-123-9999
- Email: francoicleduc@yahoo.com

**Employee Profile:**

- Employee ID: 2
- User Name: def456
- Password: fed654
- User Type (administrator, sales, doctor): administrator
- Manager ID (1 or 2): (empty)
- Active:  True  
 False

**Address Details:**

- Address ID: 8
- Address: 999 Wiser
- City: Montreal
- Postal Code: H3J5N6

**Buttons:**

- Add
- Update
- Cancel

**When Clicking on New Button to Create New Employee:**

The screenshot shows the "Employee Registration" window with the title bar "EmployeeDetails". The window contains two main sections: "Personal Information" and "Employee Profile".

**Personal Information:**

- Person ID: (empty)
- First Name: (empty)
- Last Name: (empty)
- DOB: Select a date (15)
- Phone Number: (empty)
- Email: (empty)

**Employee Profile:**

- Employee ID: (empty)
- User Name: (empty)
- Password: (empty)
- User Type (administrator, sales, doctor): (empty)
- Manager ID (1 or 2): (empty)
- Active:  True  
 False

**Address Details:**

- Address ID: (empty)
- Address: (empty)
- City: (empty)
- Postal Code: (empty)

**Buttons:**

- Add
- Update
- Cancel

## 11. Inventory Info Page

- Database Integration: Inventory information is fetched and displayed from the database in real time.
- Search Functionality: Users can search for details using the inventory id.
- Addition of New Entries: A ‘New’ button redirects to a detail page where users can add a new inventory product, either a lens or a frame.
- User can double click on a specific inventory to see the product’s details.

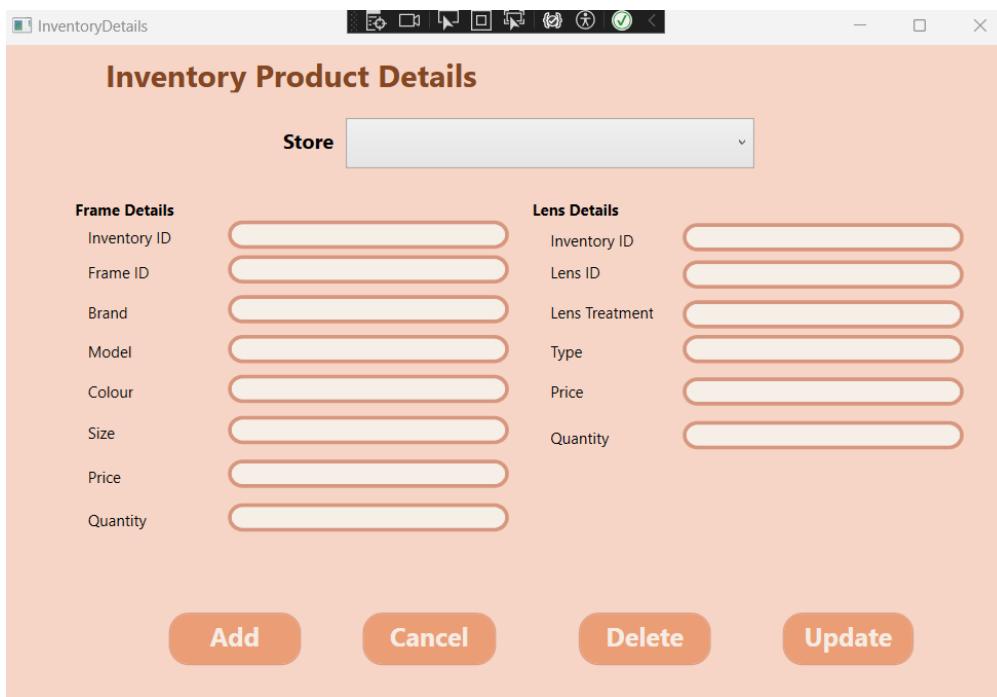
The screenshot shows a Windows-style application window titled "Inventory". The main area contains a table with columns: Inventory ID, Lens ID, Frame ID, Quantity, and Last Update. The table lists 15 rows of data. A "Search" button is located above the table, and a "New" button is to its right. The background of the window features a photograph of a pair of black-rimmed glasses. The window has standard operating system controls at the top.

Inventory ID	Lens ID	Frame ID	Quantity	Last Update
19		5	10	6/16/2023
20		1	10	6/16/2023
21		2	10	6/16/2023
23		4	10	6/16/2023
22		3	9	6/16/2023
24		5	9	6/16/2023
28	9		10	11/7/2023
29	10		10	11/7/2023
30	11		10	11/7/2023
31	12		10	11/7/2023
2	2		10	11/8/2023
16		2	11	11/8/2023
12	5		10	11/8/2023
11	4		10	11/8/2023
17		3	11	11/8/2023
9	2		10	11/8/2023

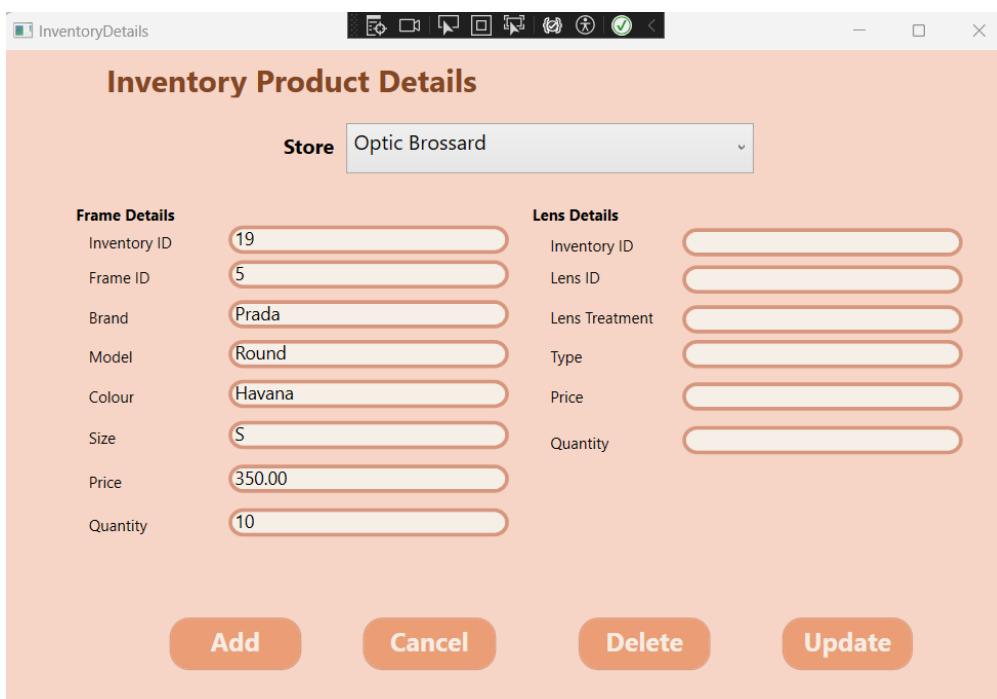
## 12. Inventory Detail Page

- Double clicking on a specific inventory product from the inventory info page will open up the inventory detail page with all of the information pulled from the database and loaded on the page.
- User can add new inventory product (either frame or lens) by clicking on New Button on the staff info page. After filling out the product’s details form, the user needs to click the add button to save the new product in the database. It will be saved in both the respective product (frame or lens) table and the inventory table.
- User can update product information by clicking the update button.
- User can delete product by clicking on the delete button. This will delete the product both from the respective (frame or lens) table and the inventory table.
- When clicking add, update or delete, if the action is successful the user will get a confirmation message.

**When Clicking New Button to Create New Inventory Product:**



**When Double Clicking on Specific Inventory Product:**

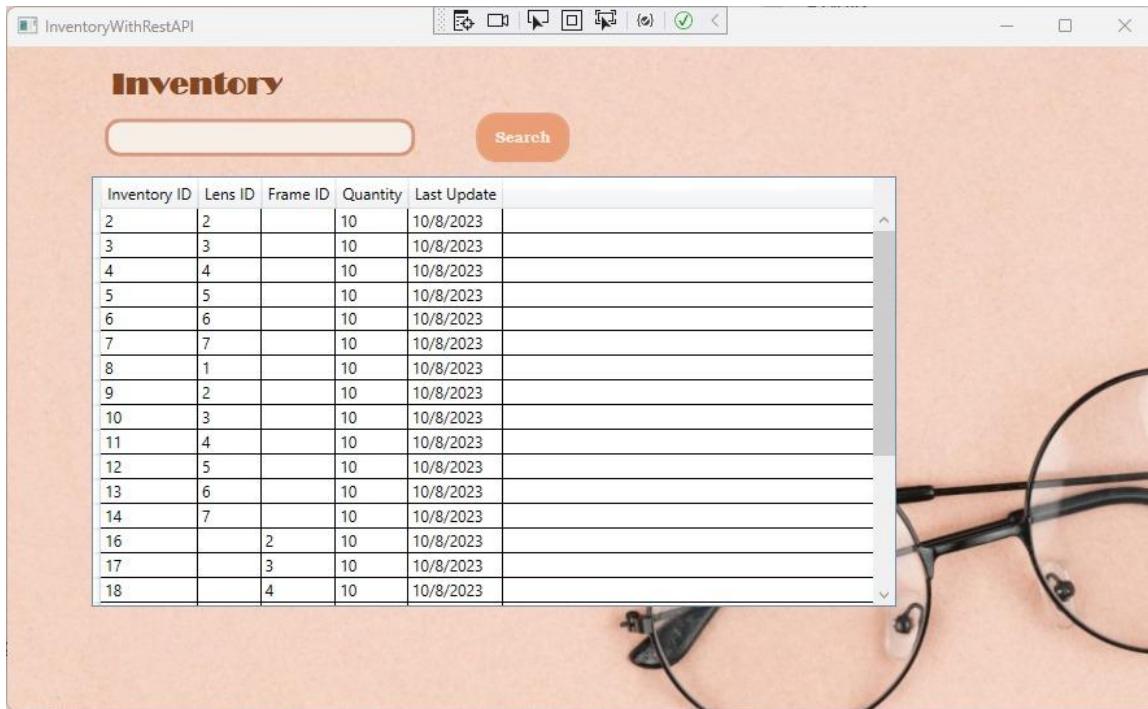


### 13. Rest API Page

- The user needs to have the RestAPI\_OCMS open first, and they need to run the API.

The screenshot shows the RestAPI\_OCMS application interface. At the top, there is a navigation bar with the OCMS logo and a search bar. Below the navigation bar, the main content area has a title "GET /OCMS/\_GetAllInventory". Underneath the title, there is a "Parameters" section which says "No parameters". To the right of this section is a "Cancel" button. Below the parameters, there are two buttons: "Execute" (highlighted in blue) and "Clear". In the "Responses" section, there is a "Curl" code block containing a command to run a curl request to the specified URL. Below the curl block is a "Request URL" field containing the URL "https://localhost:7119/OCMS/\_GetAllInventory". The "Server response" section shows a status code of 200 and a "Response body" containing JSON data. The JSON data represents five inventory items with fields like inventory\_id, lens\_id, frame\_id, quantity, and last\_update. Below the response body are "Response headers" showing content-type, date, and server information. At the bottom of the responses section, there are "Copy" and "Download" buttons.

- Then by clicking on the Rest API Test button in the main program on the dashboard page, the user will be brought to the REST API page where they can see all inventory products.
- The user can search the inventory products by id.



#### 14. Unit Test: GetInventoryById Method

- **Purpose of Test:** To check if the system correctly identifies and retrieves an inventory item when requested by its unique ID.

Arrange (Setting Up)

- **Mocks Preparation:** Fake configurations and database services are set up to mimic the application's environment. This ensures the test runs under controlled conditions without real data.
- **Expected Outcome Setup:** A 'right answer' is prepared, representing what the system should return when the inventory ID is correct.

Act (Executing the Test)

- **Controller Initialization:** A new controller, the part responsible for fetching inventory data, is created with the mocked services.
- **Method Execution:** The GetInventoryById method is called with a predetermined inventory ID to simulate a real user's action.

Assert (Verifying the Outcome)

- **Result Comparison:** The details returned by the method are compared with the 'right answer' prepared during setup.
- **Success Criteria:** If the returned details match the expected outcome, the test passes, confirming the system's ability to retrieve accurate inventory data by ID.

Conclusion

- **Test Passes:** A successful test indicates the system is reliable in finding and presenting the correct inventory item when queried with an ID.

The screenshot shows the Visual Studio IDE interface. The top part displays a C# unit test file named 'OCMS\_ControllerTests.cs' with the following code:

```

1  using Microsoft.VisualStudio.TestTools.UnitTesting;
2  using Moq;
3  using Npgsql;
4  using RestAPI_OCMS.Controllers;
5  using RestAPI_OCMS.Models;
6
7  namespace RestAPI_OCMS.Tests
8  {
9      [TestClass]
10     public class OCMS_ControllerTests
11     {
12         [TestMethod]
13         public void GetInventoryById_ReturnsCorrectResponse()
14         {
15             // Arrange
16             var mockConfiguration = new Mock<IConfigurationWrapper>();
17             var mockDBApplication = new Mock<IDBApplication>();
18             var expectedResponse = new Response();
19
20             mockConfiguration.Setup(config => config.GetConnectionString("It.IsAny<string>())).Returns("Host=localhost;Port=5432;Database=optic");
21             mockDBApplication.Setup(db => db.GetInventoryById(It.IsAny<NpgsqlConnection>(), It.IsAny<int>())).Returns(expectedResponse);
22
23             var controller = new OCMS_Controller(mockConfiguration.Object, mockDBApplication.Object);
24
25             // Act
26             var result = controller.GetInventoryById(1);
27
28             // Assert
29             Assert.AreEqual(expectedResponse, result);
30         }
31     }
32 }

```

The bottom part shows the 'Test Explorer' window with the following details:

Test	Duration	Traits
RestAPI_OCMS (1)	248 ms	
RestAPI_OCMS.Tests (1)	248 ms	
OCMS_ControllerTests (1)	248 ms	
GetInventoryById_ReturnsCorre...	248 ms	

Group Summary: OCMS\_ControllerTests  
Tests in group: 1  
Total Duration: 248 ms  
Outcomes: 1 Passed

# Project Setup

## 1. Prerequisites:

- Database: Microsoft SQL Server 2023.
- Development Environment: Visual Studio 2023 with WPF .NET support.
- Front-End Framework: WPF (Windows Presentation Foundation) using C#.
- API Development: ASP.NET Core 7.0 for RESTful API.
- Version Control: Git with GitHub for source code management.

## 2. Database Setup:

- Installation and Configuration: Install and configure Microsoft SQL Server 2023 on the client's server, setting up the necessary database schemas and tables according to OCMS requirements.
- Data Migration: If migrating from an existing system, carefully transfer data to the new SQL Server database, ensuring data integrity.

### **3. API Setup:**

- Visual Studio Configuration: Install Visual Studio 2023 on the development machine and ensure that the ASP.NET and web development workload is installed.
- API Project Setup: Clone the OCMS API repository using Git and open it in Visual Studio.
- Dependency Management: Restore all NuGet packages required for the ASP.NET Core project.
- Database Connection: Modify the connection string in the appsettings.json file to connect the API to the SQL Server database.

### **4. WPF Front-End Setup:**

- WPF Project: In Visual Studio 2023, open the WPF project from the OCMS repository.
- NuGet Packages: Ensure all necessary NuGet packages for the WPF project are installed and up-to-date.
- API Integration: Configure the WPF application to communicate with the ASP.NET Core API. This involves setting up the correct API endpoints and ensuring the WPF app can make HTTP requests to the API.
- Local Testing: Run the WPF application locally to test its connection with the API and the database.

### **5. Running the Complete Application:**

- Start the API: Run the ASP.NET Core project in Visual Studio to start the backend service.
- Launch the WPF Application: Start the WPF application. Ensure it successfully connects and interacts with the backend API.

### **6. Customization and Local Testing:**

- UI Customizations: Adjust the WPF application's user interface to meet specific client needs or branding requirements.
- Functionality Testing: Thoroughly test the entire system (front-end WPF application, API, and database) in a local environment to ensure everything functions as expected.

### **7. Deployment and Documentation:**

- Deployment: After successful testing, deploy the database and the API to the client's server environment. The WPF application can be distributed to the client's machines as needed.
- Documentation: Provide detailed documentation covering installation, configuration, and operational guidance for the OCMS.
- Ongoing Support: Offer technical support for the setup and deployment phases, and provide ongoing maintenance as required.

# Work Project in Future

Future development of OCMS will focus not only on incorporating new technologies but also on refining existing features based on user feedback and changing market dynamics:

## **1. Integration of Artificial Intelligence and Machine Learning:**

- Predictive Analytics: Implement AI algorithms to predict sales trends, inventory needs, and customer preferences. This can aid in better stock management and personalized marketing.
- Chatbots for Customer Service: Integrate AI-powered chatbots to provide instant assistance to customers online, improving engagement and freeing up human resources for more complex tasks.

## **2. Enhanced Mobile Experience:**

- Mobile App Development: Develop a dedicated mobile application for both iOS and Android platforms to provide customers and employees with on-the-go access to OCMS features.

## **3. Advanced Analytics and Reporting:**

- Customizable Dashboards: Develop more advanced, customizable dashboards for different user roles, providing tailored analytics and insights.
- Data Visualization Tools: Integrate sophisticated data visualization tools for analyzing sales data, customer demographics, and other key metrics.

## **4. Expanded Payment Options and Financial Integration:**

- Cryptocurrency Payments: Explore the integration of cryptocurrency payments to provide customers with more payment options.
- Financial Software Integration: Integrate with popular financial management software for streamlined accounting and financial reporting.

## **5. Cloud-Based Solutions and Scalability:**

- Migration to Cloud: Consider migrating OCMS to a cloud-based platform for enhanced scalability, performance, and disaster recovery options.
- Dynamic Scaling: Implement dynamic scaling solutions to efficiently handle varying loads, especially during peak business periods.