# Project Documentation

**1. Project Description:**

The Hi-Tech Order Management System is a Windows Forms Application developed using Microsoft Visual Studio 2022, C#, and SQL Server 2022/2019. The system facilitates the management of orders for computer science books supplied by Hi-Tech Distribution Inc. to colleges and universities in Quebec. It allows for the addition, update, deletion, search, and listing of various entities such as books, authors, publishers, customers, orders, and employees.

**2. Project Design:**

The Hi-Tech Order Management System consists of two projects: Hi-Tech Library and Windows Form App. Hi-Tech Library is a class library project housing the Business Logic Layer (BLL), Data Access Layer (DAL), and VALIDATION modules. Windows Form App is a .NET Windows Forms Application responsible for the Graphical User Interface (GUI).

**- Project Operations:**

|  |  |  |
| --- | --- | --- |
| Users | Operations | Technical Requirements |
| MIS Manager(Henry Brown) | * Add/update/delete user Information * Search/List user information * Add/update/delete employee information * Search/list employee information | Database Programming with  ADO.NET in Connected Mode |
| Sales Manager(Thomas Moore) | * Add/Update/Delete customer information * Search/List customer information | Database Programming with  ADO.NET in  Disconnected Mode |
| Inventory Controller(Peter Wang) | * Add/update/delete/search/list book information and related information | Entity Framework |
| Order Clerks  Mary Brown  Jennifer Bouchard | * Add/Update/Cancel customers’ orders * Search/List customers’ orders | Entity Framework |

**- Database Design:**

* Consistent naming conventions
* No data redundancy
* Proper relationships among tables
* Database Diagrams

A screenshot of a computer

Description automatically generated

A computer screen shot of several blue and white boxes

Description automatically generated

**Module 1: Connected Mode**

MIS Manager (Henry Brown) : Manages the Employee and User Information.

* **Tables:** Employees, Users, Status, Jobs
* Relationships:
  + Employees ------ many-to-one ------ Jobs
  + Employees ------ many-to-one ------ Status
  + Users ----------- one-to-one ------- Employees
  + Users ----------- many-to-one ------ Status

**Module 2: Disconnected Mode**

Sales Manager (Thomas Moore) : Manages the Customer related Information.

* **Tables:** Customers

**Module 3: Entity Framework**

Inventory Controller (Peter Wang): Manages the Books related Information.

* **Tables:** Books, Categories, Publishers, Authors, BookAuthors
* Relationships:
  + Books --------- one-to-one --------- Categories
  + Books --------- many-to-one -------- Publishers
  + Books --------- many-to-many ------- Authors

**Module 4: Entity Framework**

Order Clerks (Mary Brown and Jennifer Bouchard) : Manages Orders related Information.

* **Tables:** Orders, BookOrders
* Relationships:
  + Orders -------- many-to-many ------- Books
* **Design of Application Domain Classes**

The application domain classes represent the core entities and business logic of the Hi-Tech Order Management System. These classes encapsulate the data and behavior related to various entities such as employees, users, customers, books, orders, and more. Below is an outline of the domain classes:

* **Employee:** Represents information about employees, including their ID, name, job title, and status.
* **User:** Contains user-specific information such as login credentials and access rights.
* **Customer:** Stores details about customers, including their ID, name, contact information, and address.
* **BookController:** Represents information about books, including title, ISBN, category, publisher, and author(s).
* **PublisherController:** Handles Publisher for books, including their ID, name, and description.
* **CategoryController:** Handles categories for books, including their ID, name, and description.
* **AuthorController:** Manages information about authors, including their ID, name, and biography.
* **BookAuthorController:** Manages the relationship between books and authors, facilitating the association of multiple authors with a single book and vice versa.
* **OrderController:** Contains details about orders placed by customers, including order ID, customer ID, order date, and status.

These domain classes serve as the backbone of the application, providing a structured representation of the data and supporting the implementation of business logic.

* **Class Diagram**

**A computer screen shot of a computer screen

Description automatically generated**

**- Design of GUI Classes**

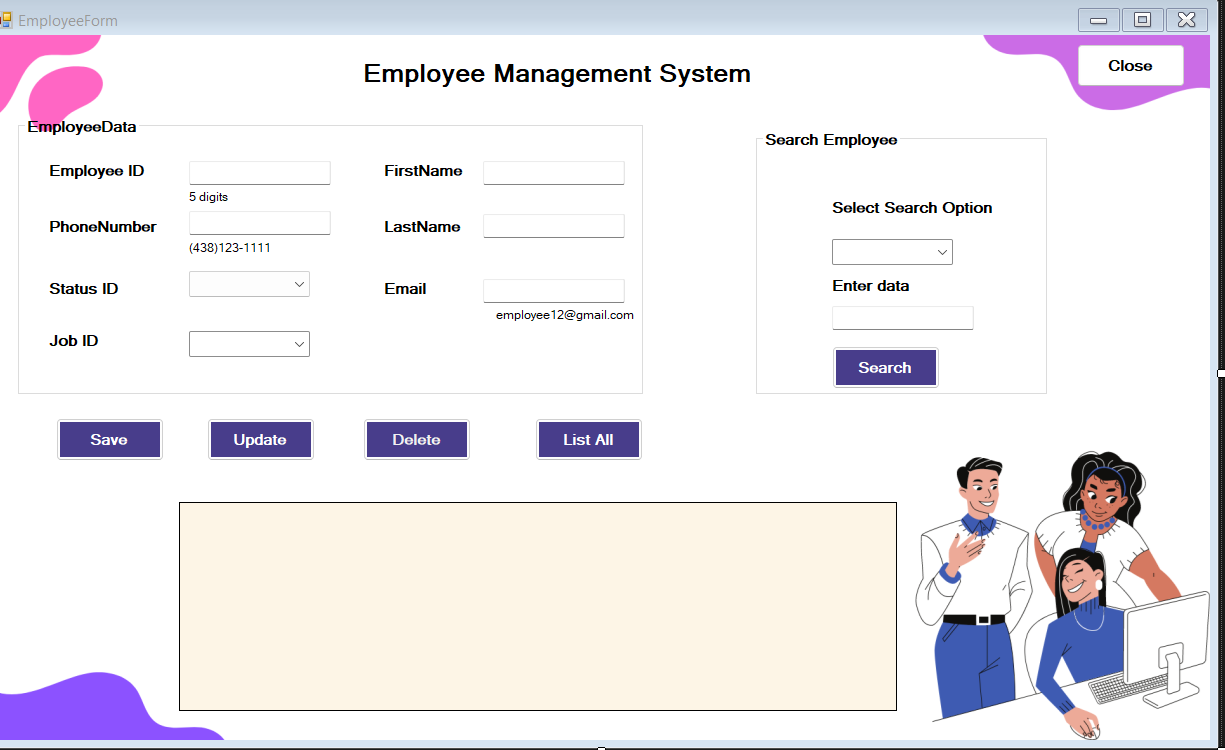
The GUI (Graphical User Interface) classes are responsible for presenting the application's functionality to users in an intuitive and visually appealing manner. These classes include forms, controls, and user interface components designed using Windows Forms technology.

* **LoginForm:** Provides the interface for user authentication, allowing users to log in with their credentials.

A computer screen with a lock

Description automatically generated

* **EmployeeForm:** Enables MIS managers to perform operations related to employee management, such as adding, updating, and deleting employee information.



* **UserForm:** Enables MIS managers to perform operations related to User management, such as adding, updating, and deleting employee information.

A screenshot of a computer

Description automatically generated

* **CustomerForm:** Allows sales managers to manage customer information, including adding, updating, and deleting customer records.

A screenshot of a computer

Description automatically generated

* **BookForm:** Provides inventory controllers with functionality to manage book and related information, such as adding, updating, and deleting books.

A computer screen shot of a book

Description automatically generated

A computer screen shot of a computer screen

Description automatically generated

A computer screen shot of a computer screen

Description automatically generated

A computer screen shot of a computer

Description automatically generated

A computer screen shot of a computer screen

Description automatically generated

* **OrderForm:** Enables order clerks to handle orders, including adding, updating, and canceling customer orders.

A screenshot of a computer

Description automatically generated

A computer screen shot of a book

Description automatically generated

These GUI classes are designed to be user-friendly and responsive, enhancing the overall usability of the system.

**- Design of Data Access Classes**

The data access classes facilitate interaction with the database, including querying, updating, and manipulating data. These classes are implemented using ADO.NET and Entity Framework for connected and disconnected modes, respectively.

* **EmployeeDB:** Provides methods for accessing and manipulating employee data in the database.
* **UserDB:** Handles interactions with user-related data, such as authentication and authorization.
* **OrderRepository:** Manages data access for orders, including creating, updating, and canceling orders.
* **BookRepository:** Handles database interactions related to book information, including CRUD operations and retrieval.
* **PublisherRepository:** Manages interactions with publisher data.
* **AuthorRepository:** Handles interactions with author data.
* **CategoryRepository**: Manages interactions with category data.
* **BookAuthorRepository:** Manages the relationship between books and authors.
* **LoginDB:** Provides methods for accessing and manipulating Login data in the database.
* **UtilityDB:** A database utility class providing common functionalities such as connection management, transaction handling, error logging, and query execution within the data access layer of the system.

These data access classes abstract the underlying database operations, promoting modularity and maintainability within the system.

**3. Project Implementation:**

The implementation involves the development of both the Hi-Tech Library and Hi-Tech Windows Form App projects according to the specified requirements. The BLL, DAL, and Validation modules are implemented in the class library, while the GUI is developed in the Windows Forms Application. Project Involves the Login Authentication. Only Active Users(Managers) in the database can access the Systems, for which they are responsible.

**Login Credentials:**

* MIS Manager (Henry Brown): User ID: 2225, Password: Henry@25
* Sales Manager (Thomas Moore): User ID: 2223, Password: Thomas@22
* Inventory Controller (Peter Wang): User ID: 2235, Password: Peter@33
* Order Clerks (Mary Brown): User ID: 2237, Password: MaryBrown@99
* Order Clerks (Jennifer Bouchard): User ID: 2238, Password: Jennifer@88

**4. Project Testing:**

Testing was conducted at various stages of development to ensure the reliability and functionality of the Hi-Tech Order Management System. The following types of testing were performed:

* Unit Testing: Each module's functions were individually tested to verify that they performed as expected.
* Integration Testing: Different modules were integrated to ensure they functioned correctly together.
* User Acceptance Testing (UAT): End-users, Other Groups, tested the system to validate its usability and suitability for their needs.
* Security Testing: The login authentication system was thoroughly tested to ensure only authorized users could access the system.
* Performance Testing: The system's performance was evaluated under various conditions to ensure it could handle the expected workload efficiently.

**6. Conclusion:**

In conclusion, our journey with the Hi-Tech Order Management System has been a valuable learning experience. We successfully navigated through Connected, Disconnected, and Entity Framework methodologies, overcoming challenges along the way. While updating entities in Entity Framework presented a notable hurdle, we honed our skills and implemented effective solutions. Our proficiency in these methodologies has not only resulted in a robust system but also equipped us with valuable insights for future projects. We are now better equipped to handle complex database interactions and look forward to applying our newfound knowledge in future endeavors.

Top of Form