|  |
| --- |
|  |
| Semester Project |
|  |
|  |
| **Caleb Iamkui**  **Matt Sunderland**  **Seiji Suenaga** |
| **12/1/2014** |

Introduction

We were hungry and thought why not make a database for a restaurant to enable people to view a menu, its items, and employees. We have all been victims to restaurants who seem much disorganized with their menus or with their staff. A database could help small restaurants around the KC area disorganized like this.

We developed our application using ASP.NET MVC to implement our tables into a very nice web page for users to navigate through the application. A user would be able to view/edit/update the employees’ information, ingredients, and menu items. An example would be if a manager wanted to terminate a current employee they would be able to go into this application and remove them.

The project should and does contain multiple relationships between entities of:

1:1 – store has one address

1:N – 1 store has multiple employees, and has multiple menu types

M:N – multiple menu types have multiple ingredients

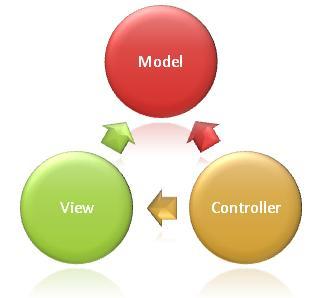
Our application allows the user to get on the site and can click to view the different menus of the restaurant. The application should allow the user to view the breakfast, lunch, and dinner menus. Our main priority is the database, so setting up security clearance depending on user will only be implemented if time permits.

Data Dependencies

|  |  |  |  |
| --- | --- | --- | --- |
| **Store** | **Employee** | **Menu\_Item** | **Ingredient** |

Architecture

Our application uses a 3-tier architectural design.



Variables

|  |  |  |
| --- | --- | --- |
| **STORE** | | |
| **Attribute** | **Type** | **Values/Notes** |
| ID | Int | Surrogate key |
| Phone | Varchar(15) | Contact info |
| Employee\_Count | Int | # of Employees |
|  |  |  |
|  |  |  |
| **ADDRESS** | | |
| **Attribute** | **Type** | **Values/Notes** |
| Street | Varchar(10) | Street |
| Zip | Varchar(10) | Zip |
| State | Varchar(20) | State |
| City | Varchar(20) | City |
| Building\_Num | Varchar(10) | Building number |
|  |  |  |
|  |  |  |
| **EMPLOYEE** | | |
| **Attribute** | **Type** | **Values/Notes** |
| ID | Int | Unique number |
| Fname | Varchar(20) | First name |
| Lname | Varchar(20) | Last name |
| Phone |  |  |
| Position | Varchar(20) | Position |
| Wage | Decimal | wage |

|  |  |  |
| --- | --- | --- |
| **MENU\_ITEM** | | |
| **Attribute** | **Type** | **Values/Notes** |
| Name | Varchar(20) | Title of item |
| Price | Decimal | Price |
| Is\_Vegan | Boolean | Vegan choice |
| Calories | Int | Calorie intake |
|  |  |  |
|  |  |  |
| **INGREDIENT** | | |
| **Attribute** | **Type** | **Values/Notes** |
| Name | Varchar(20) | Ingredient |
| Amount | Int | Amount of ingredients |
| Amount\_Unit | Varchar(20) | Pounds, bags, boxes, etc |
| Amount\_Alert | Int | Notifies when needing to be restocked |
| Date\_Restocked | Int | When restock ingredient |
| Shelf\_Life | Int | How many days an ingredient can be on the shelf |

|  |  |  |
| --- | --- | --- |
| **Menu** | | |
| **Attribute** | **Type** | **Values/Notes** |
| Name | Varchar(20) | Breakfast, lunch, dinner, drink |

External Schema

The Restaurant database is a database that stores all the different products from the menu, ingredient, and employee information. The database may be used to store newly created menu options along with the needed ingredient’s that make the product. Employees can use this database to look up recipes for all their products, and help in their decision of what types of ingredient’s are in a higher demand. We need to keep track of menu items. Along with the stuff needed to make them, and should also keep track of the cost, nutritional info, and number of calories. Each menu item is identified by product ID, name, calories, nutritional value, and price. Each ingredient is identified by name, and amount.

Schema Diagram

|  |
| --- |
| Menu\_Item |
| Name  Type  Price |

|  |
| --- |
| Ingredient |
| Name |

|  |
| --- |
| Employee |
| Name  Employee  Position  Wage |

|  |
| --- |
| Store |
| ID  Phone  Employee\_Count |

|  |
| --- |
| Address |
| Building\_Num  Street  Zip  City  State |

Non-Normalized diagram

Normalized Diagram

Future Scope

Potentially we want our application to grow not only to small restaurants in KC but also around the nation and even the world. We would first need to fix all of our current deficiencies in our application in order for us to fully have restaurants even want our application. We could then get feedback from our users and direct all that feedback towards future implementation for our application. With many ideas from user

Log in page

Detailed inventory such as calories, nutritional info

