**Abarar Sheikh**

**Features:**

Content management system –Manage movies

Contact Us Form

Movie Rating

**Content management system –Manage movies**

Manage movies feature is mean by which the cinema website can display all the movies on home page. Visitors to this website will see a link called “Movies” and be able to click on the link and view all the movies, now showing and coming soon movies.

The manage movies feature will have an administrative backend that will manage the movies. By managing the movies we mean that administrative can edit/delete or add new movies to the website. Administrative users will have to login to Web site and choose “Movies” from the available links. They would then see a list of links and option to choose one to edit, delete or add new movies.

The database will look like this:

|  |  |
| --- | --- |
| Id | Integer (Primary Key) |
| Title | Text 50 character |
| Language | Text 50 characters |
| ReleaseDate | Date |
| Director | Text 100 characters |
| Cast | Text 255 characters |

**The dataflow models**

Here is how “add movie” data flow model would look:



Here is how “Edit movie” data flow model would look:



Here is how “Delete movie” data flow model would look:



**Contact Us Form:**

Contact us future is means visitors can contact to the cinema house to ask any kind of query. visitors to the web site will see a link called “Contact Us” When user click on contact us it will go to contact us page, on this page user will see the fields options such as First name, Last name, Your Email and Your Message and last Submit button .visitors have to fill all these information and click on the Submit button then it will sent this information to the database and it also sent Email to the cinema house mail server. So, they will get the notification on their mail account. All the fields in the form are mandatory so visitors cannot left the blank space and click on Submit button.

The database will look like this:

|  |  |
| --- | --- |
| Primary Key | Numeric |
| contact\_id | Integer (Primary Key) |
| first\_name | Text 50 Character |
| last\_name | Text 50 Character |
| Email | Text 50 Character |
| Message | Text 150 Character |

Here is how “Contact Us” data flow model would look:



**Movie Rating:**

Movie Rating is mean by which the cinema website can solicit and display movie rating from satisfied people on home page. Visitors to this website will see a link called “Movie Rating” and be able to click on the this option and view rating from other visitors who have rated the movie , that will encourage them to feel good about coming to cinema to watch the movie.

The database will look like this:

|  |  |
| --- | --- |
| rating\_Id | Integer (Primary Key) |
| rating\_Name | Text 50 Character |
| rating\_Total | Numeric |
| rating\_Votes | Numeric |

Here is how “Movie Rating” data flow model would look:



**BIN LIU**

**Career page Features:**

* **Job Posting**
* **Post information Page by user**
* **Online test**

Job Posting Page

Manager will fill out a form to finish Job posting. The form will include the Job type, location, Your email, Full/Part Time, Shift, Desired pay, Job Id, Job title, job Description. After user finish fill out the form about posting page, the form will ask user are you sure you want finish? Then user will click post job or cancel to finish.

The database will look like this:

|  |  |
| --- | --- |
| **Primary key** | ID |
| Job title | CHAR(25) |
| location | VARCHAR(25) |
| Email | VARCHAR(25) |
| Shift | DATETIME() |
| Full/Part Time | TIME() |
| pay | varchar(25) |
| Job Id | varchar(25) |
| Job type | varchar(25) |
| job Description | varchar(25) |



Post information Page by user

User will choose what kind of job they want to do then fill out a form to finish information. The form will include the Job type, user first name, last name, your email, phone number, age, experience. After user finish fill out the form, Then user will click send message to go to next online text page.

The database will look like this:

|  |  |
| --- | --- |
| **Primary key** | ID |
| First name | CHAR(25) |
| Last name | VARCHAR(25) |
| phone number | VARCHAR(25) |
| Email | VARCHAR(25) |
| age | DATETIME() |
| Experience | VARCHAR(25) |



Online test

After user finish fill out their information, they will take online test, we will check their score then depend next phone interview, the test will be multiple choice.

|  |  |
| --- | --- |
| Id | (Primary Key) |
| Title | Text 50 character |
| text | Text 50 characters |
| score | VARCHAR(25) |
| question | data |



**Lena Yu**

1. **Food shopping cart Feature**

* Description

After choosing the food, The users can add and delete food in the shopping cart, or change the number of food. Also, they can see the list of the food that they want to buy and the money that they will pay or have been saved.

* Database Table

**Food table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Description | Data Type | Length | Key |
| Food\_id | Food ID | INT | 8 | Primary key |
| Food\_Name | Food name | varchar | 50 |  |
| Food\_Catagory | Food’s type | varchar | 50 |  |
| Food\_Price | Food’s price | float | 10 |  |
| Food\_Image | Food’s image’s name | varchar | 50 |  |
| Food\_Instock | The number of food in stock | int | 8 |  |
| Food\_description | Food’s description | varchar | 500 |  |
| Food\_mark | User’s average mark for the food | int | 1 |  |

**Food Order table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Description | Data Type | Length | Key |
| Order\_id | Food ID | INT | 8 | Primary key |
| User\_id | User’s id | int | 8 | Foreign key |
| Order\_time | Order time | date |  |  |
| Cinema\_id | Cinema id | int | 8 | Foreign key |
| Deliver\_time | Deliver\_time | date |  |  |
| Total\_price | Total price | float | 10 |  |
| Phone\_number | User’s phone number | varchar | 50 |  |
| Email\_address | User’s email address | varchar | 50 |  |
| Room\_id | Room ID | int | 8 | Foreign key |

**Food order item table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Description | Data Type | Length | Key |
| Order\_item\_id | Order item id | int | 8 | Primary key |
| Food\_id | Food ID | INT | 8 | Foreign key |
| Order\_id | Order\_ID | int | 8 | Foreign key |
| Quantity | Food’s quantity | int | 8 |  |
| User\_id | User ID | int | 8 | Foreign key |

* Dataflow

****

1. **Online Test Feature**

* Description

If the amount of a user’s shopping recording is over 100 dollars, the user can have a online test. And if he answer all the questions right, he will get 20 dollars.

* Database Table

**Food user recording table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Description | Data Type | Length | Key |
| Record\_id | Record id | int | 8 | Primary key |
| User\_id | User id | INT | 8 | Foreign key |
| Total\_amount | the amount of a user’s shopping recording | float | 10 |  |

**Food Test question table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Description | Data Type | Length | Key |
| Question\_id | Question\_id | int | 8 | Primary key |
| Question | Question | varchar | 100 |  |
| Option1 | Option1 | varchar | 100 |  |
| Option2 | Option2 | varchar | 100 |  |
| Option3 | Option3 | varchar | 100 |  |
| Option4 | Option4 | varchar | 100 |  |
| Answer | Answer | varchar | 50 |  |

**Food Test table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Description | Data Type | Length | Key |
| Test\_id | Test\_id | int | 8 | Primary key |
| Score | Score | int | 8 |  |
| User\_id | User id | int | 8 | Foreign key |
|  |  |  |  |  |
|  |  |  |  |  |

* Dataflow



1. **User’s comment and marking feature**

* Description

A user can make a comment for a food and mark for it.And when other users want to the details of this food, they will know the evaluation quickly and easily.

* Database Table

**Food comment table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Description | Data Type | Length | Key |
| Comment\_id | Comment Id | int | 8 | Primary key |
| Food\_id | Food\_id | int | 8 | Foreign key |
| User\_id | User id | int | 8 | Foreign key |
| Comment | Comment for food | varchar | 500 |  |
| Mark | Mark for food | int | 1 |  |

* Dataflow

****

**Yi Zhao**

Features

* Authentication system
* Payment Transaction
* Navigation and routing

**Authentication system**

It contains password log in, cookie log in, log out, register, and user roles.

Users table

|  |  |  |
| --- | --- | --- |
| id | int | Primary, unique, auto increment |
| username | Varchar(30) | Unique, not null |
| password | Varchar(255) | Not null |
| role\_id | int |  |
| Email | Varchar(50) | unique |

Tokens table

|  |  |  |
| --- | --- | --- |
| id | Varchar(255) | Primary, unique, not null |
| User\_id | int | Not null |
| token | Varchar(255) |  |



**Navigation and Routing**

Allows website admin to define create custom menus and display them on predefined position on the website. Programmers can also use it as routing system in their code.

Database

|  |  |
| --- | --- |
| name | Primary key, unique |
| menu | Text, json |



**Transaction system**

It processes user transactions, and handles user payments.

Database

|  |  |  |
| --- | --- | --- |
| Id | Int | Primary key |
| User\_id | Int | Foreign key |
| Time | Date |  |
| paid | Boolean |  |
| total | double |  |
| Items | Text | In json format |

