**ONLINE MOVIE TICKET BOOKING**

By

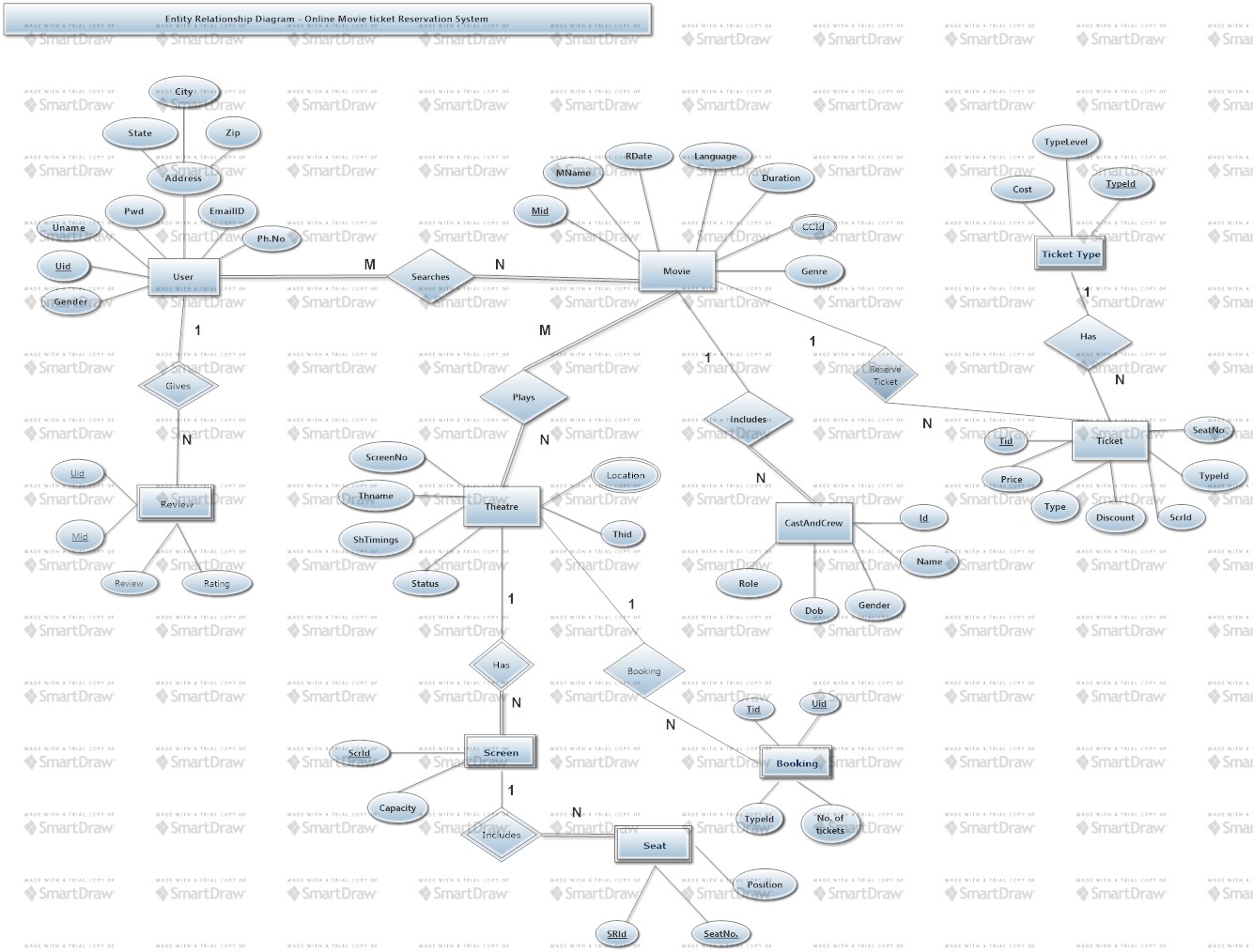
Shiva Kumar Pentyala

Ritesh Kumar Reddy Kuchukulla

Sundeep Kumar Allampally

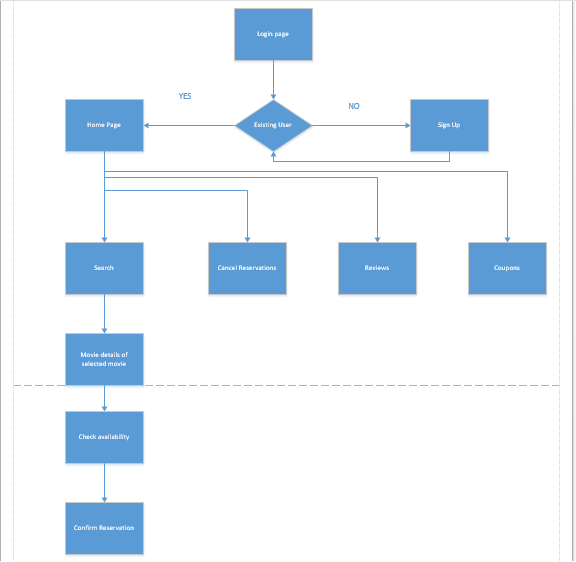
Sandeep Blv

**E-R Diagram:**



**Tool Used:Smart Draw**

**Front End Design Idea:**

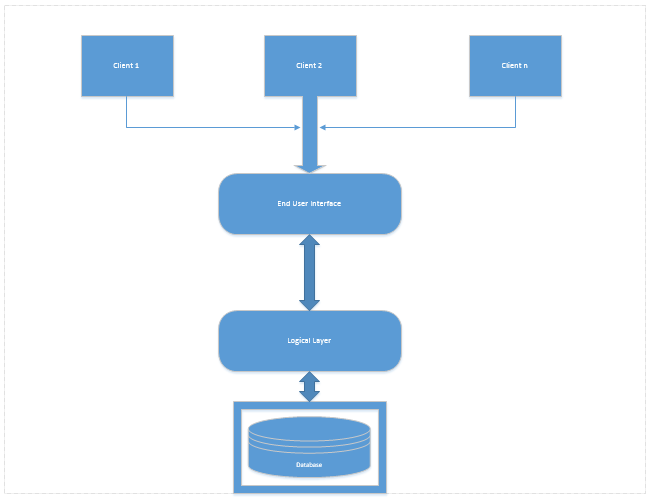


**Tool Used : Microsoft Visio.**

Online Movie Ticket booking, serves as an all in one system which deals with most of the aspects of current day online ticket booking. For any application the look and feel is very important and here goes the front end design idea for our application.

* As soon as the client enters the application the patron needs to login. If the patron is new then he first signs up and then gets logged in.
* In the home page, the client has various options to go through like booking ticket, cancelling ticket reservation , seeing reviews , coupons etc.,
* While booking the ticket, the front end will take the location and display the theaters accordingly. After that the client moves with the desired show and then the availability of seats will be shown.
* The wanted seats are hence reserved by the client.
* HTML5, JSP and CSS are used to design the webpages and javascript for validation purpose.
* We will have many dynamic pages in our application where the patrons will get up to date information.
* In the middleware we use Servlets.
* The JDBC-ODBC bridge drivers are used to connect to the backend database which is supposed to be Sql database.

**Database Architecture:**



**Tool Used: Microsoft Visio**

**USER**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| UID | UNAME | EMAIL | GENDER | ADDRESS | DOB | PASSWORD | Phone |

PRIMARY KEY: UID

NOTE: We take only one Email, phone and address per person.

**MOVIE**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| MID | MOVIENAME | CCID | LANGUAGE | MOVIE\_RATING | RELEASE\_DATE | DURATION | genre |

PRIMARY KEY: MID

**THEATER**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| THID | THNAME | MID | ScreeRoomID | SHOWTIMINGS | LOCATION | STATUS |

PRIMARY KEY: THID, FOREIGN KEY: MID

**SCREENROOM**

|  |  |
| --- | --- |
| SRID | CAPACITY |

PRIMARY KEY: SRID

**SEAT**

|  |  |  |
| --- | --- | --- |
| SEATID | SRID | POSTION |

PRIMARY KEY: {SEATID, SRID}

FOREIGN KEYS : SRID from SCREENROOM

**TICKET**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| TID | MID | TYPE | DISCOUNTS | SRID | SEATID | TYPEID |

PRIMARY KEY: TID

FOREIGN KEYS : MID is referenced from MOVIE table ; SRID is referenced from SCREENROOM table;SEATID is refrenced from SEAT;TYPEID is refrenced from Ticket\_type.

**TICKET\_TYPE**

|  |  |  |
| --- | --- | --- |
| TYPEID | TYPE\_description | COST |

PRIMARY KEY: {TYPEID}

**BOOKING**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| BID | USERID | TID | TYPEID | No\_Of\_Tickets |

PRIMARY KEY: BID

**REVIEW**

|  |  |  |  |
| --- | --- | --- | --- |
| USERID | MID | REVIEW | RATING |

**CASTS\_CREW**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | NAME | DOB | GENDER | ROLE |

PRIMARY KEY: ID

**DATA DEPENDICIES**

userID ->{username,address,emailID,gender,phone,password}

{mid,moviename}->{genre,ccid,language,movie\_rating,release\_date,duration)

Thid ->{mid,thname,srid,showtimings,location,status}

{seated,srid}->position

{bid,userid,tid) ->{typeid,no\_of\_tickets}

Typeid->{type\_description,cost}

{userid,mid} ->{review,rating}

ID ->{name,dob,gender,role}

**RELATIONAL ALGEBRAIC STATEMENTS:**

* Search for movie using language as input

ΠMID,MOVIE\_NAME(σ language=’input’(MOVIE))

* Search for movies using location as input

Result1 ->Πmid(σ location=’input’(Theater))

Result2->Πmid,moviename(Movie)

Result3->Result1 ∞ Result2

Movie\_location->Πmoviename(Result3)

* Search for movies using genre as input

Πmoviename(σ genre=’input’(Movie))

* Retrieve movies using rating as input

Πmoviename(σ rating=’input’(Movie))

* Search for movies along with duration

Πmoviename,duration(Movie)

* Search for availability of movie in particular location

Result1->Πmid,moviename(σ moviename=’input’(Movie))

Result2 ->Πmid,status(σ location=’input’(Theater))

Result3 ->Result1 ∞ Result2

Availabilty\_movie ->Πmoviename,status(Result3)

* Search for showtimings for particular movie

Result1->Πmid,moviename(σ moviename=’input’(Movie))

Result2->Πmid,showtimings(Theater))

Result3 ->Result1 ∞ Result2

Showtimings\_movie->Πmoviename,showtimings(Result3)

* Search for theater w.r.t movie & location

Result1->Πmid(σ moviename=’input’(Movie))

Result2 ->Πmid,theatername(σ location=’input’(Theater))

Result3 ->Result1 ∞ Result2

Theaters ->Πtheaternames(Result3)

**DDL & DML Commands:**

**DDL->** CREATE

Create table **movie**(mid varchar2(10),moviename varchar2(30),ccidint(10),release\_datedate,durationtimestamp,PRIMARY KEY(mid))

Create table **theater**(thid varchar2(10),thname varchar2(30),mid varchar2(10),sridint(10),show\_timings varchar2(30),location varchar2(50),status varchar2(10),PRIMARY KEY(thid) ,FOREIGN KEY(SRID) SCREENROOM(SRID))

Create table **screenroom**(sridint(10),capacity int(20) ,PRIMARY KEY(srid))

Create table **seat**(seated int(10),sridint(10),postion int(20),PRIMARY KEY(srid),FOREIGN KEY(srid) SCREENROOM(srid))

Create table **TICKET**(tid int(10),mid varchar(10),type varchar(10),discounts varchar(10),srid int(10), seatid int(10),typeid int(10) ,PRIMARY KEY(tid),  FOREIGN KEY(mid) MOVIE(mid), FOREIGN KEY(SRID) SCREENROOM(SRID), FOREIGN KEY(seatid) SEAT(seatid), FOREIGN KEY(typeid) TICKET\_TYPE(typeid))

 Create table **TICKET\_TYPE**(type\_id int(10),type\_description varchar(10),cost int(10), PRIMARY KEY(type\_id int(10)))

 Create table **BOOKING**(bid int(10),userid int(10),tid int(10),typeid int(10),no\_of\_ticket,s int(10), PRIMARY KEY(type\_id))

 Create table **REVIEW**(userid int(10),mid int(10),review varchar(100),rating int(10), PRIMARY KEY(userid), FOREIGN KEY(mid) MOVIE(mid))

 Create table **CAST\_CREW**(id int(10),name varchar(10),dob date,gender varchar(10),role varchar(10),PRIMARY KEY(id))

**DML->**{INSERT,UPDATE}

Insert into **movie** values(‘m1’,’2012’,’cc1’,’2012-01-01’,’02:36:55’)

Insert into **movie** values(‘m2’,’2012’,’cc2’,’2012-01-01’,’02:36:55’)

Insert into **movie** values(‘m3’,’2012’,’cc4’,’2012-01-01’,’02:36:55’)

Insert into **theater** values(‘th1’,’max’,’m1’,’s1’,’2:30-5:06:55’,’kc,mo 64112’,’available’)

Insert into **theater** values(‘th2’,’max’,’m1’,’s1’,’2:30-5:06:55’,’kc,mo 64112’,’available’)

Insert into **theater** values(‘th3’,’max’,’m1’,’s1’,’2:30-5:06:55’,’kc,mo 64112’,’available’)

Insert into screenroom values(101,200)

Insert into screenroom values(102,200)

Insert into seat values(1,101,’corner’)

Insert into seat values(2,101,’middle’)

Insert into Ticket values(101,’m101’,’platinum’,’30%’,101,1,’t1’)

Insert into Ticket values(102,’m102’,’platinum’,’30%’,101,1,’t2’)

Insert into ticket\_type values(‘t1’,’platinum’,’30$’)

Insert into ticket\_type values(‘t2’,’gold’,’20$’)

Insert into booking values(‘b1’,101,’t1’,’t101’,10)

Insert into review values(101,’m1’,good,’5’)

Insert into cast\_crew values(c1,’Chris evens’,’01-01-1985’,’M’,’hero’)

Set of operations we are going to perform:

* **Search for movie using language as input**

Select language from movie where language=’input’

* **Search for movies using location as input**

Select t.location from movie m ,theater t where m.mid=t.tid and m.moviename=’input’

* **Search for movies using genre as input**

Select genre from movie where genre = ‘input’.

* **Retrieve movies using rating as input**

Select moviename , movie\_rating from movie where movie\_rating =’input’

* **Search for movies along with duration**

Select moviename , duration from movie where moviename =’input’.

* **Search for availability of movie in particular movie**

Select m.moviename , t.status from movie m , theater t where t.mid=m.mid and m.moviename=’input’

* **Search for showtimings for particular movie**

Select m.moviename , t.showing from movie m , theater t where t.mid=m.mid and m.moviename=’input’

* **Search for theater w.r.t movie and location**

Select t.theatername from movie m , theater t where m.moviename =’input’ and t.location =’input’.

**Contribution:**

SujithaOnteru: E-R Diagram,Part of Data Dependency

Sindhu Koneru: Front End Design,Part of Database Architecture.

JeevanaTunuguntla: Database Architecture,Relational Algebra

SowmyaKamaraju: Data dependency, DML & DDL Operations.

**References:**

<http://www.cse.ohio-state.edu/~gurari/course/cse670/cse670Ch2.xht>

<http://en.wikipedia.org/wiki/Data_flow_diagram>

<http://www.imdb.com/>

<http://en.wikipedia.org/wiki/Codd's_12_rules>