

**Information system and Data Modeling**

**Assignment**

Topic : Online Property Sales System

Group no : MLB\_01.02\_09

Campus : Malabe

Submission Date : 15/05/2021

|  |  |
| --- | --- |
| **Registration No** | **Name** |
| IT20614676 | Sandeepa U.W.P.S |
| IT20623036 | jayashan h.s.p.c |
| IT20618872 | thisitha k.l.d |
| IT20618490 | Gunawardane p.d.t.r |
| IT20620066 | weththasinghe m.n.g |

01)

* **Hypothetical Scenario**

“New Door” is a well-known property sale system who have large number of Properties such as Lands, Houses, Apartments all over the Sri Lanka. Through our system, client can buy new lands, new houses, apartments as well as they can quickly sell their properties without any hindrance. In addition to that they can report their issues & problems to us without any hesitation.

Our system interface environment and functionality are very user friendly. So, system users can achieve their own goal easily and individually. According to our system functionality system users can be register with our system within few minutes through minimum documentation. Then after users can login to our system with their own username and password. Also, they can maintain their own user profile. They can update their profile as well as they can delete their profile. Users can browse our all advertisement without any payment. If users want any ads to watch again, compare with another one or take a decision in future they can add their favorites ads to wish list. After adding to the wish list users can view him favorites properties one by one then after he can choose the best one and go for a visit to the property.

When some users want to publish ad with our system, they can fill the details and uploads photos and submit their ads. After the system admin approval & payment verified user ad will appears on our system. If customer have any question, they can contact us our admins and management through our contact us page and feedback page.

System admin is the most powerful person in our system he can add, delete, manage our system and further he also can control the user activities in our system. System developers update our system day by day adapt to rapidly changing modern world.

**02**

* **Requirement Analysis.**
* Functional Requirements.

Visitor, Member, Administrator can access the system, but visitor cannot access everything in the system. Member can access and use system areas more than a visitor. Administrator is the most valuable and powerful person according to the system he can access all over the system as well as he can manage and control the member activities.

**Visitor Requirements:**

* Visitor can watch ads, visit to about as page as well as he can watch contact as page.
* Visitor can register the system. (then he became a member).
* Visitor needs to give valid details to the system during the registration period.
* Visitor must have a valid email address and a mobile no to the registration.
* Visitor want to verify his email and mobile no during the registration period.

**Member Requirements:**

* Member needs to login the system to access own profile and wish list.
* Member needs to use his own username and password to log in to the system.
* Member can publish ads in the system.
* Member needs to fill the ads details and submit to the administrator before the publish the ads.
* Member can access the payment gateways.
* Member needs to verify his payments details.
* Member can give feed back about the system as well as he can request a contact request to management.
* Member can submit their issues to administrator.
* Member can edit, manage, delete, update own profile.

**Administrator Requirements:**

* Administrator needs to manage the whole system.
* Administrator needs to check member feedbacks.
* Administrator needs to support member issues.
* Administrator needs to manage and attention to member activities in the system.
* Administrator can give the approval for the member’s ads administrator must verify the details about the ads before the giving approval.
* Administrator must want to verify members payment details.
* Administrator needs to manage ads, delete old ads, upload new ads to the system.
* Administrator must generate reports about the whole system.
* Administrator must find the bugs in the system and debug them and as well as update the system with new technology and modern way.

**System Requirements:**

* System needs to verify visitor’s email and mobile no through a verification code.
* System needs to create profile automatically after the visitor submit the registration.
* System must store the member Wishlist items.
* System needs to store member details.
* System needs to store ads and details.
* System needs to send ads to administrator for verification and after verification System must update with new ads automatically.
* System wants to store member’s feedbacks.
* System needs to verify the member username and password when the log in step.
* System needs to store member issues and show them to administrator.
* Non-Functional Requirements.

**Storage:**

* System must have sufficient storage to store member’s details, ads etc.
* System must have sufficient storage to register new members.

**Speed:**

* System must have a good data communication speed among the user and the system.
* System must support to concurrent access for the members.
* System do not get any delay with dealing the payment gateways.

**Security:**

* System must find the unauthorized access and avoid that.
* System must secure the payment gateways and members credit debit card details.

**Other:**

* User interfaces must be very user friendly.
* Members allows to log in to the system any time ( 24 x 7 x 365).
* System must remember the member preferences and show the ads related to him.

Data requirements for the website

Member

* Member ID (member\_ID)
* First name(F\_name)
* Second name(S\_name)
* Date Of Birth(dob)
* E-mail address(Email)
* House Number(m\_house\_num)
* Street(m\_street)
* Postal code(m\_Post\_code)

Property

* Property ID(property\_id)
* Price (Price)
* Type of property (property\_type)
* Location( location)

Payment

* Payment ID(Payment\_id)
* Type of payment(payment\_type)

Visitor

* Visitor ID (visitor\_id)
* First name (first\_n)
* Second name( second\_n)
* Date of birth (dob)
* House Number(v\_house\_num)
* Street(v\_street)
* Postal code(v\_post\_code)

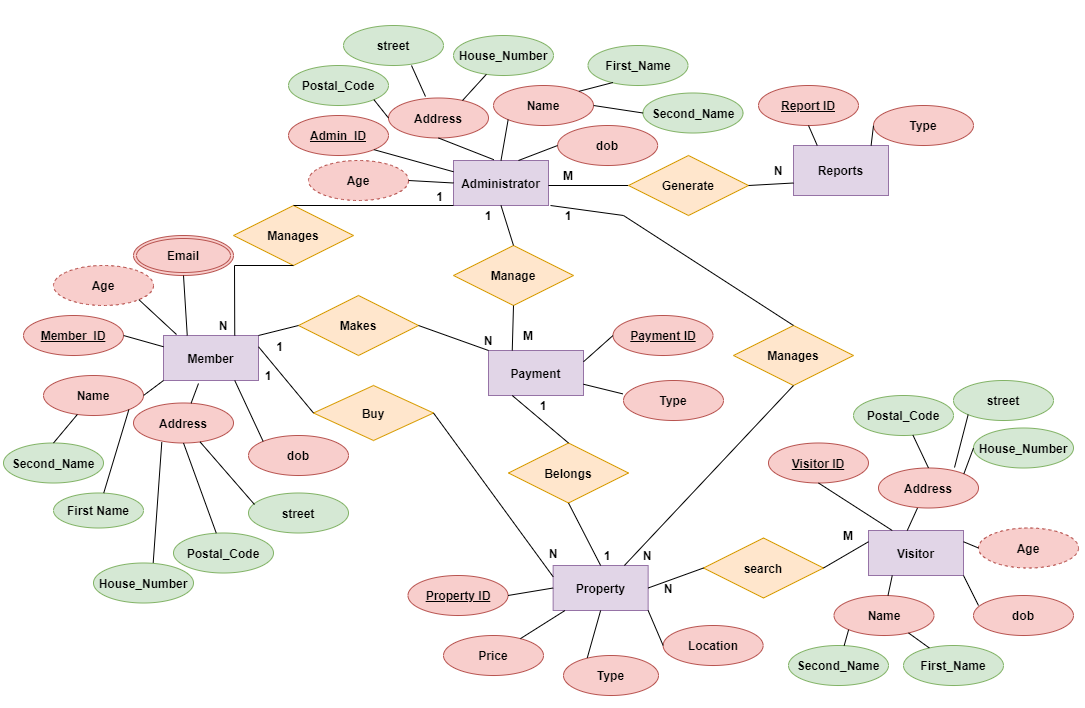
Admin

* Admin ID(admin\_id)
* First name(first\_name)
* Second name(second\_name)
* House Number(a\_house\_num)
* Street(a\_street)
* Postal code(a\_post\_code)

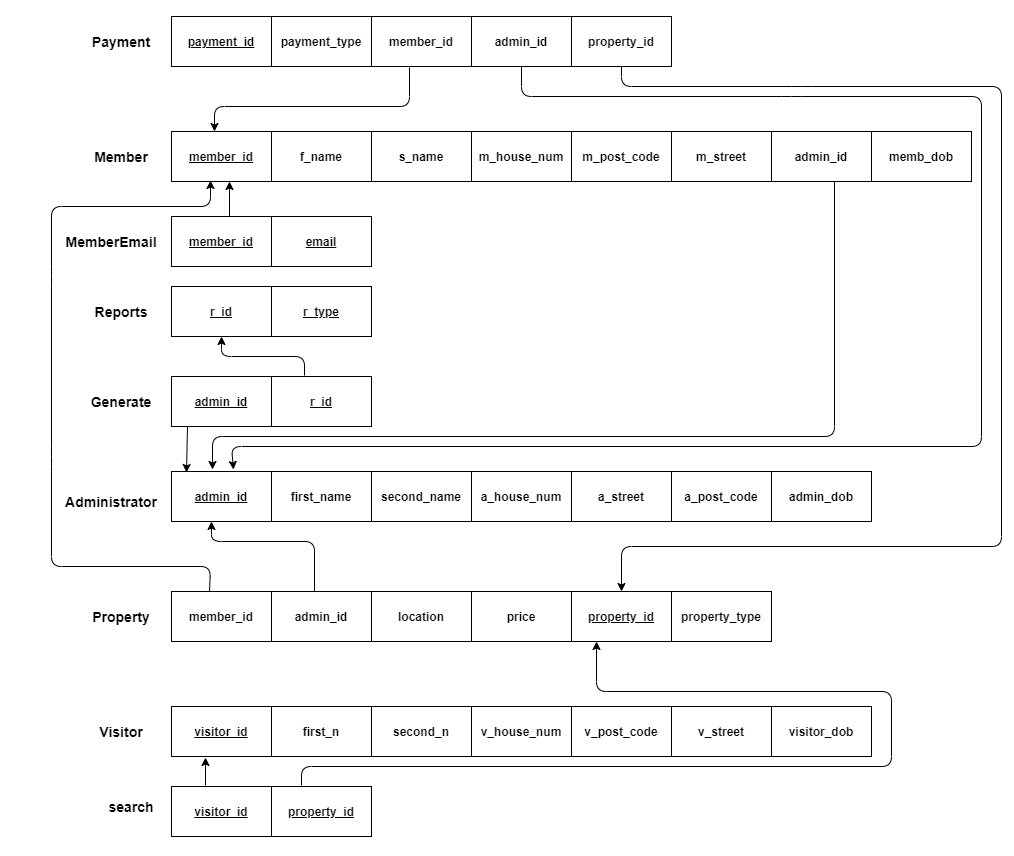
Report

* Report ID(r\_id)
* Type of report( r\_type)

**ER DIAGRAM OF ONLINE PROPERTY SALES SYSTEM**



**RELATIONAL SHEMA FOR THE DATABASE**



**03.DATABASE IMPLEMENTATION**

--Admin Table--

CREATE TABLE administrator(

admin\_id char(10)not null, --Admin Id--

first\_name char(20)not null, --Admin's First Name--

second\_name char(20)not null, --Admin's Second Name--

a\_house\_num char(10)not null, --Admin's House Number--

a\_street varchar(50)not null, --Admin's street--

a\_post\_code int not null, --Admin's Postal Code--

dob date, --Admin's Dath of Birth--

constraint admin\_pk PRIMARY KEY (admin\_id)

);

--Records Of Admin Table--

insert into administrator values('A21456','Suran','Sandeepa','A21','Ambalangoda',150,'1998/07/28');

insert into administrator values('A22456','Chamod','Subasingha','C9','Benthara',250,'1999/05/14');

insert into administrator values('A23456','DD','Dayan','B45','Dabhana',350,'1999/11/21');

insert into administrator values('A24456','GP','Tashen','S91','Malabe',450,'1985/08/12');

insert into administrator values('A25456','Kush','Minuka','F11','Thanamalwila',550,'1999/06/23');

--Member Table--

CREATE TABLE member(

member\_id char(10)not null, --Members ID--

f\_name char(20)not null, --Member's First Name--

s\_name char(20)not null, --Member's Second Name--

m\_house\_num char(10)not null, --Member's House Number--

m\_street varchar(50)not null, --Member's street--

m\_post\_code int not null, --Member's Postal Code--

memb\_dob DATE, --Member's Dath of Birth--

admin\_id char(10)not null, --Admin's Id Foreign key--

constraint member\_pk PRIMARY KEY (member\_id),

constraint member\_fk1 FOREIGN KEY (admin\_id) references administrator(admin\_id)

);

--Records Of Member Table--

insert into member values('RG123','Pasindu','Suran','A22','Balapitiya',21,'1998/07/27','A21456');

insert into member values('RG134','jayashan','Subasingha','C10','Nugegoda',22,'1999/03/12','A22456');

insert into member values('RG145','Thisitha','DD','B46','Nittabuwa',23,'1999/09/11','A22456');

insert into member values('RG156','GP','Nandana','S92','Panadura',24,'1999/03/23','A23456');

insert into member values('RG167','Navodya','Kush','F11','Elpitiya',25,'1999/06/04','A21456');

--Property Table--

CREATE TABLE property(

property\_id char(10)not null, --Property ID--

price int not null, --Price--

property\_Type char(20) not null, --Property Type--

pro\_Location char(30) not null, --Property Location--

member\_id char(10) not null, --Member's Id foreign key--

admin\_id char(10) not null, --Admin's Id Foreign key--

constraint property\_pk PRIMARY KEY (property\_id),

constraint property\_fk1 FOREIGN KEY (member\_id) references member(member\_id),

constraint property\_fk2 FOREIGN KEY (admin\_id) references administrator(admin\_id)

);

--Records Of Property Table--

insert into property values('P21',90000000,'House','Nugegoda','RG123','A21456');

insert into property values('P22',80000000,'Land','Rathmalana','RG134','A22456');

insert into property values('P23',50000000,'Land','Wallawaththa','RG145','A22456');

insert into property values('P24',60000000,'Apartment','Boralasgamuwa','RG156','A23456');

insert into property values('P25',70000000,'House','Dehiwala','RG167','A21456');

--Visitor Table--

CREATE TABLE visitor(

visitor\_id char(10) not null, --Visitor ID--

first\_n char(20) not null, --Visitor's First Name--

second\_n char(20) not null, --Visitor's Second Name--

v\_house\_num char(10) not null, --Visitor's House Number--

v\_street varchar(50)not null, --Visitor's street--

v\_post\_code int not null, --Visitor's Postal Code--

visitor\_dob DATE, --Visitor's DAth of Birth--

constraint visitor\_pk PRIMARY KEY (visitor\_id)

);

--Records Of visitor Table--

insert into visitor values('UR2','Iraj','Weerarathna','A1','Deniyaya',111,'1985/06/21');

insert into visitor values('UR3','Arnold','Subasingha','B2','Bandaragama',222,'1987/07/22');

insert into visitor values('UR4','Kasun','Chamara','C4','Horana',333,'1986/06/23');

insert into visitor values('UR5','Chandana','Rupesinghe','V5','Benthara',444,'1985/05/25');

insert into visitor values('UR6','Nihal','Gamage','N8','Malabe',555,'1992/03/21');

--Payment Table--

CREATE TABLE payment(

payment\_id char(10)not null, --Payment ID--

pay\_type char(10)not null, --Payment Type--

member\_id char(10)not null, --Member Id foreign key--

admin\_id char(10)not null, --Admin ID foreign key--

property\_id char(10)not null, --Property ID foreign key--

constraint payment\_pk PRIMARY KEY (payment\_id),

constraint payment\_fk1 FOREIGN KEY (member\_id) references member(member\_id),

constraint payment\_fk2 FOREIGN KEY (admin\_id) references administrator(admin\_id),

constraint payment\_fk3 FOREIGN KEY (property\_id) references property(property\_id)

);

--Records Of Payment Table--

insert into payment values('B14','cash','RG123','A25456','P21');

insert into payment values('B25','check','RG134','A25456','P22');

insert into payment values('B36','debit card','RG145','A25456','P23');

insert into payment values('B47','cash','RG156','A25456','P24');

insert into payment values('B58','check','RG167','A25456','P25');

--Report Table--

CREATE TABLE report(

r\_id char(10)not null, --Report ID--

r\_type char(30)not null, --Report Type--

constraint report\_pk PRIMARY KEY (r\_id)

);

--Records Of report Table--

insert into report values('R1','payments');

insert into report values('R2','payments');

insert into report values('R3','sales');

insert into report values('R4','registered members');

insert into report values('R5','sales');

--Generate Table--

CREATE TABLE generate(

admin\_id char(10)not null, --Admin ID foreign key--

r\_id char(10)not null, --Report ID foreign key--

constraint generate\_pk PRIMARY KEY (admin\_id,r\_id),

constraint generate\_fk1 FOREIGN KEY (admin\_id) references administrator(admin\_id),

constraint generate\_fk2 FOREIGN KEY (r\_id) references report(r\_id)

);

--Records Of generate Table--

insert into generate values('A25456','R1');

insert into generate values('A25456','R2');

insert into generate values('A25456','R3');

insert into generate values('A25456','R4');

insert into generate values('A25456','R5');

--Search Table--

CREATE TABLE search(

visitor\_id char(10)not null, --Visitor ID foreign key--

property\_id char(10)not null, --Property ID foreign key--

constraint search\_pk PRIMARY KEY (visitor\_id,property\_id),

constraint search\_fk1 FOREIGN KEY (visitor\_id) references visitor(visitor\_id),

constraint search\_fk2 FOREIGN KEY (property\_id) references property(property\_id)

);

--Records Of search Table--

insert into search values('UR2','P21');

insert into search values('UR3','P22');

insert into search values('UR4','P23');

insert into search values('UR5','P24');

insert into search values('UR6','P25');

--MemberEmail Table--

CREATE TABLE MemberEmail(

member\_id char(10)not null, --Member ID foreign key-

email varchar(50)not null, --Member Email--

constraint MemberEmail\_pk PRIMARY KEY (member\_id,email),

constraint MemberEmail\_fk1 FOREIGN KEY (member\_id) references member(member\_id)

);

--Records Of MemberEmail Table--

insert into MemberEmail values('RG123','pasindu@gmail.com');

insert into MemberEmail values('RG134','subasingha@gmail.com');

insert into MemberEmail values('RG145','DD@gmail.com');

insert into MemberEmail values('RG156','GP@gmail.com');

insert into MemberEmail values('RG167','Kush@gmail.com');

* **special performance**
* Registered user can edit, delete, and update his details in user account.
* Registered and Unregistered users both can view common open pages.
* Only registered user can publish ads in the system.
* System admin can delete and control the user account and as well as he can control the user activities.
* All ads want to get system admin approval before the publishing.
* **Security requirements**
* In the registration steps system needs to verify user’s email as well as user’s mobile no. system need to send verification code to mobile and email.
* Unregistered users cannot create account without confirming & verifying his details.
* Unregistered user cannot enter a pre used email for our system again.
* User’s passwords must be strong, and password must be including at least one numerical value & character value.
* Registered user must login to the system before the publish ads.
* System admin can’t change anything and not allowed to control and customize the system without login his admin account.