Use case 4:

Client{

CompanyName varchar(50) not null,

ContactFirstName varchar(50) not null,

ContactLastName varchar(50) not null,

City varchar(50) not null,

State char(2) not null,

Zip char(10) not null

}

Resources{

FirstName varchar(50) not null,

LastName varchar(50) not null,

Position varchar(50) not null,

YearOfExperience INT(50) not null

}

Allocation{

ClientCompanyName varchar(50) not null,

ResourceFirstName varchar(50) not null,

ResourceLastName varchar(50) not null

}

Use case 5,6:

User{

Username: varchar(50) not null,

Password: varchar(50) not null,

SecurityLevel: varchar(50) not null

}

public class User {

MySqlConnection connect;

public User(string username, string email, string[] roles)

{

Username = username;

Email = email; Roles = roles;

}

public string Username { get; set; }

public string Password { get; set; }

public string SecurityLevel { get; set; }

public bool UserAuthorization(){

connect.Open();

if(this.Username==”” || this.Password ==””)

{

MessageBox.Show("Username and Password should not be empty！");

return false;

}

string sql = "select \* from User where Username = '" + Username + "' and Password = '" + Password + "'";

MySqlCommand cmd = new MySqlCommand(sql, connect);

MySqlDataReader read = cmd.ExecuteReader();

If(read.Read()){

MessageBox.Show(“Login successfully”);

return true;

}

MessageBox.Show(“Login fail”);

return false;

}

public bool SecurityLevelAuthorization(){

connect.Open();

string sql = "select \* from User where SecurityLevel = PM + Or SecurityLevel = Architect ";

MySqlCommand cmd = new MySqlCommand(sql, connect);

MySqlDataReader read = cmd.ExecuteReader();

If(read.Read()){

return true;

}

return false;

}

}