

MASTERFRIDGE

TEAM MEMBERS

YASİN GÖKÇE ŞAHİN 201511210

EYÜP CAN TURGUT 201411062

BEEBUDDIES



DESCRIPTION

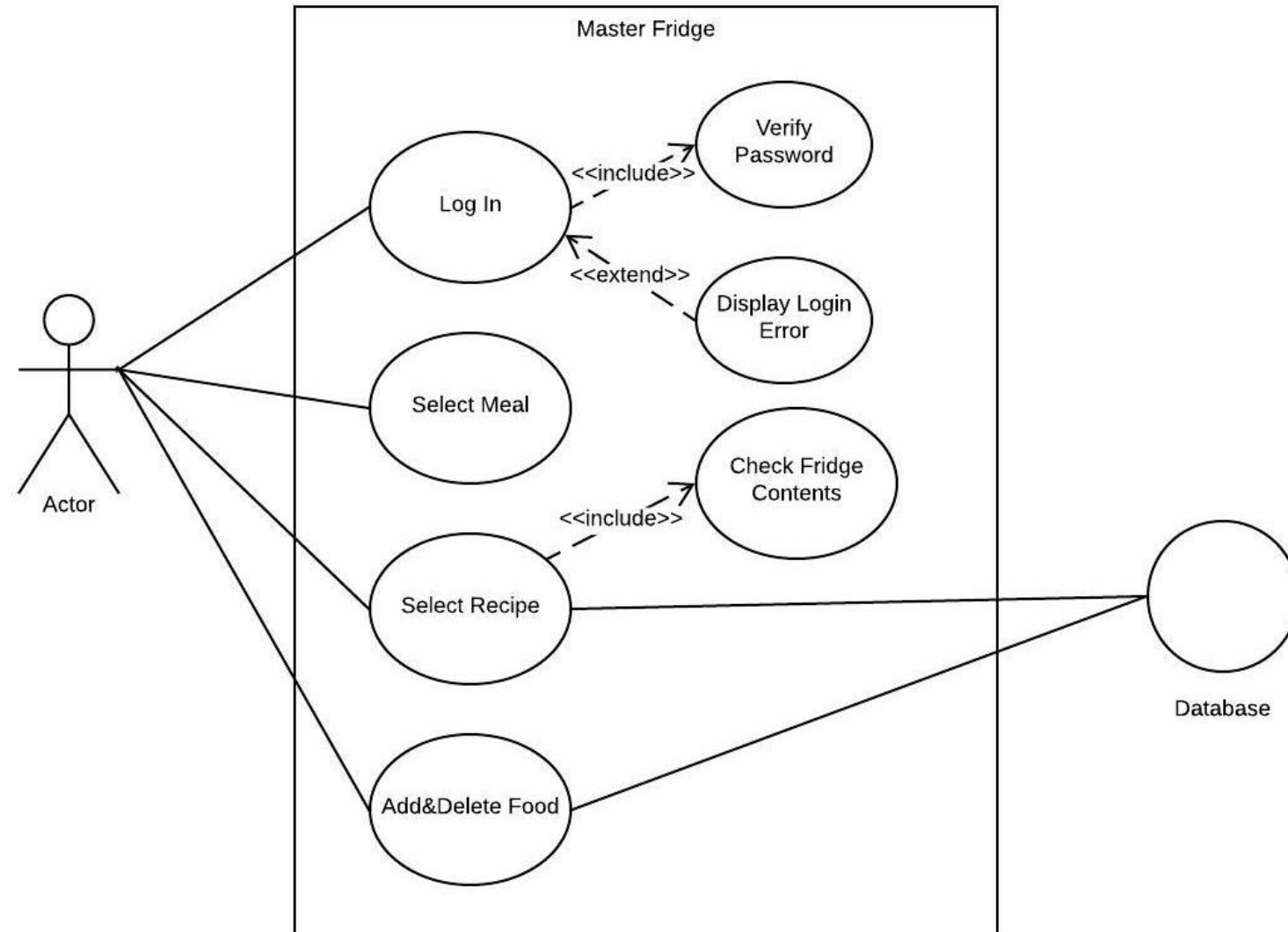
The purpose of our application is to facilitate the process of selecting food for housewives and singles at home. Imagine a system that tells you what you can do according to the materials in your fridge; for example, if you have eggplant, pepper, onion, garlic, tomato and tomato paste, your fridge will offer you to cook şakşuka. The user chooses what kind of meal to prepare and the program presents recipes according to the ingredients in the fridge.



FUNCTIONAL REQUIREMENTS

- This application can work on PC.
- Every user can input their own foods on their fridges.
- When the application is installed, users can create their own accounts.
- Users shall be able to display recipes without registering.
- Users who are enrolled can choose any categories between breakfast, lunch and dinner.

USE CASE MODEL

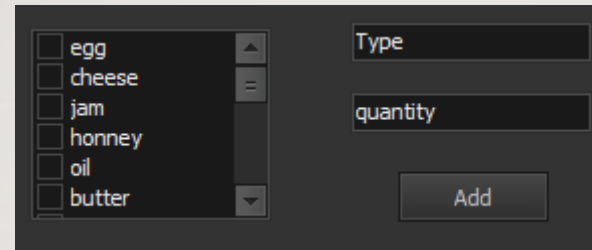


SUBSYSTEMS

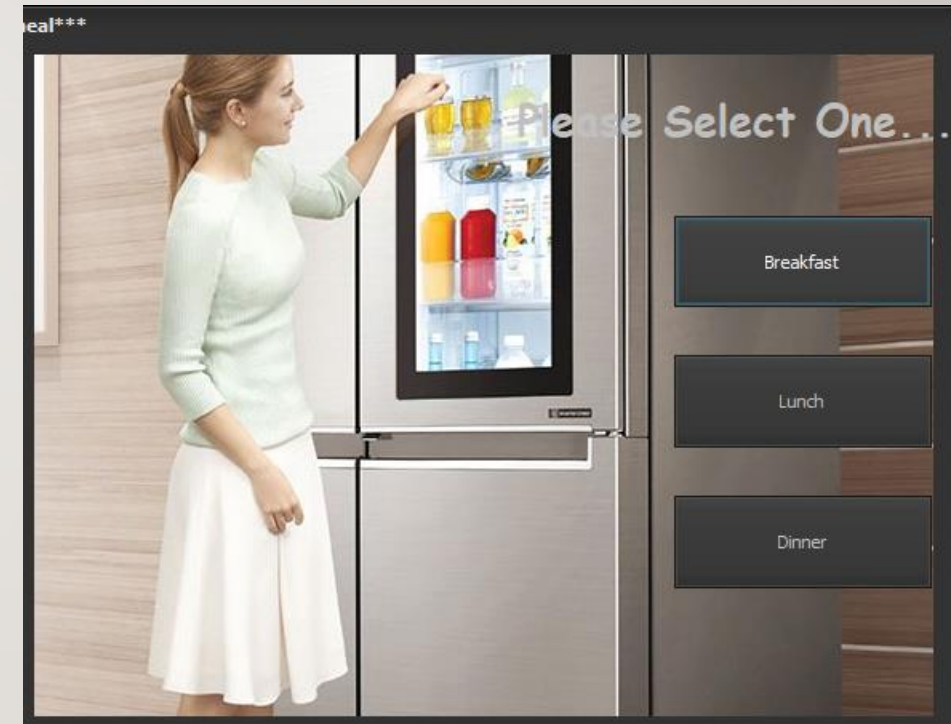
Log-In



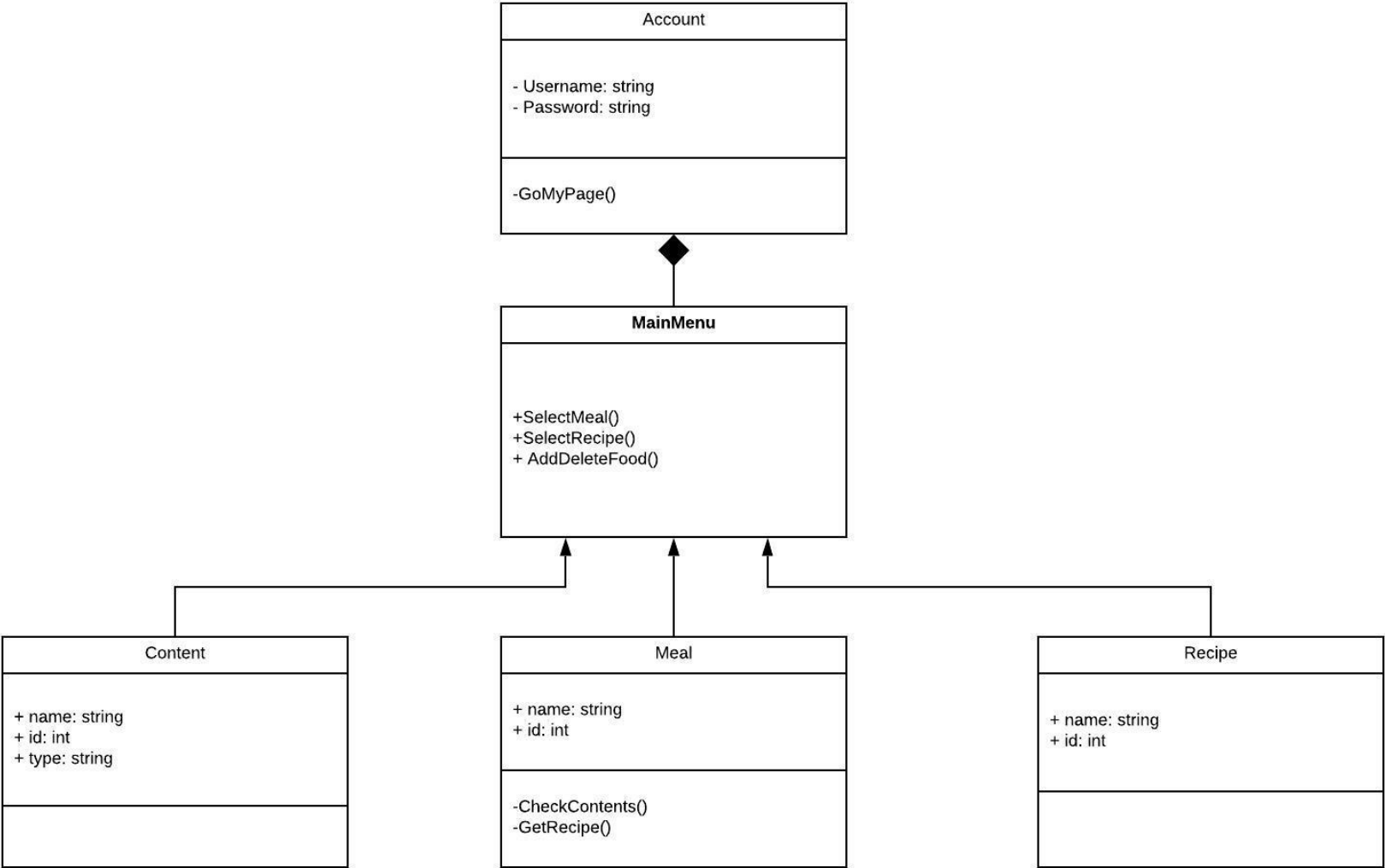
Add/Delete Food



Recipes



CLASS DIAGRAM



DATABASES

Users

	id	name	number
	1	egg	5
	2	olive	4
	3	tomato	7
	4	pepper	3
▶	5	cucumber	5
*	NULL	NULL	NULL

Foods

	id	username	password
	1	kayhan	... 12345 ...
	2	adil	... 5646 ...
	3	mustafa	... 2634 ...
	6	eyupcan	... 1234 ...
▶	7	yasin	... 1234 ...
*	NULL	NULL	NULL

CODE EXAMPLES

Registering

```
begin
    SMTP := TIdSMTP.Create(nil);
    DATA := TIdMessage.Create(nil);
    SSL := TIdSSLIOHandlerSocketOpenSSL.Create(nil);

    SSL.SSLOptions.Method := sslvTLSv1;
    SSL.SSLOptions.Mode := sslmUnassigned;
    SSL.SSLOptions.VerifyMode := [];
    SSL.SSLOptions.VerifyDepth := 0;

    DATA.From.Address := username;
    DATA.Recipients.EmailAddresses := totarget;
    DATA.subject := subject;
    DATA.body.text := body;

    SMTP.IOHandler := SSL;
    // SMTP.Host := 'smtp.gmail.com';
    // SMTP.Port := 587;
    SMTP.username := username;
    SMTP.Password := Password;
    SMTP.UseTLS := utUseExplicitTLS;

    SMTP.Connect;
    SMTP.Send(DATA);
    SMTP.Disconnect;

    SMTP.Free;
    DATA.Free;
    SSL.Free;
end;
```

Meal

```
begin
    with DataModule1.AdoQuery1 do
    begin
        if (Trim(typemeal.Text) = EmptyStr) or (Trim(quantity.Text) = EmptyStr) then
        begin
            ShowMessage('please enter the red area.');
            SRCEdit(typemeal);
            SRCEdit(quantity);
        end
        else
        begin
            ShowMessage('succeeded.');
            sql.Text :=
                'insert into breakfast (name,number) values (:b,:c)';
            Parameters.ParamByName('b').Value := typemeal.Text;
            Parameters.ParamByName('c').Value := quantity.Text;
            ExecSQL;
            close;
            sql.Text := 'select * from breakfast';
            open;
        end;
    end;
    close;
end;
```