

Báo cáo cuối kì

Môn : lập trình web với asp.net

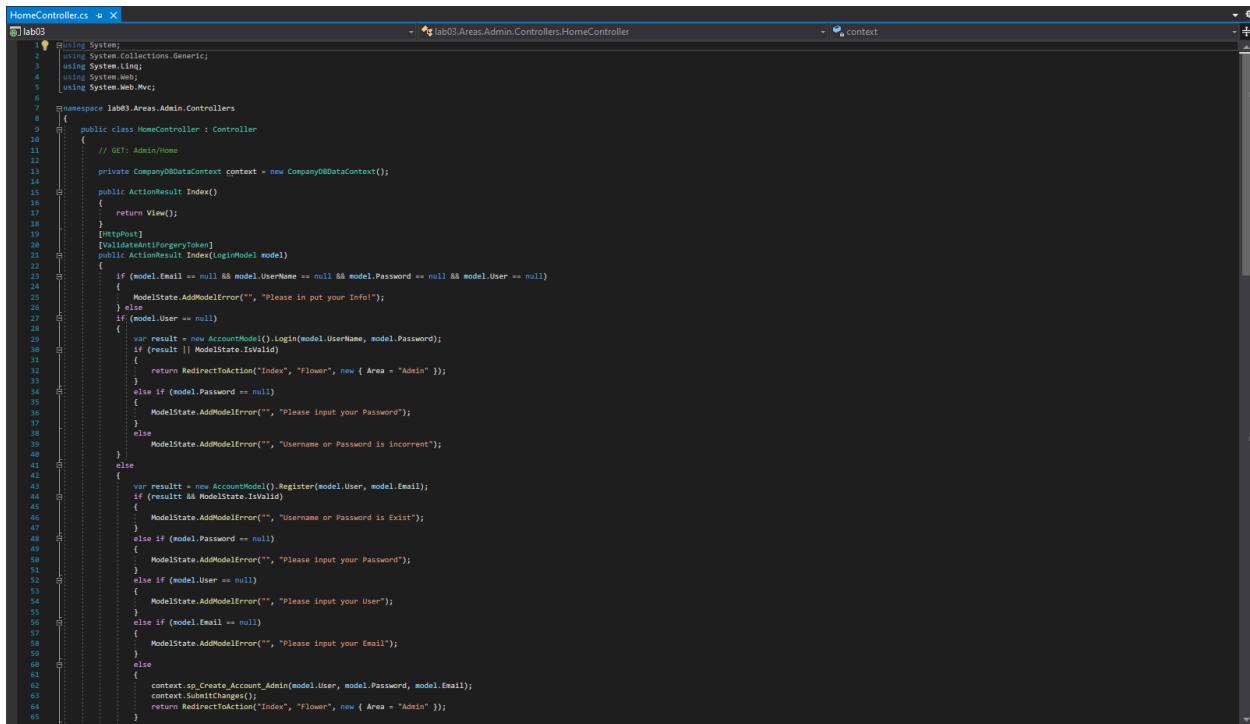
Lab 06

Quản lý flower shop

1. Scource Code:

a. Admin_Areas

- HomeController



```
HomeController.cs  lab03  lab03.Areas.Admin.Controllers.HomeController  context

1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.Mvc;
6
7  namespace lab03.Areas.Admin.Controllers
8  {
9      public class HomeController : Controller
10     {
11         // GET: Admin/Home
12
13         private CompanyDBDataContext context = new CompanyDBDataContext();
14
15         public ActionResult Index()
16         {
17             return View();
18         }
19
20         [HttpPost]
21         [ValidateAntiForgeryToken]
22         public ActionResult Index(LoginModel model)
23         {
24             if (model.Email == null && model.UserName == null && model.Password == null && model.User == null)
25             {
26                 ModelState.AddModelError("", "Please input your Info!");
27             }
28             if (modelUser == null)
29             {
30                 var result = new AccountModel().Login(model.UserName, model.Password);
31                 if (result || ModelState.IsValid)
32                 {
33                     return RedirectToAction("Index", "Flower", new { Area = "Admin" });
34                 }
35                 else if (model.Password == null)
36                 {
37                     ModelState.AddModelError("", "Please input your Password");
38                 }
39                 else
40                 {
41                     ModelState.AddModelError("", "Username or Password is incorrect");
42                 }
43             }
44             else
45             {
46                 var result = new AccountModel().Register(model.User, model.Email);
47                 if (result && ModelState.IsValid)
48                 {
49                     ModelState.AddModelError("", "Username or Password is Exist");
50                 }
51                 else if (model.Password == null)
52                 {
53                     ModelState.AddModelError("", "Please input your Password");
54                 }
55                 else if (model.User == null)
56                 {
57                     ModelState.AddModelError("", "Please input your User");
58                 }
59                 else if (model.Email == null)
60                 {
61                     ModelState.AddModelError("", "Please input your Email");
62                 }
63                 else
64                 {
65                     context.sp_Create_Account_Admin(model.User, model.Password, model.Email);
66                     context.SubmitChanges();
67                     return RedirectToAction("Index", "Flower", new { Area = "Admin" });
68                 }
69             }
70         }
71     }
```

```
66
67     }
68     return View(model);
69 }
70 public ActionResult ListAccount()
71 {
72     CompanyBDataContext context = new CompanyBDataContext();
73     var AccountInfos = context.Accounts.ToList();
74     return View(AccountInfos);
75 }
76 public ActionResult Details(int id)
77 {
78     var databyid = context.Accounts.Single(x => x.Id == id);
79     return View(databyid);
80 }
81 public ActionResult Edit(int id)
82 {
83     var databyid = context.Accounts.Single(x => x.Id == id);
84     return View(databyid);
85 }
86 [HttpPost]
87 public ActionResult Edit(int id, LoginModel account)
88 {
89     try
90     {
91         var data = context.Accounts.Single(x => x.Id == id);
92         data.Email = account.Email;
93         data.Password = account.Password;
94         data.UserName = account.UserName;
95         context.SubmitChanges();
96         return RedirectToAction("ListAccount");
97     }
98     catch
99     {
100         return View();
101     }
102 }
103
104 public ActionResult Delete(int id)
105 {
106     var databyid = context.Accounts.Single(x => x.Id == id);
107     return View(databyid);
108 }
109
110 [HttpPost]
111 public ActionResult Delete(int id, LoginModel account)
112 {
113     try
114     {
115         var data = context.Accounts.Single(x => x.Id == id);
116         context.Accounts.DeleteOnSubmit(data);
117         context.SubmitChanges();
118         return RedirectToAction("ListAccount");
119     }
120     catch
121     {
122         return View();
123     }
124 }
```

[Home](#)[Index](#)

```
Index.cshtml  ▶ HomeController.cs

2  @{
3      ViewBag.Title = "Index";
4  }
5  @model [abb].Areas.Admin.LoginModel
6
7
8
9  <h2>This is admin page</h2>
10 <!DOCTYPE html>
11 <html lang="en">
12     <head>
13         <meta charset="UTF-8">
14         <title> HTML/CSS Login Form</title>
15         <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">
16         <link rel="stylesheet" href="~/Assets/Admin/css/style.css">
17     </head>
18
19     <body>
20
21         <div class="login-page">
22             <div class="form">
23                 <using (Html.BeginForm("Index", "Home", FormMethod.Post, new { @class = "register-form" })>
24
25                     <form class="register-form">
26                         <input type="text" placeholder="name" />
27                         <input type="password" placeholder="password" />
28                         <input type="text" placeholder="email address" />
29                         <Html.AntiForgeryToken()>
30                         <input type="button" value="Sign Up" />
31                     </form>
32                 <using (Html.BeginForm("Index", "Home", FormMethod.Post, new { @class = "login-form" })>
33
34                     <form class="login-form">
35                         <Html.AntiForgeryToken()>
36                         <input type="text" placeholder="Email" />
37                         <input type="password" placeholder="password" />
38                         <input type="button" value="Login" />
39                         <p class="message">Already registered? <a href="#">Sign In</a></p>
40                     </form>
41                 </using>
42             </div>
43         </div>
44
45         <script src="https://cdn.cloudflare.com/ajax/libs/jquery/2.1.3/jquery.min.js"></script>
46
47         <script src="~/Assets/Admin/js/index.js"></script>
48
49
50     </body>
51
52     </html>
53
54     <p class="message">Not registered? <a href="#">Create an account</a></p>
55
56 </div>
57
58 </div>
59
60 <script src="~/Assets/Admin/js/index.js"></script>
61
62
63
64
65
66
67 </body>
```

Edit account

```
1 <Edit>
2     <Index>
3         <Controller>IndexController.cs
4             <Action>Index
5             <Model>Account
6
7             <ViewBag>Title = "Edit";
8             <Layout> ~/Areas/Admin/Views/Shared/Menu_Layout.cshtml;
9
10            <br />
11            <h2>Class="text-primary" Edit:</h2>
12
13            <using> Html.BeginForm()
14            </using>
15            <div>
16                <input> AntiForgeryToken()
17
18                <div class="form-horizontal">
19                    <h4>Class="text-primary" Account:</h4>
20                    <hr />
21                    <label>ValidationSummary(true, new { @class = "text-danger" })
22                    <input> HiddenFor(model => model.Id)
23
24                    <div class="form-group">
25                        <label>LabelFor(model => model.UserName, htmlAttributes: new { @class = "control-label col-md-2" })
26                        <div class="col-md-10">
27                            <input> EditorFor(model => model.UserName, new { htmlAttributes = new { @class = "form-control" } })
28                            <input> ValidationMessageFor(model => model.UserName, "", new { @class = "text-danger" })
29                        </div>
30                    </div>
31
32                    <div class="form-group">
33                        <label>LabelFor(model => model.Email, htmlAttributes: new { @class = "control-label col-md-2" })
34                        <div class="col-md-10">
35                            <input> EditorFor(model => model.Email, new { htmlAttributes = new { @class = "form-control" } })
36                            <input> ValidationMessageFor(model => model.Email, "", new { @class = "text-danger" })
37                        </div>
38                    </div>
39
40                    <div class="form-group">
41                        <label>LabelFor(model => model.FirstName, htmlAttributes: new { @class = "control-label col-md-2" })
42                        <div class="col-md-10">
43                            <input> EditorFor(model => model.FirstName, new { htmlAttributes = new { @class = "form-control" } })
44                            <input> ValidationMessageFor(model => model.FirstName, "", new { @class = "text-danger" })
45                        </div>
46                    </div>
47
48                    <div class="form-group">
49                        <label>LabelFor(model => model.LastName, htmlAttributes: new { @class = "control-label col-md-2" })
50                        <div class="col-md-10">
51                            <input> EditorFor(model => model.LastName, new { htmlAttributes = new { @class = "form-control" } })
52                            <input> ValidationMessageFor(model => model.LastName, "", new { @class = "text-danger" })
53                        </div>
54                    </div>
55
56                    <div class="form-group">
57                        <label>LabelFor(model => model.Password, htmlAttributes: new { @class = "control-label col-md-2" })
58                        <div class="col-md-10">
59                            <input> EditorFor(model => model.Password, new { htmlAttributes = new { @class = "form-control" } })
60                            <input> ValidationMessageFor(model => model.Password, "", new { @class = "text-danger" })
61                        </div>
62                    </div>
63
64                    <div class="form-group">
65                        <label>LabelFor(model => model.AccountType, htmlAttributes: new { @class = "control-label col-md-2" })
66                        <div class="col-md-10">
67                            <input> EditorFor(model => model.AccountType, new { htmlAttributes = new { @class = "form-control" } })
68                            <input> ValidationMessageFor(model => model.AccountType, "", new { @class = "text-danger" })
69                        </div>
70                    </div>
71
72                </div>
73            </div>
74        </div>
75    </div>
76
```

Detailsaccount

```
Details.cshtml <--> Edit.cshtml Index.cshtml HomeController.cs

1 model.IdeasAccount
2
3 @{
4     ViewBag.Title = "Details";
5     Layout = "~/Areas/Admin/Views/Shared/Menu_Layout.cshtml";
6 }
7
8 <div class="container">
9     <h4>Account</h4>
10    <br />
11    <div class="di-horizontal">
12        <dt>
13            @Html.DisplayNameFor(model => model.UserName);
14        </dt>
15        <dd>
16            @Html.DisplayFor(model => model.UserName)
17        </dd>
18
19        <dt>
20            @Html.DisplayNameFor(model => model.Email);
21        </dt>
22        <dd>
23            @Html.DisplayFor(model => model.Email)
24        </dd>
25
26        <dt>
27            @Html.DisplayNameFor(model => model.FirstName);
28        </dt>
29        <dd>
30            @Html.DisplayFor(model => model.FirstName)
31        </dd>
32
33        <dt>
34            @Html.DisplayNameFor(model => model.LastName);
35        </dt>
36
37        <dt>
38            @Html.DisplayNameFor(model => model.LastName);
39        </dt>
40
41        <dd>
42            @Html.DisplayFor(model => model.LastName)
43        </dd>
44
45        <dt>
46            @Html.DisplayNameFor(model => model.Password);
47        </dt>
48
49        <dd>
50            @Html.DisplayFor(model => model.Password)
51        </dd>
52
53        <dt>
54            @Html.DisplayNameFor(model => model.AccountType);
55        </dt>
56
57        <dd>
58            @Html.DisplayFor(model => model.AccountType)
59        </dd>
60
61    </div>
62    <div class="alert-danger">
63        <p>
64            @Html.ActionLink("Edit", "Edit", new { id = Model.Id }, new { @class = "btn btn-info mx-auto" }) | 
65            @Html.ActionLink("Cancel", "ListAccount", new { }, new { @class = "btn btn-danger mx-auto" })
66        </p>
67    </div>

```

- DeleteAccount

```
Delete.cshtml  Details.cshtml  Edit.cshtml  Index.cshtml  HomeController.cs
model1:Lab03.Account
1
2
3
4    ViewBag.Title = "Delete";
5    Layout = "~/Areas/Admin/Views/Shared/Menu_Layout.cshtml";
6
7    <br />
8    <h2 class="text-danger">Delete</h2>
9
10   <h3 class="text-danger">Are you sure you want to delete this?</h3>
11
12   <div class="container">
13       <h4>Account</h4>
14       <hr />
15       <dl class="dl-horizontal">
16           <dt>
17               @Html.DisplayNameFor(model => model.UserName)
18           </dt>
19           <dd>
20               @Html.DisplayFor(model => model.UserName)
21           </dd>
22
23           <dt>
24               @Html.DisplayNameFor(model => model.Email)
25           </dt>
26           <dd>
27               @Html.DisplayFor(model => model.Email)
28           </dd>
29
30           <dt>
31               @Html.DisplayNameFor(model => model.FirstName)
32           </dt>
33
34           <dd>
35               @Html.DisplayFor(model => model.FirstName)
36           </dd>
37
38           <dt>
39               @Html.DisplayNameFor(model => model.LastName)
40           </dt>
41
42           <dd>
43               @Html.DisplayFor(model => model.LastName)
44           </dd>
45
46           <dt>
47               @Html.DisplayNameFor(model => model.Password)
48           </dt>
49
50           <dd>
51               @Html.DisplayFor(model => model.Password)
52           </dd>
53
54           <dt>
55               @Html.DisplayNameFor(model => model.AccountType)
56           </dt>
57
58           <dd>
59               @Html.DisplayFor(model => model.AccountType)
60           </dd>
61
62       </dl>
63
64       <br />
65       @using (Html.BeginForm()) {
66           @Html.AntiForgeryToken()
67     
```

- ListAccount

```
ListAccount.cshtml  Delete.cshtml  Details.cshtml  Edit.cshtml  Index.cshtml  HomeController.cs
model1:Enumerable<Lab03.Account>
1
2
3
4    ViewBag.Title = "ListAccount";
5    Layout = null;
6    Layout = "~/Areas/Admin/Views/Shared/Menu_Layout.cshtml";
7
8
9    <h2 class="text-primary">ListAccount</h2>
10
11   <p>
12       @Html.ActionLink("Create New", "Index", new { }, new { @class="btn btn-success" })
13   </p>
14
15   <table class="table">
16       <tr style="background-color:#highlight">
17           <th>
18               @Html.DisplayNameFor(model => model.UserName)
19           </th>
20           <th>
21               @Html.DisplayNameFor(model => model.Password)
22           </th>
23           <th>
24               @Html.DisplayNameFor(model => model.Email)
25           </th>
26           <th>
27               @Html.DisplayNameFor(model => model.FirstName)
28           </th>
29           <th>
30               @Html.DisplayNameFor(model => model.LastName)
31           </th>
32           <th>
33               @Html.DisplayNameFor(model => model.AccountType)
34           </th>
35       </tr>
36
37       <foreach var item in Model> {
38           <tr>
39               <td>
40                   @Html.DisplayFor(modelItem => item.UserName)
41               </td>
42               <td>
43                   @Html.DisplayFor(modelItem => item.Password)
44               </td>
45               <td>
46                   @Html.DisplayFor(modelItem => item.Email)
47               </td>
48               <td>
49                   @Html.DisplayFor(modelItem => item.FirstName)
50               </td>
51               <td>
52                   @Html.DisplayFor(modelItem => item.LastName)
53               </td>
54               <td>
55                   @Html.DisplayFor(modelItem => item.AccountType)
56               </td>
57
58               <a href="#" class="text-warning" data-toggle="modal" data-target="#EditModal">
59                   @Html.ActionLink("Edit", "Edit", new { id=item.Id }, new { @class = "btn btn-warning btn-xs" })
60               </a> |
61               <a href="#" class="text-info" data-toggle="modal" data-target="#DetailsModal">
62                   @Html.ActionLink("Details", "Details", new { id=item.Id }, new { @class = "btn btn-info btn-xs" })
63               </a> |
64               <a href="#" class="text-danger" data-toggle="modal" data-target="#DeleteModal">
65                   @Html.ActionLink("Delete", "Delete", new { id=item.Id }, new { @class = "btn btn-danger btn-xs" })
66               </a>
67           </tr>
68       } </foreach>
69   </table>
```

- Layout

```
1 <!DOCTYPE html>
2 <html>
3     <head>
4         <meta charset="utf-8" />
5         <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6         <title>ViewBag.Title - My ASP.NET Application</title>
7         <link href="~/Content/bootstrap.min.css" rel="stylesheet" type="text/css" />
8         <link href="~/Content/bootstrap-theme.min.css" rel="stylesheet" type="text/css" />
9         <script src="~/Scripts/modernizr-2.8.3.js"></script>
10    </head>
11    <body>
12        <div class="navbar navbar-inverse navbar-fixed-top">
13            <div class="container">
14                <div class="navbar-header">
15                    <button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-collapse">
16                        <span class="icon-bar"></span>
17                        <span class="icon-bar"></span>
18                        <span class="icon-bar"></span>
19                    </button>
20                    <a href="#" data-actionlink="~/ApplicationName/Index" data-index="0" data-area="" data-galias="<span>My ASP.NET Application</span>">
21                    </a>
22                </div>
23                <div class="navbar-collapse collapse">
24                    <ul class="nav navbar-nav">
25                    </ul>
26                </div>
27            </div>
28        </div>
29        <div class="container body-content">
30            <div>
31                <h1>@Model</h1>
32                <p>@Model</p>
33                <p>&copy; @DateTime.Now.Year - My ASP.NET Application</p>
34            </div>
35        </div>
36        <script src="~/Scripts/jquery-3.4.1.min.js"></script>
37        <script src="~/Scripts/bootstrap.min.js"></script>
38    </body>
39 </html>
```

- Menu_Layout

```
1 <!DOCTYPE html>
2 <html lang="en">
3     <head>
4         <meta charset="utf-8" />
5         <meta http-equiv="X-UA-Compatible" content="IE=edge" />
6         <meta name="viewport" content="width=device-width, initial-scale=1" />
7         <meta name="description" content="" />
8         <meta name="author" content="" />
9         <title>Manager Flower Shop</title>
10        <!-- Bootstrap Core CSS -->
11        <link href="~/Content/bootstrap.min.css" rel="stylesheet" type="text/css" />
12        <!-- MetisMenu CSS -->
13        <link href="~/Assets/Menu/vendor/metismenu/metisMenu.min.css" rel="stylesheet" />
14        <!-- Custom CSS -->
15        <link href="~/Assets/Menu/dist/css/main.css" rel="stylesheet" type="text/css" />
16        <!-- HTML5 Shim and Respond.js support of HTML5 elements and media queries -->
17        <!-- WARNING: Respond.js doesn't work if you view the page via file:// -->
18        <!--[if lt IE 9]>
19        <script src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"></script>
20        <script src="https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.min.js"></script>
21        <![endif]-->
22    </head>
23    <body>
24        <div id="wrapper">
25            <!-- Navigation -->
26            <nav class="navbar navbar-default navbar-static-top" role="navigation" style="margin-bottom: 0; ">
27                <div class="container">
28                    <div class="navbar-header">
29                        <button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-collapse">
30                            <span class="sr-only">Toggle navigation</span>
31                            <span class="icon-bar"></span>
32                            <span class="icon-bar"></span>
33                            <span class="icon-bar"></span>
34                        </button>
35                        <a class="navbar-brand" href="#">Welcome admin to FlowerShop</a>
36                    </div>
37                    <div class="collapse navbar-collapse">
38                        <ul class="nav navbar-top-links navbar-right">
39                            <li class="dropdown">
40                                <ul class="dropdown-menu dropdown-messages">
41                                    <li>
42
```

```
Menu_Layout.cshtml  ▶ X  Layout.cshtml  ListAccount.cshtml  Delete.cshtml  Details.cshtml  Edit.cshtml  HomeController.cs
271 |     
```

```
272 |     
```

```
273 |     
```

```
274 |     
```

```
275 |     
```

```
276 |     
```

```
277 |     
```

```
278 |     
```

```
279 |     
```

```
280 |     
```

```
281 |     
```

```
282 |     
```

```
283 |     
```

```
284 |     
```

```
285 |     
```

```
286 |     
```

```
287 |     
```

```
288 |     
```

```
289 |     
```

```
290 |     
```

```
291 |     
```

```
292 |     
```

```
293 |     
```

```
294 |     
```

```
295 |     
```

```
296 |     
```

```
297 |     
```

```
298 |     
```

```
299 |     
```

```
300 |     
```

```
301 |     
```

```
302 |     
```

```
303 |     
```

```
304 |     
```

```
305 |     
```

```
306 |     
```

```
307 |     
```

```
308 |     
```

```
309 |     
```

```
310 |     
```

```
311 |     
```

```
312 |     
```

```
313 |     
```

```
314 |     
```

```
315 |     
```

```
316 |     
```

```
317 |     
```

```
318 |     
```

```
319 |     
```

```
320 |     
```

```
321 |     
```

```
322 |     
```

```
323 |     
```

```
324 |     
```

```
325 |     
```

```
326 |     
```

```
327 |     
```

```
328 |     
```

```
329 |     
```

```
330 |     
```

```
331 |     
```

```
332 |     
```

```
333 |     
```

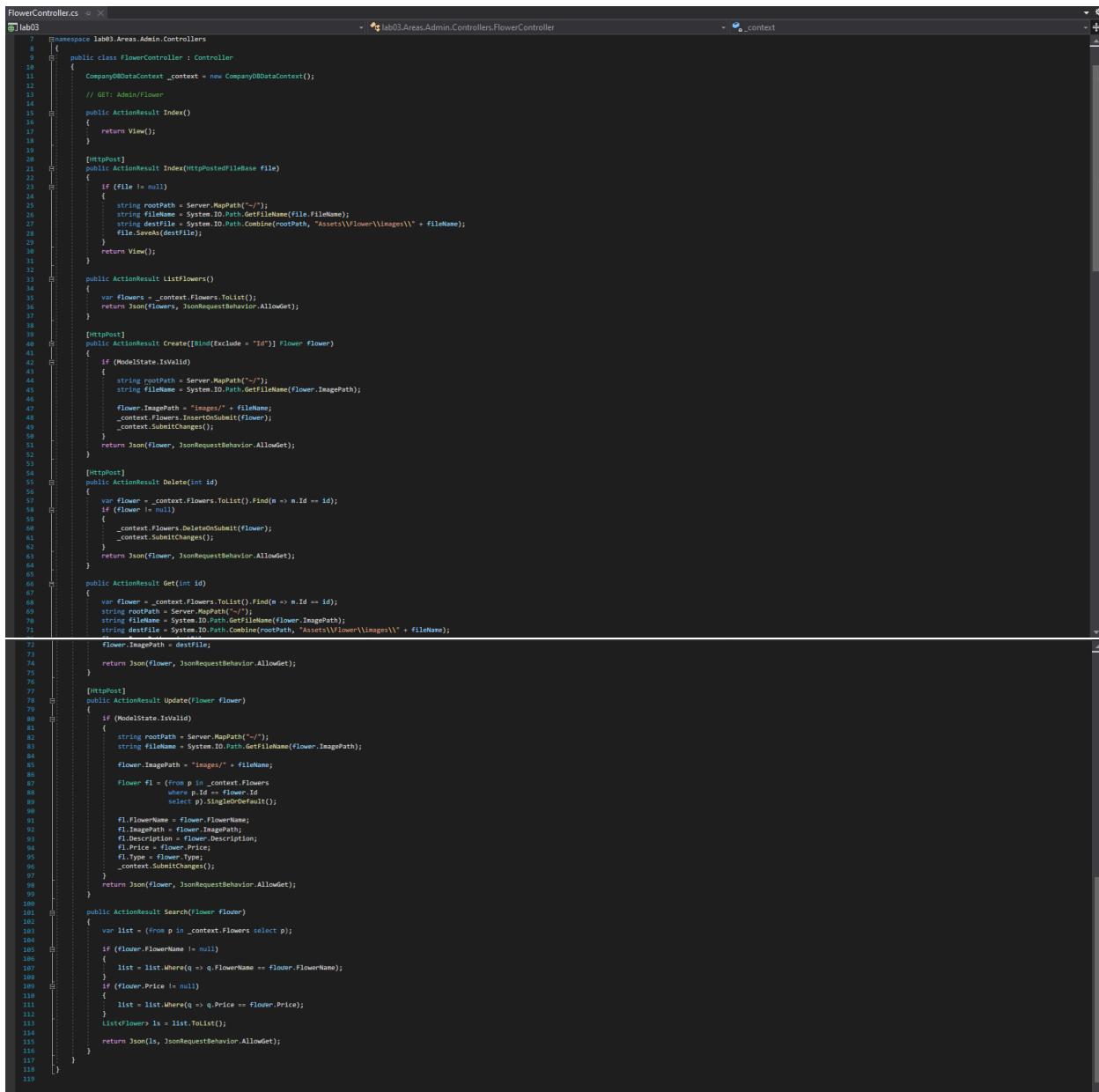
```
334 |     
```

```
335 |     
```

```
336 |     
```

```
337 |     
```

- FlowerController



The screenshot shows the code editor of Microsoft Visual Studio displaying the `FlowerController.cs` file. The code is written in C# and defines a controller for managing flowers. It includes methods for listing flowers, creating a new flower, deleting a flower, getting a flower by ID, updating a flower, and searching for flowers based on criteria like name or price.

```
1  namespace lab03.Areas.Admin.Controllers
2  {
3      public class FlowerController : Controller
4      {
5          CompanyDBODataContext _context = new CompanyDBODataContext();
6
7          // GET: Admin/Flower
8
9          public ActionResult Index()
10         {
11             return View();
12         }
13
14         [HttpPost]
15         public ActionResult Index(HttpPostedFileBase file)
16         {
17             if (file != null)
18             {
19                 string rootPath = Server.MapPath("~/");
20                 string fileName = System.IO.Path.GetFileName(file.FileName);
21                 string destfile = System.IO.Path.Combine(rootPath, "Assets\\Flower\\images\\") + fileName;
22                 file.SaveAs(destfile);
23             }
24             return View();
25         }
26
27         public ActionResult Listflowers()
28         {
29             var flower = _context.Flowers.ToList();
30             return Json(flower, JsonRequestBehavior.AllowGet);
31         }
32
33         [HttpPost]
34         public ActionResult Create([Bind(Exclude = "Id")] Flower flower)
35         {
36             if (ModelState.IsValid)
37             {
38                 string rootPath = Server.MapPath("~/");
39                 string fileName = System.IO.Path.GetFileName(flower.ImagePath);
40
41                 flower.ImagePath = "images/" + fileName;
42                 _context.Flowers.InsertOnSubmit(flower);
43                 _context.SubmitChanges();
44             }
45             return Json(flower, JsonRequestBehavior.AllowGet);
46         }
47
48         [HttpPost]
49         public ActionResult Delete(int id)
50         {
51             var flower = _context.Flowers.ToList().Find(m => m.Id == id);
52             if (flower != null)
53             {
54                 _context.Flowers.DeleteOnSubmit(flower);
55                 _context.SubmitChanges();
56             }
57             return Json(flower, JsonRequestBehavior.AllowGet);
58         }
59
60         public ActionResult Get(int id)
61         {
62             var flower = _context.Flowers.ToList().Find(m => m.Id == id);
63             string rootPath = Server.MapPath("~/");
64             string fileName = System.IO.Path.GetFileName(flower.ImagePath);
65             string destfile = System.IO.Path.Combine(rootPath, "Assets\\Flower\\images\\") + fileName;
66             flower.ImagePath = destfile;
67
68             return Json(flower, JsonRequestBehavior.AllowGet);
69         }
70
71         [HttpPost]
72         public ActionResult Update(Flower flower)
73         {
74             if (ModelState.IsValid)
75             {
76                 string rootPath = Server.MapPath("~/");
77                 string fileName = System.IO.Path.GetFileName(flower.ImagePath);
78
79                 flower.ImagePath = "images/" + fileName;
80
81                 Flower f1 = (from p in _context.Flowers
82                             where p.Id == flower.Id
83                             select p).SingleOrDefault();
84
85                 f1.FlowerName = flower.FlowerName;
86                 f1.ImagePath = flower.ImagePath;
87                 f1.Description = flower.Description;
88                 f1.Price = flower.Price;
89                 f1.Type = flower.Type;
90                 _context.SubmitChanges();
91             }
92             return Json(flower, JsonRequestBehavior.AllowGet);
93         }
94
95         public ActionResult Search(Flower flower)
96         {
97             var list = (from p in _context.Flowers select p);
98
99             if (flower.FlowerName != null)
100             {
101                 list = list.Where(q => q.FlowerName == flower.FlowerName);
102             }
103             if (flower.Price != null)
104             {
105                 list = list.Where(q => q.Price == flower.Price);
106             }
107             listFlowers ls = list.ToList();
108
109             return Json(ls, JsonRequestBehavior.AllowGet);
110         }
111     }
112 }
```

IndexFlower

```
Index.cshtml 9 ×

1  @{
2     ViewBag.Title = "Index";
3     Layout = "~/Areas/Admin/Views/Shared/_Layout.cshtml";
4 }
5 
6 
7 
8 
9 
10 <Section FLOWERS>
11     <div id="featured-content">
12         </div>
13 
14 <Section BIRTHDAY>
15     <div id="featured-content">
16         <div class="text-warning"> Birthday Flower </div>
17         <div id="birthday" style="margin-bottom:8px">
18             </div>
19         </div>
20 
21 
22 <Section Tulip>
23     <div id="featured-content">
24         <div class="text-warning">Tulip Flower</div>
25         <div id="Tulip" style="margin-bottom:8px">
26             </div>
27         </div>
28 
29 
30 <Section Another>
31     <div id="featured-content">
32         <div class="text-warning">Another Flower</div>
33         <div id="Another" style="margin-bottom: 8px">
34             </div>
35         </div>
36 
37 
38 <Section Model Dialog>
39     <div class="modal fade" id="CreateEditModal" role="dialog">
40         <div class="modal-dialog btn-lg">
41             <div class="modal-content">
42                 <div class="modal-header">
43                     <button type="button" class="close" data-dismiss="modal" data-target="#">&amptimes
```

```

139         data.ImagePath = currentUpdatePath;
140         data.Id = currentId;
141     }
142 
143     alert("success");
144     $("#fileUploadForm").submit();
145 }
146 
147 if (!isUpdatable) {
148     $().ajax({
149         url: '/Admin/Flower/Create/',
150         type: 'POST',
151         dataType: 'json',
152         data: data,
153         success: function (data) {
154             showFlowers();
155             clearModalContent();
156         },
157         error: function (err) {
158             alert("Error: " + err.responseText);
159         }
160     });
161 } else {
162     $().ajax({
163         url: '/Admin/Flower/Update/',
164         type: 'POST',
165         dataType: 'json',
166         data: data,
167         success: function (data) {
168             showFlowers();
169             clearModalContent();
170         },
171         error: function (err) {
172             alert("Error: " + err.responseText);
173         }
174     });
175 }
176 
177 $("#btnSearch").click(function (e) {
178     showSearchFlowers();
179 });
180 
181 function clearModalContent() {
182     $("#flowerName").val("");
183     $("#ImagePath").val("");
184     $("#description").val("");
185     $("#price").val("");
186     isUpdatable = false;
187 }
188 
189 function showFlowers() {
190     $("#featured-content").empty();
191 
192     $().ajax({
193         url: '/Admin/flower/ListFlowers',
194         type: 'GET',
195         dataType: 'json',
196         success: function (data) {
197             var countb = 0;
198             var countt = 0;
199             var countm = 0;
200             $.each(data, function (i, item) {
201                 var id = item.Id;
202                 var flowername = item.FlowerName;
203                 var tagimage = item.ImagePath;
204                 var tagdescription = item.Description;
205                 var tagprice = item.Price;
206                 var classname = "#item" + item.Type;
207                 if (item.type == "Mirrored") {
208                     if (countb == 0) {
209                         countb++;
210                     }
211                 }
212                 AppendNewFlower(id, countb, flowername, tagimage, tagdescription, tagprice, classname);
213             }
214             if (item.type == "Upright") {
215                 if (countt == 3) {
216                     countt++;
217                 }
218                 else {
219                     countt++;
220                 }
221                 AppendNewFlower(id, countt, flowername, tagimage, tagdescription, tagprice, classname);
222             }
223             if (item.type == "Another") {
224                 if (countm == 3) {
225                     countm++;
226                 }
227                 else {
228                     countm++;
229                 }
230                 AppendNewFlower(id, countm, flowername, tagimage, tagdescription, tagprice, classname);
231             }
232         }
233         error: function (err) {
234             alert("Error: " + err.responseText);
235         }
236     });
237 }
238 
239 function showSearchFlowers() {
240     var data = {
241         Id: "",
242         FlowerName: $("#SearchflowerName").val(),
243         Description: "",
244         Price: $("#SearchPrice").val(),
245         "": "#Mirrored").empty();
246         $("#Upright").empty();
247         $("#Another").empty();
248     };
249 
250     $().ajax({
251         url: '/Admin/flower/Search',
252         type: 'GET',
253         dataType: 'json',
254         data: data,
255         success: function (data) {
256             var countb = 0;
257             var countt = 0;
258             var countm = 0;
259             $.each(data, function (i, item) {
260                 var id = item.Id;
261                 var flowername = item.FlowerName;
262                 var tagimage = item.ImagePath;
263                 var tagdescription = item.Description;
264                 var tagprice = item.Price;
265                 var classname = "#item" + item.Type;
266             }
267         }
268     });
269 }

```

- Flower_Layout

```
FlowerLayout.cshtml -> Index.cshtml
```

```
1  @{
2      ViewBag.Title = "Flower Layout";
3      Layout = "~/Areas/Admin/Views/Shared/_Menu_Layout.cshtml";
4  }
5
6
7  <div id="wrapper">
8      <h1 class="col-md-12">Search Flower</h1>
9      <div class="col-md-12">
10          <div class="col-md-6">
11              <label>Search Flower Name</label>
12              <input type="text" id="SearchFlowerName" class="form-control" placeholder="Search ..."/>
13          </div>
14      </div>
15      <div class="form-group">
16          <div class="col-md-6">
17              <label>Search Price</label>
18              <input type="text" id="SearchPrice" class="form-control" placeholder="Search ..."/>
19          </div>
20      </div>
21      <div class="form-group">
22          <div class="col-md-12">
23              &nbsp;
24          </div>
25      </div>
26
27      <div class="form-group">
28          <div class="col-md-12">
29              <button type="button" id="btnSearch" class="btn btn-primary">Search</button>
30          </div>
31      </div>
32
33      <h1 class="col-md-12">Flower Information</h1>
34      <div class="col-md-12">
35          <button type="button" id="btCreateNew" data-toggle="modal" data-target="#CreateEditModal"
36                  class="btn btn-success">
37              Create New
38          </button>
39      </div>
40
41      <div>
42          <RenderSection("FL01", required: false)>
43          <RenderSection("BIRTHDAY", required: false)>
44          <RenderSection("TULP", required: false)>
45          <RenderSection("ANOTHER", required: false)>
46      </div>
47
48      <RenderSection("MOEL_DIALOG", required: false)>
49
50      <RenderSection("Script", required: false)>
51
52
53
```

- DealController

```
DealController.cs  DealController.cs  _context
Lab03\Areas\Admin\Controllers\  Lab03.Areas.Admin.Controllers.DealController  _context
7  namespace Lab03.Areas.Admin.Controllers
8  {
9      public class DealController : Controller
10     {
11         CompanyODBDataContext _context = new CompanyODBDataContext();
12 
13         // GET: Admin/Deal
14         public ActionResult Index()
15         {
16             return View();
17         }
18 
19         public ActionResult ListDeals()
20         {
21             var deal = _context.Deals.ToList();
22             return Json(deal, JsonRequestBehavior.AllowGet);
23         }
24 
25         [HttpPost]
26         public ActionResult Delete(int id)
27         {
28             var deal = _context.Deals.ToList().Find(s => s.Id == id);
29             if (deal != null)
30             {
31                 _context.Deals.DeleteObject(deal);
32                 _context.SubmitChanges();
33             }
34             return Json(deal, JsonRequestBehavior.AllowGet);
35         }
36 
37         public ActionResult Get(int id)
38         {
39             var deal = _context.Deals.ToList().Find(s => s.Id == id);
40             return Json(deal, JsonRequestBehavior.AllowGet);
41         }
42 
43         [HttpPost]
44         public ActionResult Update(Deal deal)
45         {
46             if (ModelState.IsValid)
47             {
48                 Deal de = (from p in _context.Deals
49                             where p.Id == deal.Id
50                             select p).SingleOrDefault();
51 
52                 de.CustomerName = deal.CustomerName;
53                 de.CustomerEmail = deal.CustomerEmail;
54                 de.DealDate = deal.DealDate;
55                 de.FlowerName = deal.FlowerName;
56                 de.Quantity = deal.Quantity;
57                 de.Total = deal.Total;
58                 _context.SubmitChanges();
59             }
60             return Json(deal, JsonRequestBehavior.AllowGet);
61         }
62 
63         public ActionResult Search(Deal deal)
64         {
65             var list = (from p in _context.Deals select p);
66             if (deal.CustomerName != null)
67             {
68                 list = list.Where(q => q.CustomerName == deal.CustomerName);
69             }
70             if (deal.CustomerEmail != null)
71             {
72                 list = list.Where(q => q.CustomerEmail == deal.CustomerEmail);
73             }
74             if (deal.DealDate != null)
75             {
76                 list = list.Where(q => q.DealDate == deal.DealDate);
77             }
78             if (deal.FlowerName != null)
79             {
80                 list = list.Where(q => q.FlowerName == deal.FlowerName);
81             }
82             listDeal = list.ToList();
83         }
84         return Json(list, JsonRequestBehavior.AllowGet);
85     }
86 }
```

DealIndex

```

135         var totalamount = 0;
136         $each(data, function (i, item) {
137             var id = item.id;
138             var customername = item.CustomerName;
139             var customeremail = item.CustomerEmail;
140             var date = item.DealDate;
141             var flowername = item.FlowerName;
142             var quantity = item.Quantity;
143             var total = item.Total;
144             totalamount += total;
145             AppendNewDeal(id, customername, customeremail, date, flowername, quantity, total);
146         });
147         rows = '';
148         rows += '<tr><td>/<td> <td>/<td> <td>/<td> <td>/<td> <td>Total amount </td>';
149         rows += '</tr>' + totalamount + '</td> </tr>';
150         $('#tbody').append(rows);
151     },
152     error: function (err) {
153         alert('Error: ' + err.responseText);
154     }
155 });
156 });
157
158 function showSearchDeal() {
159     var data = [
160         {
161             id: '',
162             customername: $('#SearchCustomerName').val(),
163             CustomerEmail: $('#SearchCustomerEmail').val(),
164             DealDate: $('#SearchDateForTransaction').val(),
165             FlowerName: $('#SearchflowerName').val(),
166         }
167     ];
168     $('#tbody').empty();
169
170     $ajax({
171         url: '/Admin/Deal/Search',
172         type: 'GET',
173         dataType: 'json',
174         data: data,
175         success: function (data) {
176             $each(data, function (i, item) {
177                 var id = item.id;
178                 var customername = item.CustomerName;
179                 var customeremail = item.CustomerEmail;
180                 var date = item.DealDate;
181                 var flowername = item.FlowerName;
182                 var quantity = item.Quantity;
183                 var total = item.Total;
184                 AppendNewDeal(id, customername, customeremail, date, flowername, quantity, total);
185             });
186         },
187         error: function (err) {
188             alert('Error: ' + err.responseText);
189         }
190     });
191 }
192
193 function AppendNewDeal(id, customername, customeremail, date, Flowername, quantity, total) {
194     var rows = '';
195     rows += '<tr>';
196     rows += '<td>' + customername + '</td>';
197     rows += '<td>' + customeremail + '</td>';
198     rows += '<td>' + date + '</td>';
199     rows += '<td>' + flowername + '</td>';
200     rows += '<td>' + quantity + '</td>';
201     rows += '<td>' + total + '</td>';
202     rows += '<td>' + '<button type="button" id="btnEdit" class="btn btn-primary" ' +
203             ' data-toggle="modal" data-target="#CreateDetailModal" ' +
204             '><span>Edit</span></button>' +
205             '<button type="button" id="btnDelete" class="btn btn-danger" ' +
206             '><span>Delete</span></button>' + '</td>';
207     rows += '</tr>';
208     rows += '</tr>';
209     $('#tbody').append(rows);
210 }
211
212 function deleteDeal(id) {
213     var confirmDelete = confirm("Do you want to delete deal?");
214     $ajax({
215         url: '/Admin/Deal/Delete/' + id,
216         type: 'POST',
217         dataType: 'json',
218         success: function (data) {
219             showDeal();
220         },
221         error: function (err) {
222             alert('Error: ' + err.responseText);
223         }
224     });
225 }
226
227 function getDetailDeal(id) {
228     currentId = 0;
229
230     $('#title').text("Edit");
231
232     $ajax({
233         url: '/Admin/Deal/Get/' + id,
234         type: 'GET',
235         dataType: 'json',
236         success: function (data) {
237             $('#CustomerName').val(data.CustomerName);
238             $('#CustomerEmail').val(data.CustomerEmail);
239             $('#DealDate').val(data.DealDate);
240             $('#FlowerName').val(data.FlowerName);
241             $('#Quantity').val(data.Quantity);
242             $('#Price').val(data.Price);
243             $('#Total').val(data.Total + '$');
244             currentId = data.Id;
245         },
246         error: function (err) {
247             alert('Error: ' + err.responseText);
248         }
249     });
250 }

```

- Deals_Layout

```
DealLayout.cshtml 4 X Index.cshtml DealController.cs
1<!DOCTYPE html>
2<html>
3    <head>
4        <meta charset="utf-8" />
5        <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6        <title>Deals</title>
7        <link href="~/Content/bootstrap.css" rel="stylesheet" />
8        <link href="~/Content/site.css" rel="stylesheet" />
9        <script src="~/Scripts/jquery-1.10.2.js" type="text/javascript"></script>
10       <script src="~/Scripts/bootstrap.js" type="text/javascript"></script>
11       <script src="~/Scripts/respond.js" type="text/javascript"></script>
12   </head>
13   <body>
14       <div id="page-wrapper">
15           <div class="row">
16               <div class="col-md-12">
17                   <div class="row">
18                       <div class="col-md-10">
19                           <div class="form-group">
20                               <label>Search Customer Name</label>
21                               <input type="text" id="searchCustomerName" class="form-control" placeholder="Search ..." />
22                           </div>
23                       </div>
24                       <div class="col-md-2">
25                           <button type="button" class="btn btn-primary" onclick="searchCustomerName()>Search</button>
26                       </div>
27                   </div>
28                   <div class="row">
29                       <div class="col-md-10">
30                           <div class="form-group">
31                               <label>Search Customer Email</label>
32                               <input type="text" id="searchCustomerEmail" class="form-control" placeholder="Search ..." />
33                           </div>
34                       </div>
35                       <div class="col-md-2">
36                           <button type="button" class="btn btn-primary" onclick="searchCustomerEmail()>Search</button>
37                       </div>
38                   </div>
39                   <div class="row">
40                       <div class="col-md-10">
41                           <div class="form-group">
42                               <label>Search Date Deal</label>
43                               <input type="text" id="searchDateTransaction" class="form-control" placeholder="Search ..." />
44                           </div>
45                       </div>
46                       <div class="col-md-2">
47                           <button type="button" class="btn btn-primary" onclick="searchDateTransaction()>Search</button>
48                       </div>
49                   </div>
50                   <div class="row">
51                       <div class="col-md-10">
52                           <div class="form-group">
53                               <label>Search Flower Name</label>
54                               <input type="text" id="searchFlowerName" class="form-control" placeholder="Search ..." />
55                           </div>
56                       </div>
57                   </div>
58                   <div class="row">
59                       <div class="col-md-12">
60                           <table border="1" style="width: 100%; border-collapse: collapse; text-align: center; border: none; font-size: small; font-weight: bold;">
61                               <thead>
62                                   <tr>
63                                       <td>Customer Name</td>
64                                       <td>Customer Email</td>
65                                       <td>Date</td>
66                                       <td>Quantity</td>
67                                       <td>Total</td>
68                                       <td>Manipulation</td>
69                                   </tr>
70                               </thead>
71                               <tbody>
72                                   <tr>
73                                       <td>@Model.CustomerName</td>
74                                       <td>@Model.CustomerEmail</td>
75                                       <td>@Model.Date</td>
76                                       <td>@Model.Quantity</td>
77                                       <td>@Model.Total</td>
78                                       <td>@Model.Manipulation</td>
79                                   </tr>
80                               </tbody>
81                           </table>
82                       </div>
83                   </div>
84               </div>
85           </div>
86           <div class="row">
87               <div class="col-md-12" style="text-align: right; margin-top: 10px;">
88                   <button type="button" class="btn btn-primary" onclick="location.href='@Url.Action("Index", "Deals")'>Back</button>
89               </div>
90           </div>
91       </div>
92   </body>
93 </html>
```

- LoginModel

```
LoginModel.cs 0 X
Lab03.Areas.Admin.LoginModel
1using System;
2using System.Collections.Generic;
3using System.ComponentModel.DataAnnotations;
4using System.Linq;
5using System.Web;
6
7namespace Lab03.Areas.Admin
8{
9    public class LoginModel
10    {
11        [Required]
12        public string User { get; set; }
13        public string UserName { get; set; }
14        public string Email { get; set; }
15        public string Password { get; set; }
16    }
17}
```

- AdminAreaRegistration

```
AdminAreaRegistration.cs 0 X LoginModel.cs
Lab03
1using System.Web.Mvc;
2
3namespace Lab03.Areas.Admin
4{
5    public class AdminAreaRegistration : AreaRegistration
6    {
7        public override string AreaName
8        {
9            get
10            {
11                return "Admin";
12            }
13        }
14
15        public override void RegisterArea(AreaRegistrationContext context)
16        {
17            context.MapRoute(
18                "Admin_default",
19                "Admin/{controller}/{action}/{id}",
20                new { Controller="Home", action = "Index", id = UrlParameter.Optional }
21            );
22        }
23    }
24}
```

- AccountModel

```
AccountModel.cs  AdminAreaRegistration.cs  LoginModel.cs
lab03  lab03.Areas.Admin.AccountModel  context

1  using System;
2  using System.Collections.Generic;
3  using System.ComponentModel.DataAnnotations;
4  using System.Linq;
5  using System.Web;
6
7  namespace lab03.Areas.Admin
8  {
9      public class AccountModel
10     {
11         private Company0BDataContext context = null;
12
13         public AccountModel()
14         {
15             context = new Company0BDataContext();
16         }
17
18         public bool Login(string UserName, string Password)
19         {
20             bool res = false;
21             context.sp_Account_Login_Check(UserName, Password, ref res);
22
23             return (res ?? false);
24         }
25
26         public bool Register(string UserName, string Email)
27         {
28             bool res = false;
29             context.sp_Check_Account(UserName, Email, ref res);
30
31             return (res ?? false);
32         }
33     }
34 }
```

- RouteConfig

```
RouteConfig.cs  AccountModel.cs  AdminAreaRegistration.cs  LoginModel.cs
lab03  lab03  lab03.RouteConfig  RegisterRoutes(RouteCollection routes)

1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.Mvc;
6  using System.Web.Routing;
7
8  namespace lab03
9  {
10     public class RouteConfig
11     {
12         public static void RegisterRoutes(RouteCollection routes)
13         {
14             routes.IgnoreRoute("{resource}.axd/{pathInfo}");
15
16             routes.MapRoute(
17                 name: "Default",
18                 url: "{controller}/{action}/{id}",
19                 defaults: new { controller = "Login", action = "Index", id = UrlParameter.Optional },
20                 namespaces : new[] { lab03.Areas.Main.Controllers }
21             ).DataTokens.Add("Area", "Main");
22         }
23     }
24 }
```

- ReportController

```
ReportController.cs  AdminAreaRegistration.cs  LoginModel.cs
lab03  lab03  lab03.Areas.Admin.Controllers.ReportController  context

1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.Mvc;
6  using lab03.Models;
7  using Microsoft.Reporting.WebForms;
8
9  namespace lab03.Areas.Admin.Controllers
10 {
11     public class ReportController : Controller
12     {
13         private Company0BDataContext context = new Company0BDataContext();
14         ProjectWICentisiesi proj = new ProjectWICentisiesi();
15         // GET: Admin/Report
16         public ActionResult Index()
17         {
18
19             return View(context.Deals.ToList());
20         }
21
22         public ActionResult Reports(string ReportType)
23         {
24             LocalReport localReport = new LocalReport();
25             localReport.ReportPath = Server.MapPath("~/Reported/Report1.rdlc");
26             reportDataSource1.Name = "Dealdata";
27             reportDataSource1.Value = context.Deals.ToList();
28             localReport.DataSources.Add(reportDataSource1);
29             string mimeType = string.Empty;
30             string encoding = string.Empty;
31             string fileNameExtension = string.Empty;
32             if(ReportType == "Excel")
33             {
34                 fileNameExtension = ".xlsx";
35             }
36             else if (ReportType == "PDF")
37             {
38                 fileNameExtension = ".pdf";
39             }
40             else if (ReportType == "Word")
41             {
42                 fileNameExtension = ".docx";
43             }
44             else
45             {
46                 fileNameExtension = ".jpg";
47             }
48
49             string[] streams;
50             Warning[] warnings;
51             byte[] renderedBytes;
52             renderedBytes = localReport.Render(reportType, "", out mimeType, out encoding, out fileNameExtension, out streams, out warnings);
53             HttpResponseBase response = Response;
54             response.AddHeader("Content-Disposition", "attachment;filename=deal_report." + fileNameExtension);
55
56             return File(renderedBytes, fileNameExtension);
57         }
58     }
59 }
60 }
```

ReportIndex

```
Index.cshtml  ReportController.cs

1  @model IEnumerable<Lab03.Deal>
2
3  @{
4      ViewBag.Title = "Index";
5      Layout = "~/Areas/Admin/Views/Shared/_Menu_Layout.cshtml";
6  }
7
8  <br />
9  <div class="text-success">All deals (</div>)
10
11  <ul style="list-style-type: none; padding-left: 0;">
12      <li>@Html.ActionLink("Export to Excel", "Reports", new { ReportType = "Excel" }, new { @class = "btn btn-info" }) |</li>
13      <li>@Html.ActionLink("Export to PDF", "Reports", new { ReportType = "PDF" }, new { @class = "btn btn-info" }) |</li>
14      <li>@Html.ActionLink("Export to word", "Reports", new { ReportType = "Word" }, new { @class = "btn btn-info" }) |</li>
15      <li>@Html.ActionLink("Export to Image", "Reports", new { ReportType = "Image" }, new { @class = "btn btn-info" }) |</li>
16  </ul>
17
18  <table class="table">
19      <tr style="background-color: #009aff; color: white;">
20          <th>Customer Name</th>
21          <th>Customer Email</th>
22          <th>Deal Date</th>
23          <th>Flower Name</th>
24          <th>Quantity</th>
25          <th>Total</th>
26          <th>Price</th>
27      </tr>
28
29      <tr>
30          <td>@Html.DisplayFor(model => model.CustomerName)</td>
31          <td>@Html.DisplayFor(model => model.CustomerEmail)</td>
32          <td>@Html.DisplayFor(model => model.DealDate)</td>
33          <td>@Html.DisplayFor(model => model.FlowerName)</td>
34          <td>@Html.DisplayFor(model => model.Quantity)</td>
35          <td>@Html.DisplayFor(model => model.Total)</td>
36          <td>@Html.DisplayFor(model => model.Price)</td>
37      </tr>
38
39      <tr>
40          <th></th>
41      </tr>
42
43      <foreach var item in Model>
44      {
45          <tr>
46              <td>@Html.DisplayFor(modelItem => item.CustomerName)</td>
47              <td>@Html.DisplayFor(modelItem => item.CustomerEmail)</td>
48              <td>@Html.DisplayFor(modelItem => item.DealDate)</td>
49              <td>@Html.DisplayFor(modelItem => item.FlowerName)</td>
50              <td>@Html.DisplayFor(modelItem => item.Quantity)</td>
51              <td>@Html.DisplayFor(modelItem => item.Total)</td>
52              <td>@Html.DisplayFor(modelItem => item.Price)</td>
53          </tr>
54      }
55  </foreach>
56
57  </table>
58
```

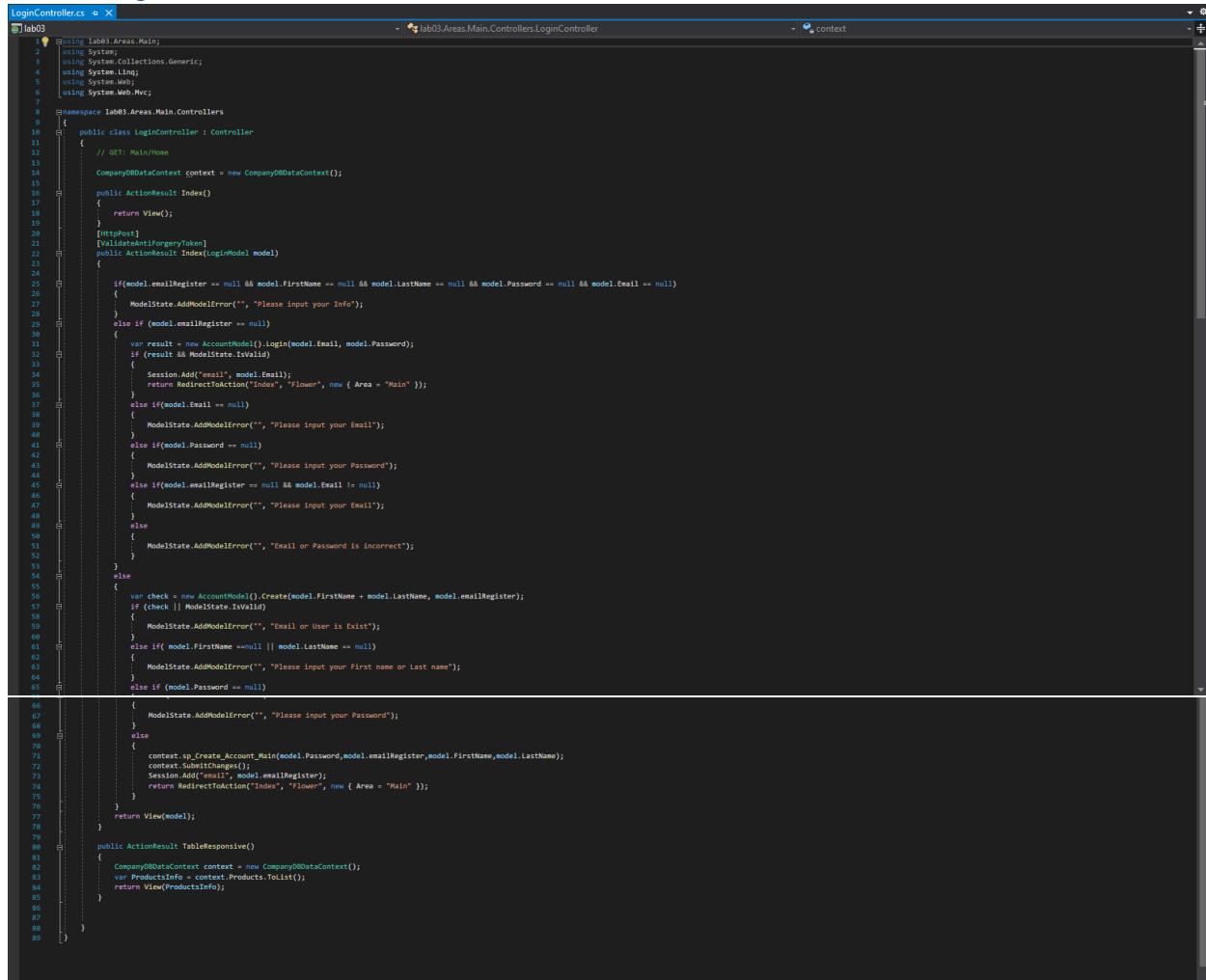
Report1.rdlc

The screenshot shows the Visual Studio interface with the following details:

- Report Data Explorer:** On the left, it lists "Built-in Fields" (Parameters, Images), "Data Sources", and "Datasets". Under "Datasets", there is a node "DealData" expanded, showing columns: Id, CustomerName, CustomerEmail, DealDate, FlowerName, Quantity, Total, and Price.
- Report1.rdlc [Design]:** The main area displays a report titled "Deals information". It features a table with columns: Id, Customer Name, Customer Email, Deal Date, Flower Name, Quantity, Total, and Price. The "Customer Name" and "Customer Email" columns are bound to "[CustomerName]" and "[CustomerEmail]" respectively. The "Deal Date" column is bound to "[DealDate]". The "Flower Name" column is bound to "[FlowerName]". The "Quantity" column is bound to "[Quantity]". The "Total" column is bound to "[Total]". The "Price" column is bound to "[Price]".
- CartController.cs:** The title bar also shows the file "CartController.cs".

b. Main_Areas

- LoginController



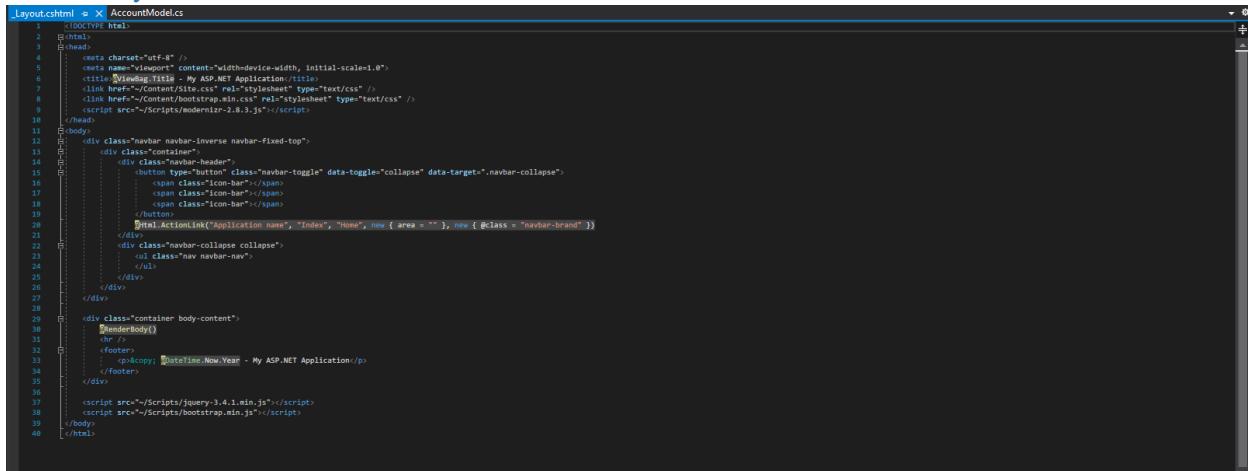
The screenshot shows the code editor in Visual Studio with the file `LoginController.cs` open. The code is written in C# and defines a controller for user login and registration. It includes imports for namespaces like `System`, `System.Collections.Generic`, `System.Linq`, `System.Web`, and `System.Web.Mvc`. The class `LoginController` has methods for `Index` (GET) and `Index` (POST). The POST method handles user input and validation, including checking for existing email addresses and displaying error messages. It also handles successful registration by creating a new account and redirecting to the main page.

```
1  @using lab03.Areas.Main;
2  using System;
3  using System.Collections.Generic;
4  using System.Linq;
5  using System.Web;
6  using System.Web.Mvc;
7
8  namespace lab03.Areas.Main.Controllers
9  {
10     public class LoginController : Controller
11     {
12         // GET: Main/Home
13         public ActionResult Index()
14         {
15             return View();
16         }
17
18         [HttpPost]
19         [ValidateAntiForgeryToken]
20         public ActionResult Index(LoginModel model)
21         {
22
23             if(model.emailRegister == null && model.FirstName == null && model.LastName == null && model.Password == null && model.Email == null)
24             {
25                 ModelState.AddModelError("", "Please input your Info");
26             }
27             else if (model.emailRegister == null)
28             {
29                 var result = new AccountModel().Login(model.Email, model.Password);
30                 if (result && ModelState.IsValid)
31                 {
32                     Session.Add("result", model.Email);
33                     return RedirectToAction("Index", "Flower", new { Area = "Main" });
34                 }
35                 else if(model.Email == null)
36                 {
37                     ModelState.AddModelError("", "Please input your Email");
38                 }
39                 else if(model.Password == null)
40                 {
41                     ModelState.AddModelError("", "Please input your Password");
42                 }
43                 else if(model.emailRegister == null && model.Email != null)
44                 {
45                     ModelState.AddModelError("", "Please input your Email");
46                 }
47                 else if(model.Email != null)
48                 {
49                     ModelState.AddModelError("", "Email or Password is incorrect");
50                 }
51             }
52             else
53             {
54                 var check = new AccountModel().Create(model.FirstName + model.LastName, model.emailRegister);
55                 if (check || ModelState.IsValid)
56                 {
57                     ModelState.AddModelError("", "Email or User is Exist");
58                 }
59                 else if( model.FirstName == null || model.LastName == null)
60                 {
61                     ModelState.AddModelError("", "Please input your First name or Last name");
62                 }
63                 else if (model.Password == null)
64                 {
65                     ModelState.AddModelError("", "Please input your Password");
66                 }
67             }
68
69             context就来看看_主(model.Password, model.emailRegister, model.FirstName, model.LastName);
70             context.SubmitChanges();
71             Session.Add("email", model.emailRegister);
72             return RedirectToAction("Index", "Flower", new { Area = "Main" });
73         }
74
75         public ActionResult TableResponsive()
76         {
77             CompanyDBDataContext context = new CompanyDBDataContext();
78             var ProductsInfo = context.Products.ToList();
79             return View(ProductsInfo);
80         }
81     }
82 }
```

- indexLogin

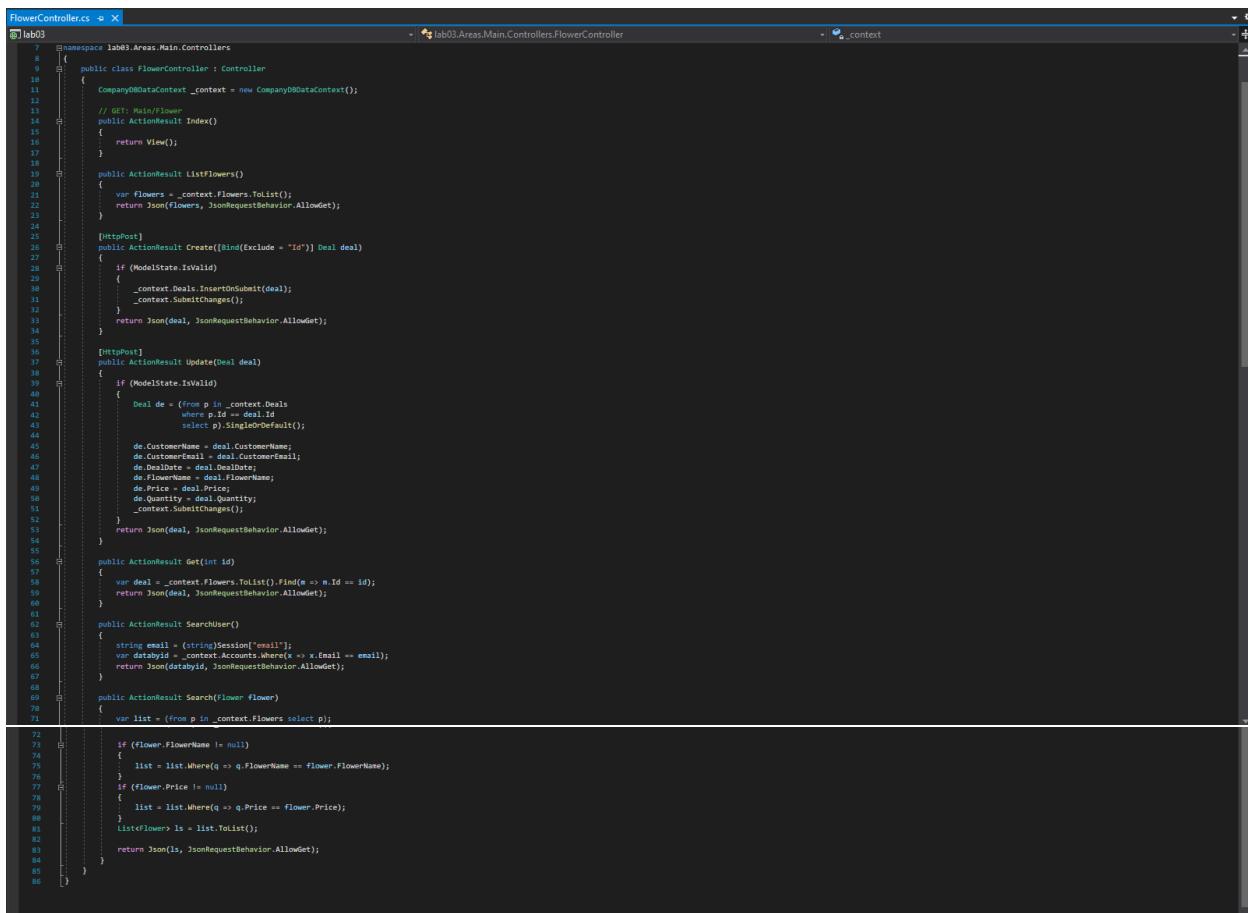
```
index.html ✘ ×
1  ||| ViewBag.Title = "Index";
2  ||
3  |
4  |model lab3.Areas.Main.LoginModel
5  |
6  <!DOCTYPE html>
7  <html lang="en">
8  |
9  |<head>
10 |    |<meta charset="UTF-8">;
11 |    |<title>Sign Up/Log In</title>;
12 |    |<link href="https://fonts.googleapis.com/css?family=Titillium+Web:400,300,600" rel="stylesheet" type="text/css">;
13 |    |<link rel="stylesheet" href="https://cdnis.cloudflare.com/zis/libs/normalize/5.0.0/normalize.min.css">;
14 |    |<link rel="stylesheet" href="~/Assets/Main/css/style.css" type="text/css">;
15 |    |<script language="JavaScript" type="text/javascript" src="~/js/jquery-1.2.6.min.js"></script>;
16 |    |<script language="JavaScript" type="text/javascript" src="~/js/jquery-ui-personalized-1.5.2.packed.js"></script>;
17 |    |<script language="JavaScript" type="text/javascript" src="~/js/sprinkle.js"></script>;
18 |</head>
19 |
20 |<body>
21 |
22 |    |<div class="form">
23 |    |
24 |        |<ul class="tab-group">
25 |        |<li class="tab active"><a href="#">Sign Up</a></li>
26 |        |<li class="tab"><a href="#">Log In</a></li>
27 |        |</ul>
28 |
29 |        |<div class="tab-content">
30 |        |<div id="signUp">
31 |        |<form action="/" method="post">
32 |        |<div>
33 |        |<input type="text" required autocomplete="off" />
34 |        |<span>First Name</span>
35 |        |<div>
36 |        |<input type="text" required autocomplete="off" />
37 |        |<span>Last Name</span>
38 |        |<div>
39 |        |<input type="text" required autocomplete="off" />
40 |        |<span>Email Address</span>
41 |        |<div>
42 |        |<input type="password" placeholder="password" />
43 |        |<span>Set A Password</span>
44 |        |<div>
45 |        |<input type="password" placeholder="password" />
46 |        |<span>Re-enter Password</span>
47 |        |<div>
48 |        |<button type="submit" class="button button-block">Get Started</button>
49 |        |</div>
50 |        |</div>
51 |        |</div>
52 |        |</div>
53 |        |</div>
54 |        |</div>
55 |        |</div>
56 |        |</div>
57 |        |</div>
58 |        |</div>
59 |        |</div>
60 |        |</div>
61 |        |</div>
62 |        |</div>
63 |        |</div>
64 |        |</div>
65 |        |</div>
66 |        |</div>
67 |        |</div>
68 |        |</div>
69 |        |</div>
70 |        |</div>
71 |        |</div>
72 |        |</div>
73 |        |</div>
74 |        |</div>
75 |        |</div>
76 |        |<div id="login">
77 |        |<h3>Welcome Back!Log In</button>
87 |        |</div>
88 |        |</div>
89 |        |</div>
90 |        |</div>
91 |        |</div>
92 |        |</div>
93 |        |</div>
94 |        |</div>
95 |        |</div>
96 |        |</div>
97 |        |</div>
98 |        |</div>
99 |        |</div>
100 |        |<p><a href="#">Forgot Password?
```

- Layout



```
<!DOCTYPE html>
<html>
    <head>
        <meta charset="utf-8" />
        <meta name="viewport" content="width=device-width, initial-scale=1.0" />
        <title>Viewing Title - My ASP.NET Application</title>
        <link href="/Content/bootstrap.min.css" type="text/css" rel="stylesheet" />
        <link href="/Content/bootstrap-theme.min.css" type="text/css" rel="stylesheet" />
        <script src="/Scripts/modernizr-2.8.3.js"></script>
    </head>
    <body>
        <div class="navbar navbar-inverse navbar-fixed-top">
            <div class="container">
                <div class="navbar-header">
                    <button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-collapse">
                        <span class="icon-bar"></span>
                        <span class="icon-bar"></span>
                        <span class="icon-bar"></span>
                    </button>
                <div class="navbar-collapse collapse">
                    <ul class="nav navbar-nav">
                        <li><a href="#">Home</a></li>
                    </ul>
                </div>
            </div>
        </div>
        <div class="container body-content">
            <div>
                <h1>Viewing Title</h1>
                <p>Copyright © <DateTime.Now.Year> - My ASP.NET Application</p>
            </div>
        </div>
        <script src="/Scripts/jquery-3.4.1.min.js"></script>
        <script src="/Scripts/bootstrap.min.js"></script>
    </body>
</html>
```

- FlowerController



```
Lab03.Areas.Main.Controllers
{
    public class FlowerController : Controller
    {
        CompanyDBContext _context = new CompanyDBContext();

        // GET: Main/Flower
        public ActionResult Index()
        {
            return View();
        }

        public ActionResult ListFlowers()
        {
            var flowers = _context.Flowers.ToList();
            return Json(flowers, JsonRequestBehavior.AllowGet);
        }

        [HttpPost]
        public ActionResult Create([Bind(Exclude = "Id")] Deal deal)
        {
            if (ModelState.IsValid)
            {
                _context.Deals.InsertOnSubmit(deal);
                _context.SubmitChanges();
            }
            return Json(deal, JsonRequestBehavior.AllowGet);
        }

        [HttpPost]
        public ActionResult Update(Deal deal)
        {
            if (ModelState.IsValid)
            {
                Deal de = (from p in _context.Deals
                           where p.Id == deal.Id
                           select p).SingleOrDefault();
                de.CustomerName = deal.CustomerName;
                de.CustomerEmail = deal.CustomerEmail;
                de.DealDate = deal.DealDate;
                de.FlowerName = deal.FlowerName;
                de.Price = deal.Price;
                de.Quantity = deal.Quantity;
                _context.SubmitChanges();
            }
            return Json(deal, JsonRequestBehavior.AllowGet);
        }

        public ActionResult Get(int id)
        {
            var deal = _context.Deals.ToList().Find(m => m.Id == id);
            return Json(deal, JsonRequestBehavior.AllowGet);
        }

        public ActionResult SearchUser()
        {
            string email = (string)Session["email"];
            var details = _context.Accounts.Where(x => x.Email == email);
            return Json(details, JsonRequestBehavior.AllowGet);
        }

        public ActionResult Search(Flower flower)
        {
            var list = (from p in _context.Flowers select p);

            if (flower.FlowerName != null)
            {
                list = list.Where(q => q.FlowerName == flower.FlowerName);
            }
            if (flower.Price != null)
            {
                list = list.Where(q => q.Price == flower.Price);
            }
            list = list.ToList();
            return Json(list, JsonRequestBehavior.AllowGet);
        }
    }
}
```

FlowerIndex

```

119         url: '/Flower/Create/',
120         type: 'POST',
121         dataType: 'json',
122         data: data,
123         success: function (data) {
124             clearModalContent();
125         },
126         error: function (err) {
127             alert('Error: ' + err.responseText);
128         }
129     });
130     else {
131         $.ajax({
132             url: '/Flower/Update/',
133             type: 'PUT',
134             dataType: 'json',
135             data: data,
136             success: function (data) {
137                 clearModalContent();
138             },
139             error: function (err) {
140                 alert("Error: " + err.responseText);
141             }
142         });
143     }
144 });
145 $("#btnSearch").click(function (e) {
146     showSearchFlowers();
147 });
148
149 function createNewCart() {
150     $.ajax({
151         url: '/Flower/SearchUser',
152         type: 'get',
153         dataType: 'json',
154         success: function (data) {
155             $.each(data, function (i, item) {
156                 $("#CustomerName").val(item.UserName);
157                 $("#CustomerEmail").val(item.Email);
158             });
159         }
160     });
161
162 function showFlowers() {
163     $("#Featured-content").empty();
164
165     $.sign({
166         url: '/Flower/ListFlowers',
167         type: 'GET',
168         dataType: 'json',
169         success: function (data) {
170             var count = 0;
171             var countb = 0;
172             var counta = 0;
173             $.each(data, function (i, item) {
174                 var id = item.Id;
175                 var flowername = item.flowerName;
176                 var tagimage = item.ImagePath;
177                 var tagdescription = item.Description;
178                 var tagprice = item.Price;
179                 var classname = "# " + item.type;
180                 if (item.type == "Birthday") {
181                     if (count == 3) {
182                         countb = 1;
183                     } else {
184                         countb++;
185                     }
186                     AppendNewFlower(id, countb, flowername, tagimage, tagdescription, tagprice, classname);
187                 }
188                 if (item.type == "Valentine") {
189                     if (count == 3) {
190                         count = 1;
191                     } else {
192                         count++;
193                     }
194                     AppendNewFlower(id, count, flowername, tagimage, tagdescription, tagprice, classname);
195                 }
196                 if (item.type == "Another") {
197                     if (count == 3) {
198                         count = 1;
199                     } else {
200                         count++;
201                     }
202                     AppendNewFlower(id, count, flowername, tagimage, tagdescription, tagprice, classname);
203                 }
204             });
205             error: function (err) {
206                 alert("Error: " + err.responseText);
207             }
208         });
209     });
210
211     function showSearchFlowers() {
212         var data = {};
213         data.FlowerName = $("#SearchFlowerName").val();
214         data.ImagePath = "";
215         data.Description = "";
216         data.Price = $("#SearchPrice").val();
217
218         $("#Featured-content").empty();
219
220         $.sign({
221             url: '/Flower/Search',
222             type: 'GET',
223             dataType: 'json',
224             data: data,
225             success: function (data) {
226                 var count = 0;
227                 $.each(data, function (i, item) {
228                     var id = item.Id;
229                     var flowername = item.flowerName;
230                     var tagimage = item.ImagePath;
231                     var tagdescription = item.Description;
232                     var tagprice = item.Price;
233                     if (count == 3) {
234                         count = 1;
235                     } else {
236                         count++;
237                     }
238                     AppendNewFlower(id, count, flowername, tagimage, tagdescription, tagprice, classname);
239                 });
240             error: function (err) {
241                 alert("Error: " + err.responseText);
242             }
243         });
244     }

```

```
    });
  });

  function AppendNewFlower(id, count, flowername, tagimage, tagdescription, tagprice, classname) {
    rows = '';
    rows += '<div id="column' + count + '">';
    rows += '  <div>' + flowername + '</div>';
    rows += '  <div class="image"></div>';
    rows += '  <div class="tagcontent">' + tagdescription + '</div>';
    rows += '  <div class="tagcost">strong>Price: $' + tagprice + '</strong></div>';
    rows += '  <div class="button"><a href="#">Read More</a></p>';
    rows += '  <button type="button" id="btnCreateNew" data-toggle="modal" data-target="#CreateEditModal" class="btn btn-success">';
    rows += '    onclick="return createNewCart(' + id + ') Add to cart</button> ';
    rows += '  </div>';
    rows += '</div>';
    $(classname).append(rows);
  }

  function createNewCart(id) {
    $.ajax({
      url: '/Flower/Get/' + id,
      type: 'GET',
      dataType: 'json',
      success: function (data) {
        $('#flowerName').val(data.FlowerName);
        $('#ppc').val(data.Price);
        $('#date').text("Create New");
        var now = new Date();
        var day = now.getUTC().toString();
        var month = (now.getMonth() + 1).toString();
        var year = now.getUTCYear().toString();
        if (day.length == 1)
          day = "0" + day;
        if (month.length == 1)
          month = "0" + month;
        $('#createDate').val(year + "-" + month + "-" + day);
        createNewCart();
      },
      error: function (err) {
        alert("Error: " + err.responseText);
      }
    });
  }

```

- Flower_Layout

```
Cart_Layout.cshtml      Index.cshtml      Flower_Layout.cshtml      Index.cshtml      FlowerController.cs

 2  Viewbag.Title = "Flower_Layout";
 3  //Layout = "~/Areas/Admin/Views/Shared/_Layout.cshtml";
 4
 5
 6  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/1999/xhtml">
 7  <html xmlns="http://www.w3.org/1999/xhtml">
 8
 9
10  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
11  <meta name="Keywords" content="" />
12  <meta name="Description" content="" />
13  <title>Flower Shop</title>
14
15  <script type="text/javascript" src="~/Scripts/jquery-3.4.1.min.js"></script>
16  <script type="text/javascript" src="~/Scripts/bootstrap.min.js"></script>
17  <link href="~/Content/bootstrap.min.css" rel="stylesheet" type="text/css" />
18  <link href="~/Assets/Menu/vendor/font-awesome/css/font-awesome.min.css" rel="stylesheet" type="text/css" />
19  <link href="~/Assets/Menu/vendor/font-awesome/css/font-awesome.min.css" rel="stylesheet" type="text/css" />
20
21  <renderSection("Script", required: false)>
22
23 </head>
24 <body>
25   <div id="wrapper">
26    <div id="page-wraper">
27     <div id="header" class="container">
28      <div id="logo">
29        <a href="#">Flower Shop</a>
30      </div>
31      <div id="menu">
32        <ul>
33          <li id="Shop"><a href="/Flower/Index">i class="fa fa-shopping-basket fa-fw">/i Shop</a></li>
34          <li id="Cart"><a href="/Cart/Index" id="Cart">i class="fa fa-shopping-cart fa-fw">/i Cart</a></li>
35          <li><a href="/Login/Index">i class="fa fa-power-off fa-fw">/i Logout</a></li>
36        </ul>
37      </div>
38    </div>
39    <div id="banner">
40      <div class="content"></div>
41    </div>
42    <div id="Header" ></div>
43    <renderBody()>
44    <renderSection("FLOWER", required: false)>
45    <renderSection("TULIP", required: false)>
46    <renderSection("BIRTHDAY", required: false)>
47    <renderSection("ANODA", required: false)>
48
49    <div id="page">
50      <div id="content">
51        <div class="post">
52          <h2 class="title"><a href="#">Welcome to Colorediallo </a></h2>
53          <div class="entry">
54            <p>This is a free, fully standards-compliant CSS template designed by <a href="http://templatemonster.com" rel="nofollow">TEMPLATED</a>. The photos in this template are from <a href="http://fotograph.com/"> Fotograph </a>.
55          </div>
56        </div>
57      </div>
58      <div id="content">
59        <div id="sideBar">
60          <ul>
61            <li>
62              <h2>Categories</h2>
63              <ul>
64                <li><a href="#">Aliquid libero</a></li>
65                <li><a href="#">Aliquam ut enim&nbsp;metus</a></li>
66                <li><a href="#">Metus aliquam pellentesque</a></li>
67                <li><a href="#">Suspendisse iaculis mauris</a></li>
68              </ul>
69            </li>
70          </ul>
71        </div>
72      </div>
73    </div>
74  </body>
75  </html>
```

```
FlowerLayout.cshtml 0 X index.cshtml FlowerController.cs

1<div>
2    <div>
3        <ul>
4            <li><a href="#">Home</a></li>
5            <li><a href="#">About</a></li>
6            <li><a href="#">Contact</a></li>
7        </ul>
8    </div>
9    <div id="content">
10        <div>
11            <h1>Flower Shop</h1>
12            <p>A flower shop website template</p>
13            <h2>Our Products</h2>
14            <ul class="list-style">
15                <li class="first"><a href="#">Maecenas luctus lectus at sapien</a></li>
16                <li><a href="#">Etiam phocous volutpat erat</a></li>
17                <li><a href="#">Etiam posuere augue sit amet nisl</a></li>
18                <li><a href="#">Mauris vulputate dolor sit amet nibh</a></li>
19                <li><a href="#">Etiam posuere augue sit amet nisl</a></li>
20            </ul>
21        </div>
22        <div id="fbbox1">
23            <h2>Nulla luctus eleifend</h2>
24            <ul class="list-style">
25                <li class="first"><a href="#">Maecenas luctus lectus at sapien</a></li>
26                <li><a href="#">Etiam phocous volutpat erat</a></li>
27                <li><a href="#">Etiam posuere augue sit amet nisl</a></li>
28                <li><a href="#">Mauris vulputate dolor sit amet nibh</a></li>
29                <li><a href="#">Etiam posuere augue sit amet nisl</a></li>
30            </ul>
31        </div>
32        <div id="fbbox2">
33            <h2>Nulla luctus lectus</h2>
34            <ul class="list-style">
35                <li class="first"><a href="#">Maecenas luctus lectus at sapien</a></li>
36                <li><a href="#">Etiam phocous volutpat erat</a></li>
37                <li><a href="#">Etiam posuere augue sit amet nisl</a></li>
38                <li><a href="#">Mauris vulputate dolor sit amet nibh</a></li>
39                <li><a href="#">Etiam posuere augue sit amet nisl</a></li>
40            </ul>
41        </div>
42        <div id="fbbox3">
43            <h2>Maecenas luctus lectus</h2>
44            <ul class="list-style">
45                <li class="first"><a href="#">Maecenas luctus lectus at sapien</a></li>
46                <li><a href="#">Etiam phocous volutpat erat</a></li>
47                <li><a href="#">Etiam posuere augue sit amet nisl</a></li>
48                <li><a href="#">Mauris vulputate dolor sit amet nibh</a></li>
49                <li><a href="#">Etiam posuere augue sit amet nisl</a></li>
50            </ul>
51        </div>
52    </div>
53    <div id="Footer">
54        <p>Copyright. Untitled. All rights reserved. Images by <a href="http://fotograph.com/">Fotograph</a>. Design by <a href="http://templated.co" rel="nofollow">TEMPLATED</a>.</p>
55    </div>
56</div> <!-- end Footer -->
57<#include section="MODEL_DIALOG", required="false" />
58<script>
59    $('#Cart').click(function () {
60        $('#Cart').addClass('current_page_item');
61        $('#Shop').removeClass('current_page_item');
62    });
63
64    $('#Shop').click(function () {
65        $('#Shop').addClass('current_page_item');
66        $('#Cart').removeClass('current_page_item');
67    });
68</script>
69</body>
70</html>
71
```

CartController

```
CartController.cs 4 x
lab03                                         lab03.Areas.Main.Controllers.CartController
[!]
7  namespace lab03.Areas.Main.Controllers
8  {
9      public class CartController : Controller
10     {
11         Company00DataContext _context = new Company00DataContext();
12
13         // GET: Main/Cart
14         public ActionResult Index()
15         {
16             return View();
17         }
18
19         public ActionResult ListCart()
20         {
21             var deal = _context.Deals.ToList();
22             string email = (string)Session["email"];
23             var dealbyid = _context.Deals.Where(x => x.CustomerEmail == email);
24             return Json(dealbyid, JsonRequestBehavior.AllowGet);
25         }
26
27         public ActionResult SearchUser()
28         {
29             string email = (string)Session["email"];
30             var dealbyid = _context.Accounts.Where(x => x.Email == email);
31             return Json(dealbyid, JsonRequestBehavior.AllowGet);
32         }
33
34         [HttpPost]
35         public ActionResult Create([Bind(Exclude = "Id")] Deal deal)
36         {
37             if (ModelState.IsValid)
38             {
39                 _context.Deals.InsertOnSubmit(deal);
40                 _context.SubmitChanges();
41             }
42             return Json(deal, JsonRequestBehavior.AllowGet);
43         }
44
45         [HttpPost]
46         public ActionResult Delete(int id)
47         {
48             var deal = _context.Deals.ToList().Find(m => m.Id == id);
49             if (deal != null)
50             {
51                 _context.Deals.DeleteOnSubmit(deal);
52                 _context.SubmitChanges();
53             }
54             return Json(deal, JsonRequestBehavior.AllowGet);
55         }
56
57         public ActionResult Get(int id)
58         {
59             var deal = _context.Deals.ToList().Find(m => m.Id == id);
60             return Json(deal, JsonRequestBehavior.AllowGet);
61         }
62
63         [HttpPost]
64         public ActionResult Update(Deal deal)
65         {
66             if (ModelState.IsValid)
67             {
68                 Deal de = (from p in _context.Deals
69                             where p.Id == deal.Id
70                             select p).SingleOrDefault();
71             }
72         }
73     }
74 }
```

```
72         de.CustomerName = deal.CustomerName;
73         de.CustomerEmail = deal.CustomerEmail;
74         de.DealDate = deal.DealDate;
75         de.DealName = deal.DealName;
76         de.Quantity = deal.Quantity;
77         de.Total = deal.Total;
78         _context.Subitchanges();
79     }
80 
81     return Json(deal, JsonRequestBehavior.AllowGet);
82 }
83 
84 [HttpPost]
85 public ActionResult Reports(string ReportType)
86 {
87     LocalReport localReport = new LocalReport();
88     localReport.ReportPath = Server.MapPath("~/Reported/Report1.rdlc");
89     var reportDataSource = localReport.DataSources.Add("reportDataSource");
90     reportDataSource.Name = "Dealdata";
91     string email = (string)Session["email"];
92     var datayield = from x in deals where x.CustomerEmail == email;
93     ReportDataSource.DataSource = datayield;
94     localReport.DataSources.Add(reportDataSource);
95     string ReportType = ReportType;
96     string mimeType = string.Empty;
97     string encoding = string.Empty;
98     string fileNameExtension = string.Empty;
99     if (ReportType == "Excel")
100    {
101        fileNameExtension = "xlsx";
102    }
103    else if (ReportType == "PDF")
104    {
105        fileNameExtension = "pdf";
106    }
107    else if (ReportType == "Word")
108    {
109        fileNameExtension = "docx";
110    }
111    else
112    {
113        fileNameExtension = "Jpg";
114    }
115    string[] streams;
116    Warning[] warnings;
117    byte[] renderedBytes;
118    renderedBytes = localReport.Render(ReportType, "", out mimeType, out encoding, out fileNameExtension, out streams, out warnings);
119    HttpResponse response = Response;
120    response.AddHeader("Content-Disposition", "attachment;filename=deal_report." + fileNameExtension);
121    return File(renderedBytes, fileNameExtension);
122 }
123 }
124 }
```

Indexcart

```

139     showDeal();
140     clearModalContent();
141   },
142   error: function (err) {
143     alert('Error: ' + err.responseText);
144   }
145 });
146 } else {
147   $.ajax({
148     url: '/Cart/Update',
149     type: 'POST',
150     dataType: 'json',
151     data: data,
152     success: function (data) {
153       showDeal();
154       clearModalContent();
155     },
156     error: function (err) {
157       alert('Error: ' + err.responseText);
158     }
159   });
160 }
161 
162 function clearModalContent() {
163   $('#UserName').val("");
164   $('#Quantity').val("");
165   isUpdatable = false;
166 }
167 
168 function createnewDeal() {
169   $.ajax({
170     url: '/Cart/SearchUser',
171     type: 'GET',
172     dataType: 'json',
173     success: function (data) {
174       $.each(data, function (i, item) {
175         $('#CustomerName').val(item.UserName);
176         $('#CustomerEmail').val(item.Email);
177       })
178     }
179   });
180 
181 function showDeal() {
182   if ($('#tbody').empty()) {
183     $.ajax({
184       url: '/Cart/ListsCart',
185       type: 'GET',
186       dataType: 'json',
187       success: function (data) {
188         var totalamount = 0;
189         $.each(data, function (i, item) {
190           var id = item.Id;
191           var customername = item.CustomerName;
192           var customeremail = item.CustomerEmail;
193           var dealname = item.DealName;
194           var flowername = item.FlowerName;
195           var quantity = item.Quantity;
196           var total = item.Total;
197           totalamount += total;
198           AppendNewDeal(id, customername, customeremail, date, flowername, quantity, total);
199         });
200         rows = '';
201         rows += '<tr><td>' + totalamount + '</td></tr>';
202         $('#tbody').append(rows);
203       },
204       error: function (err) {
205         alert('Error: ' + err.responseText);
206       }
207     });
208   }
209 
210   function AppendNewDeal(id, customername, customeremail, date, flowername, quantity, total) {
211     rows = '';
212     rows += '<td>';
213     rows += '<td>' + customername + '</td>';
214     rows += '<td>' + customeremail + '</td>';
215     rows += '<td>' + date + '</td>';
216     rows += '<td>' + flowername + '</td>';
217     rows += '<td>' + quantity + '</td>';
218     rows += '<td>' + total + '$</td>';
219     rows += '<td>';
220     rows += '<button type='button' id='btnEdit' class='btn btn-primary' +
221       '>Edit</button><button type='button' id='btnDelete' class='btn btn-danger' +
222       '>Delete</button>' + '</td>';
223     rows += '</tr>';
224     $('#tbody').append(rows);
225   };
226 
227   function deleteDeal(id) {
228     var confirmDelete = confirm("Do you want to delete deal?");
229     if (confirmDelete) {
230       $.ajax({
231         url: '/Cart/Delete/' + id,
232         type: 'POST',
233         dataType: 'json',
234         success: function (data) {
235           showDeal();
236         },
237         error: function (err) {
238           alert('Error: ' + err.responseText);
239         }
240       });
241     }
242   }
243 
244   function getDetailDeal(id) {
245     isUpdatable = true;
246     currentId = id;
247     $('#title').text("Edit");
248 
249     $.ajax({
250       url: '/Cart/Get/' + id,
251       type: 'GET',
252       dataType: 'json',
253       success: function (data) {
254         $('#CustomerName').val(data.CustomerName);
255         $('#CustomerEmail').val(data.CustomerEmail);
256         $('#DealName').val(data.DealName);
257         $('#FlowerName').val(data.FlowerName);
258         $('#Quantity').val(data.Quantity);
259         $('#Price').val(data.Price);
260         $('#Total').val(data.Total + '$');
261         currentId = data.Id;
262       },
263       error: function (err) {
264         alert("Error: " + err.responseText);
265       }
266     });
267   }
268 }
269 
```

- Cart_Layout

```
CartLayout.cshtml  Index.cshtml  CartController.cs
1 <link rel="stylesheet" href="~/Assets/Table/css/style.css">
2
3     ViewBag.Title = "Cart_Layout";
4     Layout = "~/Areas/Main/Views/Shared/Flower_Layout.cshtml";
5
6     <h2>Print report for cart</h2>
7     <div style="margin-left:280px;">
8         <a href="#">Export to Excel</a>, <a href="#">Reports</a>, new { ReportType = "Excel" }, new { @class = "btn btn-info" } ) | 
9         <a href="#">Export to PDF</a>, <a href="#">Reports</a>, new { ReportType = "PDF" }, new { @class = "btn btn-info" } ) |
10        <a href="#">Export to Word</a>, <a href="#">Reports</a>, new { ReportType = "Word" }, new { @class = "btn btn-info" } ) |
11        <a href="#">Export to Image</a>, <a href="#">Reports</a>, new { ReportType = "Image" }, new { @class = "btn btn-info" } )
12    </div>
13    <div id="wrapper" style="margin: 0 60px;">
14        <h1><span class="blue"><i>A&lt;/i></span><span class="blue">Cart</span><span class="yellow">Information</span></h1>
15
16        <table class="container">
17            <thead>
18                <tr>
19                    <th>CustomerName</th>
20                    <th>CustomerEmail</th>
21                    <th>Date</th>
22                    <th>FlowerName</th>
23                    <th>Price</th>
24                    <th>Total</th>
25                    <th>Manipulation</th>
26                </tr>
27            </thead>
28            <tbody>
29                <tr>
30                    <t>RenderSection("TABLE", required: false)</t>
31                </tbody>
32            </table>
33
34        </div>
35
36        <div>
37            <h3>RenderSection("MODEL_DIALOG", required: false)</h3>
38            <h3>RenderSection("script", required: false)</h3>
39        </div>
40
41    </div>
42
```

- AccountModel

```
AccountModel.cs  Index.cshtml  CartController.cs
1 lab03
2     using System;
3     using System.Collections.Generic;
4     using System.Linq;
5     using System.Web;
6
7     namespace lab03.Areas.Main
8     {
9         public class AccountModel
10     {
11         private CompanyDBDataContext context = null;
12
13         public AccountModel()
14         {
15             context = new CompanyDBDataContext();
16         }
17
18         public bool Login(string Email, string Password)
19         {
20             bool res = false;
21             context.sp_Account_LoginUser_Chek(Email, Password, ref res);
22
23             return res ?? false;
24         }
25
26         public bool Create(string Username, string Email)
27         {
28             bool res = false;
29             context.sp_Check_Account(Username, Email, ref res);
30
31             return res ?? false;
32         }
33     }
34 }
```

- LoginModel

```
LoginModel.cs  AccountModel.cs
1 lab03
2     using System;
3     using System.Collections.Generic;
4     using System.ComponentModel.DataAnnotations;
5     using System.Linq;
6     using System.Web;
7
8     namespace Tab03.Areas.Main
9     {
10         public class LoginModel
11         {
12             [Required]
13             public string Email { get; set; }
14             public string EmailRegister { get; set; }
15             public string Password { get; set; }
16             public string FirstName { get; set; }
17             public string LastName { get; set; }
18         }
19     }
20 }
```

- MainAreaRegistration

```
MainAreaRegistration.cs  LoginModel.cs  AccountModel.cs
1 lab03
2     using System.Web.Mvc;
3
4     namespace lab03.Areas.Main
5     {
6         public class MainAreaRegistration : AreaRegistration
7         {
8             public override string AreaName
9             {
10                 get
11                 {
12                     return "Main";
13                 }
14             }
15
16             public override void RegisterArea(AreaRegistrationContext context)
17             {
18                 context.MapRoute(
19                     "Main_default",
20                     "Main/{controller}/{action}/{id}",
21                     new { action = "Index", id = UrlParameter.Optional }
22                 );
23             }
24         }
25     }
```

2. Database:

- Dbo.Account:

The screenshot shows the 'dbo.Account' table definition in SSMS. The table has the following columns:

Column Name	Data Type	Allow Nulls
Id	int	<input type="checkbox"/>
UserName	nvarchar(50)	<input checked="" type="checkbox"/>
Email	nvarchar(50)	<input checked="" type="checkbox"/>
FirstName	nvarchar(50)	<input checked="" type="checkbox"/>
LastName	nvarchar(50)	<input checked="" type="checkbox"/>
Password	nvarchar(50)	<input checked="" type="checkbox"/>
AccountType	nvarchar(20)	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

In the 'Column Properties' pane, the 'Identity Specification' section is expanded, showing the following settings:

Setting	Value
(Is Identity)	Yes
Identity Increment	1
Identity Seed	1
Indexable	Yes
Is Columnset	No

- Dbo.Deals:

Column Name	Data Type	Allow Nulls
Id	int	<input type="checkbox"/>
CustomerName	nvarchar(50)	<input checked="" type="checkbox"/>
CustomerEmail	nvarchar(50)	<input checked="" type="checkbox"/>
DealDate	nvarchar(11)	<input checked="" type="checkbox"/>
FlowerName	nvarchar(50)	<input checked="" type="checkbox"/>
Quantity	int	<input checked="" type="checkbox"/>
Total	decimal(12, 2)	<input checked="" type="checkbox"/>
Price	decimal(12, 2)	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

Column Properties

Full-text Specification	
Has Non-SQL Server Subscriber	No
Identity Specification	
(Is Identity)	Yes
Identity Increment	1
Identity Seed	1

- Dbo.Flowers

The screenshot shows the 'Flowers' table structure in SQL Server Management Studio. The table has the following columns:

Column Name	Data Type	Allow Nulls
Id	int	<input type="checkbox"/>
FlowerName	nvarchar(50)	<input checked="" type="checkbox"/>
ImagePath	nvarchar(50)	<input checked="" type="checkbox"/>
Description	text	<input checked="" type="checkbox"/>
Price	nvarchar(20)	<input checked="" type="checkbox"/>
Type	nvarchar(50)	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

Below the table, the 'Column Properties' window is open for the 'Id' column. The 'Identity Specification' section is selected, showing the following values:

Property	Value
(Is Identity)	Yes
Identity Increment	1
Identity Seed	1

- Query

Ngoài ra còn 4 Stored Procedure để kiểm tra đăng nhập là admin hay client và create admin và client.

```
SQLQuery1.sql - LA...PNUF7\Hunter (65) ▾ X
1 CREATE TABLE Flowers (
2     Id int identity,
3     FlowerName NVARCHAR(50) not null,
4     ImagePath NVARCHAR(50) not null,
5     [Description] text not null,
6     Price nvarchar(20) not null,
7
8     PRIMARY KEY (Id)
9 )
10
11 create table Account(
12     Id int identity,
13     UserName nvarchar(50) not null,
14     Email nvarchar(50) not null,
15     FirstName nvarchar(50) not null,
16     LastName nvarchar(50) not null ,
17     Password nvarchar(50) not null,
18     AccountType nvarchar(20) not null
19
20     PRIMARY KEY (Id)
21 )
22
23 create table Deal(
24     Id int identity,
25     CustomerName nvarchar(50) not null,
26     CustomerEmail nvarchar(50) not null,
27     DealDate nvarchar(11) not null,
28     FlowerName nvarchar(50) not null,
29     Quantity int not null,
30
31     primary key (Id)
32 )
33
34 create procedure [dbo].[sp.Account_Admin_Login_Check]
35     @Email nvarchar(50),
36     @Password nvarchar(50),
37     @res bit output
```

Stored kiểm tra:

```
34  create procedure [dbo].[sp.Account_Admin_Login_Check]
35      @Email nvarchar(50),
36      @Password nvarchar(50),
37      @res bit output
38  as
39  begin
40      declare @count int
41
42      select @count = count(*)
43      from Account
44      where Email = @Email and [Password] = @Password and LOWER(AccountType) = 'admin'
45
46      if(@count > 0 )
47          set @res = 1
48      else
49          set @res = 0
50  end
51
52  create procedure [dbo].[sp.Account_Customer_Login_Check]
53      @Email nvarchar(50),
54      @Password nvarchar(50),
55      @res bit output
56  as
57  begin
58      declare @count int
59
60      select @count = count(*)
61      from Account
62      where Email = @Email and [Password] = @Password and LOWER(AccountType) = 'customer'
63
64      if(@count > 0 )
65          set @res = 1
66      else
67          set @res = 0
68  end
69
70
```

SQLQuery3.sql - LA...PNUF7\Hunter (67) X SQLQuery2.sql - LA...PNUF7\Hunter (70) SQLQuery1.sql - LA...PNUF7\Hunter (68)

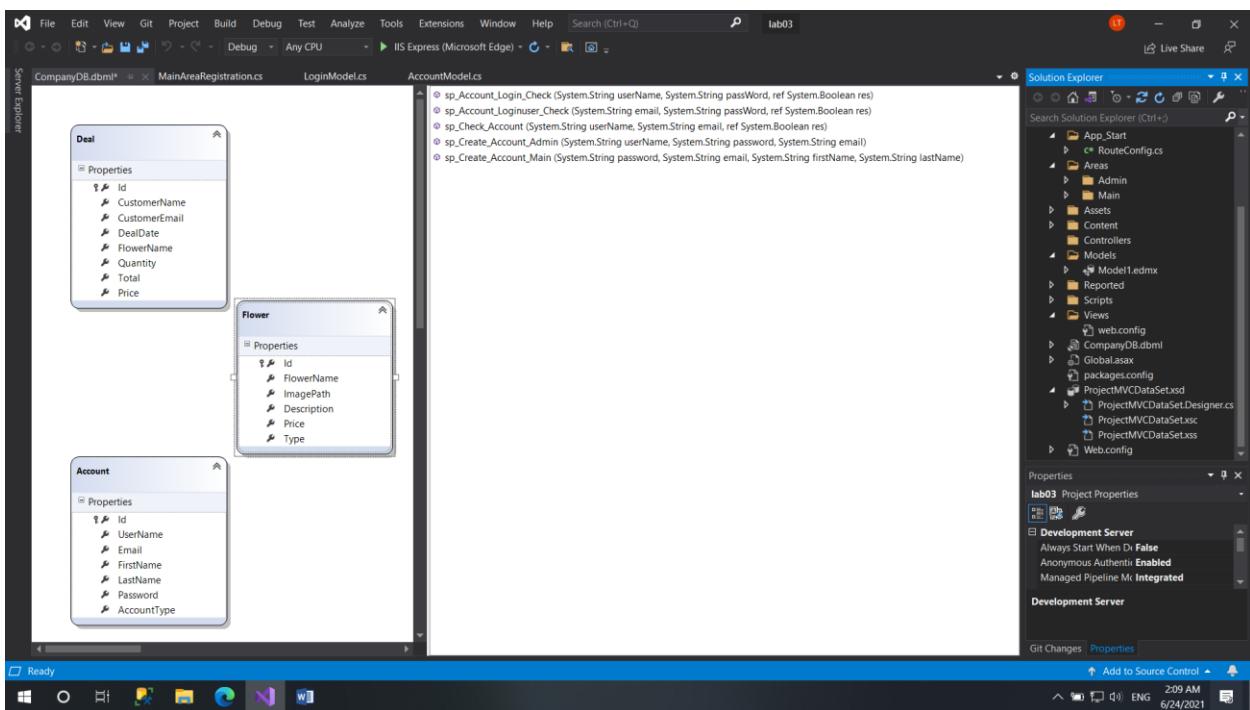
```
1 USE [ProjectMVC ]
2 GO
3 /*===== Object: StoredProcedure [dbo].[sp_Check_Account]    Script Date: 6/24/2021 1:40:37 AM =====*/
4 SET ANSI_NULLS ON
5 GO
6 SET QUOTED_IDENTIFIER ON
7 GO
8 ALTER PROCEDURE [dbo].[sp_Check_Account]
9     @UserName NVARCHAR(20),
10    @Email NVARCHAR(30),
11    @res BIT OUTPUT
12 as
13 begin
14     declare @count INT
15
16     select @count = COUNT(*) FROM Account where @UserName = UserName or @Email = Email
17
18     if @count > 0
19         set @res = 1
20     else
21         set @res = 0
22 end
```

Stored create:

```
SQLQuery1.sql - LA...PNUF7\Hunter (68)  X  SQLQuery1.sql - LA...PNUF7\Hunter (65)
1 USE [ProjectMVC]
2 GO
3 /* Object: StoredProcedure [dbo].[sp_Create_Account_Admin]   Script Date: 6/24/2021 1:39:57 AM *****/
4 SET ANSI_NULLS ON
5 GO
6 SET QUOTED_IDENTIFIER ON
7 GO
8 =====
9 -- Author: <Author,,Name>
10 -- Create date: <Create Date,,>
11 -- Description: <Description,,>
12 =====
13 ALTER PROCEDURE [dbo].[sp_Create_Account_Admin]
14     @UserName NVARCHAR(30),
15     @Password NVARCHAR(30),
16     @Email NVARCHAR(30)
17 AS
18 BEGIN
19     SET NOCOUNT ON;
20     INSERT INTO Account(UserName,Password,Email,FirstName,LastName,AccountType) VALUES (@UserName,@Password,@Email,'Admin','Admin','Admin')
21 END
22

SQLQuery2.sql - LA...PNUF7\Hunter (70)  X  SQLQuery1.sql - LA...PNUF7\Hunter (68)  SQLQuery1.sql - LA...PNUF7\Hunter (65)
1 USE [ProjectMVC]
2 GO
3 /* Object: StoredProcedure [dbo].[sp_Create_Account_Main]   Script Date: 6/24/2021 1:40:16 AM *****/
4 SET ANSI_NULLS ON
5 GO
6 SET QUOTED_IDENTIFIER ON
7 GO
8 =====
9 -- Author: <Author,,Name>
10 -- Create date: <Create Date,,>
11 -- Description: <Description,,>
12 =====
13 ALTER PROCEDURE [dbo].[sp_Create_Account_Main]
14     @Password NVARCHAR(30),
15     @Email NVARCHAR(30),
16     @firstName NVARCHAR(30),
17     @lastName NVARCHAR(30)
18 AS
19 BEGIN
20     -- SET NOCOUNT ON added to prevent extra result sets from
21     -- interfering with SELECT statements.
22     SET NOCOUNT ON;
23
24     -- Insert statements for procedure here
25     INSERT INTO Account(UserName,Password,Email,FirstName,LastName,AccountType) VALUES (@FirstName + ' ' + @LastName,@Password,@Email,@FirstName,@LastName)
26 END
27
```

- Database in visual



3. Result:

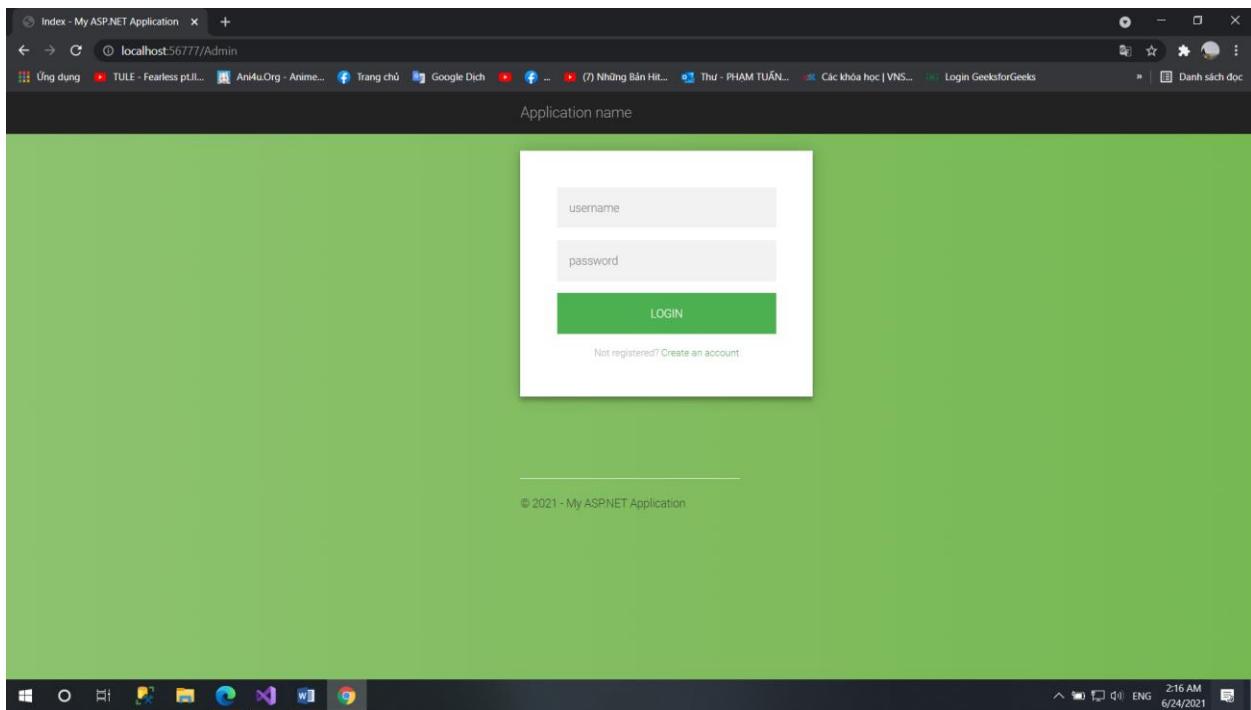
Phần bài chia làm 2 phần phần admin và phần của client.

Phần của admin có thể chỉnh sửa thêm xóa và tìm kiếm sản phẩm và chỉ có thể chỉnh sửa và xóa đơn hàng của khách hàng, có thể quản lý tất cả các tài khoản của admin cũng như của user (Edit, Details, Delete). Và cũng có thể in report các đơn hàng và tổng tiền của chúng.

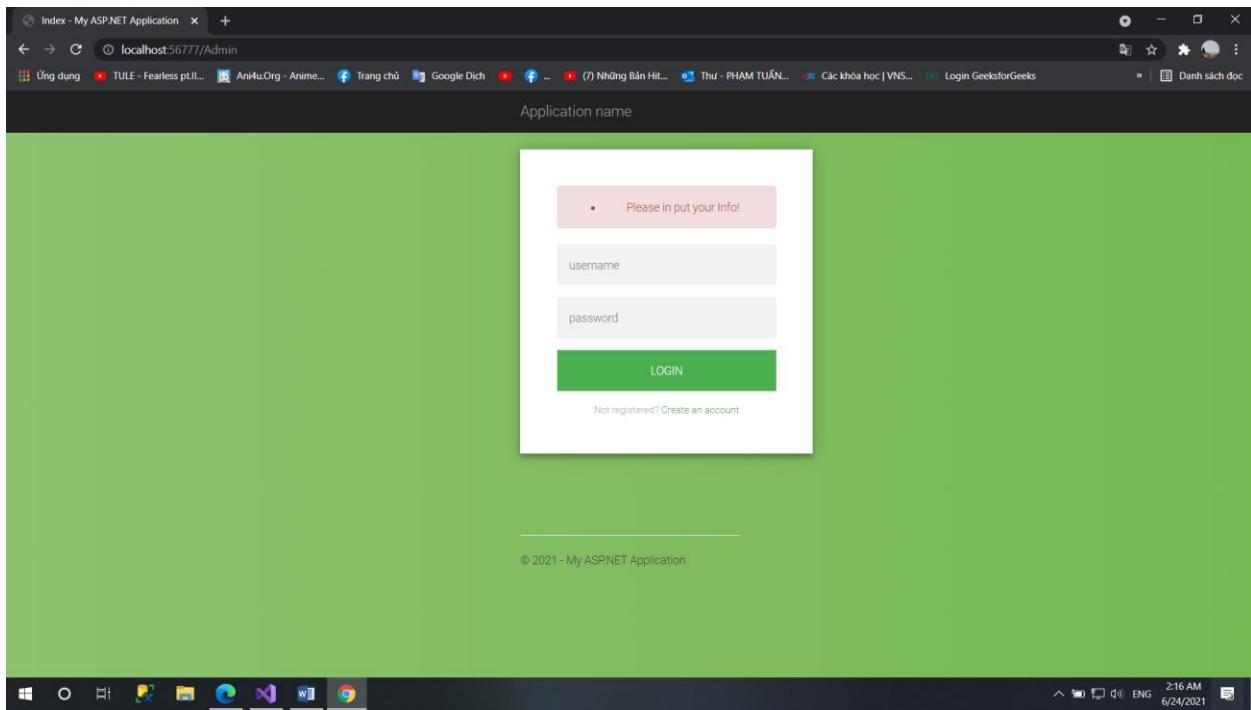
Phần của user, người dùng có thể thêm đơn hàng bằng cách ấn vào Add to cart và điền số lượng vào và chỉ cần ấn xác nhận. Về phần giỏ hàng thì người dùng chỉ có thể nhìn thấy được giỏ hàng của bản thân mình và cũng có thể in report về đơn hàng của bản thân.

a. Admin

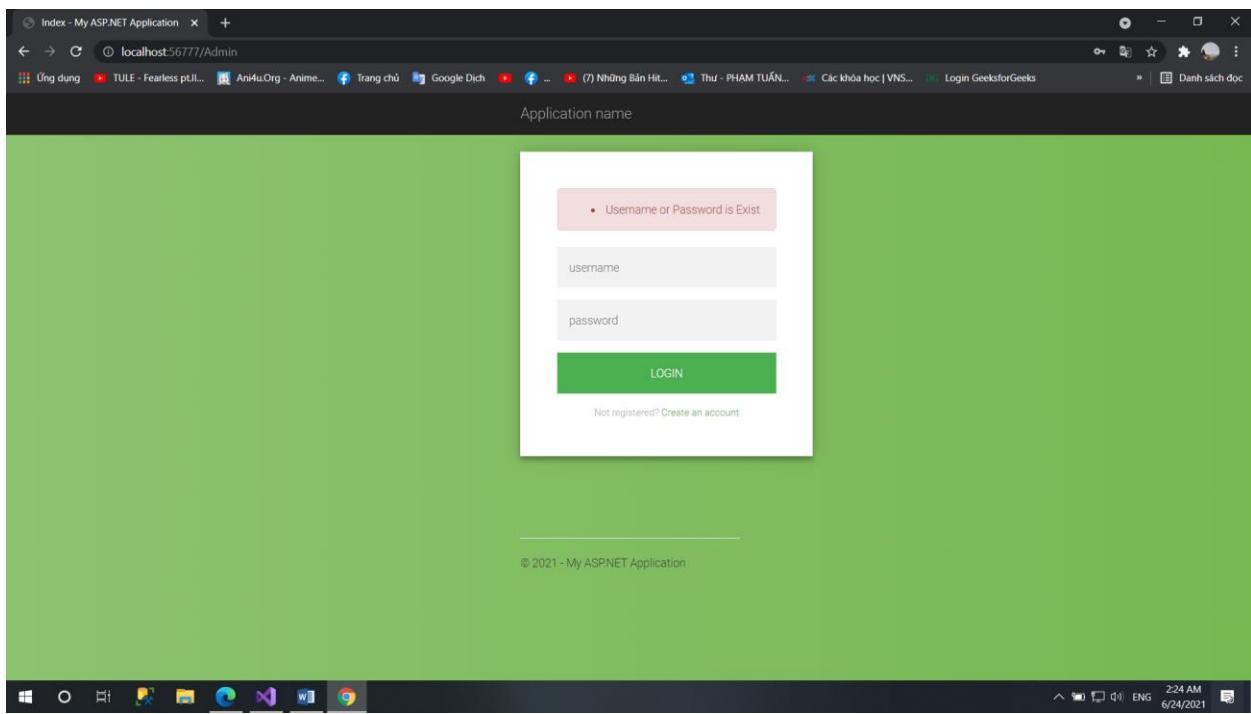
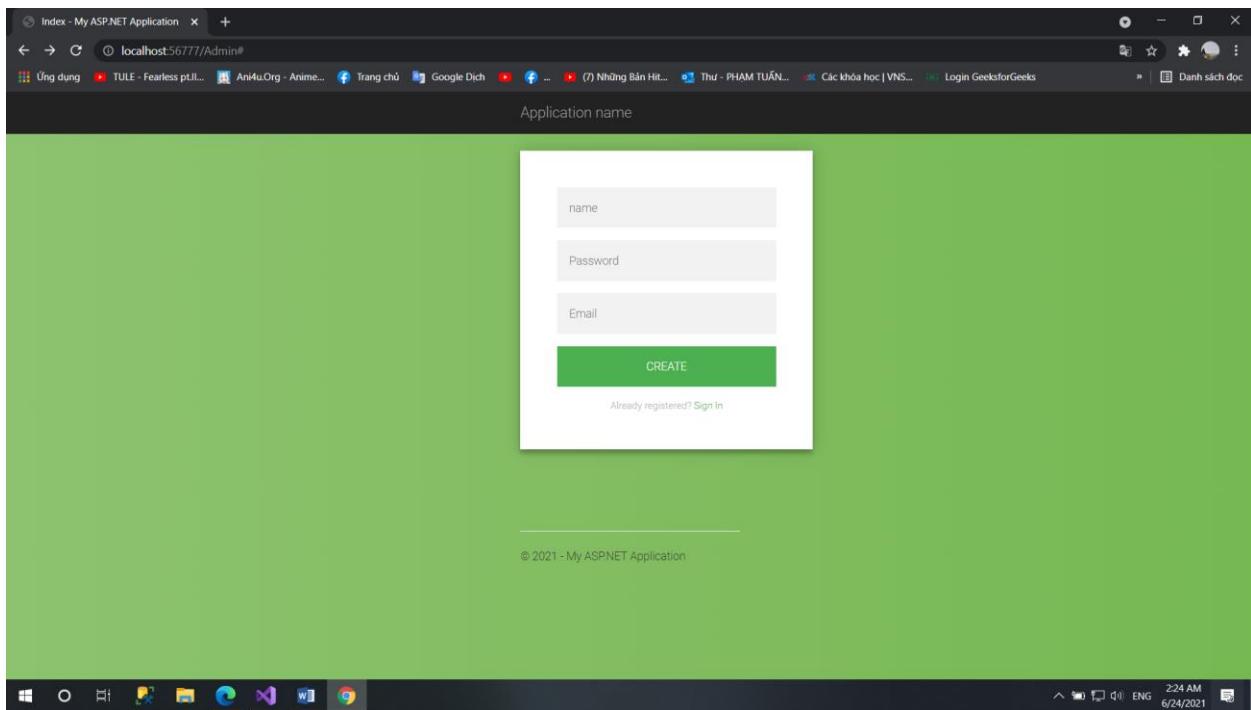
- Login:



- Thông báo lỗi khi người dùng nhập sai.



- Create admin account



- Khi admin đăng nhập vào:

- Manager Flower:

Welcome admin to FlowerShop

Search...

Search Flower

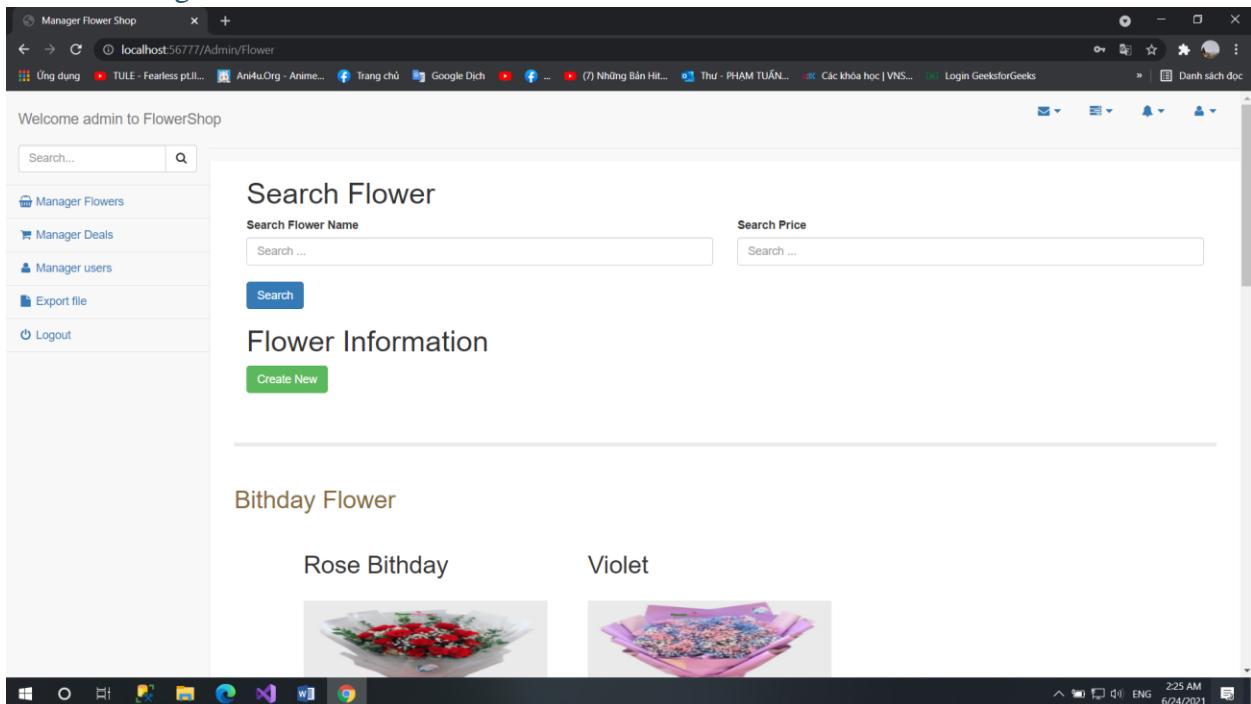
Search Flower Name Search Price

Flower Information

Bithday Flower

Rose Bithday

Violet



Price: 4.0\$

Tulip Flower

Tulip

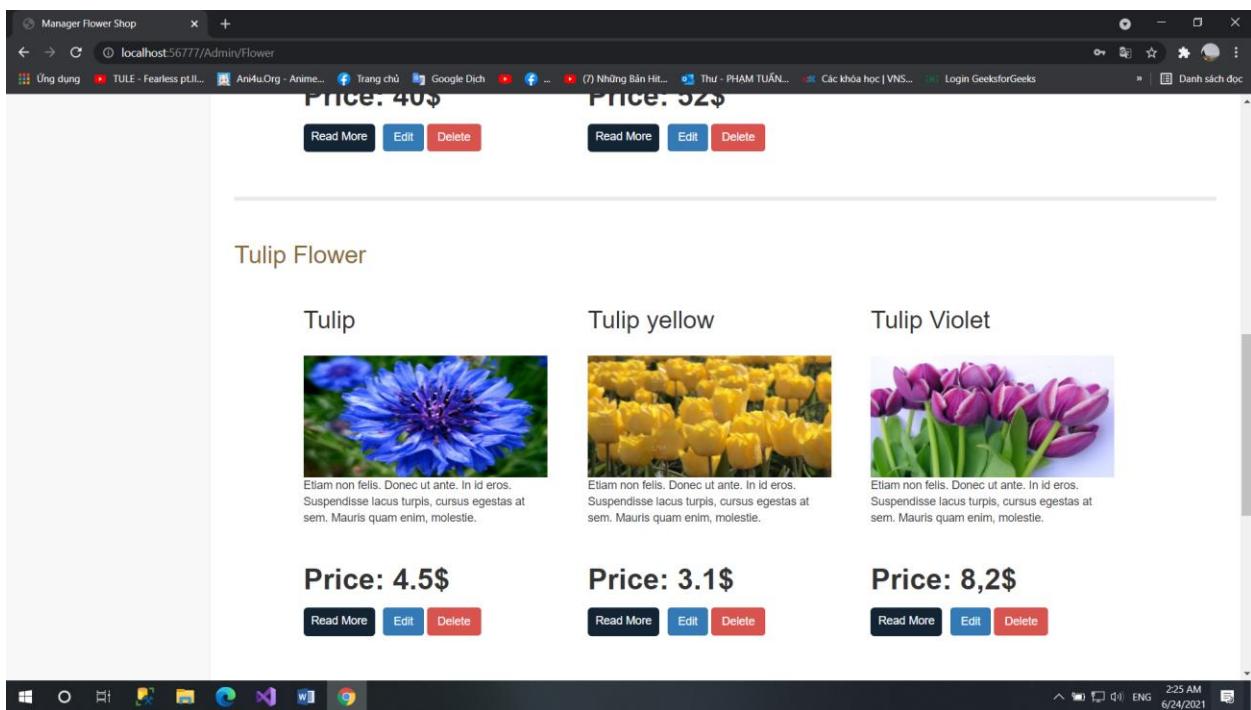
Tulip yellow

Tulip Violet

Price: 4.5\$

Price: 3.1\$

Price: 8.2\$



Screenshot of a web application titled "Manager Flower Shop" showing a list of flower products. The products are displayed in three columns:

Product Name	Image	Price
FUSCE FRINGILLA		Price: 4.0\$
PRAESENT MATTIS		Price: 5.1\$
MAECENAS LUCTUS		Price: 6.2\$

Each product row contains "Read More", "Edit", and "Delete" buttons.

Khi admin tìm kiếm 1 loại sản phẩm:

Screenshot of a web application titled "Manager Flower Shop" showing a search interface for flowers. The search results for "Violet" are displayed:

Flower Information
Create New

Bithday Flower

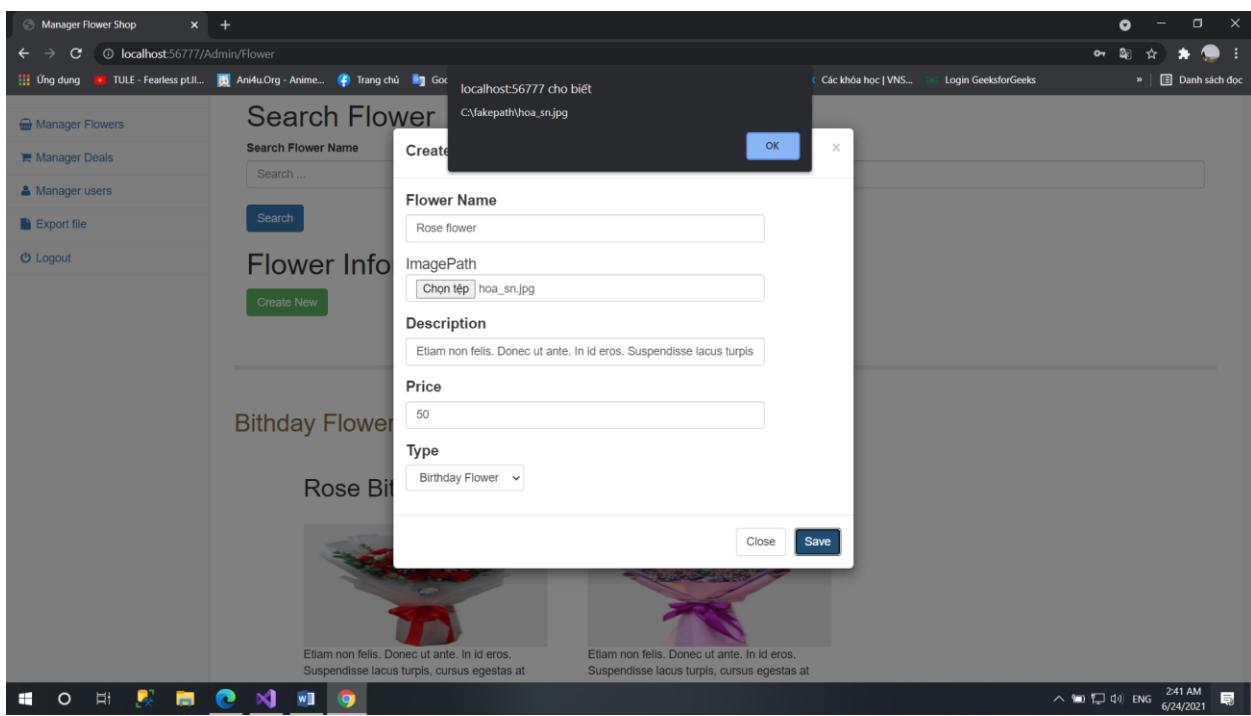
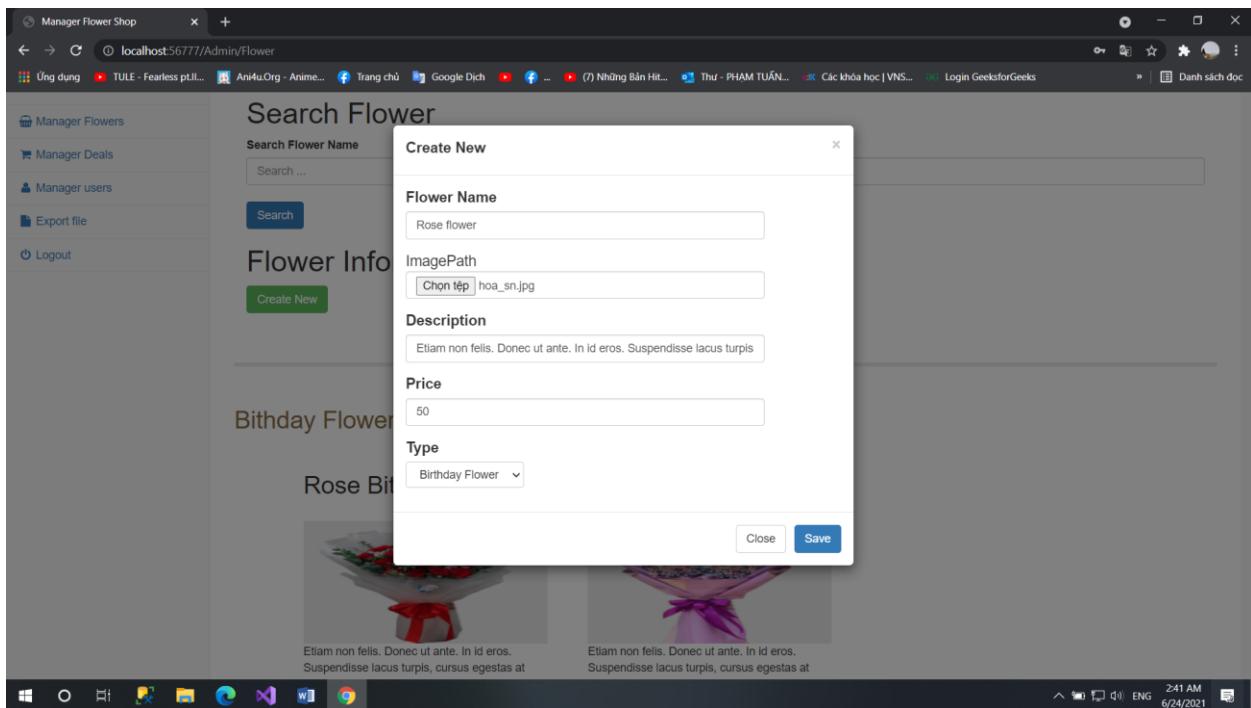
Violet

Etim non felis. Donec ut ante. In id eros. Suspendisse lacus turpis, cursus egestas at sem. Mauris quam enim, molestie.

Price: 52\$

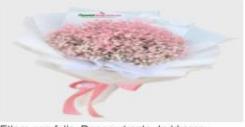
[Read More](#) [Edit](#) [Delete](#)

Khi admin thêm 1 loại sản phẩm.



Manager Flower Shop localhost:56777/Admin/Flower

Bithday Flower

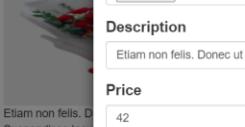
Rose Bithday	Violet	Rose flower
		
Etiam non felis. Donec ut ante. In id eros. Suspendisse lacus turpis, cursus egestas at sem. Mauris quam enim, molestie.	Etiam non felis. Donec ut ante. In id eros. Suspendisse lacus turpis, cursus egestas at sem. Mauris quam enim, molestie.	Etiam non felis. Donec ut ante. In id eros. Suspendisse lacus turpis, cursus egestas at sem. Mauris quam enim, molestie.
Price: 40\$	Price: 52\$	Price: 50\$
Read More Edit Delete	Read More Edit Delete	Read More Edit Delete

Tulip Flower

Khi admin edit 1 loại sản phẩm:

Manager Flower Shop localhost:56777/Admin/Flower

Bithday Flower

Rose Bithday	Rose flower
	
Etiam non felis. Donec ut ante. In id eros. Suspendisse lacus turpis, cursus egestas at sem. Mauris quam enim, molestie.	Etiam non felis. Donec ut ante. In id eros. Suspendisse lacus turpis, cursus egestas at sem. Mauris quam enim, molestie.
Price:	Price: 50\$
Read More Edit Delete	Read More Edit Delete

Edit

Flower Name: Red Rose Bithday

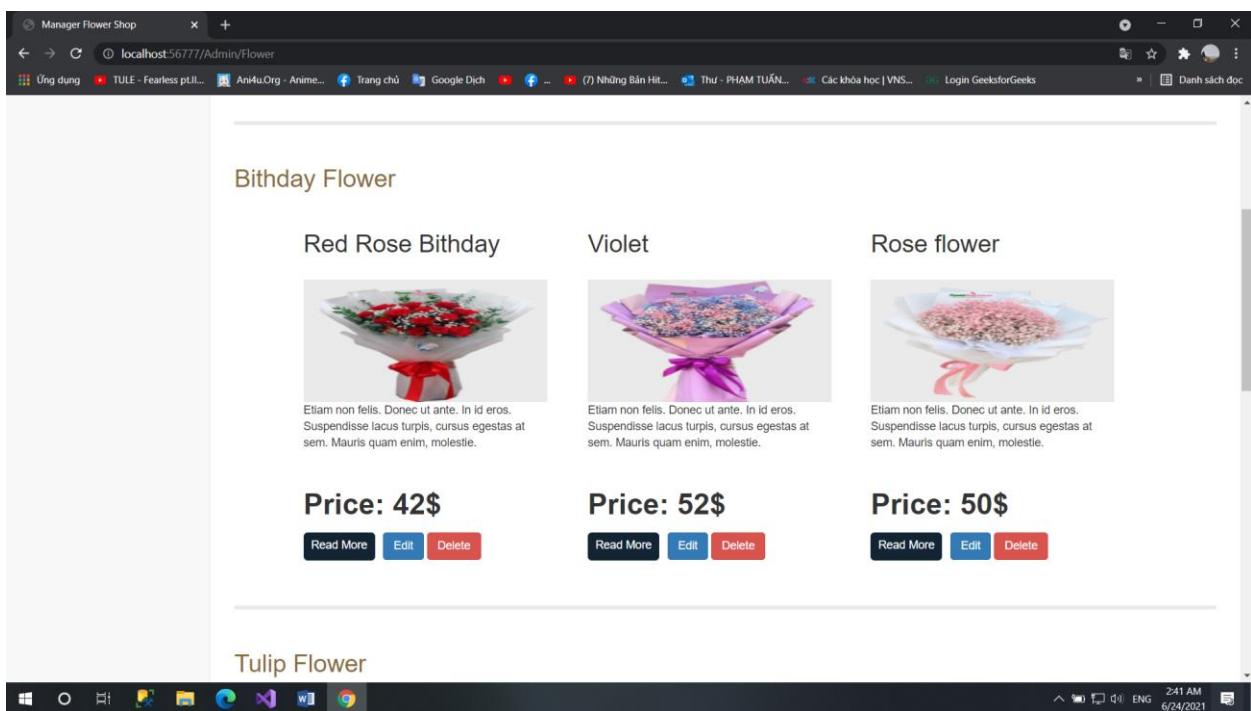
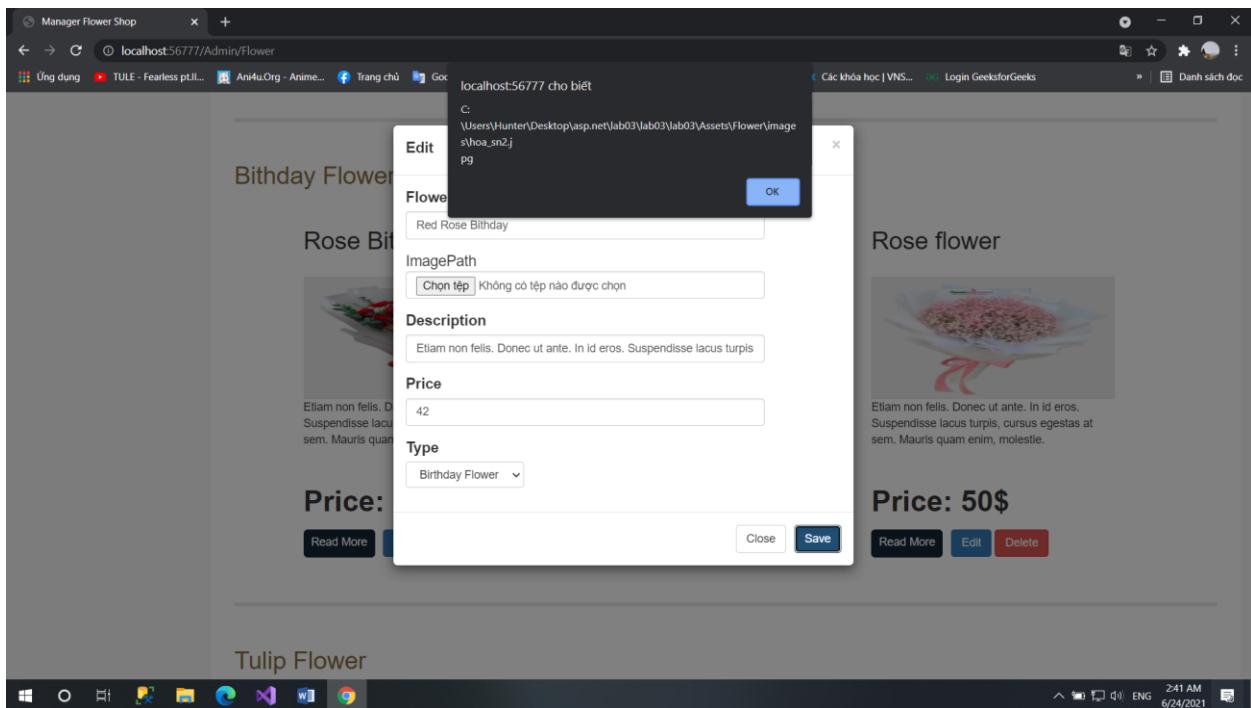
ImagePath: Chọn tệp | Không có tệp nào được chọn

Description: Etiam non felis. Donec ut ante. In id eros. Suspendisse lacus turpis

Price: 42

Type: Birthday Flower

[Close](#) [Save](#)



- ManagerDeals:
- Có thêm chức năng search Deal cho admin có thể tìm kiếm 1 cách nhanh chóng.
- Và sẽ có phần tổng kết số tiền thu về.

Manager Flower Shop

localhost:56777/Admin/Deal/Index

Welcome admin to FlowerShop

Search...

Manager Flowers

Manager Deals

Manager users

Export file

Logout

Search Deals

Search Customer Name
Search ...

Search Customer Email
Search ...

Search Date Deal
Search ...

Search Flower Name
Search ...

Deals Information

CustomerName	CustomerEmail	Date	FlowerName	Quantity	Total	Manipulation
user	user@gmail.com	2021-06-13	FUSCE FRINGILLA	3	9 \$	<input type="button" value="Edit"/> <input type="button" value="Delete"/>
user1	user1@gmail.com	2021-06-14	FUSCE FRINGILLA	2	6 \$	<input type="button" value="Edit"/> <input type="button" value="Delete"/>
user	user@gmail.com	2021-06-14	FUSCE FRINGILLA	10	30 \$	<input type="button" value="Edit"/> <input type="button" value="Delete"/>
admin	admin@gmail.com	2021-06-14	FUSCE FRINGILLA	23	69 \$	<input type="button" value="Edit"/> <input type="button" value="Delete"/>

228 AM 6/24/2021

Manager Flower Shop

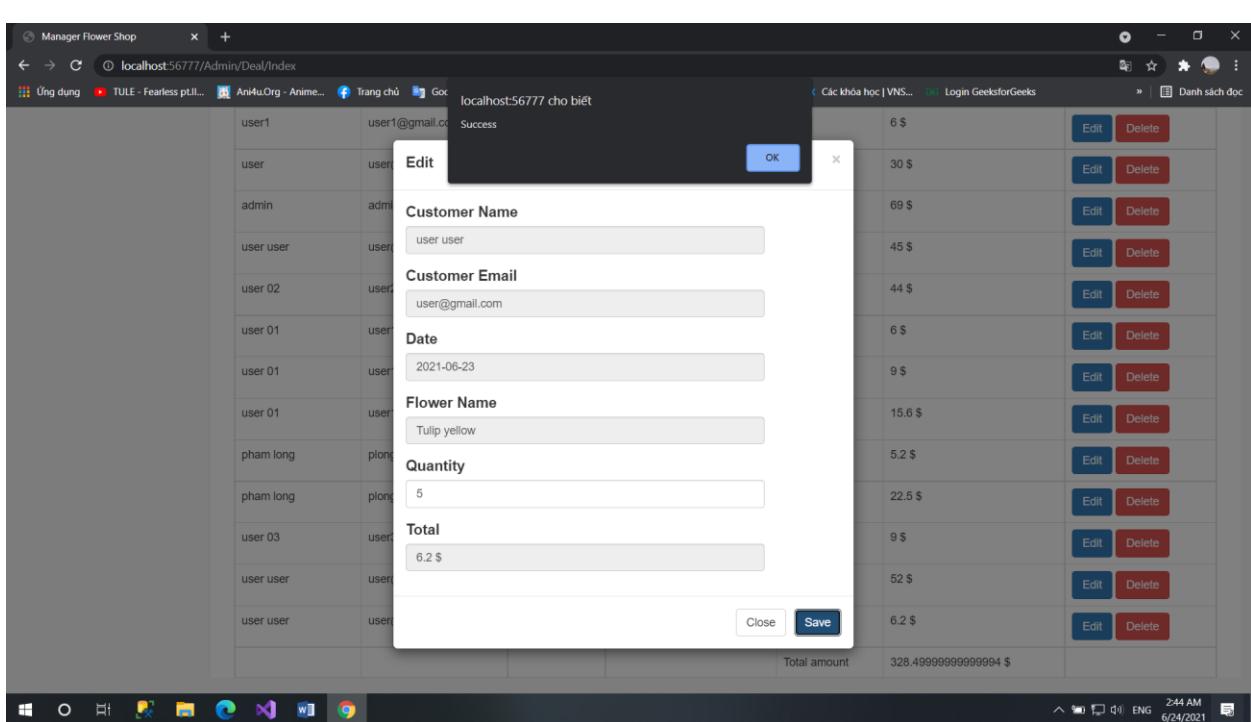
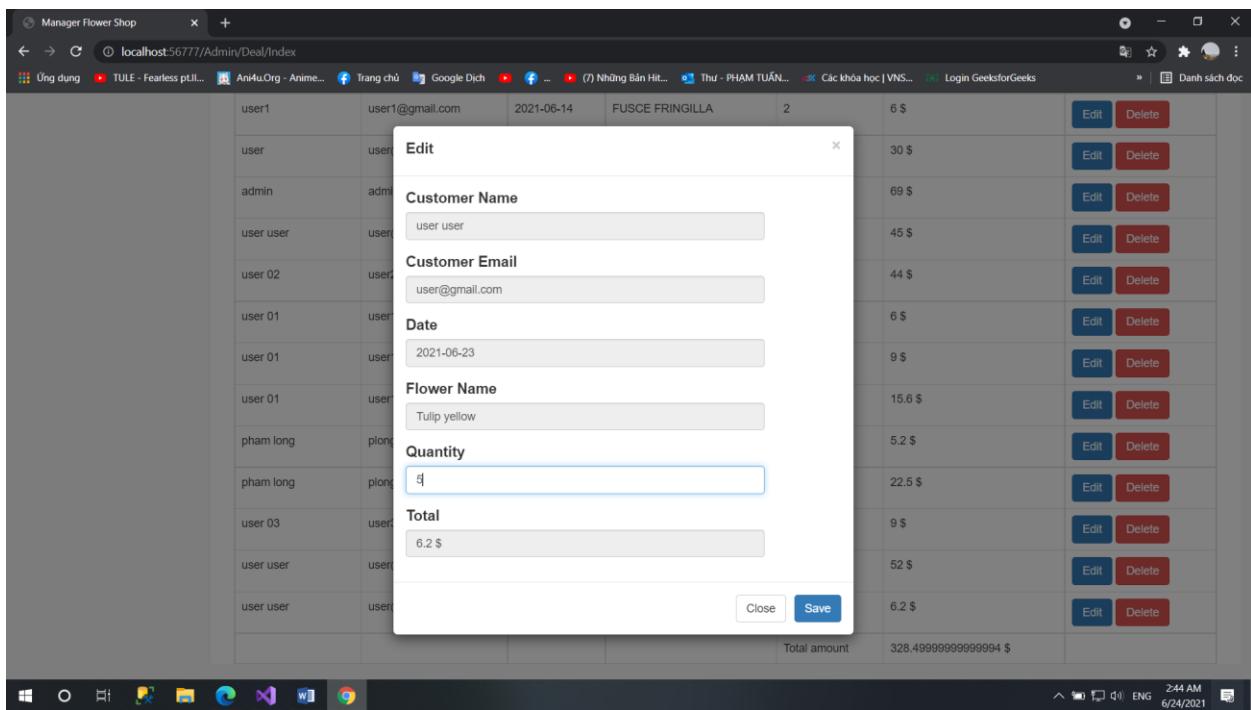
localhost:56777/Admin/Deal/Index

Manager Deals

user1	user1@gmail.com	2021-06-14	FUSCE FRINGILLA	2	6 \$	<input type="button" value="Edit"/> <input type="button" value="Delete"/>
user	user@gmail.com	2021-06-14	FUSCE FRINGILLA	10	30 \$	<input type="button" value="Edit"/> <input type="button" value="Delete"/>
admin	admin@gmail.com	2021-06-14	FUSCE FRINGILLA	23	69 \$	<input type="button" value="Edit"/> <input type="button" value="Delete"/>
user user	user@gmail.com	2021-06-14	Tulip	10	45 \$	<input type="button" value="Edit"/> <input type="button" value="Delete"/>
user 02	user2@gmail.com	2021-06-15	MAECENAS LUCTUS	22	44 \$	<input type="button" value="Edit"/> <input type="button" value="Delete"/>
user 01	user1@gmail.com	2021-06-15	FUSCE FRINGILLA	2	6 \$	<input type="button" value="Edit"/> <input type="button" value="Delete"/>
user 01	user1@gmail.com	2021-06-15	Tulip	2	9 \$	<input type="button" value="Edit"/> <input type="button" value="Delete"/>
user 01	user1@gmail.com	2021-06-15	PRAESENT MATTIS	3	15.6 \$	<input type="button" value="Edit"/> <input type="button" value="Delete"/>
pham long	plong@gmail.com	2021-06-15	PRAESENT MATTIS	1	5.2 \$	<input type="button" value="Edit"/> <input type="button" value="Delete"/>
pham long	plong@gmail.com	2021-06-15	Tulip	5	22.5 \$	<input type="button" value="Edit"/> <input type="button" value="Delete"/>
user 03	user3@gmail.com	2021-06-17	Tulip	2	9 \$	<input type="button" value="Edit"/> <input type="button" value="Delete"/>
user user	user@gmail.com	2021-06-21	Violet	1	52 \$	<input type="button" value="Edit"/> <input type="button" value="Delete"/>
user user	user@gmail.com	2021-06-23	Tulip yellow	2	6.2 \$	<input type="button" value="Edit"/> <input type="button" value="Delete"/>
				Total amount	328.49999999999994 \$	

228 AM 6/24/2021

Khi admin chỉnh sửa 1 đơn hàng chỉ chỉnh sửa được đúng số lượng đơn hàng còn mọi thông tin về khách hàng không thể chỉnh sửa được.



Manager Flower Shop

localhost:56777/Admin/Deal/Index

user1	user1@gmail.com	2021-06-14	FUSCE FRINGILLA	2	6 \$	<button>Edit</button>	<button>Delete</button>
user	user@gmail.com	2021-06-14	FUSCE FRINGILLA	10	30 \$	<button>Edit</button>	<button>Delete</button>
admin	admin@gmail.com	2021-06-14	FUSCE FRINGILLA	23	69 \$	<button>Edit</button>	<button>Delete</button>
user user	user@gmail.com	2021-06-14	Tulip	10	45 \$	<button>Edit</button>	<button>Delete</button>
user 02	user@gmail.com	2021-06-15	MAECENAS LUCTUS	22	44 \$	<button>Edit</button>	<button>Delete</button>
user 01	user1@gmail.com	2021-06-15	FUSCE FRINGILLA	2	6 \$	<button>Edit</button>	<button>Delete</button>
user 01	user1@gmail.com	2021-06-15	Tulip	2	9 \$	<button>Edit</button>	<button>Delete</button>
user 01	user1@gmail.com	2021-06-15	PRAESENT MATTIS	3	15.6 \$	<button>Edit</button>	<button>Delete</button>
pham long	plong@gmail.com	2021-06-15	PRAESENT MATTIS	1	5.2 \$	<button>Edit</button>	<button>Delete</button>
pham long	plong@gmail.com	2021-06-15	Tulip	5	22.5 \$	<button>Edit</button>	<button>Delete</button>
user 03	user3@gmail.com	2021-06-17	Tulip	2	9 \$	<button>Edit</button>	<button>Delete</button>
user user	user@gmail.com	2021-06-21	Violet	1	52 \$	<button>Edit</button>	<button>Delete</button>
user user	user@gmail.com	2021-06-23	Tulip yellow	5	15.5 \$	<button>Edit</button>	<button>Delete</button>
				Total amount	337.79999999999995 \$		

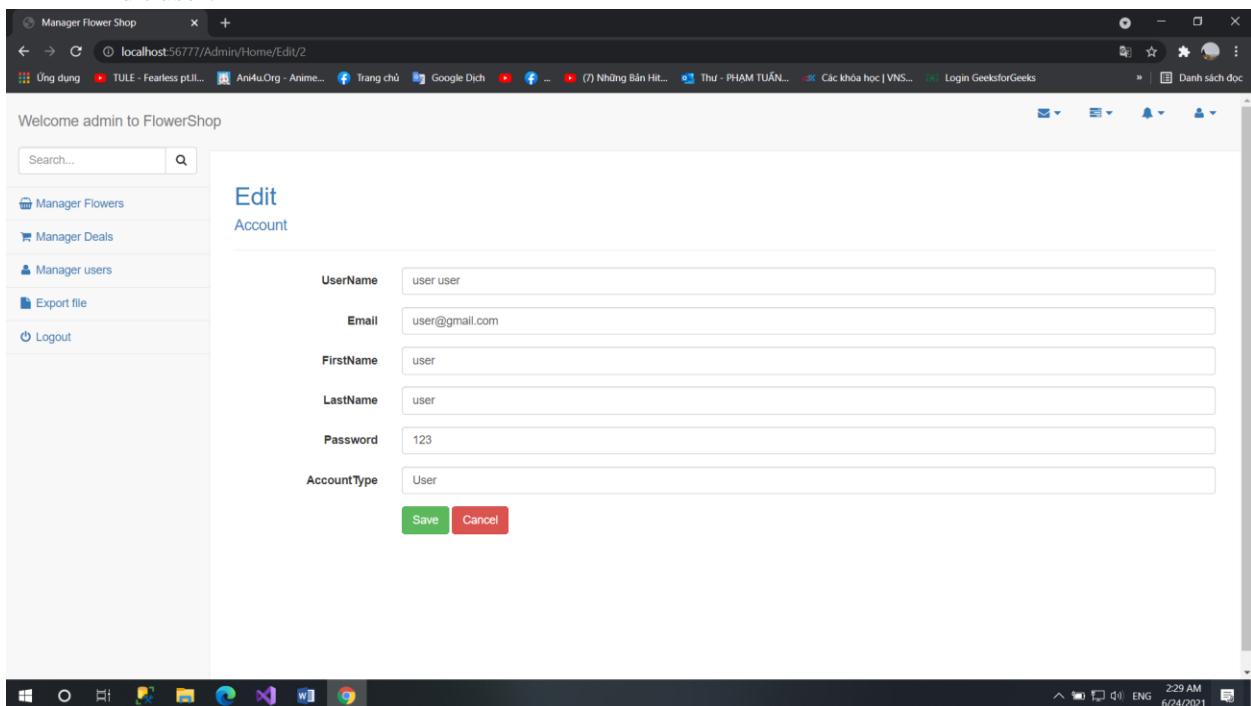
- Manager User:

Welcome admin to FlowerShop

localhost:56777/Admin/Home>ListAccount

ListAccount						
UserName	Password	Email	FirstName	LastName	AccountType	
admin	123	admin@gmail.com	Admin	Admin	Admin	<button>Edit</button> <button>Details</button> <button>Delete</button>
user user	123	user@gmail.com	user	user	User	<button>Edit</button> <button>Details</button> <button>Delete</button>
user 01	123	user1@gmail.com	user	01	User	<button>Edit</button> <button>Details</button> <button>Delete</button>
pham long	123	plong@gmail.com	pham	long	User	<button>Edit</button> <button>Details</button> <button>Delete</button>
user 02	12345	user2@gmail.com	user	02	User	<button>Edit</button> <button>Details</button> <button>Delete</button>
user 03	123456	user3@gmail.com	user	03	User	<button>Edit</button> <button>Details</button> <button>Delete</button>

- Edit user:



Welcome admin to FlowerShop

Search...

Manager Flowers

Manager Deals

Manager users

Export file

Logout

Edit

Account

UserName: user user

Email: user@gmail.com

FirstName: user

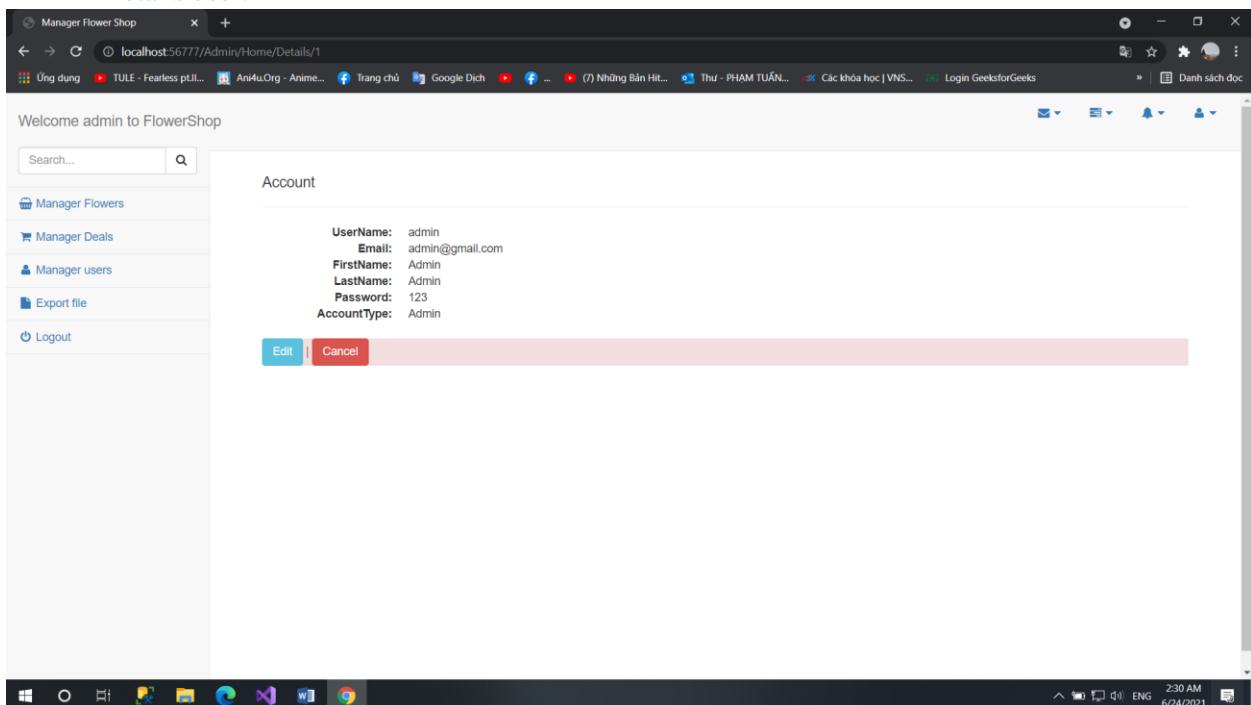
LastName: user

Password: 123

AccountType: User

Save Cancel

- Details user:



Welcome admin to FlowerShop

Search...

Manager Flowers

Manager Deals

Manager users

Export file

Logout

Account

UserName: admin

Email: admin@gmail.com

FirstName: Admin

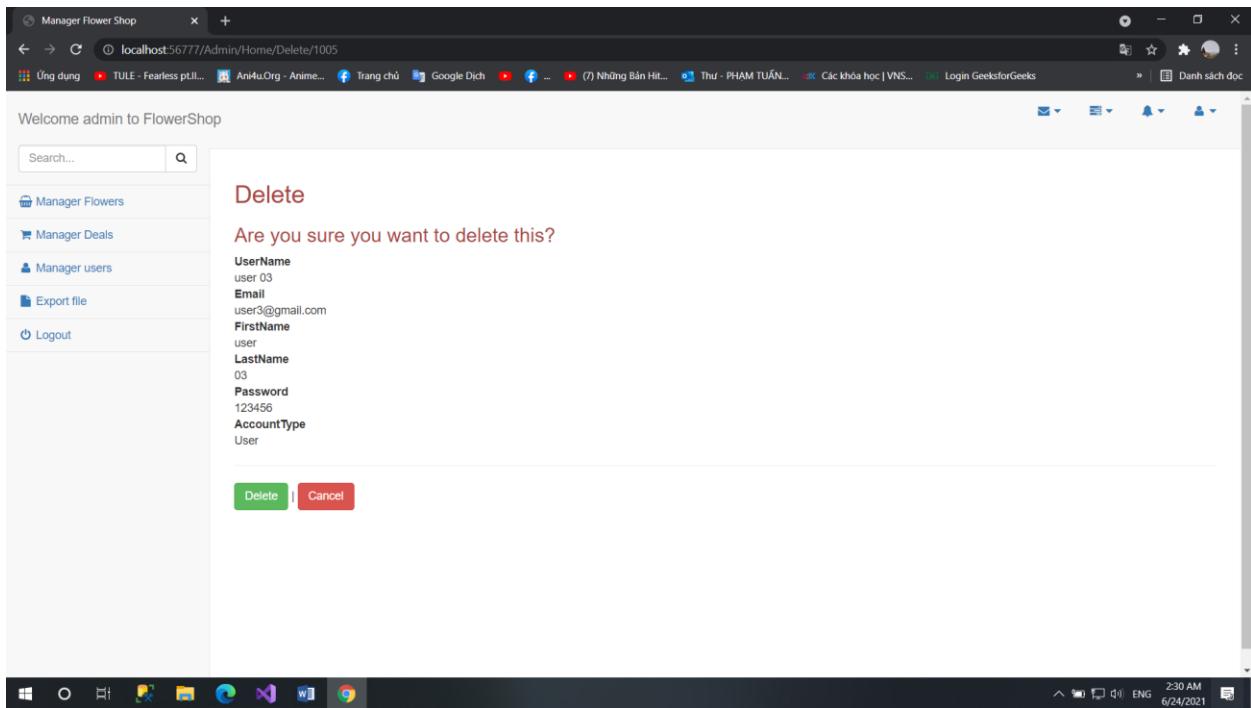
LastName: Admin

Password: 123

AccountType: Admin

Edit Cancel

- Delete user:



- Export file:

Có chức năng nén về file excel, pdf, word, image.

The screenshot shows a web browser window titled "Manager Flower Shop" at the URL "localhost:56777/Admin/Report/Index". The main content area displays a table titled "All deals" with the following columns: CustomerName, CustomerEmail, DealDate, FlowerName, Quantity, Total, and Price. There are four buttons above the table: "Export to Excel", "Export to PDF", "Export to Word", and "Export to Image". The table contains 18 rows of deal data. At the bottom of the page, there is a navigation bar with links like "Home", "Flowers", "Deals", "Users", "Logout", and "Help".

CustomerName	CustomerEmail	DealDate	FlowerName	Quantity	Total	Price
user	user@gmail.com	2021-06-13	FUSCE FRINGILLA	3	9.00	3.00
user1	user1@gmail.com	2021-06-14	FUSCE FRINGILLA	2	6.00	3.00
user	user@gmail.com	2021-06-14	FUSCE FRINGILLA	10	30.00	3.00
admin	admin@gmail.com	2021-06-14	FUSCE FRINGILLA	23	69.00	3.00
user user	user@gmail.com	2021-06-14	Tulip	10	45.00	4.50
user 02	user2@gmail.com	2021-06-15	MAECENAS LUCTUS	22	44.00	2.00
user 01	user1@gmail.com	2021-06-15	FUSCE FRINGILLA	2	6.00	3.00
user 01	user1@gmail.com	2021-06-15	Tulip	2	9.00	4.50
user 01	user1@gmail.com	2021-06-15	PRAESENT MATTIS	3	15.60	5.20
pham long	plong@gmail.com	2021-06-15	PRAESENT MATTIS	1	5.20	5.20
pham long	plong@gmail.com	2021-06-15	Tulip	5	22.50	4.50
user 03	user3@gmail.com	2021-06-17	Tulip	2	9.00	4.50
user user	user@gmail.com	2021-06-21	Violet	1	52.00	52.00
user user	user@gmail.com	2021-06-23	Tulip yellow	2	6.20	3.10

- Excel

Screenshot of a web application showing a report titled "All deals". The report includes a table with columns: CustomerName, CustomerEmail, DealDate, FlowerName, Quantity, Total, and Price. Below the table, a Microsoft Excel window displays the same data in a spreadsheet format.

Web Application Screenshot:

CustomerName	CustomerEmail	DealDate	FlowerName	Quantity	Total	Price
user	user@gmail.com	2021-06-13	FUSCE FRINGILLA	3	9.00	3.00
user1	user1@gmail.com	2021-06-14	FUSCE FRINGILLA	2	6.00	3.00
user	user@gmail.com	2021-06-14	FUSCE FRINGILLA	10	30.00	3.00
admin	admin@gmail.com	2021-06-14	FUSCE FRINGILLA	23	69.00	3.00
user user	user@gmail.com	2021-06-14	Tulip	10	45.00	4.50
user 02	user2@gmail.com	2021-06-15	MAECENAS LUCTUS	22	44.00	2.00
user 01	user1@gmail.com	2021-06-15	FUSCE FRINGILLA	2	6.00	3.00
user 01	user1@gmail.com	2021-06-15	Tulip	2	9.00	4.50
user 01	user1@gmail.com	2021-06-15	PRAESENT MATTIS	3	15.60	5.20
pham long	plong@gmail.com	2021-06-15	PRAESENT MATTIS	1	5.20	5.20
pham long	plong@gmail.com	2021-06-15	Tulip	5	22.50	4.50
user 03	user3@gmail.com	2021-06-17	Tulip	2	9.00	4.50
user user	user@gmail.com	2021-06-21	Violet	1	52.00	52.00
user user	user@gmail.com	2021-06-23	Tulip yellow	2	6.20	3.10

Microsoft Excel Screenshot:

Id	Customer Name	Customer Email	Deal Date	Flower Name	Quantity	Total	Price
1	user	user@gmail.com	2021-06-13	FUSCE FRINGILLA	3	9	3
2	user1	user1@gmail.com	2021-06-14	FUSCE FRINGILLA	2	6	3
3	user	user@gmail.com	2021-06-14	FUSCE FRINGILLA	10	30	3
4	admin	admin@gmail.com	2021-06-14	FUSCE FRINGILLA	23	69	3
5	user user	user@gmail.com	2021-06-14	Tulip	10	45	4.5
6	1002 user 02	user2@gmail.com	2021-06-15	MAECENAS LUCTUS	22	44	2
7	1003 user 01	user1@gmail.com	2021-06-15	FUSCE FRINGILLA	2	6	3
8	1004 user 01	user1@gmail.com	2021-06-15	Tulip	2	9	4.5
9	1005 user 01	user1@gmail.com	2021-06-15	PRAESENT MATTIS	3	15.6	5.2
10	1006 pham long	plong@gmail.com	2021-06-15	PRAESENT MATTIS	1	5.2	5.2
11	1007 pham long	plong@gmail.com	2021-06-15	Tulip	5	22.5	4.5
12	1012 user 03	user3@gmail.com	2021-06-17	Tulip	2	9	4.5
13	2010 user user	user@gmail.com	2021-06-21	Violet	1	52	52
14	2011 user user	user@gmail.com	2021-06-23	Tulip yellow	2	6.2	3.1

Total amount: 328.5

- Pdf

Report1

Deals information

ID	Customer Name	Customer Email	Deal Date	Flower Name	Quantity	Total	Price
1 user	user@gmail.com	2021-06-13	FUSCE FRINGILLA	3	9.00	3.00	
2 user1	user1@gmail.com	2021-06-14	FUSCE FRINGILLA	2	6.00	3.00	
3 user	user@gmail.com	2021-06-14	FUSCE FRINGILLA	10	30.00	3.00	
4 admin	admin@gmail.com	2021-06-14	FUSCE FRINGILLA	23	69.00	3.00	
5 user user	user@gmail.com	2021-06-14	Tulip	10	45.00	4.50	
1002 user 02	user2@gmail.com	2021-06-15	MAECENAS LUCTUS	22	44.00	2.00	
1003 user 01	user1@gmail.com	2021-06-15	FUSCE FRINGILLA	2	6.00	3.00	
1004 user 01	user1@gmail.com	2021-06-15	FUSCE FRINGILLA	2	9.00	4.50	
1005 user 01	user1@gmail.com	2021-06-15	PRAESENT MATTIS	3	15.60	5.20	
1006 pham long	plong@gmail.com	2021-06-15	PRAFATIS	1	5.20	5.20	
1007 pham long	plong@gmail.com	2021-06-15	Tulip	5	22.50	4.50	
1012 user 03	user3@gmail.com	2021-06-17	Tulip	2	9.00	4.50	
2910 user user	user@gmail.com	2021-06-21	Violet	1	52.00	52.00	
2911 user user	user@gmail.com	2021-06-23	Tulip yellow	2	6.20	3.10	
Total amount:							328.50

deal_report(1).TIF deal_report.doc deal_report.pdf deal_report.xls

Hiển thị tất cả

deal_report(1).TIF deal_report.doc deal_report.pdf deal_report.xls

232 AM ENG 6/24/2021

- Word

deal_report.doc [Compatibility Mode] - Word (Product Activation Failed)

Deals information

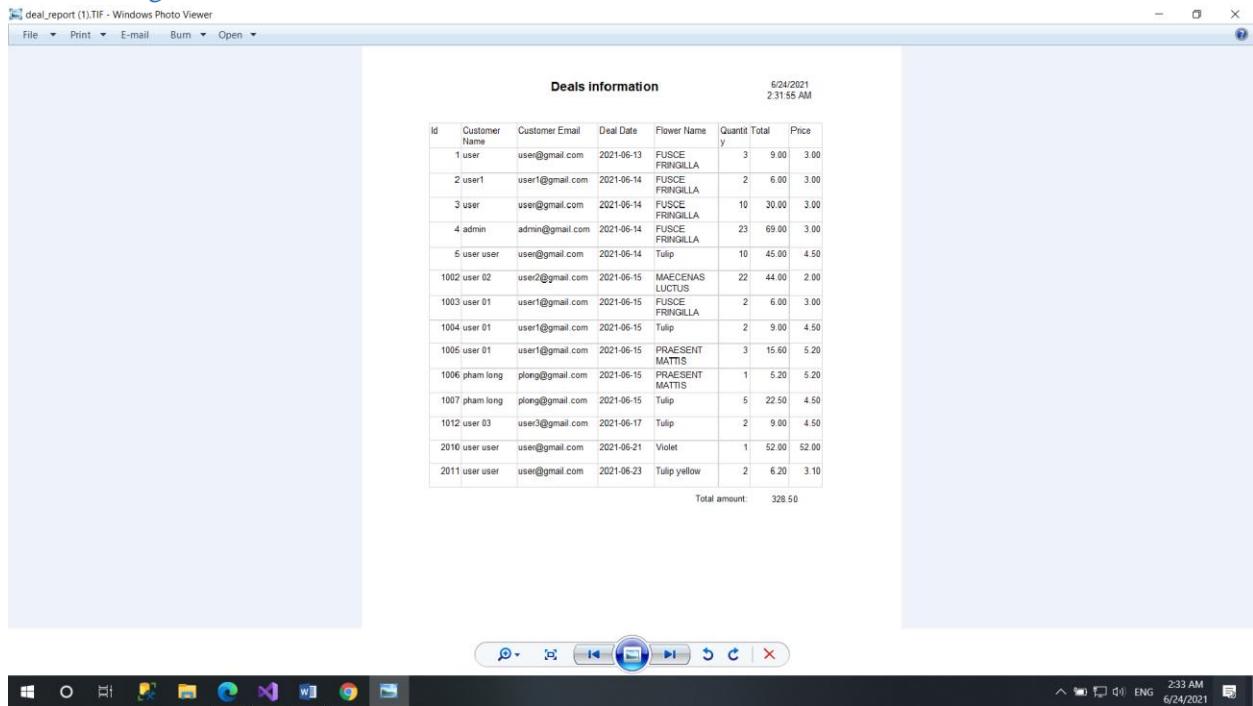
ID	Customer Name	Customer Email	Deal Date	Flower Name	Quantity	Total	Price
1 user	user@gmail.com	2021-06-13	FUSCE FRINGILLA	3	9.00	3.00	
2 user1	user1@gmail.com	2021-06-14	FUSCE FRINGILLA	2	6.00	3.00	
3 user	user@gmail.com	2021-06-14	FUSCE FRINGILLA	10	30.00	3.00	
4 admin	admin@gmail.com	2021-06-14	FUSCE FRINGILLA	23	69.00	3.00	
5 user user	user@gmail.com	2021-06-14	Tulip	10	45.00	4.50	
1002 user 02	user2@gmail.com	2021-06-15	MAECENAS LUCTUS	22	44.00	2.00	
1003 user 01	user1@gmail.com	2021-06-15	FUSCE FRINGILLA	2	6.00	3.00	
1004 user 01	user1@gmail.com	2021-06-15	Tulip	2	9.00	4.50	
1005 user 01	user1@gmail.com	2021-06-15	PRAESENT MATTIS	3	15.60	5.20	
1006 pham long	plong@gmail.com	2021-06-15	PRAFATIS	1	5.20	5.20	
1007 pham long	plong@gmail.com	2021-06-15	Tulip	5	22.50	4.50	
1012 user 03	user3@gmail.com	2021-06-17	Tulip	2	9.00	4.50	
2910 user user	user@gmail.com	2021-06-21	Violet	1	52.00	52.00	
2911 user user	user@gmail.com	2021-06-23	Tulip yellow	2	6.20	3.10	
Total amount:							328.50

deal_report.doc: 929 characters (an approximate value).

deal_report.doc deal_report.pdf deal_report.xls

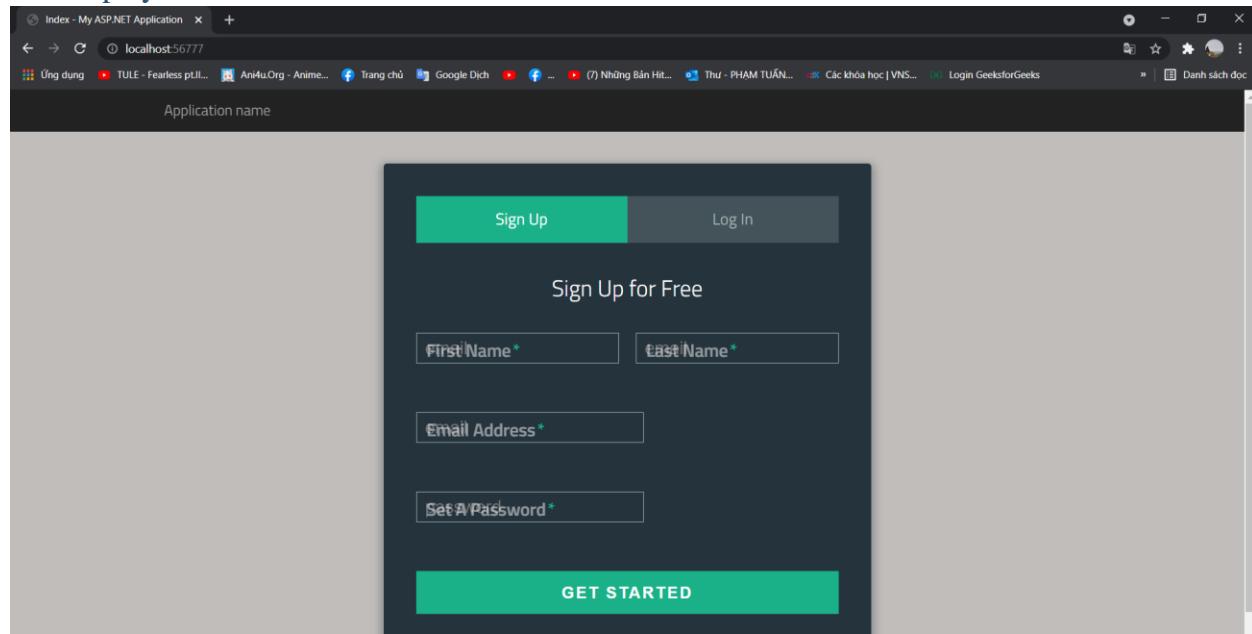
233 AM ENG 6/24/2021

- Image:

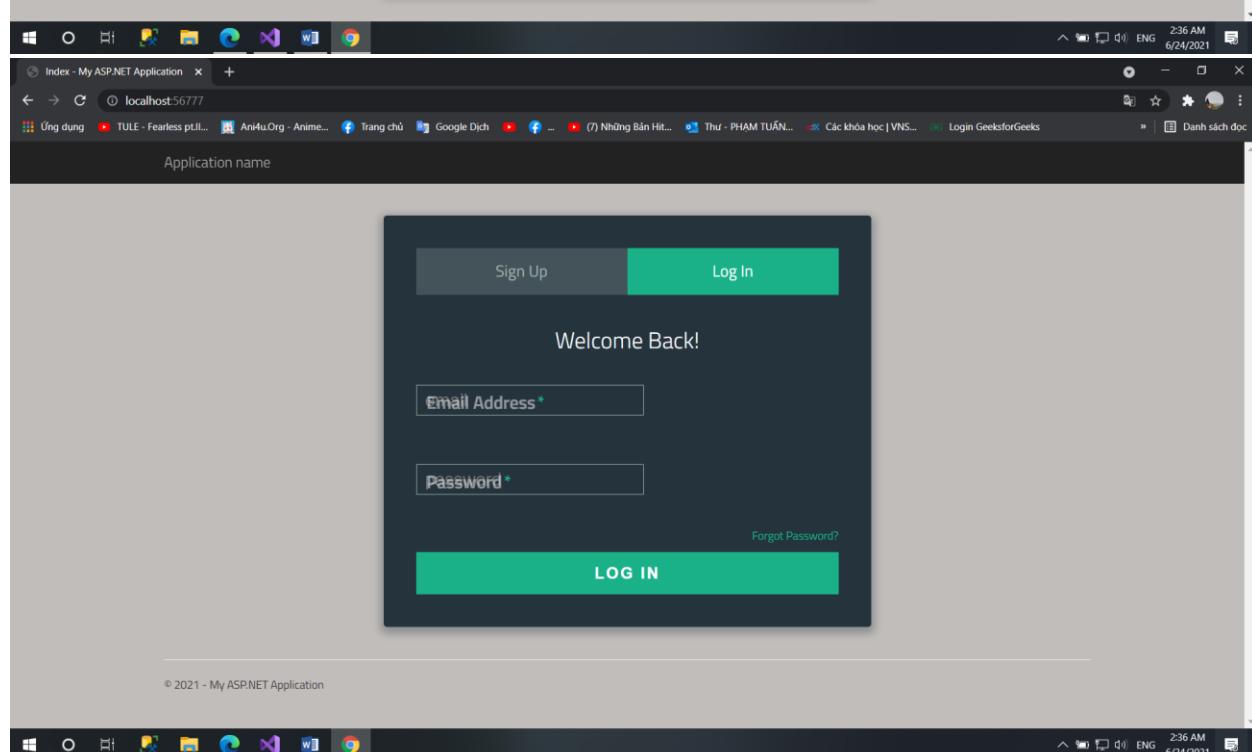


b. User

- Display

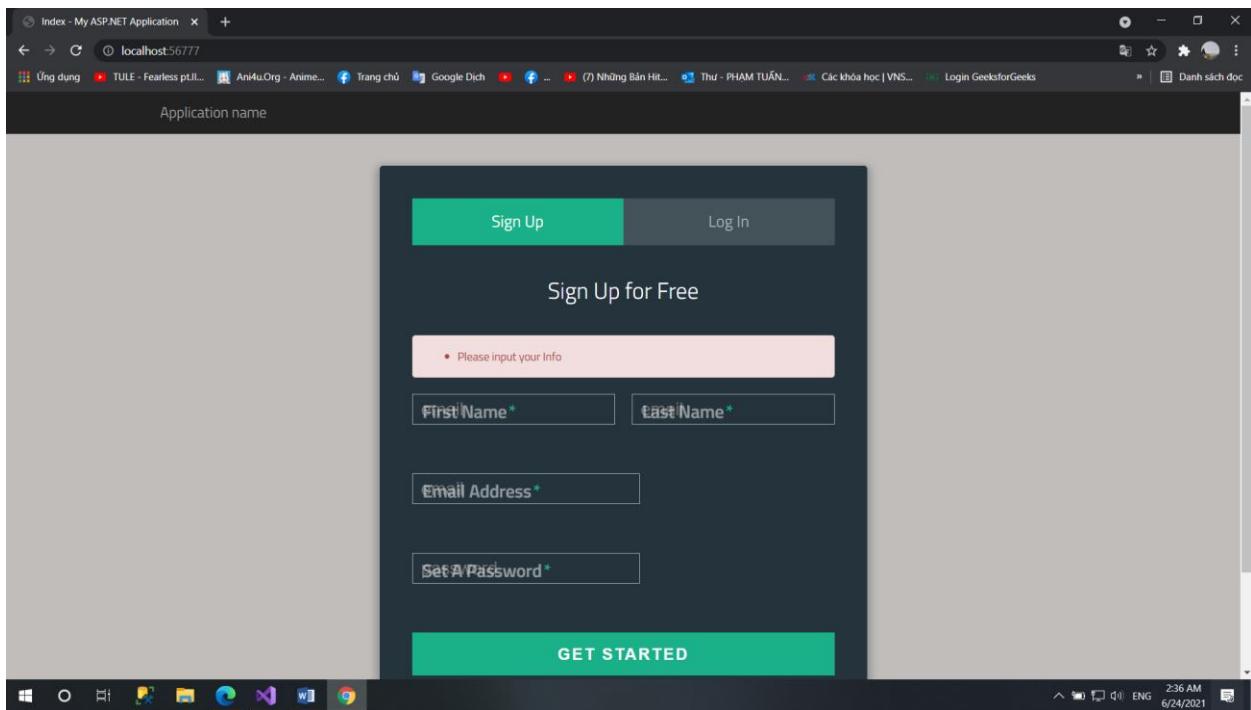


The screenshot shows a web browser window with the URL `localhost:56777`. The page displays a sign-up form titled "Sign Up for Free". It includes fields for "First Name*", "Last Name*", "Email Address*", and "Set A Password*". There are "Sign Up" and "Log In" buttons at the top, and a "GET STARTED" button at the bottom.

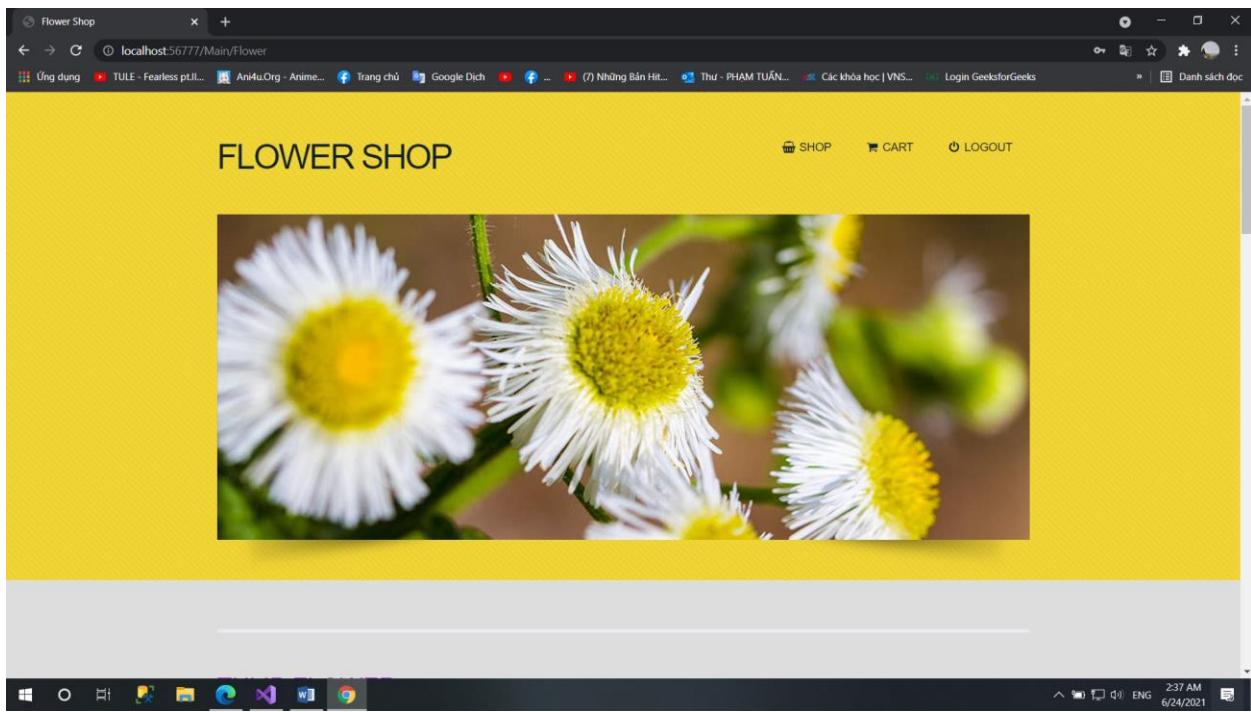


The screenshot shows a web browser window with the URL `localhost:56777`. The page displays a login form titled "Welcome Back!". It includes fields for "Email Address*" and "Password*". There are "Sign Up" and "Log In" buttons at the top, and a "Forgot Password?" link below the password field. The "LOG IN" button is highlighted in green.

Khi người dùng nhập sai



Khi người dùng đăng nhập vào:



[Flower Shop](#) [localhost:56777/Main/Flower](#)

Ứng dụng TULE - Fearless pt.II... Anh4u.Org - Anime... Trang chủ Google Dịch ... () Những Bản Hit... Thư - PHẠM TUẤN... Các khóa học | VNS... Login GeeksforGeeks Danh sách đọc

TULIP FLOWER

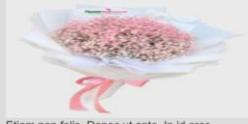
TULIP  <p>Etim non felis. Donec ut ante. In id eros. Suspendisse lacus turpis, cursus egestas at sem. Mauris quam enim, molestie. Price: \$4.5</p> <p>Read More Add to cart</p>	TULIP YELLOW  <p>Etim non felis. Donec ut ante. In id eros. Suspendisse lacus turpis, cursus egestas at sem. Mauris quam enim, molestie. Price: \$3.1</p> <p>Read More Add to cart</p>	TULIP VIOLET  <p>Etim non felis. Donec ut ante. In id eros. Suspendisse lacus turpis, cursus egestas at sem. Mauris quam enim, molestie. Price: \$8.2</p> <p>Read More Add to cart</p>
--	---	--

BIRTHDAY FLOWER

[ROSE BIRTHDAY](#) [VIOLET](#)

Flower Shop [localhost:56777/Main/Flower](#)

Ứng dụng TULE - Fearless pt.II... Anh4u.Org - Anime... Trang chủ Google Dịch ... () Những Bản Hit... Thư - PHẠM TUẤN... Các khóa học | VNS... Login GeeksforGeeks Danh sách đọc

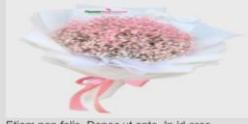
ROSE BIRTHDAY  <p>Etim non felis. Donec ut ante. In id eros. Suspendisse lacus turpis, cursus egestas at sem. Mauris quam enim, molestie. Price: \$4.5</p> <p>Read More Add to cart</p>	VIOLET  <p>Etim non felis. Donec ut ante. In id eros. Suspendisse lacus turpis, cursus egestas at sem. Mauris quam enim, molestie. Price: \$3.1</p> <p>Read More Add to cart</p>	VIOLET  <p>Etim non felis. Donec ut ante. In id eros. Suspendisse lacus turpis, cursus egestas at sem. Mauris quam enim, molestie. Price: \$8.2</p> <p>Read More Add to cart</p>
--	---	--

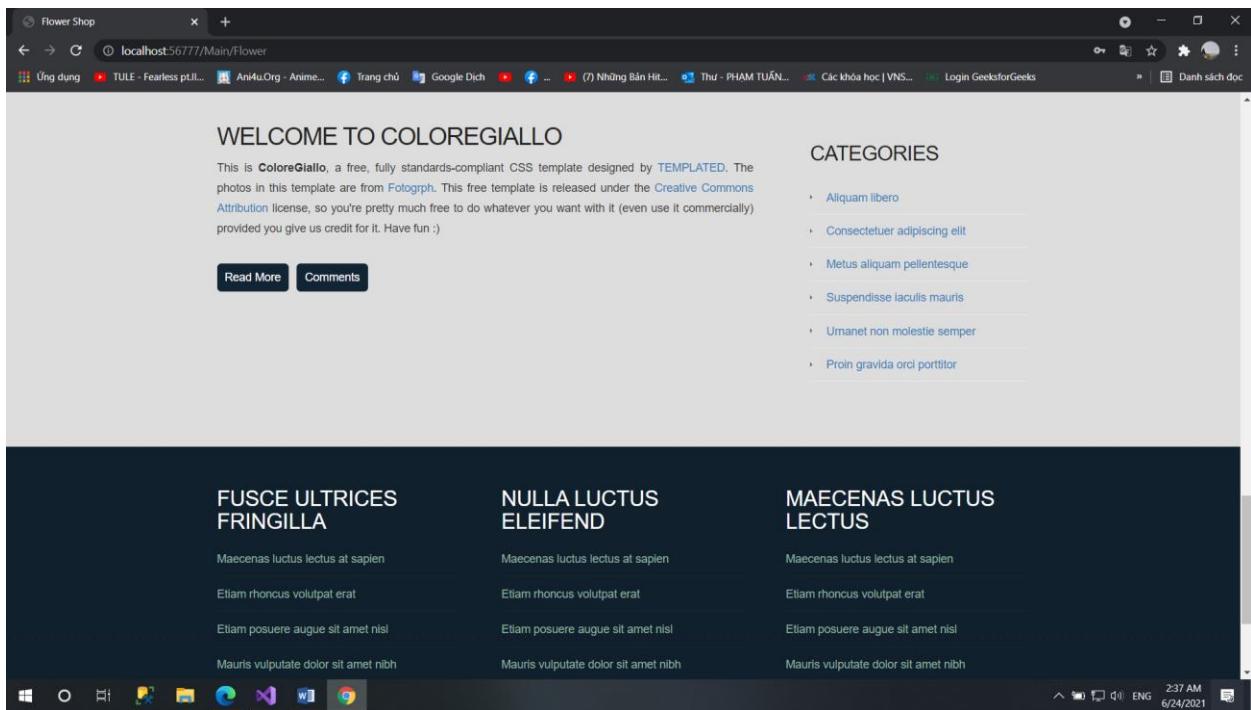
BIRTHDAY FLOWER

[RED ROSE BIRTHDAY](#) [VIOLET](#) [ROSE FLOWER](#)

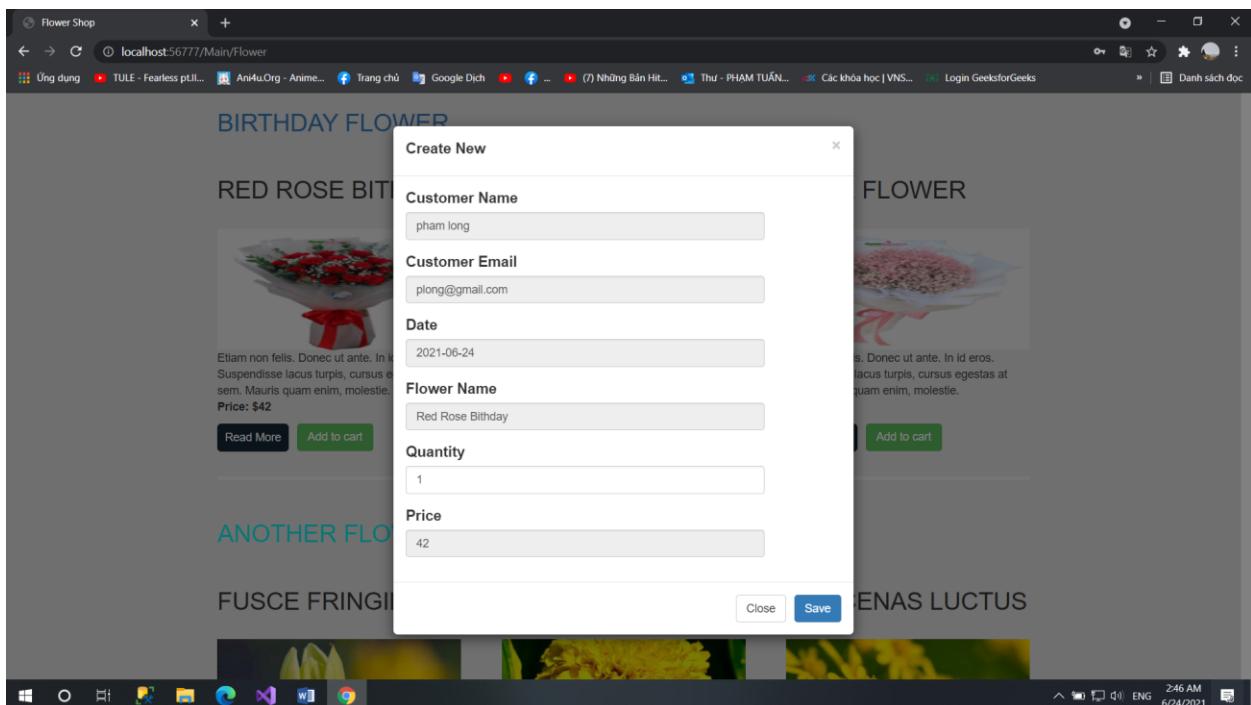
Flower Shop [localhost:56777/Main/Flower](#)

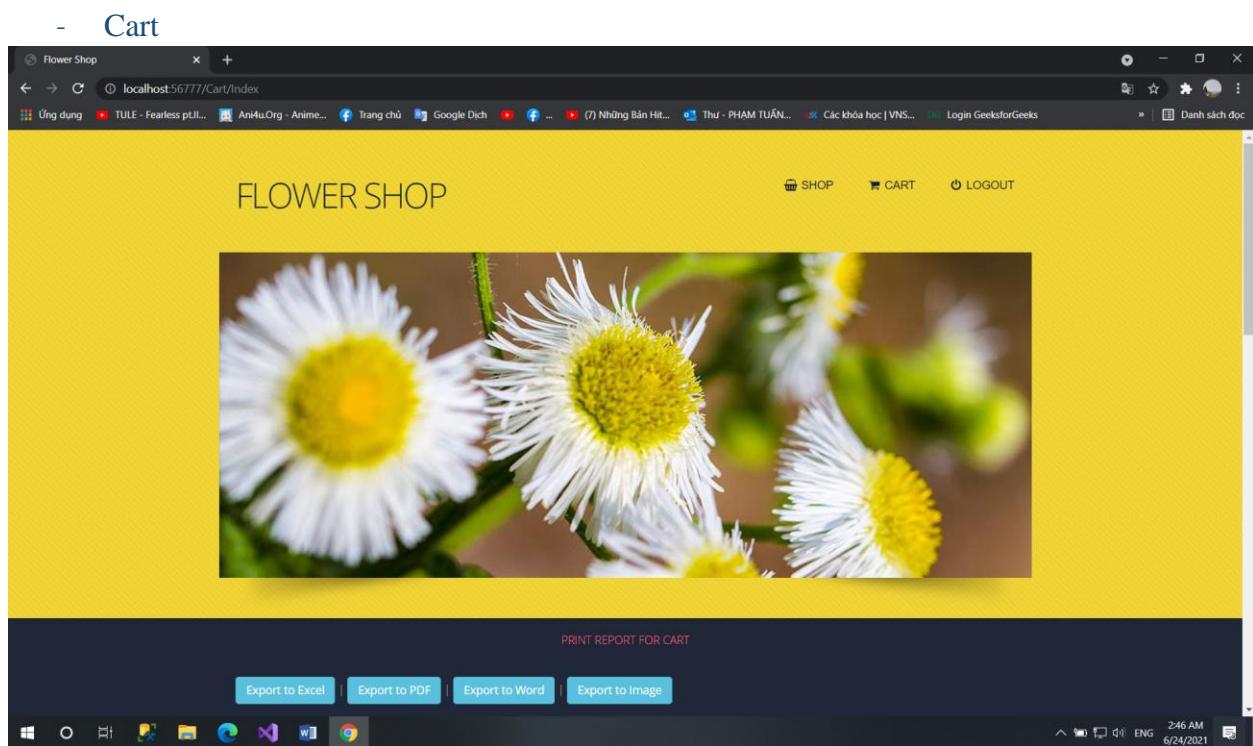
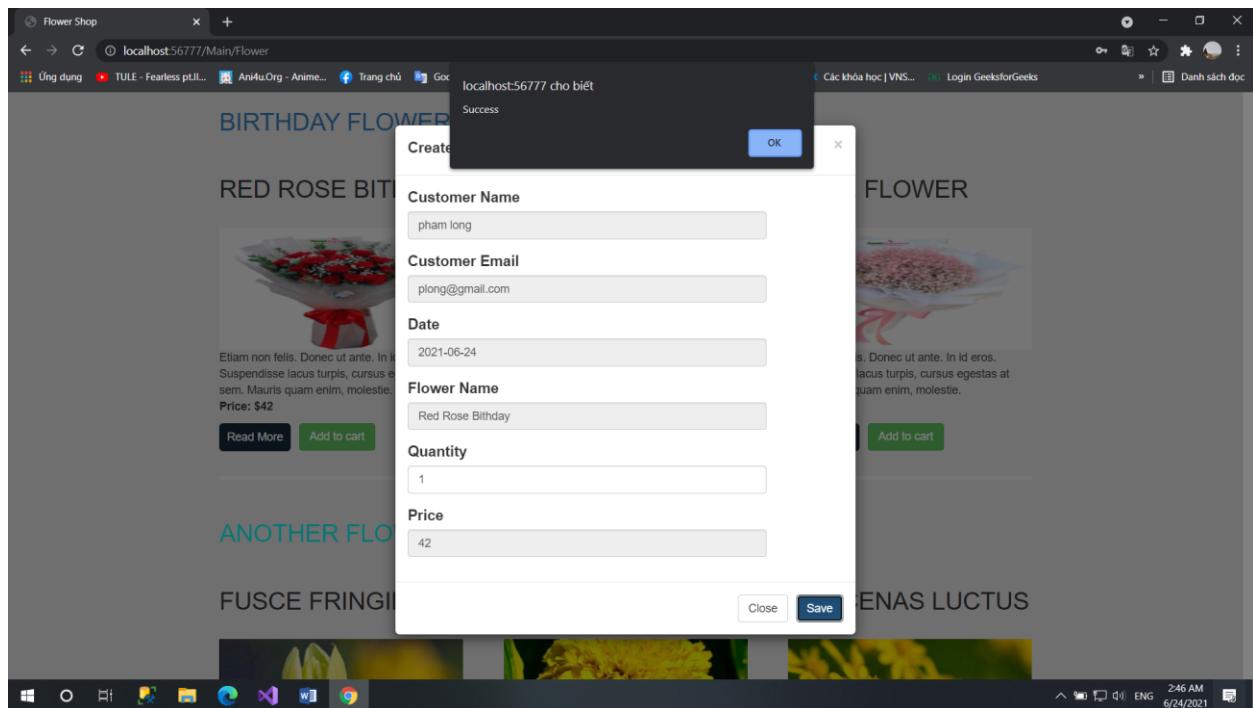
Ứng dụng TULE - Fearless pt.II... Anh4u.Org - Anime... Trang chủ Google Dịch ... () Những Bản Hit... Thư - PHẠM TUẤN... Các khóa học | VNS... Login GeeksforGeeks Danh sách đọc

RED ROSE BIRTHDAY  <p>Etim non felis. Donec ut ante. In id eros. Suspendisse lacus turpis, cursus egestas at sem. Mauris quam enim, molestie. Price: \$4.2</p> <p>Read More Add to cart</p>	VIOLET  <p>Etim non felis. Donec ut ante. In id eros. Suspendisse lacus turpis, cursus egestas at sem. Mauris quam enim, molestie. Price: \$5.2</p> <p>Read More Add to cart</p>	ROSE FLOWER  <p>Etim non felis. Donec ut ante. In id eros. Suspendisse lacus turpis, cursus egestas at sem. Mauris quam enim, molestie. Price: \$5.0</p> <p>Read More Add to cart</p>
--	---	---



Khi người dùng thêm 1 đơn hàng nào đó người dùng chỉ cần nhập đúng số lượng cần mua còn về thông tin khách hàng đã được điền sẵn mọi thứ.





The screenshot shows a web browser window titled "Flower Shop" with the URL "localhost:56777/Cart/Index". The page is titled "PRINT REPORT FOR CART" and features four export buttons: "Export to Excel", "Export to PDF", "Export to Word", and "Export to Image". Below this is a section titled "<CART> INFOMATION" containing a table of flower purchases. The table has columns: CustomerName, CustomerEmail, Date, FlowerName, Quantity, Total, and Manipulation (with "Edit" and "Delete" buttons). The data in the table is as follows:

CustomerName	CustomerEmail	Date	FlowerName	Quantity	Total	Manipulation
pham long	plong@gmail.com	2021-06-15	PRAESENT MATTIS	1	5.2 \$	<button>Edit</button> <button>Delete</button>
pham long	plong@gmail.com	2021-06-15	Tulip	5	22.5 \$	<button>Edit</button> <button>Delete</button>
pham long	plong@gmail.com	2021-06-24	Red Rose Birthday	1	42 \$	<button>Edit</button> <button>Delete</button>

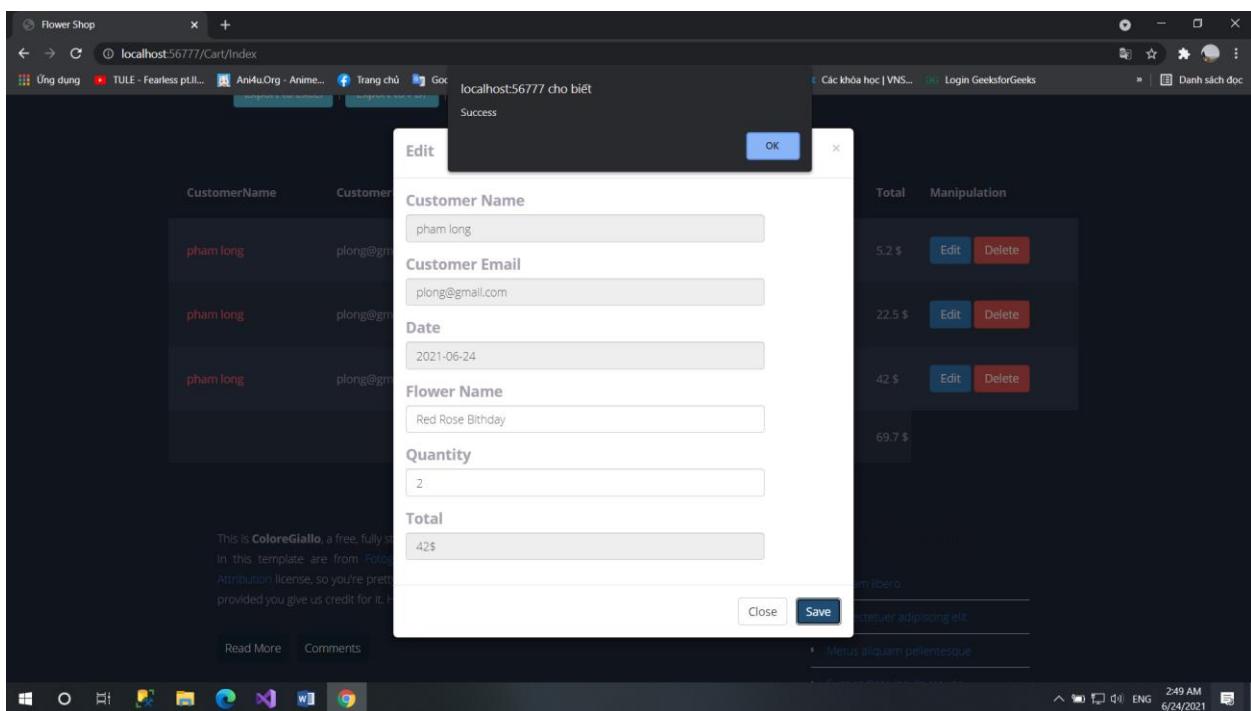
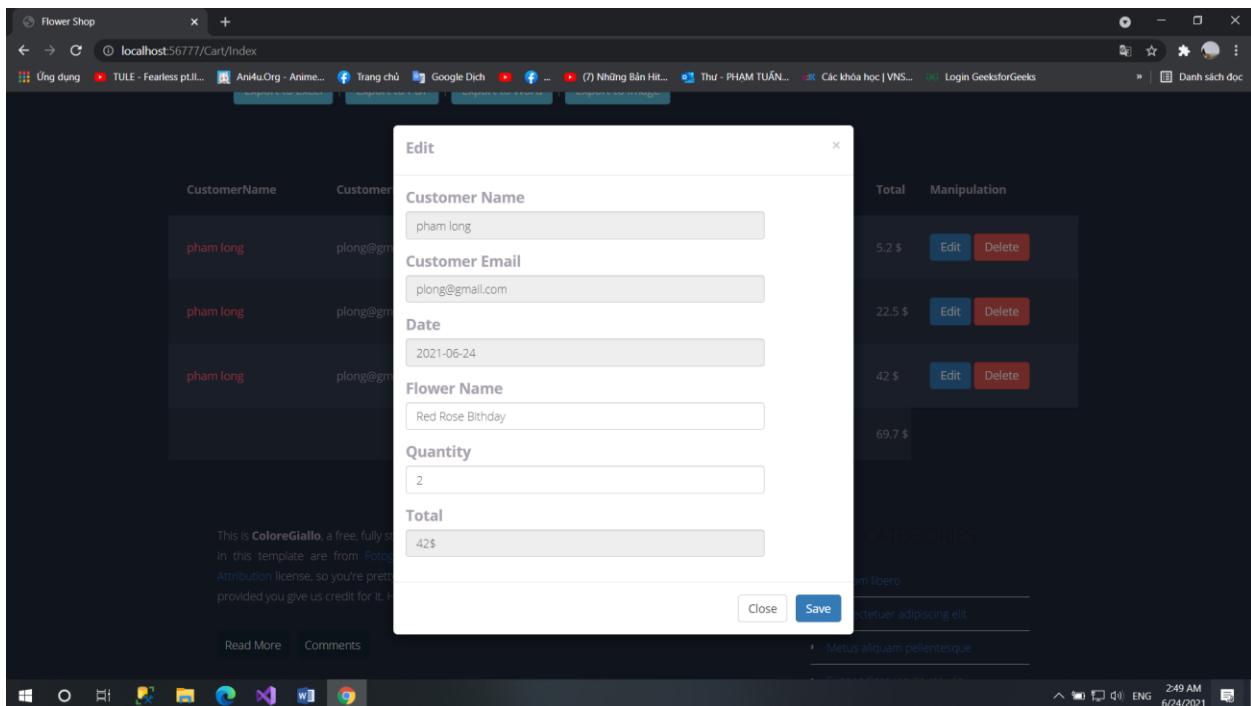
Total amount: 69.7 \$

Below the table, there is a "WELCOME TO COLOREGALLO" message and a "CATEGORIES" sidebar with a single item: "Aliquam libero".

This screenshot is identical to the one above, showing the same "Flower Shop" application interface. It includes the "PRINT REPORT FOR CART" section, the "<CART> INFOMATION" table with the same data, and the "WELCOME TO COLOREGALLO" message. The "CATEGORIES" sidebar also contains the single item "Aliquam libero".

Người dùng chỉ có thể chỉnh sửa và xóa những đơn hàng của chính người dùng và còn có thể xuất hóa đơn khi cần thiết.

Khi người dùng chỉnh sửa chỉ có thể chỉnh sửa tên hoa và số lượng mua hoa.



Khi người dùng xóa 1 đơn hàng:

Flower Shop localhost:56777/Cart/Index

localhost:56777 cho biết
Do you want to delete deal?

CustomerName	CustomerEmail	Date	FlowerName	Quantity	Total	Manipulation	
pham long	plong@gmail.com	2021-06-15	PRAESENT MATTIS	1	5.2 \$	<button>Edit</button> <button>Delete</button>	
pham long	plong@gmail.com	2021-06-15	Tulip	5	22.5 \$	<button>Edit</button> <button>Delete</button>	
pham long	plong@gmail.com	2021-06-24	Red Rose Birthday	2	84 \$	<button>Edit</button> <button>Delete</button>	
						Total amount	111.7 \$

WELCOME TO COLORGIALLO.COM
This is **ColorGiallo**, a free, fully standards-compliant CSS template designed by [TEMPLATED](#). The photos in this template are from [Fotograph](#). This free template is released under the [Creative Commons Attribution](#) license, so you're pretty much free to do whatever you want with it (even use it commercially) provided you give us credit for it. Have fun :)

[Read More](#) [Comments](#)

Windows Taskbar: File Explorer, File Manager, FileZilla, Visual Studio, Word, Google Chrome

Flower Shop localhost:56777/Cart/Index

localhost:56777 cho biết
Do you want to delete deal?

[Export to Excel](#) [Export to PDF](#) [Export to Word](#) [Export to Image](#)

<CART> INFOMATION

CustomerName	CustomerEmail	Date	FlowerName	Quantity	Total	Manipulation	
pham long	plong@gmail.com	2021-06-15	Tulip	5	22.5 \$	<button>Edit</button> <button>Delete</button>	
pham long	plong@gmail.com	2021-06-24	Red Rose Birthday	2	84 \$	<button>Edit</button> <button>Delete</button>	
						Total amount	106.5 \$

WELCOME TO COLORGIALLO.COM
This is **ColorGiallo**, a free, fully standards-compliant CSS template designed by [TEMPLATED](#). The photos in this template are from [Fotograph](#). This free template is released under the [Creative Commons Attribution](#) license, so you're pretty much free to do whatever you want with it (even use it commercially) provided you give us credit for it. Have fun :)

[Read More](#) [Comments](#)

Windows Taskbar: File Explorer, File Manager, FileZilla, Visual Studio, Word, Google Chrome

Người dùng xuất hóa đơn:

Word:

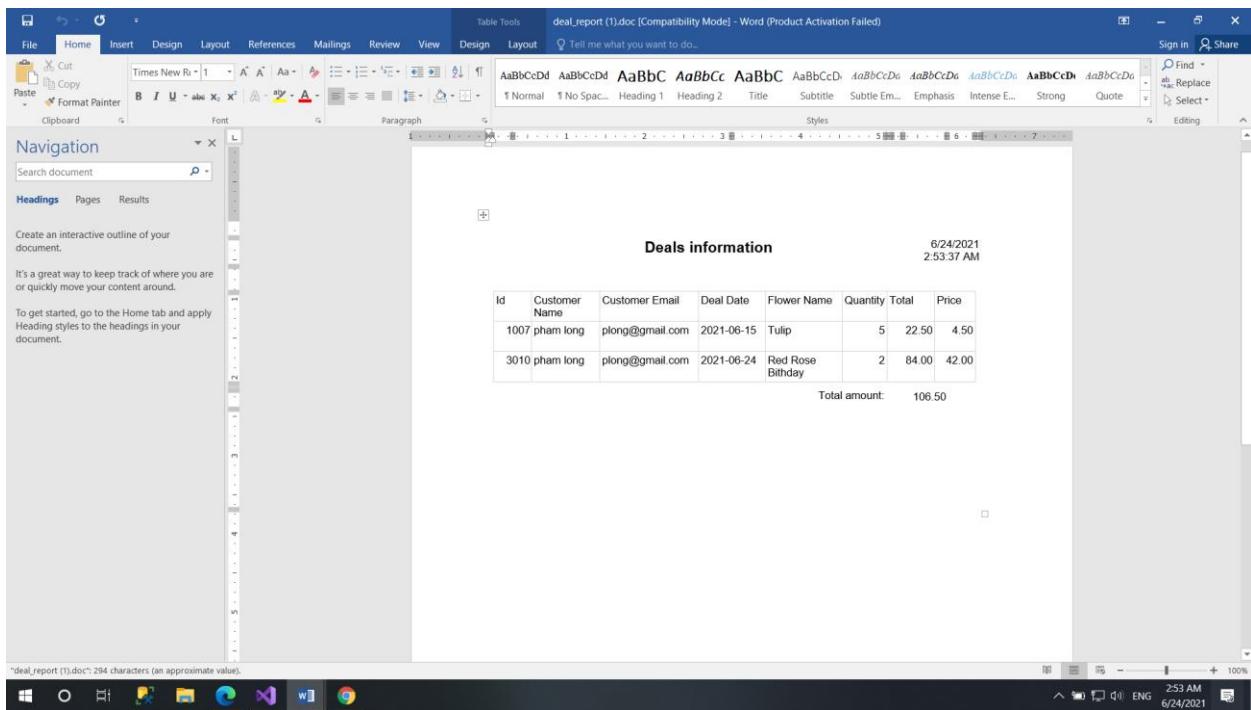
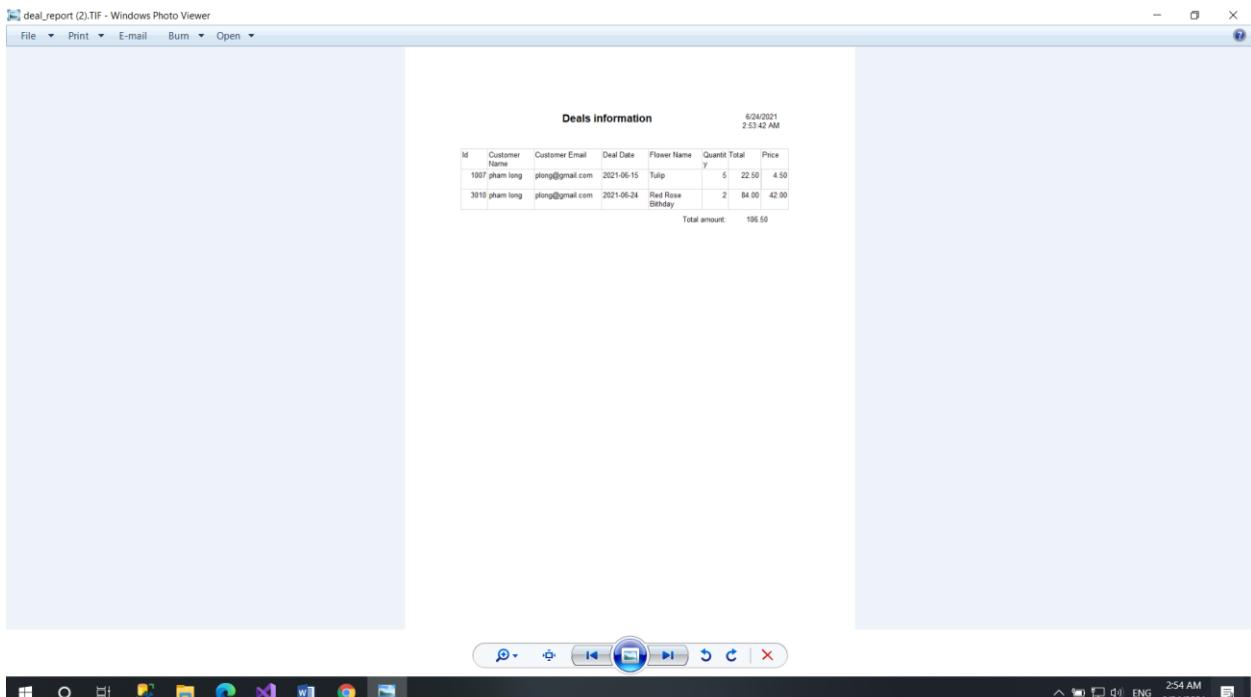
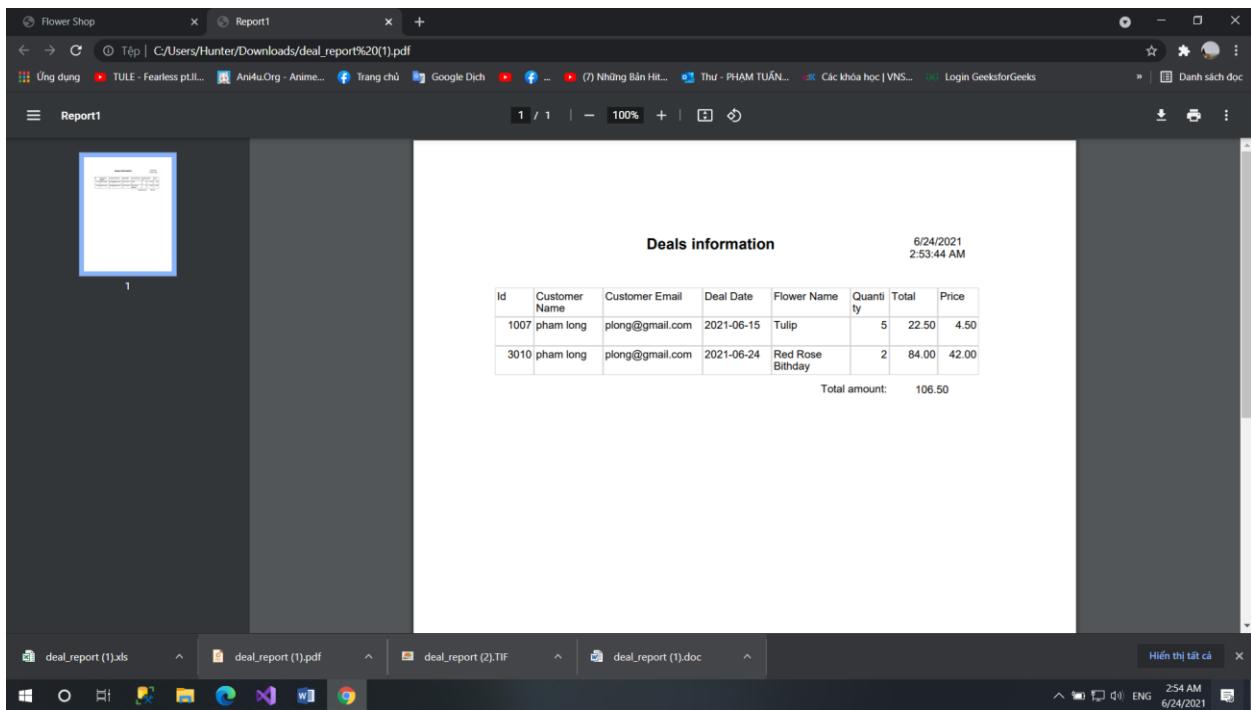


Image:



Pdf:



Excel:

The screenshot shows a Microsoft Excel spreadsheet titled "deal_report (1).xls". The ribbon menu is visible with tabs like File, Home, Insert, Page Layout, Formulas, Data, Review, View, and Tell me what you want to do... The main content area displays the "Deals information" data in a table format:

ID	Customer Name	Customer Email	Deal Date	Flower Name	Quantity	Total	Price
1007	pham long	plong@gmail.com	2021-06-15	Tulip	5	22.5	4.5
3010	pham long	plong@gmail.com	2021-06-24	Red Rose Birthday	2	84	42

Below the table, it says "Total amount: 106.5". The Excel interface includes a ribbon, a formula bar, and a status bar at the bottom indicating "Ready", "2:54 AM", and "6/24/2021".