

## Lập trình web với ASP.net Lab 05

Code:

Index.cshtml

```
FlowersDB.dml  FlowersController.cs  Index.cshtml  FlowersLayout.cshtml
1  ViewBag.Title = "Index";
2  Layout = "~/Views/Shared/FlowersLayout.cshtml";
3
4
5
6  <h2>Index</h2>
7
8  <section TAG>
9  <div id="featured-content">
10
11  </div>
12
13  <section MODEL DIALOG>
14  <div class="modal fade" id="CreateEditModal" role="dialog">
15    <div class="modal-dialog btn-lg">
16      <div class="modal-content">
17        <div class="modal-header">
18          <button type="button" class="close" data-dismiss="modal">&times;</button>
19          <h4 class="modal-title"><label id="title">No Title</label></h4>
20        </div>
21        <div class="modal-body">
22          <div class="form-horizontal">
23            <div class="form-group">
24              <div class="col-md-10">
25                <input type="hidden" class="form-control" id="Id" />
26                <label>Tag Name</label>
27                <input type="text" id="TagName" class="form-control" />
28              </div>
29            </div>
30            <div class="form-group">
31              <div class="col-md-10">
32                <button type="button" class="btn btn-default" data-dismiss="modal">Close</button>
33                <button type="button" class="btn btn-primary" data-dismiss="modal">Save</button>
34              </div>
35            </div>
36            <div class="form-group">
37              <div class="col-md-10">
38                <input type="text" id="TagContent" class="form-control" />
39              </div>
40            </div>
41            <div class="form-group">
42              <div class="col-md-10">
43                <input type="text" id="ImagePath" class="form-control" />
44              </div>
45            </div>
46          </div>
47        </div>
48      </div>
49    </div>
50  </div>
51
52  <script type="text/javascript">
53
54  $(document).ready(function () {
55    showFlowersTag();
56  });
57
58  $("#btnCreateNew").click(function (e) {
59    isUpdatable = false;
60    currentUpdatePath = "";
61    currentId = 0;
62    $(document).ready(function () {
63      showFlowersTag();
64    });
65  });
66
67  $("#btnClose").click(function (e) {
68    clearModalContent();
69  });
70
71  $("#btnSave").click(function (e) {
72    var data = {
73      Id: $("#Id").val(),
74      TagName: $.trim($("#TagName").val()),
75      ImagePath: $("#ImagePath").val(),
76      TagContent: $.trim($("#TagContent").val()),
77    };
78
79    if (!isUpdatable) {
80      if (data.ImagePath || !data.TagName) {
81        alert("You should input Tag Name and choose image flowers");
82        return;
83      }
84    } else {
85      if (data.ImagePath) {
86        data.ImagePath = currentUpdatePath;
87        data.Id = currentId;
88      }
89    }
90
91    alert(data.ImagePath);
92    $("#uploadForm").submit();
93  });
94
95  if (!isUpdatable) {
96    $.ajax({
97      url: "/Flowers/Create",
98      type: "POST",
99      dataType: "json",
100      data: data,
101      success: function (data) {
102        showFlowersTag();
103        clearModalContent();
104      },
105      error: function (err) {
106        alert("Error: " + err.responseText);
107      }
108    });
109  } else {
110    $.ajax({
111      url: "/Flowers/Update",
112      type: "POST",
113      dataType: "json",
114      data: data,
115    });
116  }
117
118  </script>
119
```

```

119         success: function (data) {
120             showFlowersTag();
121             clearModalContent();
122         },
123         error: function (err) {
124             alert("Error: " + err.responseText);
125         }
126     });
127
128
129
130     function clearModalContent() {
131         $("#tagName").val("");
132         $("#ImagePath").val("");
133         $("#tagContent").val("");
134     };
135
136
137     function showFlowersTag() {
138         $("#featured-content").empty();
139
140         $.ajax({
141             url: '/Flowers/ListTagFlowers',
142             type: 'GET',
143             dataType: 'json',
144             success: function (data) {
145                 var count = 0;
146                 $.each(data, function (i, item) {
147                     var id = item.Id;
148                     var tagName = item.TagName;
149                     var tagImage = item.ImagePath;
150                     var tagContent = item.TagContent;
151                     if (count == 3) { // limit of column
152                         count = 1;
153                     }
154                     else {
155                         count++;
156                     }
157                     AppendNewTag(id, count, tagName, tagImage, tagContent);
158                 });
159             },
160             error: function (err) {
161                 alert("Error: " + err.responseText);
162             }
163         });
164
165         function AppendNewTag(id, count, tagName, tagImage, tagContent) {
166             rows = '';
167             rows += '<div id="column" + count + ">';
168             rows += '<h2>' + tagName + '</h2>';
169             rows += '<div class="image"></div>';
170             rows += '<div class="tagcontent">' + tagContent + '</div>';
171             rows += '<div class="groupbutton">';
172             rows += '<a class="button">a href="#">Read More</a></p>';
173             rows += '<button type="button" id="btnEdit" class="btn btn-primary" + "';
174                 "data-toggle="modal" data-target="#CreateEditModal" + "';
175                 "onclick="return getDetailTag(" + id + ")">Edit</button>&nbsp;&nbsp;&";
176             rows += '<button type="button" id="btnDelete" class="btn btn-danger" + "';
177                 "onclick="return deleteTag(" + id + ")">Delete</button>";
178
179             rows += '</div>';
180             rows += '</div>';
181             $("#featured-content").append(rows);
182         };
183
184     function deleteTag(id) {
185         var confirmDelete = confirm("Do you want to delete tag?");
186         $.ajax({
187             url: '/Flowers/Delete/' + id,
188             type: 'POST',
189             dataType: 'json',
190             success: function (data) {
191                 showFlowersTag();
192             },
193             error: function (err) {
194                 alert("Error: " + err.responseText);
195             }
196         });
197     };
198
199     function getDetailTag(id) {
200         isUpdatable = true;
201         currentUpdatePath = "";
202         currentId = 0;
203         $("#title").text("Edit");
204
205         $.ajax({
206             url: '/Flowers/Get/' + id,
207             type: 'GET',
208             dataType: 'json',
209             success: function (data) {
210                 $("#tagName").val(data.TagName);
211                 $("#tagContent").val(data.TagContent);
212                 currentUpdatePath = data.ImagePath;
213                 currentId = data.Id;
214             },
215             error: function (err) {
216                 alert("Error: " + err.responseText);
217             }
218         });
219     };
220
221 </script>

```

## FlowersConttoller.cs

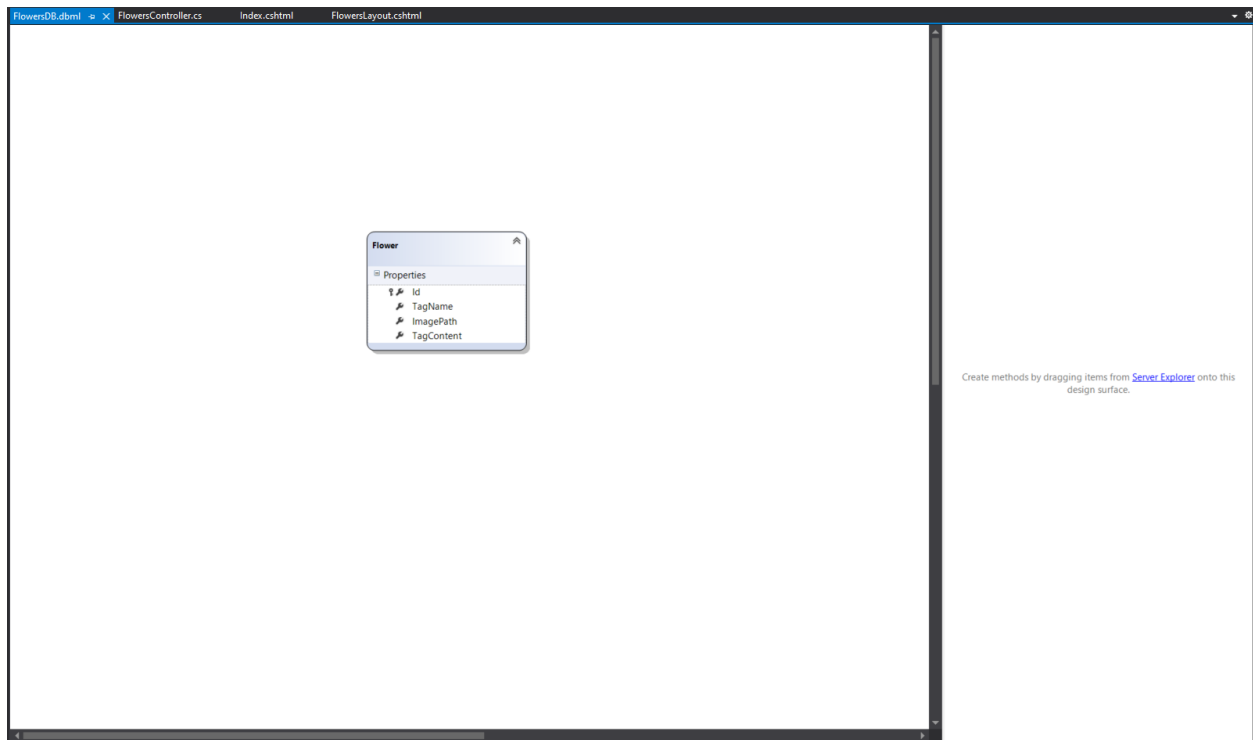
```
FlowersDB.dbml | FlowersController.cs | Index.cshtml | FlowersLayout.cshtml
Lab05_Flowers
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Web;
5 using System.Web.Mvc;
6
7 namespace Lab05_Flowers.Controllers
8 {
9
10     public class FlowersController : Controller
11     {
12         FlowersDBDataContext _context = new FlowersDBDataContext();
13
14         // GET: Flowers
15         public ActionResult Index()
16         {
17             return View();
18         }
19
20         [HttpPost]
21         public ActionResult Index(HttpPostedFileBase file)
22         {
23             if (file != null)
24             {
25                 string rootPath = Server.MapPath("~/");
26                 string fileName = System.IO.Path.GetFileName(file.FileName);
27                 string destFile = System.IO.Path.Combine(rootPath, "Assets\\images\\" + fileName);
28                 file.SaveAs(destFile);
29             }
30             return View();
31         }
32
33         public ActionResult ListTagFlowers()
34         {
35             var tags = _context.Flowers.ToList();
36             return Json(tags, JsonRequestBehavior.AllowGet);
37         }
38
39         [HttpPost]
40         public ActionResult Create([Bind(Exclude = "Id")] Flower tag)
41         {
42             if (ModelState.IsValid)
43             {
44                 string rootPath = Server.MapPath("~/");
45                 string fileName = System.IO.Path.GetFileName(tag.ImagePath);
46
47                 tag.ImagePath = "images/" + fileName;
48                 _context.Flowers.InsertOnSubmit(tag);
49                 _context.SubmitChanges();
50             }
51             return Json(tag, JsonRequestBehavior.AllowGet);
52         }
53
54         [HttpPost]
55         public ActionResult Delete(int id)
56         {
57             var tag = _context.Flowers.ToList().Find(m => m.Id == id);
58
59             if (tag != null)
60             {
61                 _context.Flowers.DeleteOnSubmit(tag);
62                 _context.SubmitChanges();
63             }
64             return Json(tag, JsonRequestBehavior.AllowGet);
65         }
66
67         public ActionResult Get(int id)
68         {
69             var tag = _context.Flowers.ToList().Find(m => m.Id == id);
70             string rootPath = Server.MapPath("~/");
71             string fileName = System.IO.Path.GetFileName(tag.ImagePath);
72             string destFile = System.IO.Path.Combine(rootPath, "Assets\\images\\" + fileName);
73             tag.ImagePath = destFile;
74
75             return Json(tag, JsonRequestBehavior.AllowGet);
76         }
77
78         [HttpPost]
79         public ActionResult Update(Flower tag)
80         {
81             if (ModelState.IsValid)
82             {
83                 string rootPath = Server.MapPath("~/");
84                 string fileName = System.IO.Path.GetFileName(tag.ImagePath);
85                 tag.ImagePath = "images/" + fileName;
86
87                 Flower tg = (from p in _context.Flowers
88                             where p.Id == tag.Id
89                             select p).SingleOrDefault();
90
91                 tg.TagName = tag.TagName;
92                 tg.ImagePath = tag.ImagePath;
93                 tg.TagContent = tag.TagContent;
94                 _context.SubmitChanges();
95             }
96             return Json(tag, JsonRequestBehavior.AllowGet);
97         }
98     }
99 }
100
```

## FlowersLayout.cshtml

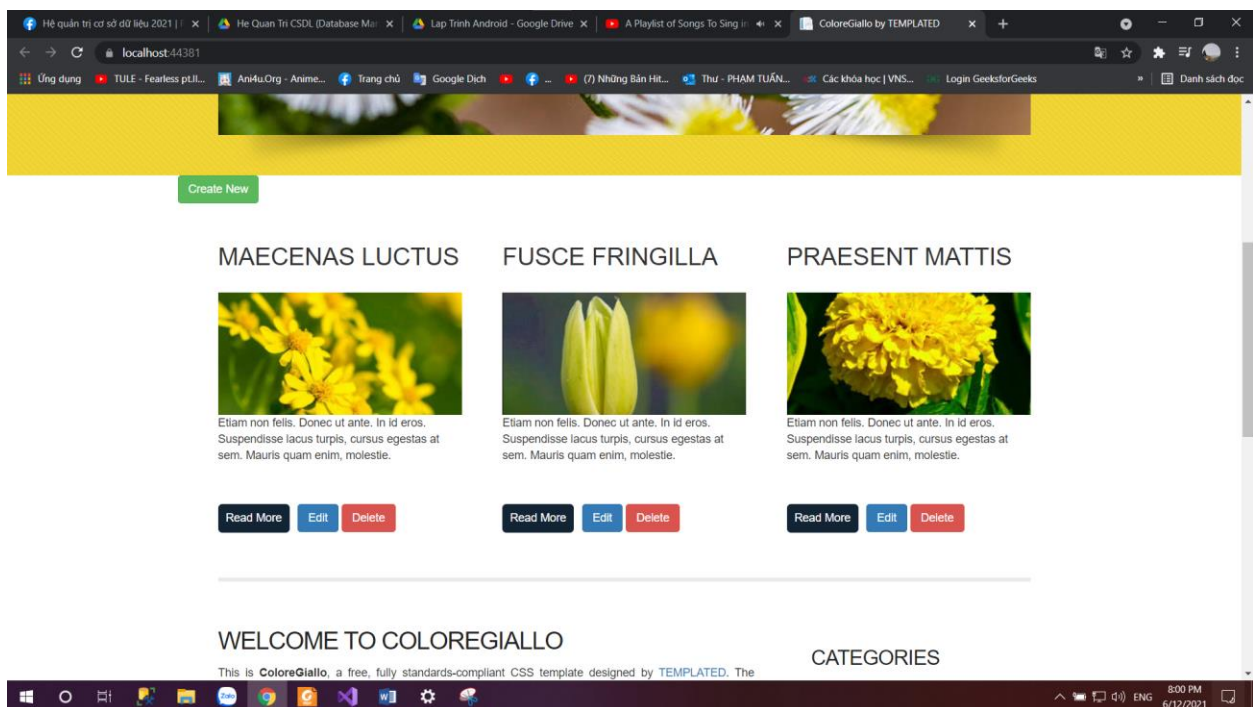
```

FlowersDB.dml  FlowersController.cs  Index.cshtml  FlowersLayout.cshtml  X
1
2  Layout = null;
3
4 <DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
5 <html xmlns="http://www.w3.org/1999/xhtml">
6 <head>
7
8 <meta http-equiv="content-type" content="text/html; charset=utf-8" />
9 <meta name="keywords" content="" />
10 <meta name="description" content="" />
11 <title>Colorediallo by TEMPLATED</title>
12 <script type="text/javascript" src="/Scripts/jquery-3.4.1.min.js"></script>
13 <script type="text/javascript" src="/Scripts/bootstrap.min.js"></script>
14 <link href="/Assets/css/style.css" rel="stylesheet" type="text/css" media="screen" />
15 <link href="/Content/bootstrap.min.css" rel="stylesheet" type="text/css" />
16 <@renderSection("script", required: false)>
17 </head>
18 <body>
19 <div id="wrapper">
20 <div id="header-wrapper">
21 <div id="header" class="container">
22 <div id="logo">
23 <h1><a href="#">Colore Giallo</a></h1>
24 </div>
25 <div id="menu">
26 <ul>
27 <li><a href="#">Homepage</a></li>
28 <li><a href="#">Services</a></li>
29 <li><a href="#">Portfolio</a></li>
30 <li><a href="#">About</a></li>
31 <li><a href="#">Contact</a></li>
32 </ul>
33 </div>
34 <div id="banner">
35 <div class="content"></div>
36 </div>
37 </div>
38 <!-- end #header -->
39 <div><button type="button" id="btnCreateNew" data-toggle="modal" data-target="#CreateEditModal"
40 class="btn btn-success" style="margin-left: 210px;">Create New</button></div>
41
42 <@renderSection("TAG", required: false)>
43
44 <div id="page">
45 <div id="content">
46 <div class="post">
47 <h2 class="title"><a href="#">Welcome to Colore Giallo</a></h2>
48 <div class="entry">
49 <p>This is <strong>Colore Giallo</strong>, a free, fully standards-compliant CSS template designed by <a href="http://templated.co" rel="nofollow">TEMPLATED</a>. The photos in this template are from <a href="http://www.fotograf.com/">Fotograf</a>.</p>
50 <p><a href="#">Read More</a></p>
51 </div>
52 </div>
53 </div>
54 <!-- end #content -->
55 <div id="sidebar">
56 <ul>
57 <li>
58 <h2>Categories</h2>
59 <ul>
60 <li><a href="#">Aliquam libero</a></li>
61 <li><a href="#">Consectetur adipiscing elit</a></li>
62 <li><a href="#">Metus aliquam pellentesque</a></li>
63 <li><a href="#">Suspendisse laculis mauris</a></li>
64 <li><a href="#">Ultricies semper</a></li>
65 <li><a href="#">Proin gravida orci porttitor</a></li>
66 </ul>
67 </li>
68 </ul>
69 </div>
70 <!-- end #sidebar -->
71 <div style="clear: both;"></div>
72 </div>
73 <!-- end #page -->
74
75 <div id="footer-content-wrapper">
76 <div id="footer-content">
77 <div id="fbbox1">
78 <h2>Fusce ultrices fringilla</h2>
79 <ul class="style1">
80 <li><a href="#">Maecenas luctus lectus at sapien</a></li>
81 <li><a href="#">Etiam rhoncus volutpat erat</a></li>
82 <li><a href="#">Etiam posuere augue sit amet nisl</a></li>
83 <li><a href="#">Mauris vulputate dolor sit amet nibh</a></li>
84 <li><a href="#">Etiam posuere augue sit amet nisl</a></li>
85 </ul>
86 </div>
87 <div id="fbbox2">
88 <h2>Nulla luctus eleifend</h2>
89 <ul class="style1">
90 <li><a href="#">Maecenas luctus lectus at sapien</a></li>
91 <li><a href="#">Etiam rhoncus volutpat erat</a></li>
92 <li><a href="#">Etiam posuere augue sit amet nisl</a></li>
93 <li><a href="#">Mauris vulputate dolor sit amet nibh</a></li>
94 <li><a href="#">Etiam posuere augue sit amet nisl</a></li>
95 </ul>
96 </div>
97 <div id="fbbox3">
98 <h2>Maecenas luctus lectus</h2>
99 <ul class="style1">
100 <li><a href="#">Maecenas luctus lectus at sapien</a></li>
101 <li><a href="#">Etiam rhoncus volutpat erat</a></li>
102 <li><a href="#">Etiam posuere augue sit amet nisl</a></li>
103 <li><a href="#">Mauris vulputate dolor sit amet nibh</a></li>
104 <li><a href="#">Etiam posuere augue sit amet nisl</a></li>
105 </ul>
106 </div>
107 </div>
108 </div>
109 <div id="footer">
110 <p><small>Copyright. Untitled. All rights reserved. Images by <a href="http://fotograf.com/">Fotograf</a>. Design by <a href="http://templated.co" rel="nofollow">TEMPLATED</a></small></p>
111 </div>
112 <!-- end #footer -->
113
114 <@renderSection("MODEL_DIALOG", required: false)>
115 </body>
116 </html>

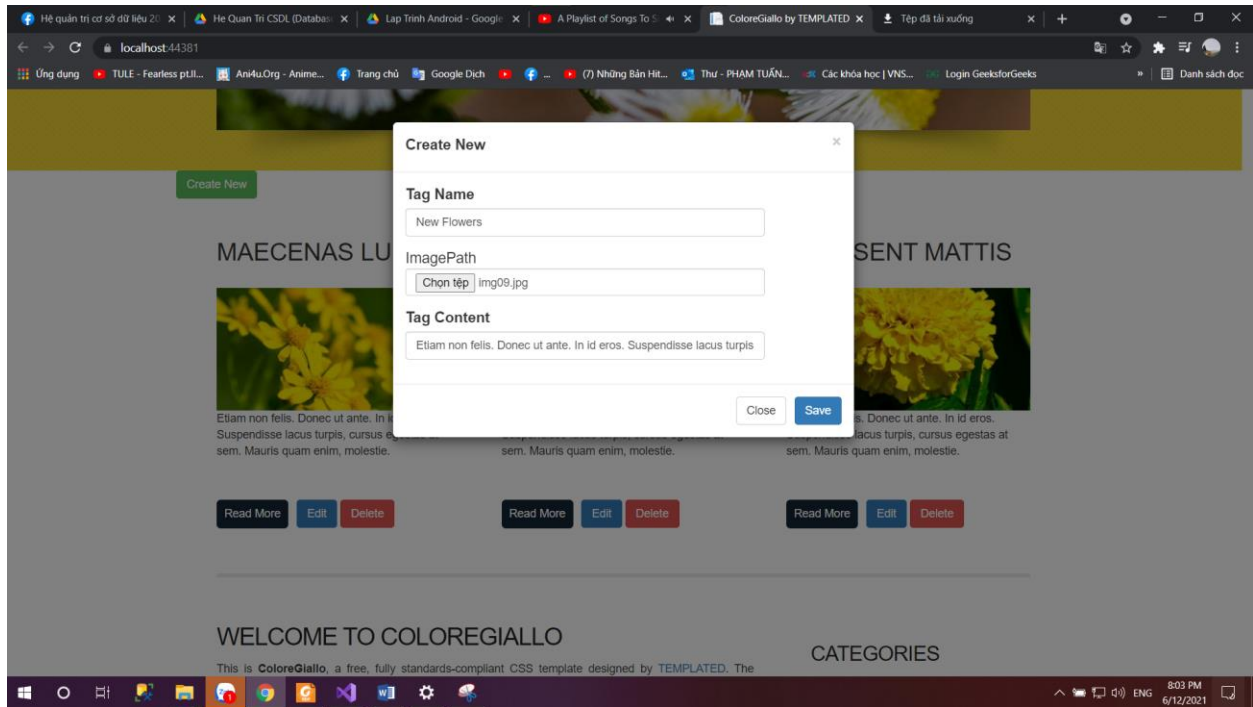
```

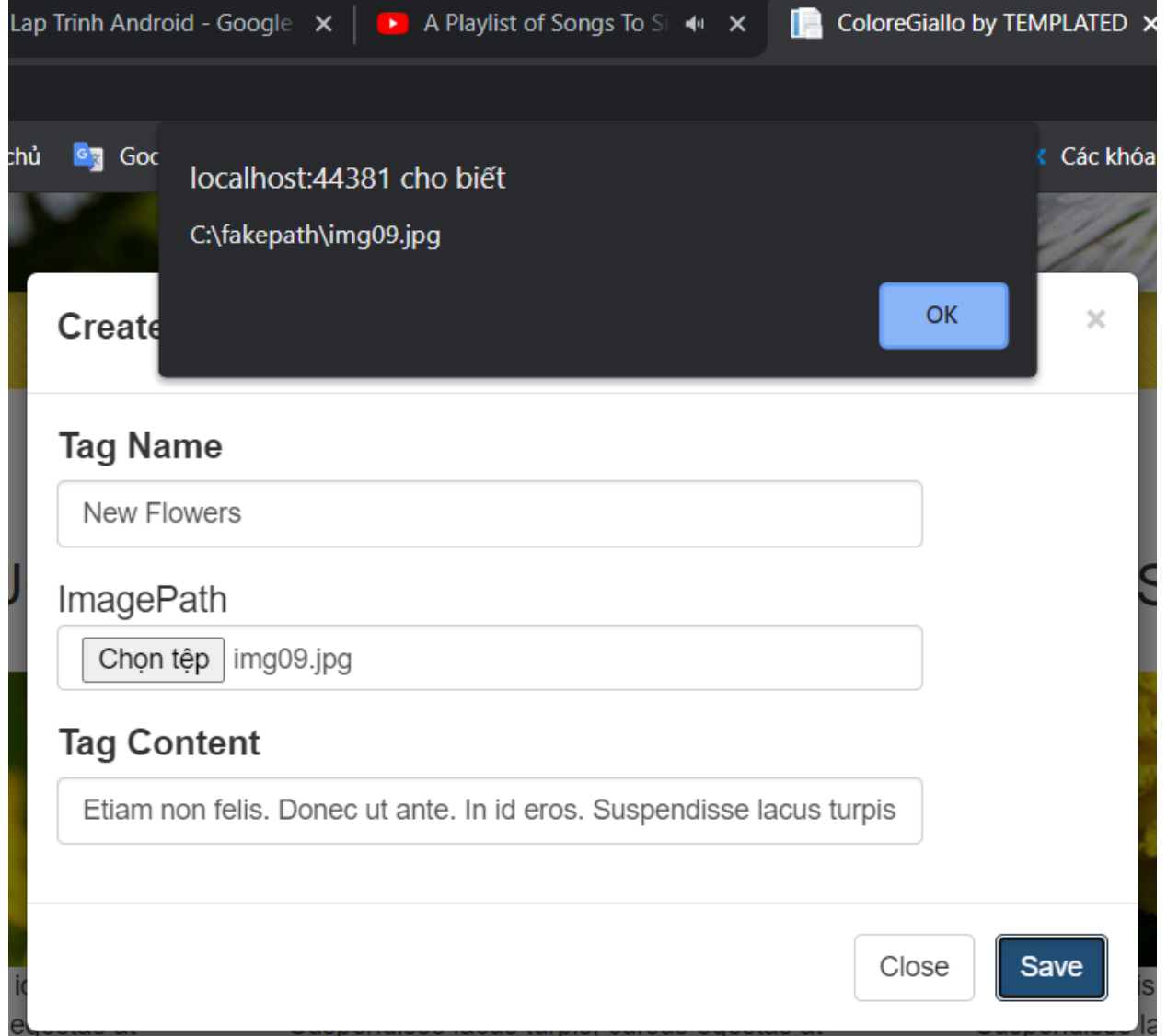


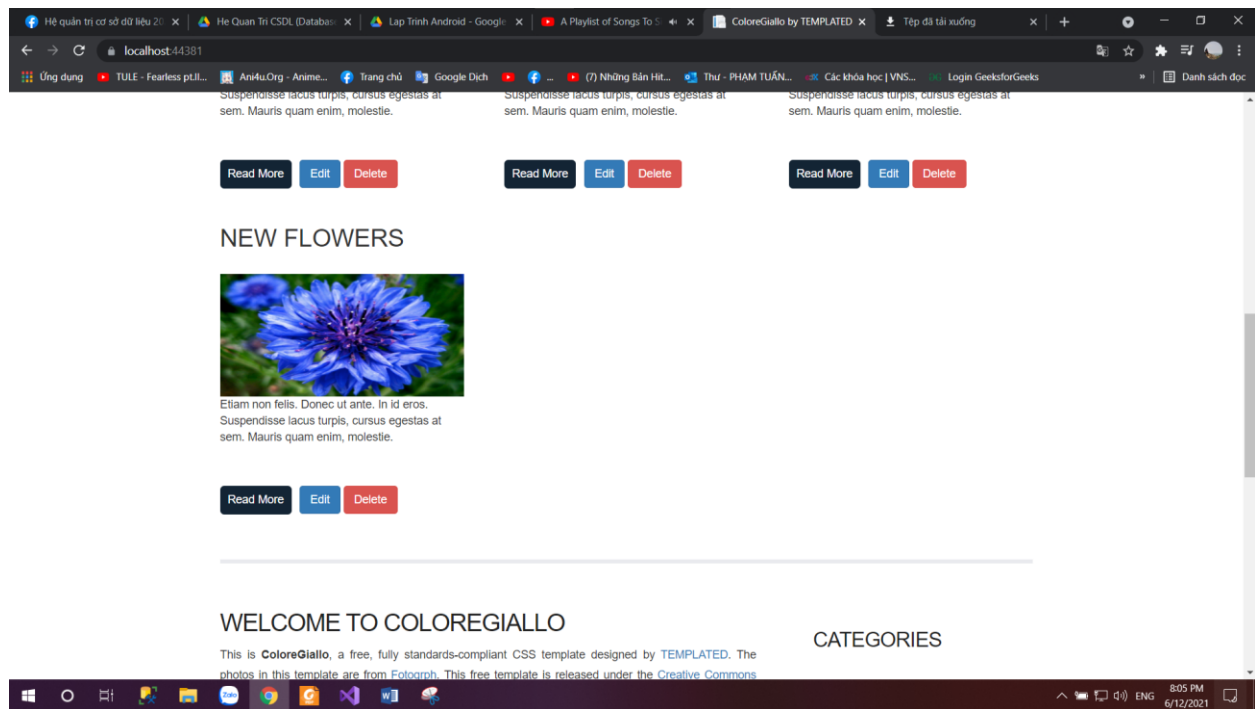
## Result:



## Create new



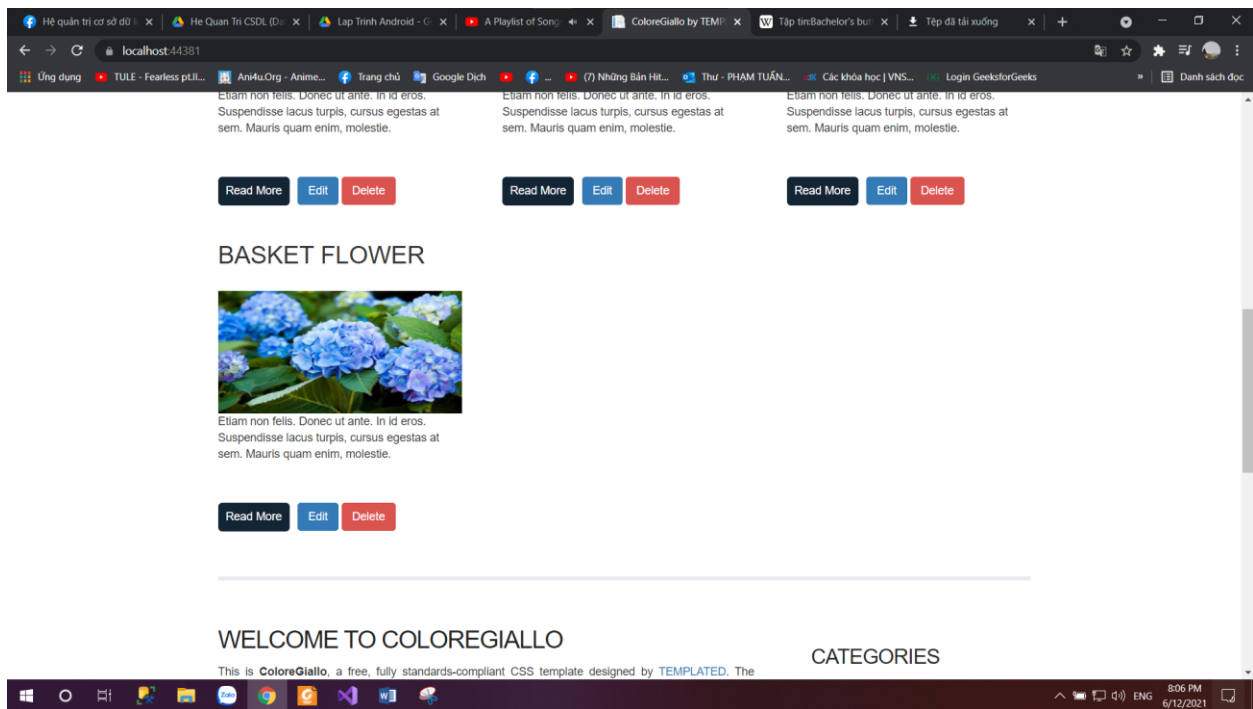




**Edit:**







Delete

