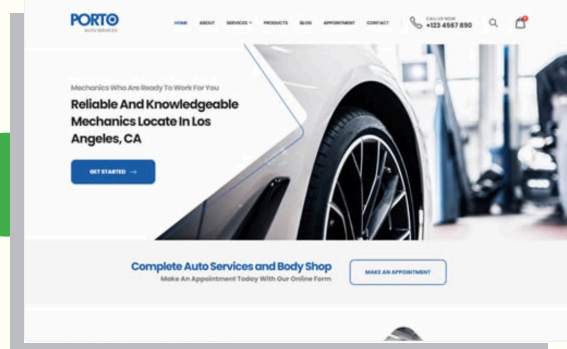
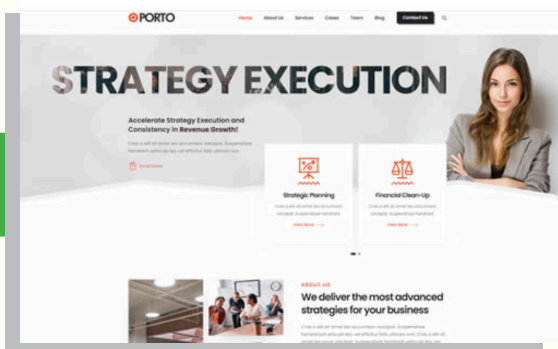
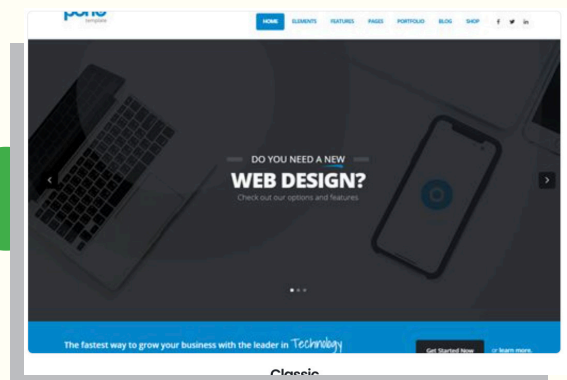
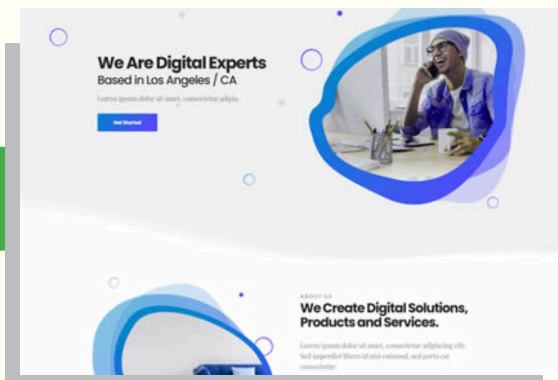


# SUMMARY

## Software Infrastructure

### Website Infrastructure:

- Website: C# MVC .NET Framework
- Web API Solution
- MSSQL Database
- Frontend will use the demo themes available at
- <https://www.okler.net/previews/porto/10.1.0/>.

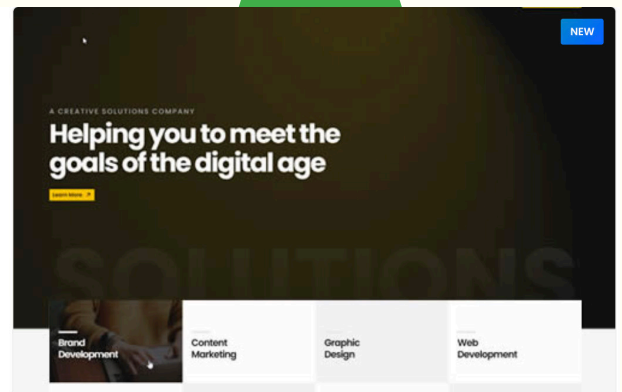


# SUMMARY

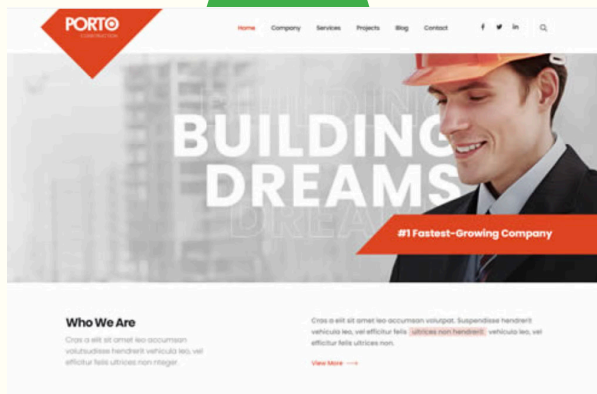
## Software Infrastructure



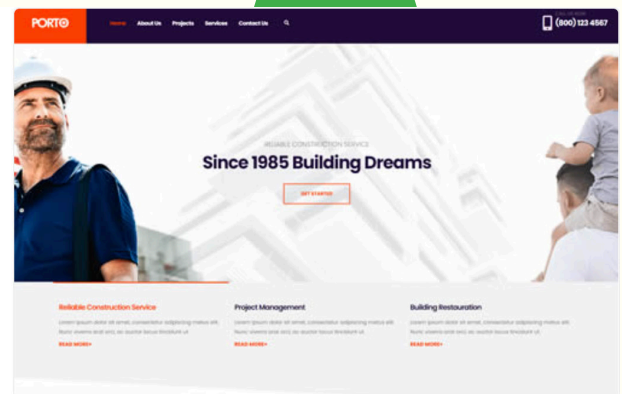
Creative Agency 1



Creative Agency 2



Construction



Construction 2

### Admin Panel Infrastructure:

- ▶ Admin Panel: C# MVC .NET Framework
- ▶ Web API Solution
- ▶ MSSQL Database
- ▶ The admin theme was purchased from
- ▶ <https://themesbrand.com/velzon/html/master/index.html> and it comes with a C# MVC .NET Framework project file. However, C# ORM (Object-Relational Mapping) code is missing.

# Admin Panel

## Project Details

- The code has not been written yet; only the pages such as header, footer, script, CSS, etc., have been formatted to the MVC layout. Each page has its own controller and action, but these are the ones from the theme. We will customize them according to our needs.

```

49  <div class="col-12">
50  <div class="fs-15 fst-italic">"Great! Clean code, clean design, easy for customization. Thanks very
51  </div>
52  <div class="fs-15 fst-italic">"The theme is really great with an amazing customer support."</div>
53  </div>
54  <div class="col-12">
55  <div class="fs-15 fst-italic">"Great! Clean code, clean design, easy for customization. Thanks very
56  </div>
57  </div>
58  </div>
59  </div>
60  </div>
61  </div>
62  </div>
63  </div>
64  </div>
65  </div>
66  </div>
67  <div class="col-12">
68  <div class="p-12">
69  <div>
70  <div class="text-primary">@Resources.Language.WelcomeBack</div>
71  <div class="text-muted">@Resources.Language.SignInToContinueToInfoIntegrate</div>
72  </div>
73  <div class="mt-4" id="formfail">
74  </div>
75  </div>
76  </div>
77  </div>
78  </div>
79  <div class="mb-3">
80  <div class="form-label">@Resources.Language.Email</div>
81  <input type="text" class="form-control" id="email" name="email" placeholder="@Resources.Language.Email">
82  </div>
83  </div>
84  <div class="mb-3">
85  <div class="float-end">
86  <a href="@Url.Action("PasswordResetCover", "Authentication")" class="text-muted">@Resources.Language
87  </div>
88  <div class="form-label">@Resources.Language.Password</div>
89  <div class="position-relative auth-pass-inputgroup mb-3">
90  <input type="password" class="form-control pe-5 password-input" placeholder="@Resources.Language.Pa
91  <button class="btn btn-link position-absolute end-0 top-0 text-decoration-none text-muted password-

```

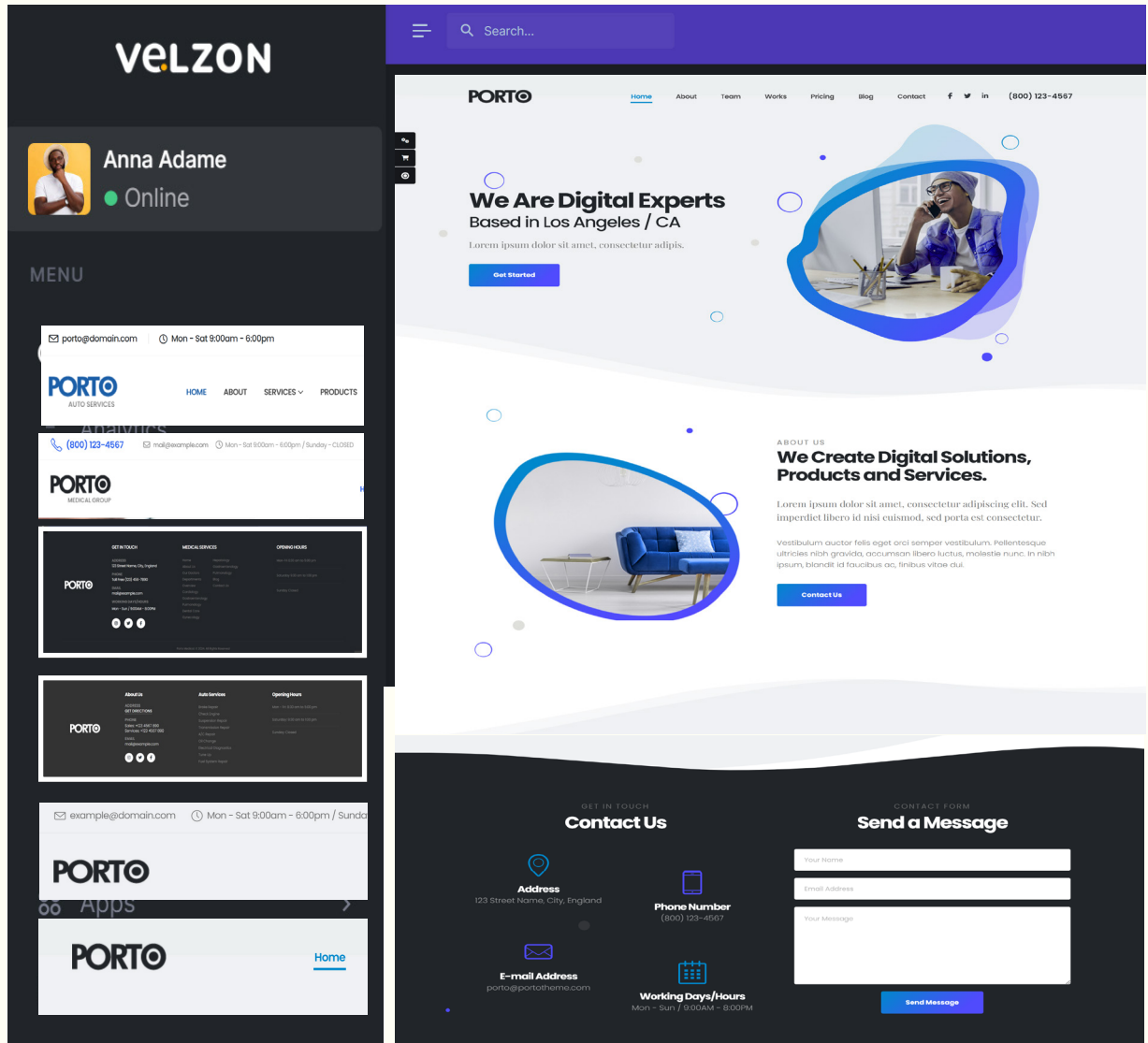
```

7  namespace InfoIntegrate.Controllers
8  {
9  public class AuthenticationController : Controller
10  {
11
12  [ActionName("SignInBasic")]
13  public ActionResult SignInBasic()
14  {
15  return View();
16  }
17
18  [ActionName("SignInCover")]
19  public ActionResult SignInCover()
20  {
21  return View();
22  }
23
24  [HttpPost, ActionName("SignInCoverUpdate")]
25  public JsonResult SignInCoverUpdate(Model.Authentication.SignInCoverUpdate form)
26  {
27  var afr = new AjaxFormResult();
28  afr.Success = false;
29  if (!ModelState.IsValid)
30  {
31  // Model hatalarını topluyoruz
32  // var errors = ModelState.SelectMany(x => x.Value.Errors.Select(z => z.ErrorMessage));
33
34
35  afr.Messages.AddRange(ModelState.SelectMany(x => x.Value.Errors.Select(z => z.ErrorMessage)));
36  return Json(afr, JsonRequestBehavior.AllowGet);
37  // Hata mesajlarını JSON olarak geri döndür
38  }
39
40
41  // Model geçerli ise, işlemlere devam edin
42  var apidata = Helper.restService.Post("/users/login?" + string.Format("email={0}&password={1}", form.Email, form.Password), "", "");
43
44  return Json(new { success = true, data = apidata });
45  }
46
47  [ActionName("SignUpBasic")]
48  public ActionResult SignUpBasic()
49  {

```

# Admin Panel

## Project Details



- ADMIN
- .REGISTER
- .FORGOT PASSWORD
- .LOG IN
- .MY WEBSITES
- ..CREATE NEW SITE
- ...CHOOSE THEME
- ...CHOOSE SECTION
- ...SAVE
- ..UPDATE SITE THEME
- ...CHOOSE THEME
- ...CHOOSE SECTION
- ...TRANSFER CONTENT OF THE SAME MODULES
- ...SAVE
- ..UPDATE SITE CONTENT
- ...UPDATE ALL PAGE CONTENTS LIKE
- HOMEPAGE, CONTACT, GALLERY

- JSON
- .SECTIONS HEADER, FOOTER, PAGE-HEADER,
- GET IN TOUCH, TESTIMONIALS
- .(PAGES) HOME - SELECTED SECTION - VISIBLE/
- INVISIBLE FIELDS
- .(PAGES) ABOUT US - SELECTED SECTION -
- VISIBLE/INVISIBLE FIELDS
- .(PAGES) SERVICES > SUB PAGES - SELECTED
- SECTION - VISIBLE/INVISIBLE FIELDS
- .(PAGES) CASES - SELECTED SECTION - VISIBLE/
- INVISIBLE FIELDS
- .(PAGES) TEAM - SELECTED SECTION - VISIBLE/
- INVISIBLE FIELDS
- .(PAGES) CONTACT US - SELECTED SECTION -
- VISIBLE/INVISIBLE FIELDS
- MSSQL
- .BLOG

# WEBSITE DESIRED FEATURES

Imagine a system similar to WordPress. Users will select one of our themes from the admin panel, make the payment, and launch their website. Then, they will start adding content. All themes are compatible with each other. This means that if you place the header of one theme into another, the menu style will change to match the style of the new theme. Components like the header, footer, content, etc., are all sections. In the selected theme, users will have the option to change sections like the header, footer, gallery, contact form, etc., to match those from other themes.

