

```
1 using DatabaseConnect;
2 using DatabaseConnect.Entities;
3 using Microsoft.AspNetCore.Cryptography.KeyDerivation;
4 using Microsoft.AspNetCore.Mvc;
5 using Microsoft.Extensions.Logging;
6 using System;
7 using System.Collections.Generic;
8 using System.Linq;
9 using System.Security.Cryptography;
10 using System.Threading.Tasks;
11 using static LibraryAppMVC.Models.Models;
12
13 namespace LibraryAppMVC.Controllers
14 {
15     [Route("/dev/")] // All endpoints checked 2/25/18
16     public class DevController : Controller
17     {
18         private Context _ctx;
19         private readonly ILogger _logger;
20         public DevController(Context context, ILogger<UserController> logger)
21         {
22             _ctx = context;
23             _logger = logger;
24         }
25
26         [Route("adduser")]
27         [HttpPost]
28         public IActionResult AddUser([FromBody]NewUser newuser) // Checked 2/25/18 ↗
29             working
30         {
31             if (newuser.UserTypeInt == 0) { newuser.UserTypeInt = 1; }
32             User user = new User() { SchoolID = newuser.Username, Password = ↗
33                 newuser.Password };
34             byte[] salt = new byte[128 / 8];
35             using (var rng = RandomNumberGenerator.Create())
36             {
37                 rng.GetBytes(salt);
38             }
39             string hashed = Convert.ToBase64String(KeyDerivation.Pbkdf2(
40                 password: user.Password,
41                 salt: salt,
42                 prf: KeyDerivationPrf.HMACSHA1,
43                 iterationCount: 10000,
44                 numBytesRequested: 256 / 8));
45             user.Salt = Convert.ToBase64String(salt);
46             user.PasswordHash = hashed;
47             _ctx.Users.Add(user);
48             _ctx.SaveChanges();
49             int UserID = _ctx.Users
50                 .Single(u => u.SchoolID == user.SchoolID)
51                 .UserID;
52             _ctx.UserUType_rel
```

```
51         .Add(new UserUType { UserID = UserID, UTypeID = 1 });
52         _ctx.SaveChanges();
53         return Ok();
54     }
55 }
56 }
57
```