









Process: [15908] iisexpress.exe

Debug Any CPU Continue Lifecycle Events Thread Stack Frames

User\_DetailsController.cs MemberProvConvoesController.cs User\_RoleController.cs ResetPasswordsController.cs UQP.Controllers.ResetPasswordsController GetResetPasswords

```
68         await db.SaveChangesAsync();
69     }
70     catch (DbUpdateConcurrencyException)
71     {
72         if (!ResetPasswordExists(id))
73         {
74             return NotFound();
75         }
76         else
77         {
78             throw;
79         }
80     }
81
82     return StatusCode(HttpStatusCode.NoContent);
83 }
84
85 // POST: api/ResetPasswords
86 [ResponseType(typeof(ResetPassword))]
87 public async Task<IHttpActionResult> PostResetPassword(ResetPassword resetPassword)
88 {
89     using (UQPEntities db = new UQPEntities(HttpContext.Current.Request.Headers.GetValues("IPA").FirstOrDefault()))
90     {
91         if (!ModelState.IsValid)
92         {
93             return BadRequest(ModelState);
94         }
95     }
```

Debug Team Tools Test Analyze Window Help

Process: [15908] iisexpress.exe

Debug - Any CPU - Continue - Lifecycle Events - Thread: Stack Frame

DetailsController.cs MemberProvConvoesController.cs User\_RoleController.cs ResetPasswordsController.cs

UQP.Controllers.ResetPasswordsController

```
92 if (!ModelState.IsValid)
93 {
94     return BadRequest(ModelState);
95 }
96
97 db.ResetPasswords.Add(resetPassword);
98 await db.SaveChangesAsync();
99
100 return CreatedAtRoute("DefaultApi", new { id = resetPassword.id }, resetPassword);
101 }
102
103 // DELETE: api/ResetPasswords/5
104 [ResponseType(typeof(ResetPassword))]
105
106 public async Task<IHttpActionResult> DeleteResetPassword(long id)
107 {
108     using (UQPEntities db = new UQPEntities(HttpContext.Current.Request.Headers.GetValues("IPA").FirstOrDefault()))
109     {
110         ResetPassword resetPassword = await db.ResetPasswords.FindAsync(id);
111         if (resetPassword == null)
112         {
113             return NotFound();
114         }
115
116         db.ResetPasswords.Remove(resetPassword);
117         await db.SaveChangesAsync();
118
119         return Ok(resetPassword);
120     }
121 }
```



