* Users
  + UserID
  + UserName
  + OpenIdClaim?
  + Email
  + EmailHash
  + DateOfRegistration
  + LastSeenDate – make a function call for updating this
  + [ProfileInformation]
    - AboutMe
    - IsSuspended
    - ReinstateDate
* UserOpenIDs
  + See SEDE
* Roles
  + RoleID
  + RoleName
* UsersInRoles
  + UserID
  + RoleID
* ~~Resumes~~
  + ~~ResumeID~~
  + ~~ResumeType~~
  + ~~UserID~~
  + ~~IsFiled~~
  + ~~FiledStartDate~~
  + ~~FiledEndDate~~
  + ~~[ResumeInfo]~~
    - ~~Name~~
    - ~~About (Markdown)~~
    - ~~Achievements (Markdown)~~
    - ~~Education (Markdown)~~
    - ~~Experience (Markdown)~~
    - ~~Other (optional)~~
* ~~ResumeTypes~~
  + ~~TypeID~~
  + ~~TypeName~~
    - ~~Examples: Musician Resume, Teacher Resume~~
* ~~Jobs~~
  + ~~JobID~~
  + ~~UserID~~
  + ~~Location (optional)~~
    - ~~StreetAddress~~
    - ~~Lat~~
    - ~~Long~~
  + ~~Description (Markdown)~~
  + ~~Requirements (Markdown)~~
  + ~~Other (Markdown)~~
* Stores tables (will we handle online stores? I don’t want to)
  + Listing:
    - ID
    - Address
    - Lat
    - Long
    - Name
    - Description
    - UserID
  + Review
    - ID
  + ReviewRevision
    - ID
    - ?
* Discuss
  + Boards
    - ID
    - Location (city name, board name)
  + Threads
    - ID
    - BoardID
    - Title
  + Posts
    - ID
  + PostRevisions
    - ID
    - ?
* InstrumentsForSale: tables:
  + Duplicate:
    - Listing
    - Review
    - ReviewRevisions
  + Add:
    - Optional SoldByStoreID to Listing
    - Edit Listing without revision tracking

OpenID process:

* Authentication
* If already registered, redirect to home page or return url (**NOTE:** Use GeekDinner code where it verifies that return url is in fact one of our own urls, not to a different domain)
* If not registered, show registration form. User specifies:
  + Email
  + UserName
  + AboutMe (optional)