Group 5 Technical Prototype

Technical Prototype Questions and Challenges: Source for current prototype is on our github and will be tagged as 2 (for end iteration 2)

* How are we going to handle database backups?
  + We currently have a database backup on github. At the end of each iteration, we will include a current version of the database. We will also investigate scheduling programmatic database backups.
* How are we going to handle user permissions to access budgets and other user roles?
* How are we going to quickly and efficiently populate the database?
  + It is possible to seed the database from a CSV file.
* Is our current setup valid for the entire project?
  + Not entirely, but the general framework is very easy to redo and we now know what works. The general idea of C#, asp.net, and everything else with Microsoft SQL is definitely valid and this prototype helped a lot with that.
* What changes would we consider necessary for project success?
  + No technical ones, really. Just some streamlining and refactoring of code.
* What is the best-practice workflow for adding features?
  + Github individually throughout week, meet up for a few hours for a group coding session when necessary.
* What naming conventions should we adhere to?
  + We have discovered some naming conflicts among our initial models, SQL, and asp.net Identity. These have been resolved by refactoring our models, but this is an issue we will have to monitor as we develop additional models.
* What should we add to the ASP.Net user table, and what should be stored in separate tables?
  + This was autogenerated for .net Identity, though we are likely going to have a separate table for user permissions (like guest/child accounts, not Admin/user accounts)
* Should we allow SQL Server to auto-generate Id values in each table, or should we programmatically control the creation of Ids?
  + Auto-generation is simpler to implement, but may add superfluous columns to tables. For example, the ACCOUNTS table would have a auto-generated Id and an accountNumber that we add.