Auto Build’s 3-Tier Architecture abstraction layer definitions

Presentation Tier

UI: This layer is the user interface, and is all elements regarding data display to the user, as well as the starting point for user interaction and input.

Logic Tier

Entry Point: The entry point is the layer of abstraction at which the users input data is caught and handled before passing it down to the lower layers. This layer should catch all remaining errors that have been thrown by any lower levels (if necessary). This layer should also handle any translation of private data to safe data that will not reveal privacy, inner workings of the system, or naming conventions.

Business Rule/Business Logic: This layer handles all business rules and business logic that is to be implemented, in accordance with the BRD and the project proposal. This is largely app specific code.

Reusable: This layer is where our reusable code, and generic modules reside. We are to implement several generic modules, and they will take in translated data (from the business layer) to handle generic functions. This layer will likely throw errors, however, if we can handle generic obvious errors here, this would be good.

DAL: The Data Access Layer handles all data access objects and methods. These will handle any database errors and will communicate with the Database. (We intend to use an ORM and are unsure if this is necessary)

Data Storage Tier

DS: This layer is the database itself, and will likely have a denial of direct access but allow granted access for some specific queries and methods to perform desired tasks (Such as add user), these permissions will be granted based on roles.