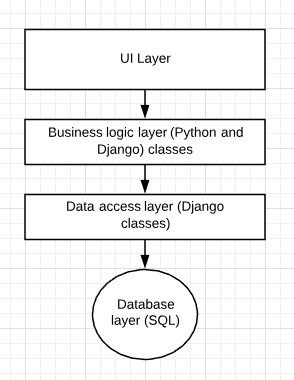
The architecture pattern chosen for the case study project is known as a layered pattern. This pattern is used to construct programs that can be deconstructed into subtasks, with each being at a level of abstraction. Then each layer provides services to the layer above it. The layers in this architecture pattern are:

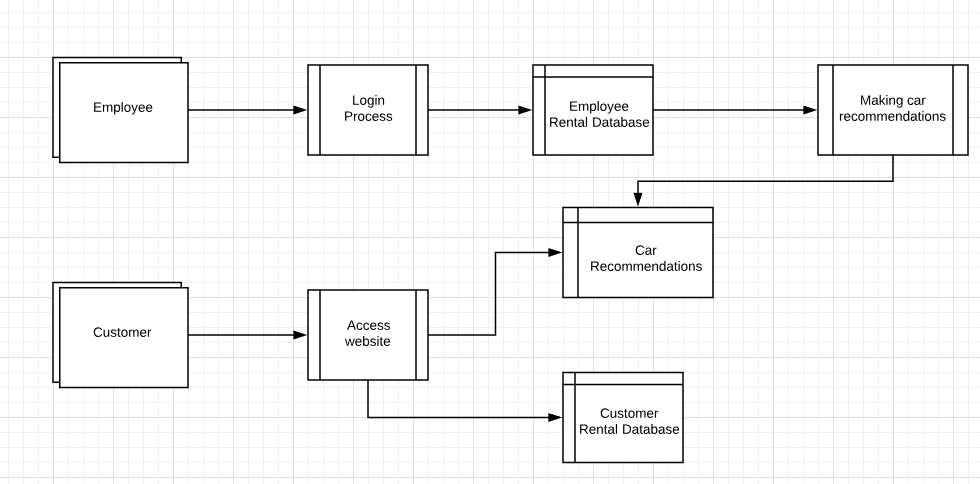
* The User interface layer (Presentation layer)
* Service layer (Application layer)
* Business logic layer (Domain layer)
* Data access layer (database layer)

This software architecture pattern was chosen because it is best suited for a desktop web application like the case study project. Layered pattern focuses around the database and because this application is essentially to view information from a database, this architecture pattern is best suited. A positive of layered architecture is that here is a separation of concerns, this means that each layer can focus solely on its role in the application. This makes the application:

* Easy to maintain and update
* Easy to test each individual layer separately
* Easy to assign each layer separate roles

If the architecture is layered correctly it will have separate layers that aren’t affected by changes in other layers, which can make certain changes easier because it won’t affect the application as a whole.

****

****

