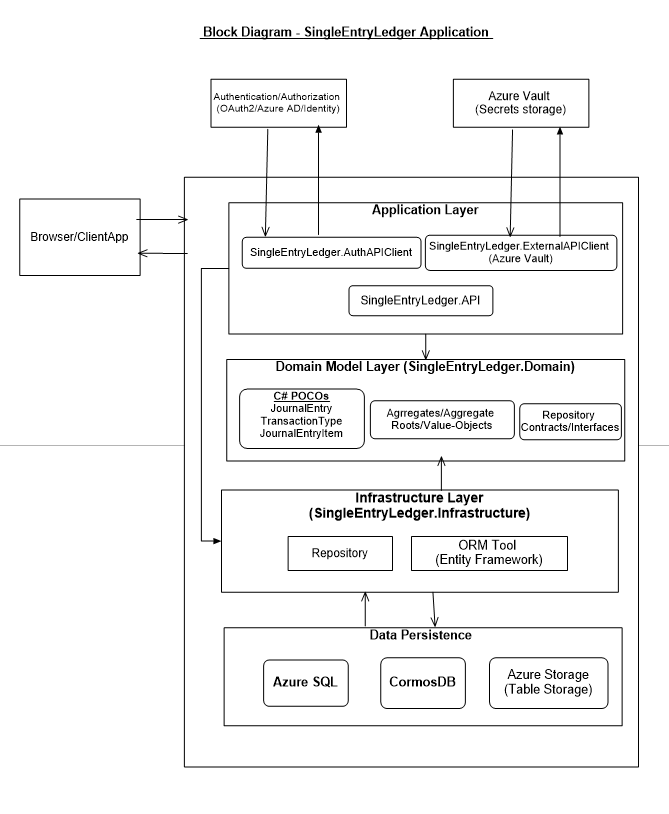
**SingleEntryLeder Application**

SingleEntryLedger application system can be designed as follows: 

The application is composed of the following layers:

1. ***Application* *Layer***

This layer is responsible for

* communication with the client applications through a set of API methods,
* communicate with the Authentication/Authorization systems for user authentication/authorization, and communicate with the AzureVault for storing/retreiving application secrets
* communication with any external dependent API
* propagate the application flow and interact with the subsequent layers.
* It is created as a .NET Core Web API and can be deployed as Azure WebApps using Azure AppService.

1. ***Domain Model Layer***

This layer holds all the Entities, Aggregate roots, Aggregates and value objects that resembles the business domain and complexities. All the business rules will be defined here and should be developed as a class library, which will be referred in and be part of the SingleEntryLedger system. In this application, this layer holds the following Domain entities and aggregates:

* JournalEntry
* JournalEntryItem
* TransactionType

The repository interface/contracts which enable references for the infrastructure are also defined in this layer. This layer will be part of the solution and does not have any dependencies with the other layers of the application, This layer will be part of the application and deployed as an AppService in Azure.

1. ***Infrastructure layer***

This layer defines the Repository classes for the EntityFramework Core dbContext and further takes care of persisting data to the persistence data store such as for relational data (Azure SQL), NoSQL data (CosmosDB), for key-value pair (Azure storage – Table storage)