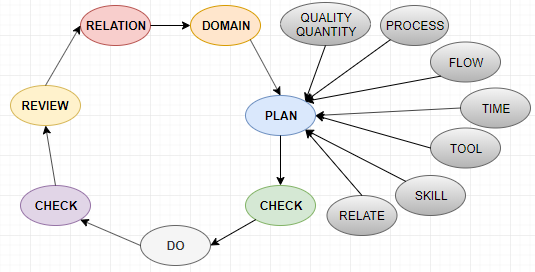
**KICKOFF SUMMARY**

**INTRANET APPLICATION:**

* HR
* Admin
* Visitors
* Materials
* Service Ticketing
* Other Service which serves as an aid for an organization in daily management activity.

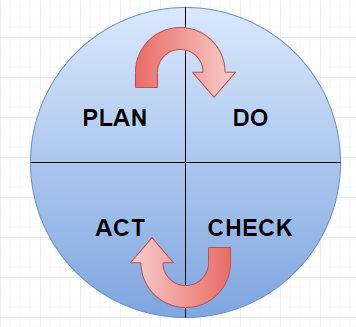
**Approach Model:**



The planning process should include the following aspects,

* Requirement of the Business Ideas.
* Process is the way of approach to execute the Business requirement.
* Flow is the pattern in which Process goes on.
* Relationship between the process to be fulfilled.
* Skill required to carry the process.
* Tool - Application needed to complete the Application.
* Domain - Intranet Portal.

All these business aspects should follow a principle of **2Q1T** ( Quality Quantity reversed with Time).



**Deming’s Wheel**

The planning model should take care of the unexpected problems. As that the planned model should be futuristic and error free one.

Check has to be done once the planning model is designed, so that the probability of arising error should be minimized.

DO - Implements the ideas and executes the planned modeled process.The primary requirements should be met first.

The checking has to be done in a regular interval with unit testing and with some other QA techniques, so that probability of bugs is reduced.

The type of testing that been carried out are,

* + - * Unit Testing
      * Integration Testing
      * Performance Testing
      * Load Testing

In the Review phase the final checking is done and the process should possess a effective relationship with one another.

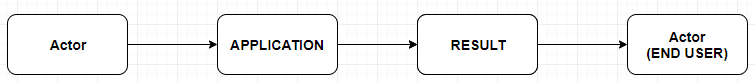
**DATA**

The Data input should be validated across the application starting from the entry level to the Destination level of the Application.

The Relationship of these data input should be related with each other in a properly and an orderly manner such that respective data inputs should be equipped in each section.

Persistence - the availability(lifetime) of the data should be maintained across the application. Eg: The password data input shouldn’t be shown in the final result page as in case of booking a product in Online Store. The longevity and the location of the data inputs should be maintained.

**APPLICATION ARCHITECTURE**



The Application Architecture 00 is designed. The business idea should be converted into the Application by the actor ( by developer) the final result of the system application is accessed by the end user.

**Application:**

1. **Client** :

It focuses on the front end of the application which includes the UI/UX designing part . The web designs are done using Angular JS framework , Bootstrap is to be used for enhancing UI. Angular JS allows you to use plugins which injects built-in features into the application.

The Storage and cache should be taken care at the front end. These preserves the data input of the end user.

The Session Token should be used which act as a trigger which **kickoffs** the user after the session timeout. Remember me option can be included so that it’s not necessary to enter the login credentials each and every time.

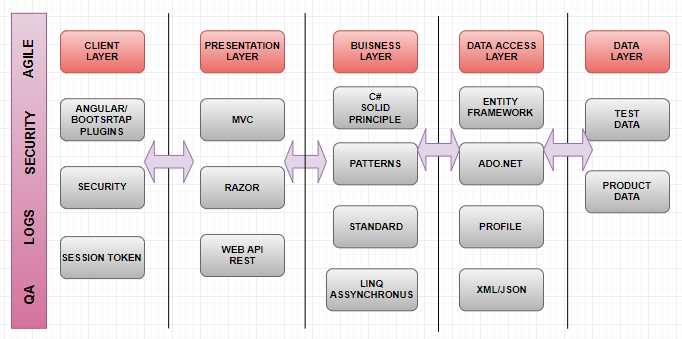
The Security should be ensured across all the verticals in the application. As that data should preserved and not to be shared across the application. OTP authentication and E-Mail verification has to be done for every time user login’s and forget password feature has to be enabled by means of some security questions.

Various social network login platforms have to be integrated to this such as Facebook, Google, Twitter etc..

The Performance of the application should be maintained across all vertical in the application.

HTTPS protocol should be used to ensure the security.

All layers should have Agility across the verticals.



**Architectural Diagram**

1. **Presentation**:

Model View Control (MVC) Architecture, REST Architecture should be used in designing the Application.

Razor pages can be used in the application to make the web page dynamic.

WEB API is used in designing the application to fetch the data in a more secured manner. It acts as a platform between the front the back end. Application protocol interface call is done to fetch data from database which converts it into a required format and send it into front-end in more secure manner.

1. **Business Layer:**

In this layer the C# takes control which follow the SOLID Principle.

The various SOLID Principles are,

* SINGLE RESPONSIBILITY PRINCIPLE
* OPEN/CLOSED PRINCIPLE
* LISCOV SUBSTITUTION PRINCIPLE
* INTERFACE SEGREGATION PRINCIPLE
* DEPENDENCY INVENSION PRINCIPLE

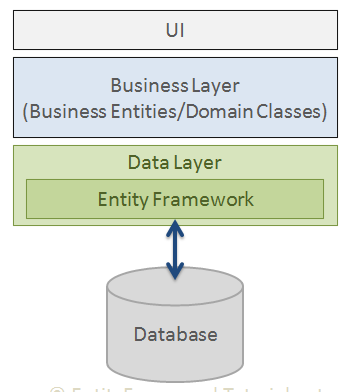
Other principle such as GRASP can also be followed in the Business layer such as Low Coupling, High Cohesion,Controller, Information Expert etc..

The Industry standard and design pattern should be maintained across the Application.

The various patterns in software development are,

* Architectural Patterns
* Design Patterns
* Idioms

Asynchronous and LINQ should be used in the Application.All the possible cases should be tested and application should be agile in nature.



**LAYERS OF APPLICATION DEVELOPMENT**

1. **Data Access Layer:**

The Entity Framework and ADO.net should be used which acts as a platform between the database and data Layer. It saves data stored in the properties of business entities and also retrieves data from the database and converts it to business entities objects automatically.

The data can be stored in the form of JSON/XML depends on the use case we possess.

1. **DATA:**

The database should be populated with test data during the time of application development and after the release it should be able to access it with the production data(real-time) data.

**POC**

Proof of Concepts - It represent the prototype of the ideologies we learn in this Application Development Phase.