**Notebook Application Demo**

Architectural choices –

**Choice#1-**

Can we build a web based application with client support (thick client)? Silverlight? Yes this can be a choice but it is outdated and want to go with this. Also local storage capacity, ability to perform, can we make rich UI, time to develop a rich client UI are concerns. The pros are we can host both thin and thick client at same time of development. Microsoft Sync framework is available which can be used for syncing. Multiple platform support is also an advantage.

We can either move to silver light or any equivalent technology to achieve this architecture

**Choice#2**

We can build a thick client (WPF, Forms) and client which have all offline capabilities as well as ability to sync with cloud storage. We need additional development in web based UI development to make it to web compatible. Have to implement multiple platform support and this can be run over any platform if .net framework is supported. Considering the rich UI and offline capabilities, performance criteria and ability to store locally (local cache) this can be a better choice.

I have developed an application both with silver light and Windows forms/WPF to do a POC study to learn on the architecture perspective.

I decided to move with choice#2 –

Which is having below modules in the solution – NoteAppUI (this is an executable run on client system) which is based on forms/WPF

1. Offline capability
2. Detect network status via NoteBookCommn.DLL
3. Local persistent cache/storage NoteBookObj.DLL (on a binary file)
4. Sync to cloud via NoteBookSync.DLL which is based on WCL and will be using Microsoft Sync framework
5. Serialization/de serialization support for objects stored and can be used for attaching notes over outlook extension

**Testing strategies**

**Unit testing**

Visual studio C# unit testing framework can be used for UT

Unit Test Using Mock Object and Dependency Injection can be covered like this

Online and offline sync emulation?