* **Utilize MVC or MVVVM design patterns** - Develop your application’s User Interface using the Model/View/Controller pattern. Architect your application using a Model/View/Controller approach or a Model/View/ViewModel approach where there is a clear separation between the “Model” and the rest. Determine which parts of your application will be using native user interface elements of each platform (iOS, Android, Windows, Mac) and use this as a guideline to split your application into two components: “Core” and “User-Interface”

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**.

To increase the amount of code re-use, you can adopt cross-platform components that provide common services across all these systems such as:

1. [SQLite-net](https://www.nuget.org/packages/sqlite-net-pcl/) for local SQL storage,
2. [Xamarin Plugins](https://xamarin.com/plugins) for accessing device-specific capabilities including the camera, contacts and geolocation,
3. [NuGet packages](https://nuget.org/) that are compatible with Xamarin projects, such as [Json.NET](https://www.nuget.org/packages/Newtonsoft.Json/),
4. Using .NET framework features for networking, web services, IO and more.

Some of these components are implemented in the *Tasky* case study.

)\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* **Android** – C# is compiled to IL and packaged with MonoVM + JIT’ing. Unused classes in the framework are stripped out during linking. The application runs side-by-side with Java/ART (Android runtime) and interacts with the native types via JNI (see [Xamarin.Android Limitations](https://developer.xamarin.com/guides/android/advanced_topics/limitations) ).
* **Business Layer** – (sometimes called the Business Logic Layer or BLL) contains business entity definitions (the Model) and business logic. Candidate for Business Façade pattern.

Part5 foreach first code snippet

Last two code snippets

------------------------------------------------------------------------------------------------------------

* What is meant by Cannot use compiler directives.