**Technical Specifications**

**Visual Studio 2012 – Development Environment**

Microsoft Visual Studio is an [integrated development environment](https://en.wikipedia.org/wiki/Integrated_development_environment) (IDE) from [Microsoft](https://en.wikipedia.org/wiki/Microsoft). It is used to develop [console](https://en.wikipedia.org/wiki/Console_application) and [graphical user interface](https://en.wikipedia.org/wiki/Graphical_user_interface) [applications](https://en.wikipedia.org/wiki/Application_software) along with [Windows Forms](https://en.wikipedia.org/wiki/Windows_Forms) or [WPF](https://en.wikipedia.org/wiki/Windows_Presentation_Foundation) applications, [web sites](https://en.wikipedia.org/wiki/Web_site), [web applications](https://en.wikipedia.org/wiki/Web_application), and [web services](https://en.wikipedia.org/wiki/Web_service) in both [native code](https://en.wikipedia.org/wiki/Native_code) together with [managed code](https://en.wikipedia.org/wiki/Managed_code) for all platforms supported by [Microsoft Windows](https://en.wikipedia.org/wiki/Microsoft_Windows), [Windows Mobile](https://en.wikipedia.org/wiki/Windows_Mobile), [Windows CE](https://en.wikipedia.org/wiki/Windows_CE), [.NET Framework](https://en.wikipedia.org/wiki/.NET_Framework), [.NET Compact Framework](https://en.wikipedia.org/wiki/.NET_Compact_Framework) and [Microsoft Silverlight](https://en.wikipedia.org/wiki/Microsoft_Silverlight).

Microsoft Visual Studio 2012 is loaded with new capabilities for Windows 8, the web, SharePoint, mobile, and cloud development—as well as the application management lifecycle tools you need to break down team barriers and reduce cycle times to deliver value continuously. You’ll even find a redesigned UI to streamline everyday tasks and kick your productivity into gear.

When it comes to web development, Visual Studio 2012 also has you covered with new templates, better publishing tools, and full support for emerging standards, like HTML5 and CSS3, as well as the latest advances in ASP.NET. We've also made it easier to debug with the Page Inspector by interacting with the page you're coding, right in the IDE. Going mobile? With ASP.NET you can now create applications with controls that optimize for phones, tablets, and other small screens.

* This tool is used as our main development environment in creating our web applications as well as our mobile application.
* The main reason for this is for faster app creation due to its rich intelligence on various languages including C#, HTML and other languages that we are using.

**ASP.NET Web API**

ASP.NET Web API is a framework that makes it easy to build HTTP services that reach a broad range of clients, including browsers and mobile devices. ASP.NET Web API is an ideal platform for building REST full applications on the .NET Framework.

WebAPI is an ideal platform for building pure HTTP based services where the request and response happens with HTTP protocol. The client can make a GET, PUT, POST, and DELETE request and get the WebAPI response appropriately.

In summary, the WebAPI is

* An HTTP Service
* Designed for broad reach
* Uses HTTP as an Application protocol, not a transport protocol
* This tool is primarily used in building our http services.
* The main reason for this is because it allows us to reach a broad range of clients easier.

**HTML 5 / CSS3 / JQuery / JQuery Mobile**

* [**HTML5**](http://en.wikipedia.org/wiki/HTML5) - provides developers with tools such as Offline Web Storage, Geo Location API, Canvas Drawing, CSS3, and many more. HTML5 applications run on [Tizen](http://en.wikipedia.org/wiki/Tizen" \o "Tizen) and [Firefox OS](http://en.wikipedia.org/wiki/Firefox_OS), [Web OS](http://en.wikipedia.org/wiki/WebOS) and [Ubuntu Touch](http://en.wikipedia.org/wiki/Ubuntu_Touch) without a browser.
* **CSS** - Cascading Style Sheets (CSS) is a [style sheet language](http://en.wikipedia.org/wiki/Style_sheet_language) used for describing the [presentation semantics](http://en.wikipedia.org/wiki/Presentation_semantics) (the look and formatting) of a document written in a [markup language](http://en.wikipedia.org/wiki/Markup_language). Its most common application is to style [web pages](http://en.wikipedia.org/wiki/Web_page) written in [HTML](http://en.wikipedia.org/wiki/HTML) and [XHTML](http://en.wikipedia.org/wiki/XHTML), but the language can also be applied to any kind of [XML](http://en.wikipedia.org/wiki/XML) document, including [plain XML](http://en.wikipedia.org/wiki/Plain_Old_XML), [SVG](http://en.wikipedia.org/wiki/Scalable_Vector_Graphics) and [XUL](http://en.wikipedia.org/wiki/XUL).
  + **CSS3** - is divided into several separate documents called "modules". Each module adds new capabilities or extends features defined in CSS 2, over preserving backward compatibility. Work on CSS level 3 started around the time of publication of the original CSS 2 recommendation.
* **JQuery -** is a fast, small, and feature-rich JavaScript library. It makes things like HTML document traversal and manipulation, event handling, animation, and Ajax much simpler with an easy-to-use API that works across a multitude of browsers. With a combination of versatility and extensibility, jQuery has changed the way that millions of people write JavaScript.
* **JQuery Mobile -** jQuery mobile framework takes the "write less, do more" mantra to the next level: Instead of writing unique apps for each mobile device or OS, the jQuery mobile framework allows you to design a single highly-branded web site or application that will work on all popular smartphone, tablet, and desktop platforms.
* These tools were used to develop our mobile applications.
* The main reason for this is so we can effectively target multiple platforms with a single code base. Any browser, including mobile Web browsers, can render the HTML pages created with these technologies.

**ASP .NET MVC 4**

The ASP.NET MVC framework provides an alternative to the ASP.NET Web Forms pattern for creating ASP.NET MVC-based Web applications. The ASP.NET MVC framework is a lightweight, highly testable presentation framework that (as with Web-forms-based applications) is integrated with existing ASP.NET features, such as master pages and membership-based authentication so you get all the power of the core .NET framework. This is useful if you are already familiar with ASP.NET Web Forms because all the libraries that you already use are available in ASP.NET MVC 4 as well.

ASP.NET MVC 4is a framework for building scalable, standard-based web applications using well-established design pattern and the power of ASP.NET and the .NET Framework.

* **Models**: Model objects are the parts of the application that implement the domain logic. Often, model objects also retrieve and store model state in a database.
* **Views:** Views are the components that display the application's user interface (UI). Typically, this UI is created from the model data. An example would be the edit view of Albums that displays text boxes and a drop-down list based on the current state of an Album object.
* **Controllers:** Controllers are the components that handle user interaction, manipulate the model, and ultimately select a view to render the UI. In an MVC application, the view only displays information; the controller handles and responds to user input and interaction.

The MVC pattern helps you to create applications that separate the different aspects of the application (input logic, business logic, and UI logic), while providing a loose coupling between these elements. This separation helps you manage complexity when you build an application, as it allows you to focus on one aspect of the implementation at a time. In addition, the MVC pattern makes it easy to test applications, also encouraging the use of test-driven development (TDD) for creating applications.

* This tool is primarily used for our front end websites.
* The main reason for this is because of its testability, its separation of concerns feature, its extensibility and the fact that it is released under an open source. Basically, this tool allows us to have absolute control over the HTML Mark-up.

**MS SQL Server 2012**

MS SQL Server 2012 as the foundation of the cloud-ready information platform, SQL Server 2012 will help organizations unlock breakthrough insights across the organization as well as quickly build solutions and extend data across on-premises and public cloud backed by mission critical confidence. It is a relational database management system ([RDBMS](http://searchsqlserver.techtarget.com/definition/relational-database-management-system)) designed for the [enterprise](http://searchwinit.techtarget.com/definition/enterprise) environment. Like its predecessors, SQL Server 2012 comprises a set of programming extensions to enhance the Structured Query Language ([SQL](http://searchsqlserver.techtarget.com/definition/SQL)), a standard interactive and programming language for getting information from and updating a [database](http://searchsqlserver.techtarget.com/definition/database).

Microsoft SQL Server 2012, which supplants SQL Server 2008 R2, offers new capabilities, notable among them the following.

* Column Store indexes: Read-only indexes that group data, streamlining the processing of large data warehouse queries.
* Support for Windows Server Core: This is a stripped-down version that places a far lower demand on computer resources than a full install does.
* Power View: Makes it possible to generate [mash-ups](http://whatis.techtarget.com/definition/mash-up) of business intelligence ([BI](http://searchbusinessanalytics.techtarget.com/definition/self-service-business-intelligence-BI)) reports.
* Enhanced Auditing: Users can customize their audit logs to accommodate a wider range of events with greater flexibility than was previously possible.
* Always On: Users can fail over multiple databases and read secondary copies, enhancing disaster recovery (DR) operations.
* Distributed Replay: A workload can be taken from a production server and played on another server to test it under realistic conditions before deploying it.
* This tool is used for the database of our application.
* The reasons for this is because it offers Business Intelligence to help companies analyze business data, an Always On availability and uptime enhancement, Contained Databases for managing databases as a group and a quick-query tool called Column Store Index.

**Entity Framework**

The Microsoft® ADO.NET Entity Framework is an Object/Relational Mapping (ORM) framework that enables developers to work with relational data as domain-specific objects, eliminating the need for most of the data access plumbing code that developers usually need to write. Using the Entity Framework, developers issue queries using LINQ, then retrieve and manipulate data as strongly typed objects. The Entity Framework’s ORM implementation provides services like change tracking, identity resolution, lazy loading, and query translation so that developers can focus on their application-specific business logic rather than the data access fundamentals.

High-level capabilities of the Entity Framework:

* Works with a variety of database servers (including Microsoft SQL Server, Oracle, and DB2)
* Includes a rich mapping engine that can handle real-world database schemas and works well with stored procedures
* Provides integrated Visual Studio tools to visually create entity models and to auto-generate models from an existing database. New databases can be deployed from a model, which can also be hand-edited for full control
* Provides a Code First experience to create entity models using code. Code First can map to an existing database or generate a database from the model.
* Integrates well into all the .NET application programming models including ASP.NET, Windows Presentation Foundation (WPF), Windows Communication Foundation (WCF), and WCF Data Services (formerly ADO.NET Data Services).

The Entity Framework is built on the existing ADO.NET provider model, with existing providers being updated additively to support the new Entity Framework functionality. Because of this, existing applications built on ADO.NET can be carried forward to the Entity Framework easily with a programming model that is familiar to ADO.NET developers.

* This tool is mainly used in organizing Object Relational Mapping or Objects (classes) to Database (tables) Mapping.
* The reason for this is because it allows us to easily bind the data to control and manage relationships between tables. The framework can provide an additional layer of abstraction allowing us more freedom in how the data is presented without spending a lot of time on storage and access.

**C#**

C# is a type-safe, object-oriented language that is simple yet powerful, allowing programmers to build a breadth of applications. Combined with the .NET Framework, Visual C# enables the creation of Windows applications, Web services, database tools, components, controls, and more.

C# (pronounced "C sharp") is a programming language that is designed for building a variety of applications that run on the .NET Framework. C# is simple, powerful, type-safe, and object-oriented. The many innovations in C# enable rapid application development while retaining the expressiveness and elegance of C-style languages.

Visual C# is an implementation of the C# language by Microsoft. Visual Studio supports Visual C# with a full-featured code editor, compiler, project templates, designers, code wizards, a powerful and easy-to-use debugger, and other tools. The .NET Framework class library provides access to many operating system services and other useful, well-designed classes that speed up the development cycle significantly.

* This language is used for the scripting of the server side of the application and a primarily tool that used for transferring information to and from the database.
* The main reason for this is because it has a great amount of tiny details that helps us when using IDEs like Visual Studio, Sharp Develop, and Mono develop. We get tool write code much more robust, clean, faster and comfortably than when doing it in other languages.