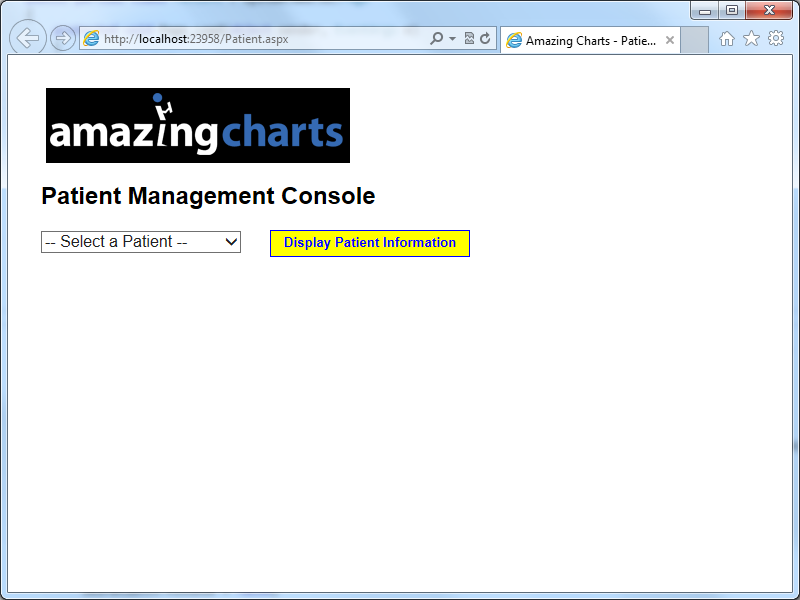
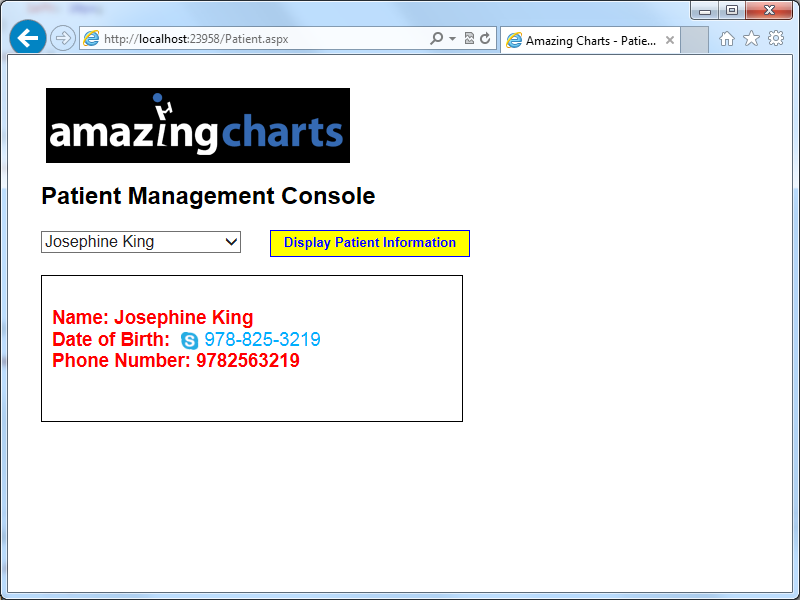
Output for Amazing Charts Sample Application

**Candidate: Martin D. Chicknavorian**

Upon initial client start, the following screen will appear:



After a patient has been selected from the dropdownlist and the yellow button clicked, you will see this screen:



All of the code has been attached to this submittal. Please note that I ran the project utilizing the ASP.NET Development Server, under port 8000. You may or may not have success if you attempt to import the code into Visual Studio and run it; however, it is likely that there will have to be configuration changes in order for it to run successfully. In this case, I used Visual Studio 2012 (.NET 4.5) and EF 6.1.

Below is a schematic of the program design. I loaded the sample data into a SQL Server 2008 database and used Entity Framework 6.1 as the ORM to map the objects to .NET types. The data was then stored in a repository from where it was retrieved by a Management Layer. The WCF web service exposed all of the methods via a host server, which was an ASP.NET application project. The client was also written in ASP.NET. Given that the spec stated that all actions must take place on the server side, no asynchronous code was written. As a result, ASP.NET was the most expedient UI to employ. If the spec called for a thick client, I most likely would have used a pure HTML UI with third-party controls (Telerik or Kendo) and used AJAX calls to fetch the data.

Thank you very much, and I look forward to speaking with you personally!

Marty Chicknavorian

**AmazingCharts.mdf**

**ACModel.edmx**

**IPatientsRepository.cs**

**IPatientsRepository.cs**

**IPatientManagement.cs**

**PatientManagement.cs**

**EF**

**REPOSITORY LAYER**

**MANAGEMENT LAYER**

**Database**

**6.1**

**HOST SERVER**

**CLIENT (UI)**

**ASP.NET**