# Massive Interactive .NET candidates coding challenge

## Problem statement

Using the provided input data representing an undirected graph, you are required to:

1. Load the graph into a database
2. Build a service layer to access the graph nodes
3. Build a thin client for graph visualization
4. Implement a function to find the shortest path between any two nodes

### Input data

You are provided with a set of XML files to be used for input data. Each file represents a single graph node in the following format:

<node>

<id>XXX</id>

<label>Some label</label>

<adjacentNodes>

<id>YYY</id>

<id>ZZZ</id>

</adjacentNodes>

</node>

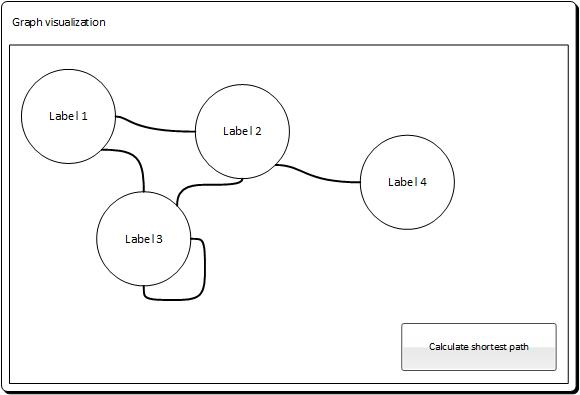
1. ID – a unique string identifying the node
2. Label – human readable text to be displayed in UI
3. Adjacent Nodes – IDs of nodes adjacent to current

Node relationships could be described in uni- or bidirectional mode. Since we are operating on an undirected graph, bidirectional relationships should be treated the same way as a unidirectional one.

Input files are located in the root of a directory and their order is not guaranteed.

Sample set of input files is provided in input.zip

### UI requirements

The client application UI/UX should look as follows:

The application should preserve the original screen size when more nodes are added, and display scrollbars when required.

After selecting any two nodes and clicking “Calculate shortest path”, the application should show the shortest path between the selected nodes, and change/highlight the selected nodes border color.

### Solution components

Your solution should contain the following main components. You are free to create extra projects/libraries within the solution for better modularity and/or re-usability.

##### Data loader

Data loader is a simple **console application** which parses input folder and loads data into the database.

The loader should be able to re-execute over the same folder again when new node files are added or existing files removed.

##### Web Services

Web Services should be a **web application** containing a set of **WCF services** for both data loader and client app. Services should be split into at least 3 categories:

1. Data management – to be used by Data Loader
2. Front-end oriented – to provide data for thin client application
3. Domain specific – used to implement a path finding algorithm

##### Client

Client application could be either a **Web,** **WPF** or **Sliverlight** application as long as it follows the thin client approach. You are expected to implement most data mining and path traversing logic in a service-oriented manner at Service Layer web app.

##### Database

You are free to pick a database engine of your choice as long as:

1. The instructions are provided for database installation and schema creation
2. The database engine could run on Windows machine and installed from a simple package
3. The database engine is available in free version and/or trial mode.

Database could be a classic RDBMS or modern NoSQL.

### Output package requirements

The solution has to be provided as a Git repository including all the implementation commits history. Please follow the **frequent commit** practice while working on the task so that your local Git repository indicates reasonable milestones of your implementation.

You are more than welcome to share your Git repository on a free service such as GitHub or BitBucket.

The repository should contain:

1. Source code with no binaries.
   1. It should be buildable using plain Visual Studio
   2. In case you need to use external libraries, please add them as Nuget
2. Installation and deployment instructions for apps and components

### Coding and development standards

You are expected to apply industry best practices and techniques such as:

1. Unit testing
2. Inversion of control and dependency injection
3. Source code documentation

### Technologies and tools

Please use the following technologies and tools:

1. .NET 3.5+
2. Visual Studio 2012+. In case you do not have a license, please download an [express version](http://www.visualstudio.com/en-us/products/visual-studio-express-vs.aspx)
3. IIS 7+