**Minesight**

**Evaluation**

**Application**

**Version 1.0.0.0**

**Table of Contents**

[Building the Application 3](#_Toc429037720)

[How to build the ‘Minesight’ solution? 3](#_Toc429037721)

[How to start the ‘Minesight’ console application? 3](#_Toc429037722)

[Visual Studio 2013 3](#_Toc429037723)

[Command Line 3](#_Toc429037724)

[Command Line Parameters 3](#_Toc429037725)

[Application Defaults 3](#_Toc429037726)

[Example: 3](#_Toc429037727)

[Story-1: to query for the closest point id in my point set compared to a query point 4](#_Toc429037728)

[Acceptance Criteria 4](#_Toc429037729)

[1. Acceptance Criteria 4](#_Toc429037730)

[2. and 3. Acceptance Criteria 4](#_Toc429037731)

[4. Acceptance Criteria 5](#_Toc429037732)

[5. Acceptance Criteria 5](#_Toc429037733)

[Story-2: be able to shift and translate all my points by a vector 5](#_Toc429037734)

[Acceptance Criteria 6](#_Toc429037735)

[1. Acceptance Criteria 6](#_Toc429037736)

[2. Acceptance Criteria 6](#_Toc429037737)

[Story-3: be able to specify a number of points as closest point 6](#_Toc429037738)

[Acceptance Criteria 6](#_Toc429037739)

[1. Acceptance Criteria 7](#_Toc429037740)

[2. Acceptance Criteria 7](#_Toc429037741)

[Known issues and caveats 7](#_Toc429037742)

# Building the Application

## How to build the ‘Minesight’ solution?

* Open the file ‘Minesight.sln’ with Visual Studio 2013 and build it.
* Run the NUnit test to make sure everything is working.

## How to start the ‘Minesight’ console application?

### Visual Studio 2013

* After building the solution in Visual Studio 2013 click the ‘Start’ or ‘Debug’ button.  
  *This will start the application using the following default options.*

### Command Line

* You can also start the application from a command window.
  + Open a command window
  + Press the Windows key + R and type
  + In the search window type: cmd
  + Click ‘Ok’
  + Change into the folder containing the binaries
    - In the command window type: cd <folder>  
      *e.g.* cd C:\Minesight\Evaluation.Console\bin\Debug\
  + Start the application by typing: Evaluation.Console.exe  
    *This will start the application using the following default options.*

## Command Line Parameters

The Evaluation.Console.exe application accepts the following options:

|  |  |  |  |
| --- | --- | --- | --- |
| **Option (short)** | **Option  (long)** | **Description** | **Example** |
| -s | -source | The name of the CSV file containing points. | –s Points.csv |
| -q | -queryPoint | The query point’s coordinates. | –q=”1.0 2.0 3.0” |
| -m | -moveByVector | The move/shift by vector’s coordinates. | –m=”1.0 2.0 3.0” |
| -n | -numberOfClosestPoints | The number of closest points ids to return. | -n 1 |
| -v | -verbose | Show additional information | -v |

### Application Defaults

The application is using the following default options:

|  |  |
| --- | --- |
| * –s Points.csv * –q=”0.0 0.0 0.0” | * –m=”0.0 0.0 0.0” * -p 1 |

### Example:

Evaluation.Console.exe -s Points.csv –q=”1.0 2.0 3.0” –m=”4.0 5.0 6.0” -n 5

# Story-1: to query for the closest point id in my point set compared to a query point

As a User, I want to query for the closest point id in my point set compared to a query point, so that I find locations for building my drill plan.

It is important for me to be able to enter a co-ordinate and compare it against all my known points in the database, and locate the closest point in that database.

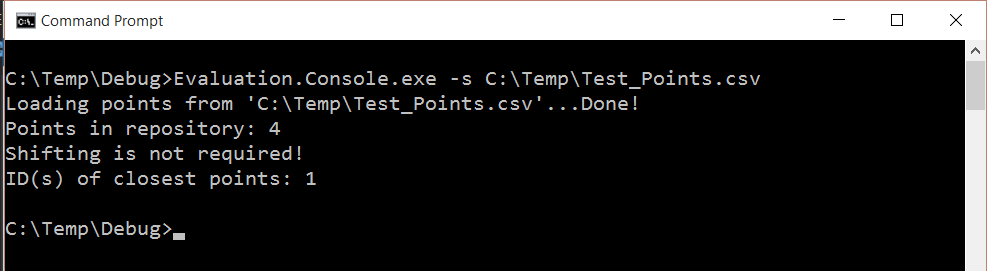
## Acceptance Criteria

1. I expect to be able to specify a source for my points (see Appendix I: Source File Format).
2. I expect to be able to specify a query point.
3. I expect to be able to use positive or negative values of the query point. o i.e. X=1000,Y=10,Z=12 or X=-12, Y=2000, Z=-1201
4. I expect that the point ID returned will be the closest point to the query point.
5. I expect that this is documented.

#### 1. Acceptance Criteria

The user can specify the CSV file to use as the source of points by using the command line option –s or –source followed by the filename. For example:

Evaluation.Console.exe –s C:\Temp\Test\_Points.csv



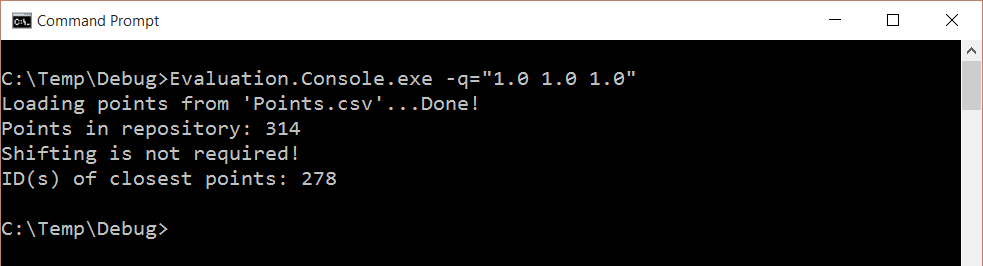
#### 2. and 3. Acceptance Criteria

The user can specify the query point by using the command line option –q or –queryPoint followed by the X, Y, Z coordinates of the point separated by spaces. For example:

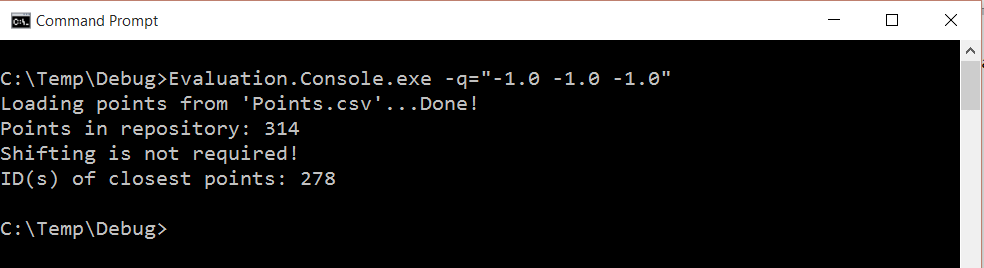
Evaluation.Console.exe –q=”1.0 1.0 1.0”

Or

Evaluation.Console.exe –q=”-1.0 -1.0 -1.0”

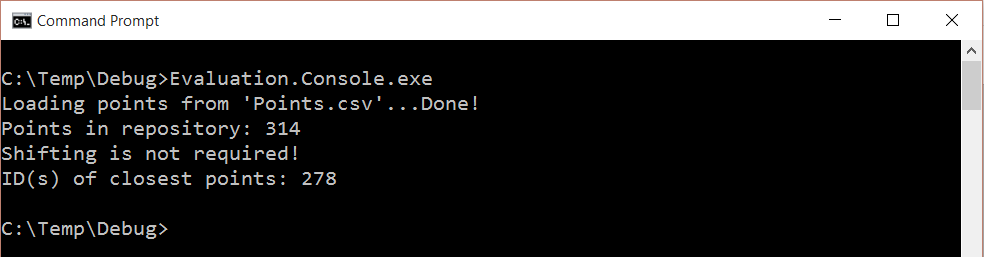


Or



#### 4. Acceptance Criteria

By default the application is returning the ID of closed point to the query point. The example below returns the closest point for the default query point (X=0.0, Y=0.0, Z=0.0) and the test CSV file Points.csv. The closet point ID is 278.



#### 5. Acceptance Criteria

This is the documentation.

# Story-2: be able to shift and translate all my points by a vector

As a User, I want to be able to translate all my points by a vector, so that I can apply a primitive adjustment to my coordinate system to match my query point’s coordinate system.

The point I am querying might be in a different coordinate system than the point data itself, I need a way to shift that data.

## Acceptance Criteria

1. I expect to be able to enter an X, Y, and Z coordinate, and have my entire source data shift by that.
2. I expect that this is documented.

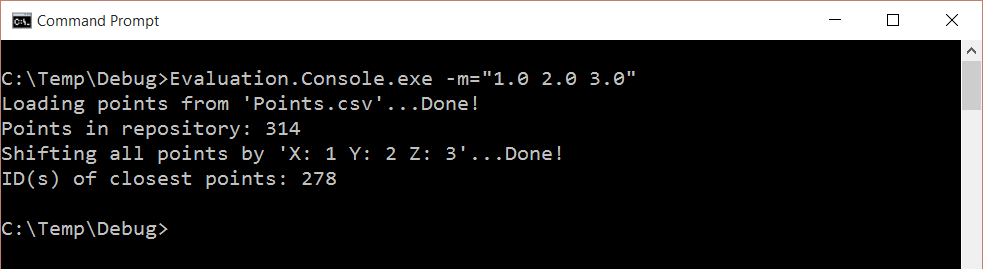
#### 1. Acceptance Criteria

The user can specify the shift vector by using the command line option –m or –moveByVector followed by the X, Y, Z coordinates of the point separated by spaces. For example:

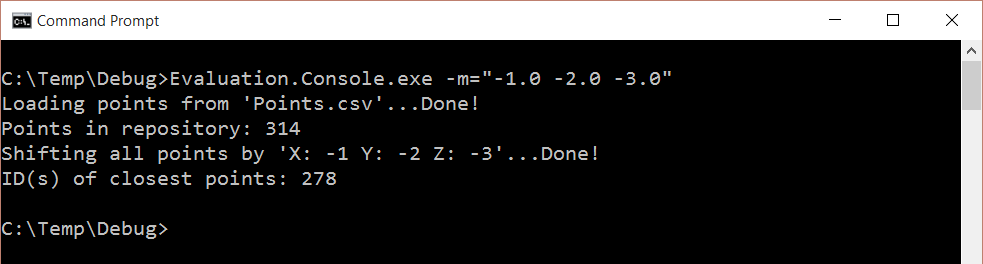
Evaluation.Console.exe –m=”1.0 2.0 3.0”

Or

Evaluation.Console.exe –m=”-1.0 -2.0 -3.0”



Or



#### 2. Acceptance Criteria

This is the documentation.

# Story-3: be able to specify a number of points as closest point

As a User, I want to be able to specify a number of points as closest point, so that I can select the best point to work from.

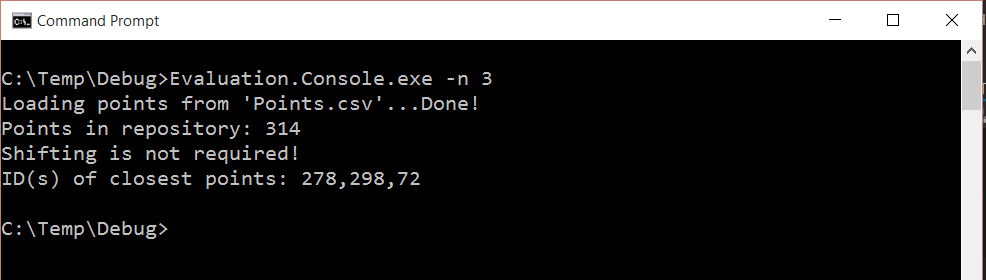
## Acceptance Criteria

1. I expect to be able to specify the number of points to report as closest.
2. I expect that this is documented.

#### 1. Acceptance Criteria

The user can specify the number of closest point ids to be displayed by using the command line option –n or –numberOfClosestPoints. For example:

Evaluation.Console.exe –n 3



#### 2. Acceptance Criteria

This is the documentation.

# Known issues and caveats

* The command line help option -?, -h, -HELP, -Help is currently not working.