|  |
| --- |
| Gastaldo TS2020 Production |
| TS2020KMLtoCSV Converter |
| Progetto per applicazione in C# |

|  |
| --- |
| Daniele Gastaldo  23 settembre 2020 |

SOMMARIO

[1 Introduzione 2](#_Toc51753501)

[1.1 Nome progetto 2](#_Toc51753502)

[1.2 Obiettivi progetto 2](#_Toc51753503)

[1.3 Framework utilizzato 2](#_Toc51753504)

[1.4 Linguaggio utilizzato 2](#_Toc51753505)

[1.5 Tipo di app 2](#_Toc51753506)

[2 Struttura dei files kml 3](#_Toc51753507)

[2.1 Xml Namespaces 3](#_Toc51753508)

[2.2 Radice del file kml 3](#_Toc51753509)

[2.3 Marker puntuali 3](#_Toc51753510)

[2.4 Marker lineari 3](#_Toc51753511)

[2.5 Marker poligonali 4](#_Toc51753512)

# Introduzione

## Nome progetto

TS2020KMLtoCSVConverter

## Obiettivi progetto

Scrivere un programma informatico capace di trasformare uno o più files di markers geografici in formato kml (estensione del formato di files proprietario di Google e utilizzato per importare ed esportare i dati geografici da Google Maps e Google Earth Pro) in uno o più files in formato csv utilizzabili successivamente con il blueprint editor di Railworks Train Simulator 2020 (RWTS2020 o TS2020).

## Framework utilizzato

Versione 0: .Net Core Framework (perché in questo modo l’app può essere distribuita con poco sforzo su ambienti di diverse piattaforme).

## Linguaggio utilizzato

Versione 0: C# (C-sharp).

## Tipo di app

Versione 0.1: App console di .Net Core Framework (scritta in C#).

# Struttura dei files kml

In questo paragrafo sono riportati quindi gli appunti riguardanti la struttura dei files kml (così come vengono generati da Google Earth Pro). Tuttavia, ho riportato qui solo quelle parti di tale struttura che sono d’interesse per il progetto KMLToCSVConverter, tralasciando di analizzarla completamente. La motivazione di tale scelta è che un tipico file kml contiene molti elementi che, pur essendo necessari a Google Earth Pro (che li genera e deve essere in grado di interpretarli correttamente) non sono in alcun modo utili agli scopi che tale progetto si pone.

## Xml Namespaces

Lo schema di questa struttura sarà utilizzato per estrarre dal file in esame in valore del namespace “xlmns”, il namespace cui appartengono gli elementi di nostro successivo interesse.

1. RADICE: un singolo elemento di nome “kml”;
   1. ATTRIBUTO: un singolo namespace xml di nome “xmlns”.

## Radice del file kml

Questo schema di struttura contiene gli elementi dell’albero di file kml da percorrere, a partire dalla radice del documento kml, per arrivare al livello dei singoli placemark, il livello di nostro interesse.

1. RADICE: un singolo elemento di nome “Document”;
   1. ELEMENTO: un singolo elemento di nome “Folder”;
      1. ELEMENTO: uno o più elementi di nome “Placemark”.

## Marker puntuali

Lo schema seguente è quello di un marker puntuale (le puntine da disegno sulla mappa di Google Earth Pro).

1. RADICE: un singolo elemento di nome “Placemark”;
   1. ELEMENTO: un singolo elemento di nome “name”;
   2. ELEMENTO: un singolo elemento di nome “Point”;
      1. ELEMENTO: un singolo elemento di nome “coordinates”.

## Marker lineari

Di seguito riportiamo la struttura di dati per un marker lineare (le linee rette colorate della mappa di Google Earth Pro).

1. RADICE: un singolo elemento di nome “Placemark”;
   1. ELEMENTO: un singolo elemento di nome “name”;
   2. ELEMENTO: un singolo elemento di nome “LineString”;
      1. ELEMENTO: un singolo elemento di nome “coordinates”.

## Marker poligonali

Infine, riportiamo lo schema della struttura di un marker poligonale (le forme colorate della mappa di Google Earth Pro).

1. RADICE: un singolo elemento di nome “Placemark”;
   1. ELEMENTO: un singolo elemento di nome “name”;
   2. ELEMENTO: un singolo elemento di nome “Polygon”;
      1. ELEMENTO: un singolo elemento di nome “OuterBoundaryIs”;
         1. ELEMENTO: un singolo elemento di nome “LinearRing”;
            1. ELEMENTO: un singolo elemento di nome “coordinates”.