



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería Informática



**TFG del Grado en Ingeniería
Informática**

SurveyingPointCode

**Automatización del proceso
de delineación a partir de
datos de un Levantamiento
Topográfico.**



Presentado por
José Eduardo Risco Sánchez-Cortés
en la Universidad de Burgos — 6 de mayo
de 2019

Tutores: Dr. César García-Osorio
y Dr. Carlos López Nozal



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería Informática



D. César García-Osorio y D. Carlos López Nozal, profesores del departamento de Ingeniería Civil, área de Lenguajes y Sistemas Informáticos.

Exponen:

Que el alumno D. José Eduardo Risco Sánchez-Cortés, con DNI 40441256B , ha realizado el Trabajo final de Grado en Ingeniería Informática titulado: "SurveyingPointCode - Automatización del proceso de delineación a partir de datos de un Levantamiento Topográfico"

Y que dicho trabajo ha sido realizado por el alumno bajo la dirección del que suscribe, en virtud de lo cual se autoriza su presentación y defensa.

En Burgos, 6 de mayo de 2019

Vº. Bº. del Tutor:

Vº. Bº. del tutor:

D. César García-Osorio

D. Carlos López Nozal

Resumen

En el campo de la Topografía, hay trabajos cuyo objetivo es obtener el plano de un levantamiento topográfico. Este trabajo se divide en dos partes, trabajo de campo y delineación. Es importante entender que la relación entre las dos partes es fundamental para poder reducir al máximo los recursos invertidos tanto en la parte del trabajo de campo como en la de delineación.

Con este trabajo, se pretende que mediante una codificación definida por el usuario en el trabajo de campo, (para cada punto medido), se optimice la medición en campo y posteriormente, la codificación permita la generación automática del plano final, en un archivo de **CAD** (con formato **DXF**), teniendo en cuenta la distribución de los puntos en diferentes capas, generación de líneas, curvas, simbología, etc.

Con este propósito se ha desarrollado la aplicación Surveying-PointCode, aplicación Web que realiza la conversión de un archivo que contiene los datos de un levantamiento topográfico a un archivo **DXF**.

Descriptores

Levantamiento topográfico, CAD, DXF, código punto, delineación, aplicación Web.

Abstract

In the field of Topography, there are projects whose objective is to obtain the plan of a topographic survey. This project is divided in two parts; survey work and delineation. It is important to understand that the relationship between these two parts will be fundamental to be able to reduce to the maximum the resources invested in both the survey work and the delineation work.

The main purpose of this work is to optimize the data collection through a coding defined by the user in the field work (for each measured point). Subsequently the coding will allow the automatic generation of the final plan, in a **CAD** file. (with **DXF** format), taking into account the distribution of points in different layers, generation of lines, curves, symbology, etc.

For this purpose, the SurveyingPointCode application has been developed, a Web application that performs the conversion of a file that contains the data from a topographic survey to a **DXF** file

Keywords

Topographical survey, CAD, DXF, point code, delineation, Web application.

Índice general

Índice general	III
Índice de figuras	IV
Índice de tablas	V
Introducción	1
1.1. Introducción	1
1.2. Contenido del proyecto	2
Objetivos del proyecto	5
2.1. Objetivos generales	5
2.2. Objetivos técnicos	6
2.3. Objetivos personales	6
Conceptos teóricos	7
3.1. Levantamiento topográfico	7
3.2. Archivo DXF. Elementos del dibujo	7
3.3. Codificación de los puntos	11
Técnicas y herramientas	17
Aspectos relevantes del desarrollo del proyecto	19
Trabajos relacionados	21
Conclusiones y Líneas de trabajo futuras	23

Índice de figuras

3.1. Elementos gráficos	9
3.2. Capas en un programa de CAD	10
3.3. Símbolos creados como entidades bloques de CAD	11
3.4. Representación de una línea	12
3.5. Representación de una curva Spline	13
3.6. Representación de un rectángulo, un cuadrado y un círculo . . .	14
3.7. Representación de como se crearía una linea formada por puntos inaccesibles	15

Índice de tablas

3.1. Caracteres obligatorios y significado	16
--	----

Introducción

1.1. Introducción

Uno de los trabajos que se realiza habitualmente en el campo de la Topografía y Cartografía, es la representación en un plano de los datos obtenidos en un levantamiento topográfico. Pongamos como ejemplo que se nos pide como producto final, un plano basado en un levantamiento topográfico de una zona de una ciudad para realizar el estudio de unas futuras obras. El plano debe contener todos los elementos existentes que sean de interés para este estudio, como, por ejemplo: edificios, aceras, viales, redes de saneamiento, abastecimiento, eléctricas, mobiliario urbano, etc.

Este trabajo se realizará en dos fases; la primera, el trabajo de campo, donde se realiza el levantamiento topográfico, adquisición de datos, que consiste en la medición georreferenciada de todos los elementos que sean de interés y que deben aparecer en el plano. La segunda fase es la de delineación, en la que se ‘dibuja’ el plano. Hay que aclarar que, en la fase del levantamiento topográfico, solo se miden puntos, y es en la fase de delineación donde tenemos que crear todos los elementos, como líneas, curvas, símbolos, etc

Un problema importante al que nos enfrentamos es la gran cantidad de tiempo y recursos que invertimos en gestionar los datos obtenidos en campo para obtener un plano, alargando en exceso el desarrollo y entrega de un proyecto. Con este planteamiento surge la necesidad de crear una aplicación que consiga automatizar este trabajo el máximo posible y así obtener un ahorro de tiempo y de recursos, fundamental en nuestros proyectos. Puede parecer a priori un trabajo sencillo y rápido de realizar, con una herramienta de CAD, uniendo puntos manualmente para crear líneas, u otros elementos, claro está, si tienes un levantamiento topográfico con 30 puntos y recuerdas

como lo has hecho en campo, pero cuando tienes un levantamiento con 1000 puntos, e intervienen varias personas en él, se puede convertir en un trabajo muy complicado e insufrible de realizar. Una herramienta así no existe mercado, algunos programas como AutoCAD disponen de algún módulo a mayores del programa, que permiten hacer algo similar, pero con un resultado muy pobre respecto a lo que aquí se plantea.

La principal ventaja que aporta el uso de esta aplicación es el importante ahorro de tiempo y recursos que obtenemos en la realización de un trabajo. Por ejemplo, un trabajo de campo con 1000 puntos, con una codificación compleja, es decir con múltiples y diferentes elementos, una persona con habilidad en manejo de aplicaciones CAD, puede invertir alrededor de 2 horas. Esta aplicación pretende que ese proceso sea automático e inmediato. No debemos olvidar que el uso de una codificación totalmente aleatoria por el usuario a la hora de realizar el trabajo de campo también va a permitir obtener un importante ahorro de recursos, en esa parte del trabajo. Otras ventajas son:

- Es una aplicación web, por lo que los usuarios no necesitan instalar nada en sus equipos y pueden acceder desde cualquier lugar.
- Facilidad de uso, no hacen falta conocimientos técnicos en el uso de herramientas CAD.

1.2. Contenido del proyecto

La estructura de la memoria es la siguiente:

- **Introducción:** breve descripción del contenido del trabajo y de la estructura de la memoria.
- **Objetivos del proyecto:** objetivos que se persiguen con la realización del proyecto.
- **Conceptos teóricos:** breve explicación de los conceptos necesarios para el desarrollo de la solución propuesta.
- **Técnicas y herramientas:** presentación de las técnicas metodológicas y las herramientas de desarrollo que se han utilizado para llevar a cabo el proyecto.
- **Aspectos relevantes del desarrollo:** descripción de los aspectos más importantes ocurridos a lo largo del desarrollo del proyecto.

- **Trabajos relacionados:** pequeño resumen de los trabajos y proyectos ya realizados en el campo del proyecto en curso.
- **Conclusiones y líneas de trabajo futuras:** resumen acerca de los conocimientos adquiridos y posibles aspectos de mejora o expansión de la solución aportada.

Además, se han desarrollado los siguientes anexos:

- **Plan de Proyecto Software:** explicación sobre la planificación temporal llevada a cabo, así como un estudio de viabilidad del proyecto.
- **Especificación de requisitos:** descripción de la fase de análisis de la aplicación.
- **Especificación de diseño:** explicación y descripción del diseño de la aplicación.
- **Documentación técnica de programación:** recoge los aspectos más relevantes relacionados con el código fuente (estructura, compilación, instalación, ejecución, pruebas, etc.).
- **Documentación de usuario:** manual de usuario que permita conocer el funcionamiento la aplicación.

Por último, hay que indicar que el proyecto se encuentra disponible en el siguiente enlace:

<https://github.com/EduardoRisco/SurveyingPointCode>

Objetivos del proyecto

A continuación, se detallan los diferentes objetivos que han motivado la realización del proyecto.

2.1. Objetivos generales

- Desarrollar una aplicación web que permita obtener un archivo DXF a partir de los datos de un levantamiento topográfico.
- Definir un tipo de codificación para los puntos medidos en campo, que sea interpretada por la aplicación, automatice el proceso de dibujo y mejore el rendimiento en el trabajo de campo.
- Permitir al usuario crear una cuenta.
- Permitir al usuario subir : archivos de campo, archivos de configuración personalizados de la transformación y archivos DXF, con símbolos creados por el usuario.
- Permitir al usuario modificar o elegir, en la interfaz, los nombres de las capas de CAD, colores y símbolos existentes, y asociarlos a los códigos de campo
- Permitir al usuario elegir la versión de CAD para generar el archivo DXF.
- Permitir al usuario almacenar en su equipo, el archivo DXF generado.

2.2. Objetivos técnicos

- Desarrollar una aplicación web utilizando el framework **Flask**.
- Utilizar **PLY**¹ (Python Lex-Yacc), como analizador sintáctico, para reconocer la validez de los archivos de entrada.
- Utilizar **ezdxf**², para leer, modificar y crear archivos DXF.
- Utilizar **Bootstrap**, para crear el sitio web.
- Utilizar **TinyColor**³ para poder definir los colores con los estándares de AutoCAD.
- Utilizar **PostGIS**⁴ como sistema de base de datos.
- Realizar test unitarios.
- Hacer uso de la herramienta **CODEBEAT** para comprobar la calidad del código.
- Desplegar la aplicación usando **Docker**.
- Utilizar **Git** como sistema de control de versiones distribuido junto con la plataforma **GitHub**.
- Aplicar la metodología ágil **Scrum** para el desarrollo del software.
- Utilizar **ZenHub** como herramienta de gestión de proyectos.

2.3. Objetivos personales

- Mejorar los recursos invertidos y facilitar el trabajo, a la hora de realizar un levantamiento topográfico.
- Realizar el proyecto como un trabajo real, aplicando los conocimientos adquiridos en la realización del Grado.
- Utilizar herramientas y metodologías demandadas en el mercado laboral; Git, desarrollo Web, Docker, etc.

¹PLY: <https://pypi.org/project/ply/>

²ezdxf: <https://ezdxf.readthedocs.io/en/latest/index.html>

³TinyColor: <https://github.com/bgrins/TinyColor>

⁴PostGIS: <https://postgis.net/>

Conceptos teóricos

En este apartado se explican conceptos los básicos necesarios para entender el contexto en que se desarrolla este proyecto, respecto a lo que aquí se pretende, haciendo hincapié en la topografía y el dibujo en CAD.

3.1. Levantamiento topográfico

Un levantamiento topográfico consiste en la medición de distintos puntos, mediante cálculos topográficos, donde se obtienen las coordenadas de estos, generalmente las coordenadas X,Y y Z .Estas coordenadas nos indican su posición exacta en el espacio. El software que tienen los equipos de topografía; como teodolitos, estaciones totales o GPS, permiten asociar a cada punto un texto en el momento de medir el punto y así, el punto queda registrado con ese campo. Este campo de texto, que de aquí en adelante llamaremos código de punto, es la clave de este proyecto. El algoritmo diseñado debe saber interpretar este código de punto. Como veremos más adelante, este nos puede indicar una o múltiples cosas a la vez, lo que conlleva mayor dificultad a la hora de definir el formato que debe tener este. Tiene que aportar la máxima información posible, pero a la vez tiene que ser muy sencillo en su estructura y de una longitud lo más corta posible. No es muy viable tener que estar mucho tiempo tecleando, ya que se perdería mucho tiempo y alargaría en exceso el trabajo de campo.

3.2. Archivo DXF. Elementos del dibujo

DXF (acrónimo del inglés Drawing Exchange Format) es un formato de archivo para dibujos de diseño asistido por computadora, creado por Auto-

CAD. Formato muy extendido y utilizado por la mayoría de programas de CAD, como por ejemplo: AutoCAD, MicroStation, FreeCAD, DraftSight, . . . y también por programas de GIS, como: ArcGIS, QGIS, gvSIG, . . .

Para entender de forma sencilla los elementos de los que está compuesto un archivo DXF, solo se tratarán aquellos elementos que se utilizan en esta aplicación, de otra forma, la explicación sería muy extensa. Simplificando, se procede a hacer una diferenciación entre los elementos que queremos dibujar que llamaremos elementos gráficos, y los elementos no gráficos.

Elementos gráficos

- **Puntos:** En este caso es el elemento básico, ya que en una medición topográfica solo se obtienen puntos. El resto de los elementos que se exponen a continuación están basados en la posición de estos puntos.
- **Líneas:** Están formadas por una sucesión de puntos, pueden ser abiertas o cerradas. Deben tener un punto inicial y otro final. Los cuadrados y rectángulos, en esta aplicación, se pueden englobar en este grupo, al ser polígonos o líneas cerradas.
- **Splines:** Curvas interpoladas en una sucesión de puntos, pueden ser abiertas o cerradas. Deben tener un punto inicial y otro final.
- **Círculos:** En esta aplicación los definiremos por un punto, que es el centro del círculo y un radio.
- **Textos:** Es un texto cuya posición en el dibujo se define, por la posición de un punto y una distancia a este.

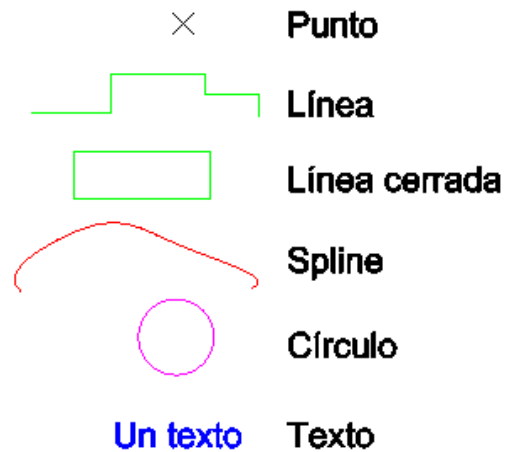


Figura 3.1: Elementos gráficos

Elementos no gráficos

- **Capas:** Sirven para organizar mejor los objetos en nuestro dibujo ayudando a un mejor manejo a la hora de mover, copiar o identificarlos en el plano. Estas son un aspecto fundamental en el desarrollo del dibujo para poder obtener un plano bien organizado y con excelente presentación. En un buen trabajo, es casi imprescindible el uso de capas. Las capas son una forma de clasificar objetos en CAD, si tienes un plano muy extenso con muchos objetos, la forma más fácil de identificarlos es mediante su capa, cada objeto dibujado en CAD se le asociara una capa y esa capa la puedes crear asignándole un nombre, un color, entre otras cosas. Es la mejor forma de identificar cierto objeto dentro de tu dibujo. En esta aplicación, las capas tendrán un nombre y un color.

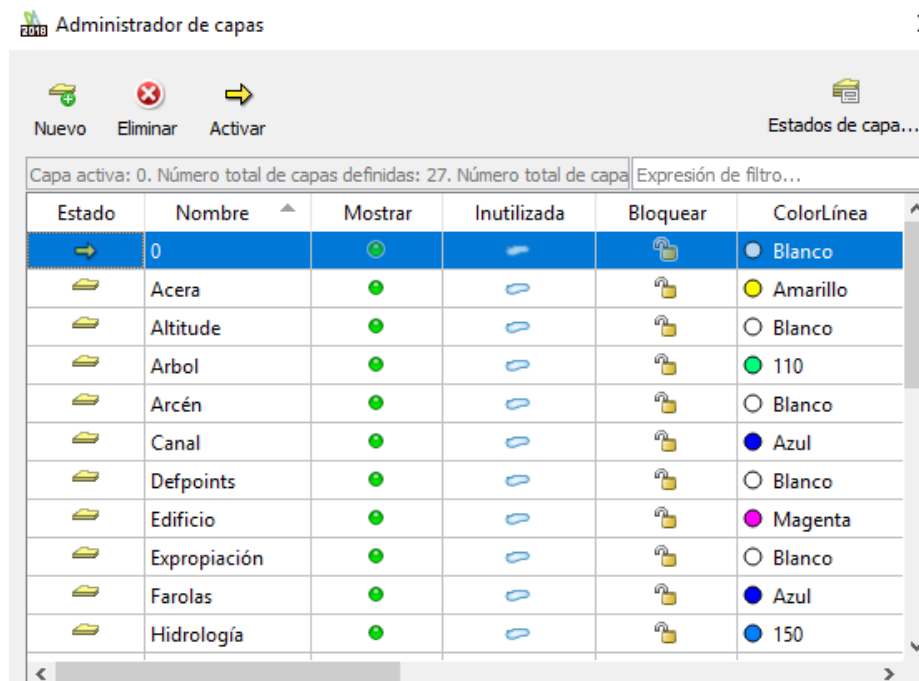


Figura 3.2: Capas en un programa de CAD

- Bloques:** Un bloque en CAD es un conjunto de objetos (llamados entidades), agrupados como un todo. Es decir, que podemos dibujar líneas, arcos, círculos, cada uno con propiedades distintas que los demás y luego invocar un comando para “juntarlos” a todos bajo un mismo nombre y asignarle un punto de inserción. En esta aplicación se pretende que el usuario pueda subir un archivo DXF personalizado con sus bloques, y poder asignarlo a diferentes puntos del levantamiento topográfico a través del código de punto



Figura 3.3: Símbolos creados como entidades bloques de CAD

3.3. Codificación de los puntos

Como se ha mencionado anteriormente, la codificación de los puntos es la parte clave para el desarrollo de este proyecto, ya que en ella se va a basar la aplicación a la hora de crear los elementos del dibujo. Debe poder crear líneas y splines, interpretando donde empiezan y acaban. Crear círculos, cuadrados y rectángulos, insertar símbolos o bloques en determinados puntos y, por último, asociar cada elemento su capa correspondiente, con su color asignado. Todo esto, si eres un usuario con conocimientos en manejo de aplicaciones CAD, se haría de forma manual en su mayor parte. Con una buena definición del código de punto, se tratará de automatizar todo este proceso. Como ya sabemos, el código debe ser lo mas sencillo posible y a la vez aportar la máxima información posible, dejando que el usuario tenga libertad para elegir los códigos, salvo algunas partes, que deben de tener una sintaxis concreta.

A continuación, veremos algunos ejemplos de codificación para distintos elementos del dibujo:

- Se quiere dibujar la línea que define el arcén de una carretera, podríamos escribir en el código del punto; 'ARCEN', 'ARC', 'A', o lo que

quisiéramos. Como se pretende que el código sea lo más sencillo posible, tomamos como mejor opción 'A'. Para definir que la línea empieza en un punto concreto, ese texto debe ir seguido obligatoriamente del carácter 'I'. Un ejemplo de cómo sería una línea de arcén formada por 4 puntos es: ('A I', 'A', 'A', 'A').

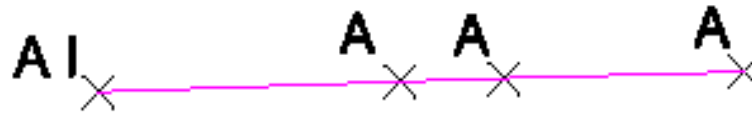


Figura 3.4: Representación de una línea

- Siguiendo con el ejemplo anterior, ahora queremos que dibuje una curva tipo spline. Para definir que la línea empieza en un punto concreto y que sea spline, ese texto debe ir seguido obligatoriamente del carácter 'IC' y el resto de los puntos de esa curva, el texto inicial debe ir seguido obligatoriamente del carácter 'C'. Un ejemplo de cómo sería una línea de arcén tipo spline formada por 4 puntos es: ('A IC', 'AC', 'AC', 'AC').

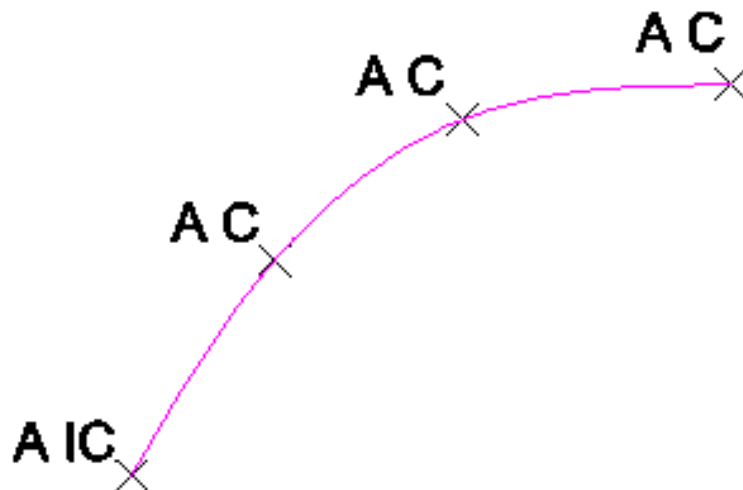


Figura 3.5: Representación de una curva Spline

- Se quiere dibujar un cuadrado, por ejemplo, una tapa de registro perteneciente a la red de saneamiento. El código de punto debe comenzar obligatoriamente por 'TC', seguido de la descripción que queramos asignarle, en este caso como es la red de saneamiento usaremos 'SAN'. Así finalmente, el código para ese punto queda 'TC SAN'. Como se pretende optimizar también el trabajo de campo, un cuadrado solo se definirá mediante 2 puntos y la aplicación al interpretar el código, deberá siempre dibujar un cuadrado a la derecha de esos dos puntos. La explicación es sencilla, si en un trabajo hay que medir 500 tapas de registro, no es lo mismo medir 1000 puntos que 2000. Con esta codificación nos hemos ahorrado la mitad del tiempo en el trabajo de campo.
- Lo mismo que en el caso anterior, para dibujar rectángulos. El código de punto debe comenzar obligatoriamente por 'TR', seguido de la descripción que queramos asignarle, en este caso como es la red de saneamiento usaremos 'SAN'. Así finalmente, el código para ese punto queda 'TR SAN'. Como se pretende optimizar también el trabajo de campo, un rectángulo solo se definirá mediante 3 puntos y la aplicación al interpretar el código, deberá siempre dibujar un rectángulo.

- Se quiere dibujar un círculo, por ejemplo, una tapa de registro perteneciente a la red eléctrica. El código de punto debe comenzar obligatoriamente por 'TX', seguido de la descripción que queramos asignarle, en este caso como es la red de eléctrica usaremos 'RE'. En este código necesitaremos también incluir en radio de la circunferencia, por ejemplo 0.5 metros, por lo que al final el código queda 'TX 0.5 RE'.

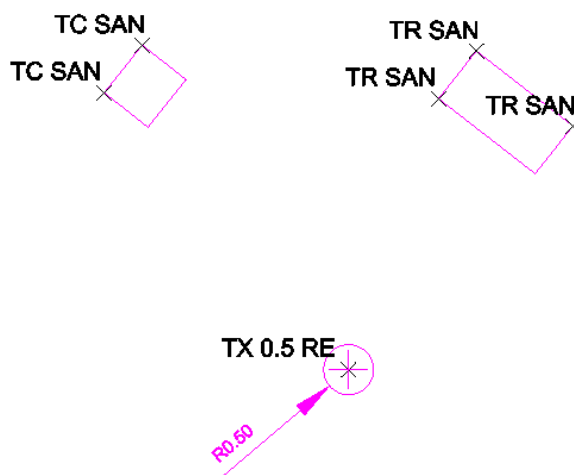


Figura 3.6: Representación de un rectángulo, un cuadrado y un círculo

- Por último, planteamos una opción más complicada y suele ser habitual en los trabajos de campo, la representación de elementos que realmente no se han medido en campo, porque pueden no ser accesibles. Queremos representar la línea de la fachada de un edificio, donde existen dos puntos inaccesibles, que no podemos medir topográficamente, pero conocemos su longitud (por ejemplo, hemos extraído estos datos de un plano catastral). Como es un edificio, podríamos representarlo con el texto 'E'. La siguiente secuencia de 4 puntos con su codificación, construirá la línea final formada por 6 puntos, ('E I', 'E 1.5 -2', 'E', 'E'). El código el punto 2, 'E 1.5 -2', indica que cuando la línea llegue a ese punto, debe continuar perpendicular a su derecha durante 1.5 metros, después, 2 metros, también en perpendicular, hacia su izquierda, y por último debe unirse con el siguiente punto que forma la línea.

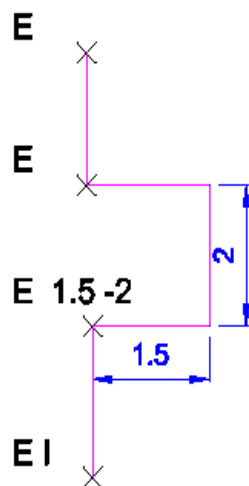


Figura 3.7: Representación de como se crearía una línea formada por puntos inaccesibles

Otra gran ventaja que aporta esta codificación es que los puntos en campo que forman un elemento no estás obligado a medirlos de forma consecutiva, lo que supone un importantísimo ahorro de tiempo. Vamos a explicar esto con un ejemplo. Se pretende dibujar el plano de una carretera de 50 km, las líneas que nos interesan son; el eje de la carretera y las líneas de los arceles. No podemos plantearnos el medir las líneas de forma consecutiva, por ejemplo, medir el arcén derecho, volver por el eje y, por último, el arcén izquierdo. Si hiciéramos esto, recorreríamos 150 km. Esta aplicación es capaz de crear distintas líneas, aunque no estén medidas en campo en un orden consecutivo. Una solución de codificación podría ser la siguiente secuencia de puntos: ('AD I', 'E I', 'AI I', 'AI', 'E', 'AD', 'E', 'AD', 'AI', ...), donde 'AD' es el arcén derecho, 'E' el eje, 'AI' el arcén izquierdo. Así solo recorreríamos la carretera una sola vez.

Vemos que este tipo de codificación definida no solo aporta la ventaja de generar automáticamente el plano, si no que, el tiempo ahorrado en el trabajo de campo, es la mayor ventaja. Esta parte del trabajo es la que tiene mayores costes y, por lo tanto, más peso a la hora de decidirse por un proyecto.

Character	Significado
'I'	Punto inicio de línea
'IC'	Punto inicio de curva
'C'	Punto perteneciente a una curva
'TC'	Punto perteneciente a un cuadrado
'TR'	Punto perteneciente a un rectángulo
'TX'	Punto que define un círculo
'+'	En puntos inaccesibles, giro a la derecha 90°
'-'	En puntos inaccesibles, giro a la izquierda 90°

Tabla 3.1: Caracteres obligatorios y significado

Técnicas y herramientas

Esta parte de la memoria tiene como objetivo presentar las técnicas metodológicas y las herramientas de desarrollo que se han utilizado para llevar a cabo el proyecto. Si se han estudiado diferentes alternativas de metodologías, herramientas, bibliotecas se puede hacer un resumen de los aspectos más destacados de cada alternativa, incluyendo comparativas entre las distintas opciones y una justificación de las elecciones realizadas. No se pretende que este apartado se convierta en un capítulo de un libro dedicado a cada una de las alternativas, sino comentar los aspectos más destacados de cada opción, con un repaso somero a los fundamentos esenciales y referencias bibliográficas para que el lector pueda ampliar su conocimiento sobre el tema.

Aspectos relevantes del desarrollo del proyecto

Este apartado pretende recoger los aspectos más interesantes del desarrollo del proyecto, comentados por los autores del mismo. Debe incluir desde la exposición del ciclo de vida utilizado, hasta los detalles de mayor relevancia de las fases de análisis, diseño e implementación. Se busca que no sea una mera operación de copiar y pegar diagramas y extractos del código fuente, sino que realmente se justifiquen los caminos de solución que se han tomado, especialmente aquellos que no sean triviales. Puede ser el lugar más adecuado para documentar los aspectos más interesantes del diseño y de la implementación, con un mayor hincapié en aspectos tales como el tipo de arquitectura elegido, los índices de las tablas de la base de datos, normalización y desnormalización, distribución en ficheros³, reglas de negocio dentro de las bases de datos (EDVHV GH GDWRV DFWLYDV), aspectos de desarrollo relacionados con el WWW... Este apartado, debe convertirse en el resumen de la experiencia práctica del proyecto, y por sí mismo justifica que la memoria se convierta en un documento útil, fuente de referencia para los autores, los tutores y futuros alumnos.

Trabajos relacionados

Este apartado sería parecido a un estado del arte de una tesis o tesina. En un trabajo final grado no parece obligada su presencia, aunque se puede dejar a juicio del tutor el incluir un pequeño resumen comentado de los trabajos y proyectos ya realizados en el campo del proyecto en curso.

Conclusiones y Líneas de trabajo futuras

Todo proyecto debe incluir las conclusiones que se derivan de su desarrollo. Éstas pueden ser de diferente índole, dependiendo de la tipología del proyecto, pero normalmente van a estar presentes un conjunto de conclusiones relacionadas con los resultados del proyecto y un conjunto de conclusiones técnicas. Además, resulta muy útil realizar un informe crítico indicando cómo se puede mejorar el proyecto, o cómo se puede continuar trabajando en la línea del proyecto realizado.