



Escuela Politécnica de Ingeniería de Ferrol

M. U. en Eficiencia Energética y Sostenibilidad

Trabajo Fin de Máster

TFM N°: XXX

TÍTULO: TÍTULO DEL TRABAJO FINAL

AUTOR: NOMBRE DEL AUTOR DEL TRABAJO

TUTOR: NOMBRE DEL TUTOR

FECHA: NOVIEMBRE DE 2022

Título Título del Trabajo Final

Índice general

Peticionario Escuela Politécnica de Ingeniería de Ferrol
Rúa Mendizábal, s/n, Campus de Esteiro,
15403, Ferrol

Fecha Noviembre de 2022

Autor El Alumno

Fdo. Nombre del autor del Trabajo

Índice general

1	Índice general	1
2	Memoria	5
2.1	Objeto	7
2.2	Alcance	7
2.3	Antecedentes	7
2.4	Normas y referencias	7
2.4.1	Disposiciones legales y normas aplicadas	7
2.4.2	Bibliografía	8
2.4.3	Programas de cálculo	8
2.4.4	Otras referencias	8
2.5	Definiciones y abreviaturas	8
2.6	Requisitos de diseño	9
2.7	Análisis de las soluciones	9
2.8	Resultados finales	9
2.9	Orden de prioridad entre los documentos	9
3	Anexos	11
3.1	Contenido	13
3.1.1	Documentación de partida	13
3.1.2	Cálculos	13
3.1.3	Anexos en función del ámbito de aplicación del Trabajo Final	13
3.2	Código de MatLab	13
3.2.1	DAQ_Start	13
3.2.2	DAQ_Stop	14
3.2.3	DAQ_Read	14
3.2.4	DAQ_Write	15
3.3	Código de Arduino	17
3.4	Código VHDL	21
4	Estudios con entidad propia	25
5	Otros anexos que justifiquen y aclaren conceptos expresados en el Trabajo Final	29
6	Planos	33
6.1	Contenido	35
	Nombre del plano A4	37
	Nombre del plano A3	39
	Nombre del plano A2	41
Noviembre de 2022		Índice general
		2

7	Pliego de condiciones	43
7.1	Contenido	45
8	Mediciones	47
8.1	Contenido	49
9	Presupuesto	51
9.1	Contenido	53

Título Título del Trabajo Final

Memoria

Peticionario Escuela Politécnica de Ingeniería de Ferrol
Rúa Mendizábal, s/n, Campus de Esteiro,
15403, Ferrol

Fecha Noviembre de 2022

Autor El Alumno

Fdo. Nombre del autor del Trabajo

Índice Memoria

2.1	Objeto	7
2.2	Alcance	7
2.3	Antecedentes	7
2.4	Normas y referencias	7
2.4.1	Disposiciones legales y normas aplicadas	7
2.4.2	Bibliografía	8
2.4.3	Programas de cálculo	8
2.4.4	Otras referencias	8
2.5	Definiciones y abreviaturas	8
2.6	Requisitos de diseño	9
2.7	Análisis de las soluciones	9
2.8	Resultados finales	9
2.9	Orden de prioridad entre los documentos	9

La Memoria es uno de los documentos que constituyen el Trabajo Final y asume la función fundamental de nexo de unión entre todos ellos. Tiene como misión justificar las soluciones adoptadas, su adecuación a la normativa legal aplicable y, conjuntamente con los Planos y el Pliego de Condiciones, debe describir de forma unívoca el objeto del Trabajo Final. La Memoria debe ser claramente comprensible, no sólo por profesionales especialistas sino por terceros y especialmente en lo que se refiere a los objetivos del Trabajo Final, las alternativas estudiadas, sus ventajas e inconvenientes, y las razones que han conducido a la solución elegida. La Memoria se iniciará con un índice que hará referencia a cada uno de los capítulos y apartados que la componen, con el fin de facilitar su utilización. En los puntos que siguen se indica el título y contenido de los capítulos y apartados en los que descompone la Memoria del Trabajo Final.

2.1. Objeto

En este capítulo de la Memoria se debe indicar el objetivo del Trabajo Final y su justificación.

2.2. Alcance

En este capítulo se debe indicar el ámbito de aplicación del Trabajo Final.

2.3. Antecedentes

En este capítulo de la Memoria se deben indicar los condicionantes de partida, por ejemplo cuando el lugar, obra, instalación, infraestructura, actividad, etc. objeto del Trabajo Final dispone de permisos o legalizaciones previas, cuando se trata de una ampliación, rehabilitación o reforma, cuando constituye un cambio de actividad o uso, etc.

2.4. Normas y referencias

En este capítulo figurarán los siguientes apartados. Se incluye aquí un pequeño ejemplo sobre como se deberían de citar los libros [2], artículos [1], congresos [4], sitios web [3] y normas [Norma1].

2.4.1. Disposiciones legales y normas aplicadas

En este apartado se deben incluir las referencias de las disposiciones legales (leyes, reglamentos, ordenanzas, normas de obligado cumplimiento por su inclusión en disposiciones legales, etc.) y de las normas que se deben tener en cuenta para la realización del Trabajo Final.

[Norma1] Normalización Española. Criterios generales para la elaboración formal de los documentos que constituyen un proyecto técnico. UNE 157001:2014. 2014/06/18.

2.4.2. Bibliografía

En este apartado se debe incluir un listado con las páginas web, libros, revistas u otros textos que el autor considere de interés para justificar las soluciones adoptadas en el Trabajo Final.

- [1] Roque Aguado, José Luis Casteleiro-Roca, David Vera, and José Luis Calvo-Rolle. A hybrid intelligent model to predict the hydrogen concentration in the producer gas from a downdraft gasifier. *International Journal of Hydrogen Energy*, Early publish:–, 2022.
- [2] Roque Aguado Molina, José Luis Casteleiro-Roca, Esteban Jove, Francisco Zayas Gato, Héctor Quintián, and José Luis Calvo-Rolle. *Hidrógeno y su almacenamiento: el futuro de la energía eléctrica*. Servizo de Publicacións, Universidade da Coruña, A Coruña, Spain, 2021.
- [3] Universidade da Coruña. Escuela politécnica de ingeniería de ferrol. url = <http://www.udc.es/epef>. Accessed: 2022-11-01.
- [4] Juan Aurelio Montero-Sousa, Esteban Jove, José Luis Casteleiro-Roca, Héctor Quintián, José Luis Calvo-Rolle, Héctor Aláiz-Moretón, and Tomás González-Ayuso. Sistema híbrido para la predicción del funcionamiento de una celda de combustible basada en hidrógeno, empleada en el almacenamiento de energía. In *XL Jornadas de Automática; Septiembre 4–6*, pages 200–210, Ferrol, Spain, 2019.

2.4.3. Programas de cálculo

Se debe incluir la relación de programas, modelos u otras herramientas utilizadas para desarrollar los diversos cálculos del Trabajo Final.

2.4.4. Otras referencias

En este apartado se debe incluir cualquier información que, no estando relacionada en los apartados anteriores, se consideren de interés para la comprensión y materialización del Trabajo Final. Por ejemplo, se puede usar este apartado para citar las referencias de donde se hayan sacado las figuras [RefFig1].

- [RefFig1] Francisco Zayas Gato, Héctor Quintián, Esteban Jove, José-Luis Casteleiro-Roca, and José Luis Calvo-Rolle. *Diseño de controladores PID*. Servizo de Publicacións, Universidade da Coruña, A Coruña, Spain, 2020.

2.5. Definiciones y abreviaturas

Se deben relacionar todas las definiciones, abreviaturas, etc. que se han utilizado y su significado.

2.6. Requisitos de diseño

En este capítulo de la Memoria se deben describir las bases y datos de partida establecidos por el Tutor y los que derivan de:

- El emplazamiento, y su entorno socio-económico y ambiental.
- Los estudios realizados encaminados a la definición de la solución adoptada.
- Las interfaces con otros sistemas o elementos externos al Trabajo Final u otros que condicionan las soluciones técnicas del mismo.

2.7. Análisis de las soluciones

En este capítulo de la Memoria se deben indicar las distintas alternativas estudiadas, qué caminos se han seguido para llegar a ellas, ventajas e inconvenientes de cada una y cuál es la solución finalmente elegida, junto con su justificación en función de los requisitos de la legislación y reglamentación aplicables.

2.8. Resultados finales

En este capítulo se debe describir el producto, obra, edificio, instalación y servicio según la solución elegida, indicando cuáles son sus características definitorias y haciendo referencia a los planos y otros elementos del Trabajo Final que lo definen.

2.9. Orden de prioridad entre los documentos

En este capítulo de la Memoria el autor del Trabajo Final, frente a posibles discrepancias, debe establecer el orden de prioridad de los documentos del Trabajo Final. Si no se especifica otra cosa, el orden de prioridad debe ser el siguiente:

1. Planos.
2. Pliego de Condiciones.
3. Presupuesto.
4. Memoria.

Título Título del Trabajo Final

Anexos

Peticionario Escuela Politécnica de Ingeniería de Ferrol
Rúa Mendizábal, s/n, Campus de Esteiro,
15403, Ferrol

Fecha Noviembre de 2022

Autor El Alumno

Fdo. Nombre del autor del Trabajo

Índice Anexos

3.1	Contenido	13
3.1.1	Documentación de partida	13
3.1.2	Cálculos	13
3.1.3	Anexos en función del ámbito de aplicación del Trabajo Final	13
3.2	Código de MatLab	13
3.2.1	DAQ_Start	13
3.2.2	DAQ_Stop	14
3.2.3	DAQ_Read	14
3.2.4	DAQ_Write	15
3.3	Código de Arduino	17
3.4	Código VHDL	21

El documento Anexos está formado por la documentación que desarrolla, justifica o aclara apartados específicos de la Memoria u otros documentos del Trabajo Final.

3.1. Contenido

Se debe iniciar con un índice que hará referencia a cada uno de los capítulos y apartados que lo componen, con el fin de facilitar su utilización. Este documento debe contener los anexos necesarios (según proceda en cada caso) correspondientes a:

3.1.1. Documentación de partida

Este anexo debe incluir una copia de la propuesta inicial de asignación del Trabajo Final.

3.1.2. Cálculos

Este anexo o anexos tienen como misión justificar las fórmulas aplicadas, las soluciones adoptadas y, conjuntamente con los documentos Planos y el Pliego de Condiciones, debe describir de forma unívoca el objeto del Trabajo Final. Debe contener las hipótesis de partida, los criterios y procedimientos de cálculo, así como los resultados finales base del dimensionamiento o comprobación de los distintos elementos que constituyen el Trabajo Final.

3.1.3. Anexos en función del ámbito de aplicación del Trabajo Final

En este capítulo deben incluirse los anexos relativos a:

- Seguridad (prevención de incendios, sanidad, radiaciones, pública concurrencia, etc.).
- Medio ambiente (acústica, residuos, emisiones, etc.).
- Eficiencia energética.
- Emplazamiento del proyecto, Geotécnicos, etc.
- Gestión de residuos.
- Códigos de programación.

Se va a hacer una mención especial a los códigos de programación, ya que la plantilla se ha preparado para poder incluir fácilmente código tanto de MatLab, Arduino o incluso VHDL. Se muestra a continuación un ejemplo de cada tipo.

3.2. Código de MatLab

Se incluye a continuación el código de MatLab para realizar la comunicación entre MatLab y Arduino Uno y usar este último como tarjeta de adquisición de datos.

3.2.1. DAQ_Start

```
1 function []=DAQ_Start(Port)
2 %% This function configures the DAQ communication port
3
```

```

4 % The Arduino_COM is the global variable that connects the
   different scripts
5 global Arduino_COM
6
7 % In the there is no input argument to select the communication
   port, the
8 % script uses 'COM3' as standard configuration
9 if nargin==0
10     Port='COM3';
11 end
12
13 % The COM number depends on each system; it is necessary to
   confirm in the
14 % "Devices administrator"
15 Arduino_COM=serial(Port,'BaudRate',115200);
16
17 % COM port is open to communicate with Arduino board
18 fopen(Arduino_COM);
19
20 end

```

3.2.2. DAQ_Stop

```

1 function []=DAQ_Stop()
2 %% This function finalice the uses of the DAQ (close the
   communication)
3
4 % The Arduino_COM is the global variable that connects the
   different scripts
5 global Arduino_COM
6
7 % Clear the outputs channels
8 DAQ_Write(); % Ensure the output channels are resetted
9
10 % The COM port is closed
11 fclose(Arduino_COM);
12
13 % The variable of the DAQ is deleted
14 delete(Arduino_COM);
15
16 end

```

3.2.3. DAQ_Read

```

1 function [Input_0 , Input_1 , Input_2 , Input_3 , Input_4 , Input_5]=
   DAQ_Read()
2 %% This function read the value in the inputs of the DAQ

```

```
3
4 % The Arduino_COM is the global variable that connects the
   different scripts
5 global Arduino_COM
6
7 Command=uint8(85); % The command to request the values from DAQ
8
9 fwrite(Arduino_COM,Command,'uchar'); % Send the Command to
   Arduino
10
11 Input_0=fscanf(Arduino_COM,'%d');
12 % Arduino reading range is 0@255; the measure is scaled to 0@100
13 Input_0=100*Input_0/255;
14 Input_1=fscanf(Arduino_COM,'%d');
15 % Arduino reading range is 0@255, the measure is scaled to 0@100
16 Input_1=100*Input_1/255;
17 Input_2=fscanf(Arduino_COM,'%d');
18 % Arduino reading range is 0@255; the measure is scaled to 0@100
19 Input_2=100*Input_2/255;
20 Input_3=fscanf(Arduino_COM,'%d');
21 % Arduino reading range is 0@255, the measure is scaled to 0@100
22 Input_3=100*Input_3/255;
23 Input_4=fscanf(Arduino_COM,'%d');
24 % Arduino reading range is 0@255; the measure is scaled to 0@100
25 Input_4=100*Input_4/255;
26 Input_5=fscanf(Arduino_COM,'%d');
27 % Arduino reading range is 0@255, the measure is scaled to 0@100
28 Input_5=100*Input_5/255;
29
30 end
```

3.2.4. DAQ_Write

```
1 function []=DAQ_Write(Output_0,Output_1,Output_2,Output_3,
   Output_4,Output_5)
2 %% This function write de desired values in the outputs of the
   DAQ
3
4 % The Arduino_COM is the global variable that connects the
   different scripts
5 global Arduino_COM
6
7 % If the user don't need all the channels, there will not be
   sufficient
8 % argument in the function
9 switch nargin
```



```
10     case 5
11         Output_5=0;
12     case 4
13         Output_5=0;
14         Output_4=0;
15     case 3
16         Output_5=0;
17         Output_4=0;
18         Output_3=0;
19     case 2
20         Output_5=0;
21         Output_4=0;
22         Output_3=0;
23         Output_2=0;
24     case 1
25         Output_5=0;
26         Output_4=0;
27         Output_3=0;
28         Output_2=0;
29         Output_1=0;
30     case 0
31         Output_5=0;
32         Output_4=0;
33         Output_3=0;
34         Output_2=0;
35         Output_1=0;
36         Output_0=0;
37 end
38
39 % Ensure that the value is inside the operation range
40 if Output_0>100
41     Output_0=100;
42 elseif Output_0<0
43     Output_0=0;
44 end
45 if Output_1>100
46     Output_1=100;
47 elseif Output_1<0
48     Output_1=0;
49 end
50 if Output_2>100
51     Output_2=100;
52 elseif Output_2<0
53     Output_2=0;
54 end
55 if Output_3>100
```

```
56     Output_3=100;
57 elseif Output_3<0
58     Output_3=0;
59 end
60 if Output_4>100
61     Output_4=100;
62 elseif Output_4<0
63     Output_4=0;
64 end
65 if Output_5>100
66     Output_5=100;
67 elseif Output_5<0
68     Output_5=0;
69 end
70
71 % As the range of the values is 0@100, the values are scaled to
    0@255
72 Output_0=round(255*Output_0/100);
73 Output_1=round(255*Output_1/100);
74 Output_2=round(255*Output_2/100);
75 Output_3=round(255*Output_3/100);
76 Output_4=round(255*Output_4/100);
77 Output_5=round(255*Output_5/100);
78
79 Command=uint8(170); % The command to send the values to DAQ
80
81 fwrite(Arduino_COM,Command,'uchar');
82 fwrite(Arduino_COM,Output_0,'uchar');
83 fwrite(Arduino_COM,Output_1,'uchar');
84 fwrite(Arduino_COM,Output_2,'uchar');
85 fwrite(Arduino_COM,Output_3,'uchar');
86 fwrite(Arduino_COM,Output_4,'uchar');
87 fwrite(Arduino_COM,Output_5,'uchar');
88
89 end
```

3.3. Código de Arduino

Se incluye a continuación el código que tiene que tener cargado el Arduino Uno para usarlo como tarjeta de adquisición de datos.

```
1 /* Programa para que el Arduino se comporte como una tarjeta
    de adquisición de datos.
2  * El programa trabaja en conjunto con MatLab, con unos
    scripts diseñados específicamente.
3  * KST-DAQ - Keep Simple Task - Data Adquisition card
4  *
```

```
5 * José Luis Casteleiro Roca - jose.luis.casteleiro@udc.es
6 * Octubre 2020
7 */
8
9 // Definición de los canales de Salida
10 int Output_0 = 3; // Definimos el pin del canal 0 para
    generación - Compatible con el Arduino Uno
11 int Output_1 = 5; // Definimos el pin del canal 1 para
    generación - Compatible con el Arduino Uno
12 int Output_2 = 6; // Definimos el pin del canal 2 para
    generación - Compatible con el Arduino Uno
13 int Output_3 = 9; // Definimos el pin del canal 3 para
    generación - Compatible con el Arduino Uno
14 int Output_4 = 10; // Definimos el pin del canal 4 para
    generación - Compatible con el Arduino Uno
15 int Output_5 = 11; // Definimos el pin del canal 5 para
    generación - Compatible con el Arduino Uno
16
17 // Definición de los canales de Entrada
18 int Input_0 = 0; // Definimos el pin del canal 0 para
    adquisición - Compatible con el Arduino Uno
19 int Input_1 = 1; // Definimos el pin del canal 1 para
    adquisición - Compatible con el Arduino Uno
20 int Input_2 = 2; // Definimos el pin del canal 2 para
    adquisición - Compatible con el Arduino Uno
21 int Input_3 = 3; // Definimos el pin del canal 3 para
    adquisición - Compatible con el Arduino Uno
22 int Input_4 = 4; // Definimos el pin del canal 4 para
    adquisición - Compatible con el Arduino Uno
23 int Input_5 = 5; // Definimos el pin del canal 5 para
    adquisición - Compatible con el Arduino Uno
24
25 // Definimos las variables necesarias
26 int Comando=0; // Valor inicial para el comando 0 - El
    comando se divide en 8 bits según:
27         // R | W | v2 | v1 | v0 | ch2 | ch1 | ch0 - El
            comando básico son los 2 primeros bits
28         // 0 | 0 | x | x | x | x | x | x - Reiniciar
            la tarjeta
29         // 0 | 1 | x | x | x | ch2 | ch1 | ch0 -
            Generar las señales en tantos canales de
            salida como indica (ch2ch1ch0)
30         // 1 | 0 | x | x | x | ch2 | ch1 | ch0 -
            Adquirir las señales de tantos canales de
            salida como indica (ch2ch1ch0)
```

```
31          // 1 | 1 | v2 | v1 | v0 | x | x | x - Envío de
           // la versión del Firmware
32 int Dato=0; // Valor inicial para el Dato 0
33 int ver=
34
35 // Configuración del Arduino
36 void setup() {
37     // Defino los pines de las señales PWM como salida
38     pinMode(Output_0,OUTPUT); // Canal 0
39     pinMode(Output_1,OUTPUT); // Canal 1
40     pinMode(Output_2,OUTPUT); // Canal 2
41     pinMode(Output_3,OUTPUT); // Canal 3
42     pinMode(Output_4,OUTPUT); // Canal 4
43     pinMode(Output_5,OUTPUT); // Canal 5
44     // Definimos un nuevo preescaler para los temporizadores.
           // Esto hará que la frecuencia de la señal PWM sea más alta
45     bitClear(TCCR0B,CS02);
46     bitClear(TCCR0B,CS01);
47     bitSet(TCCR0B,CS00); // Entrada de reloj al contador sin
           // preescaler
48     bitClear(TCCR1B,CS12);
49     bitClear(TCCR1B,CS11);
50     bitSet(TCCR1B,CS10); // Entrada de reloj al contador sin
           // preescaler
51     bitClear(TCCR2B,CS22);
52     bitClear(TCCR2B,CS21);
53     bitSet(TCCR2B,CS20); // Entrada de reloj al contador sin
           // preescaler
54     // Se inicializa la comunicación serie con el ordenador
55     Serial.begin(115200); // La velocidad tiene que ser la
           // misma tanto en el Arduino como en MatLab
56 }
57
58 // Programa principal del Arduino
59 void loop() {
60     // Primero esperaremos a que se reciba algún byte por el
           // puerto serie
61     if (Serial.available()>0){
62         // Leemos el valor recibido
63         Comando=Serial.read();
64         // Comprobamos si lo que queremos es hacer una Generar se
           // ñales
65         if (bitRead(Comando,7)==1) { // Generar señales PWM
66             // Generar señal por Canal 0
67             while (Serial.available()<=0);
```

```
68     Dato=Serial.read(); // Recibimos el dato
        correspondiente al Canal 0
69     analogWrite(Output_0,Dato); // Generamos la señal PWM
        con el valor recibido
70     // Generar señal por Canal 1
71     while (Serial.available()<=0);
72     Dato=Serial.read(); // Recibimos el dato
        correspondiente al Canal 1
73     analogWrite(Output_1,Dato); // Generamos la señal PWM
        con el valor recibido
74     // Generar señal por Canal 2
75     while (Serial.available()<=0);
76     Dato=Serial.read(); // Recibimos el dato
        correspondiente al Canal 2
77     analogWrite(Output_2,Dato); // Generamos la señal PWM
        con el valor recibido
78     // Generar señal por Canal 3
79     while (Serial.available()<=0);
80     Dato=Serial.read(); // Recibimos el dato
        correspondiente al Canal 3
81     analogWrite(Output_3,Dato); // Generamos la señal PWM
        con el valor recibido
82     // Generar señal por Canal 4
83     while (Serial.available()<=0);
84     Dato=Serial.read(); // Recibimos el dato
        correspondiente al Canal 4
85     analogWrite(Output_4,Dato); // Generamos la señal PWM
        con el valor recibido
86     // Generar señal por Canal 5
87     while (Serial.available()<=0);
88     Dato=Serial.read(); // Recibimos el dato
        correspondiente al Canal 5
89     analogWrite(Output_5,Dato); // Generamos la señal PWM
        con el valor recibido
90 }
91 // Comprobamos si lo que queremos es hacer una Adquirir
    señales
92 if (bitRead(Comando,7)==0) { // Adquirir señales analó
    gicas
93     Dato=analogRead(Input_0); // Adquirimos la señal analó
        gica presente en el Canal 0
94     Dato=Dato>>2; // Desplazamos el valor para quedarnos
        con los 8 bits más significativos
95     Serial.println(lowByte(Dato)); // Enviamos el dato
        correspondiente al Canal 0
```

```
96     Dato=analogRead(Input_1); // Adquirimos la señal analó
      gica presente en el Canal 1
97     Dato=Dato>>2; // Desplazamos el valor para quedarnos
      con los 8 bits más significativos
98     Serial.println(lowByte(Dato)); // Enviamos el dato
      correspondiente al Canal 1
99     Dato=analogRead(Input_2); // Adquirimos la señal analó
      gica presente en el Canal 2
100    Dato=Dato>>2; // Desplazamos el valor para quedarnos
      con los 8 bits más significativos
101    Serial.println(lowByte(Dato)); // Enviamos el dato
      correspondiente al Canal 2
102    Dato=analogRead(Input_3); // Adquirimos la señal analó
      gica presente en el Canal 3
103    Dato=Dato>>2; // Desplazamos el valor para quedarnos
      con los 8 bits más significativos
104    Serial.println(lowByte(Dato)); // Enviamos el dato
      correspondiente al Canal 3
105    Dato=analogRead(Input_4); // Adquirimos la señal analó
      gica presente en el Canal 4
106    Dato=Dato>>2; // Desplazamos el valor para quedarnos
      con los 8 bits más significativos
107    Serial.println(lowByte(Dato)); // Enviamos el dato
      correspondiente al Canal 4
108    Dato=analogRead(Input_5); // Adquirimos la señal analó
      gica presente en el Canal 5
109    Dato=Dato>>2; // Desplazamos el valor para quedarnos
      con los 8 bits más significativos
110    Serial.println(lowByte(Dato)); // Enviamos el dato
      correspondiente al Canal 5
111  }
112 }
113 // Si no se recibe ningún byte, no se hace nada; sólo
      esperar
114 }
```

3.4. Código VHDL

Se incluye a continuación un ejemplo de código VHDL.

```
1
2 library IEEE;
3 use IEEE.STD_LOGIC_1164.ALL;
4 use IEEE.STD_LOGIC_ARITH.ALL;
5 use IEEE.STD_LOGIC_unsigned.ALL;
```

```

6
7
8 entity Decod_7seg is
9   Port ( X : in STD_LOGIC_VECTOR (15 downto 0);
10         clr: in STD_LOGIC;
11         CLK: in STD_LOGIC;
12         ce: in STD_LOGIC;
13         a_to_g : out STD_LOGIC_VECTOR (6
14               downto 0);
15         an: out STD_LOGIC_VECTOR (3 downto 0)
16               ;
17         dp : out STD_LOGIC);
18 end Decod_7seg;
19
20 architecture Behavioral of Decod_7seg is
21   —Salida del contador de 2 bits
22   signal s: std_logic_vector(1 downto 0);
23   —Salida del MUX44
24   signal digit: std_logic_vector(3 downto 0);
25
26 begin
27   —Contador de 2 bits
28   CTR2bit: process(clr , CLK)
29     begin
30       if clr='0' then
31         s<="00";
32       elsif CLK'event and CLK
33         ='1' then
34         if ce='1' then
35           s<=s+1;
36         end if;
37       end if;
38     end process CTR2bit;
39   —MUX44
40   Mux44: process (s,X)
41     begin
42       case s is
43         when "00
           " =>
           digit
           <=x(3
           downto

```

```

44         0);
         when "01
           " =>
             digit
             <=x(7

45         downto
           4);
         when "10
           " =>
             digit
             <=x
             (11
             downto
             8);

46         when
           others
             =>
             digit
             <=x
             (15
             downto
             12);

47         end case;
48     end process Mux44;
49 —hex7seg
50     process(digit)
51     begin
52         case digit is
53             when X"0" => a_to_g <= "0000001"; —0
54             when X"1" => a_to_g <= "1001111"; —1
55             when X"2" => a_to_g <= "0010010"; —2
56             when X"3" => a_to_g <= "0000110"; —3
57             when X"4" => a_to_g <= "1001100"; —4
58             when X"5" => a_to_g <= "0100100"; —5
59             when X"6" => a_to_g <= "0100000"; —6
60             when X"7" => a_to_g <= "0001101"; —7
61             when X"8" => a_to_g <= "0000000"; —8
62             when X"9" => a_to_g <= "0000100"; —9
63             when X"A" => a_to_g <= "0001000"; —A
64             when X"B" => a_to_g <= "1100000"; —b
65             when X"C" => a_to_g <= "0110001"; —C
66             when X"D" => a_to_g <= "1000010"; —d
67             when X"E" => a_to_g <= "0110000"; —E
68             when others => a_to_g <= "0111000"; —F
69     end case;

```



```
70     end process ;
71
72     dp <= '1';
73 —ANCODE
74 Ancode : process (s)
75         begin
76             case s is
77                 when "00" => an
78                     <= "1110";
79                 when "01" => an
80                     <= "1101";
81                 when "10" => an
82                     <= "1011";
83                 when "11" => an
84                     <= "0111";
85                 when others => an
86                     <= "1111";
87             end case ;
88         end process Ancode ;
89
90 end Behavioral ;
```

Título Título del Trabajo Final

Estudios con entidad propia

Peticionario Escuela Politécnica de Ingeniería de Ferrol
Rúa Mendizábal, s/n, Campus de Esteiro,
15403, Ferrol

Fecha Noviembre de 2022

Autor El Alumno

Fdo. Nombre del autor del Trabajo

Índice Estudios con entidad propia

Este documento debe contener todos aquellos estudios que deban incluirse en el Proyecto por exigencias legales. Debe comprender, entre otros y sin carácter limitativo, los relativos a:

- Estudio Básico de Seguridad y Salud o Estudio de Seguridad y Salud, según corresponda.
- Estudio de Impacto Ambiental.

Cada anexo debe contener la justificación del cumplimiento de la normativa legal vigente aplicable y, si procede, de las fórmulas aplicadas para el cálculo.

Título Título del Trabajo Final

Otros anexos que justifiquen y aclaren conceptos expresados en el Trabajo Final

Peticionario Escuela Politécnica de Ingeniería de Ferrol
Rúa Mendizábal, s/n, Campus de Esteiro,
15403, Ferrol

Fecha Noviembre de 2022

Autor El Alumno

Fdo. Nombre del autor del Trabajo

Índice Otros anexos que justifiquen y aclaren conceptos expresados en el Trabajo Final

Se pueden incluir:

- Catálogos de los elementos constitutivos del objeto del TFG/TFM.
- Listados.
- Información en soportes lógicos, magnéticos, ópticos o cualquier otro.
- Maquetas o modelos.
- Otros documentos que se juzguen necesarios.

Título Título del Trabajo Final

Planos

Peticionario Escuela Politécnica de Ingeniería de Ferrol
Rúa Mendizábal, s/n, Campus de Esteiro,
15403, Ferrol

Fecha Noviembre de 2022

Autor El Alumno

Fdo. Nombre del autor del Trabajo

Índice Planos

6.1	Contenido	35
	Nombre del plano A4	37
	Nombre del plano A3	39
	Nombre del plano A2	41

El documento Planos es uno de los documentos que constituyen el Trabajo Final y tiene como misión, junto con la Memoria, definir de forma unívoca el objeto del Trabajo Final.

6.1. Contenido

El documento Planos se debe iniciar con un índice que haga referencia a cada uno de los planos que contiene, indicando su ubicación en el documento, con el fin de facilitar su utilización. Cada uno de los planos debe contener la información gráfica, alfanumérica, de códigos y de escala, necesaria para su comprensión y correcta ejecución de lo representado. Los planos y la documentación técnica, en cuanto a principios generales de representación, cajetines, indicaciones, escritura, rotulación, acotación, símbolos gráficos, plegado, listas de elementos, escalas, métodos de proyección, formatos y presentación de los elementos gráficos y gestión de la información técnica asistida por ordenador, deben tener en cuenta lo indicado en las siguientes Normas: UNE 1027, UNE 1032, UNE 1035, UNE 1039, UNE 1089-1, UNE 1089-2, UNE 1135, UNE 1166-1, UNE-EN ISO 3098-0, UNE-EN ISO 3098-2, UNE-EN ISO 3098-3, UNE-EN ISO 3098-4, UNE-EN ISO 3098-5, UNE-EN ISO 3098-6, UNE-EN ISO 5455, UNE-EN ISO 5456-1, UNE-EN ISO 5456-2, UNE-EN ISO 5456-3, UNE-EN ISO 5457, UNE-EN ISO 6433, UNE-EN ISO 10209-2, UNE-EN ISO 11442-1, UNE-EN ISO 11442-2, UNE-EN ISO 11442-3, UNE-EN ISO 11442-4, UNE-EN ISO 81714-1. Todos los planos deben incluir un cuadro de rotulación con la siguiente información:

- Nombre de la Escuela y la Titulación.
- Número del Trabajo Final.
- Título del Trabajo Final.
- Título, escala y número de Plano.
- Nombre completo del autor y firma.
- Fecha de la convocatoria en la cual se entrega el Trabajo Final.

1

2

3

4

A

B

C

D

E

F



GRADO EN INGENIERÍA ELECTRÓNICA INDUSTRIAL Y AUTOMÁTICA

Escuela Politécnica de Ingeniería de Ferrol



UNIVERSIDADE DA CORUÑA

TFG Nº: 770511A000

TÍTULO DEL TFG:

TÍTULO COMPLETO DEL TFG

TÍTULO DEL PLANO:

TÍTULO COMPLETO DEL PLANO

FECHA: Noviembre 2022

ESCALA: 1:0000



AUTOR:

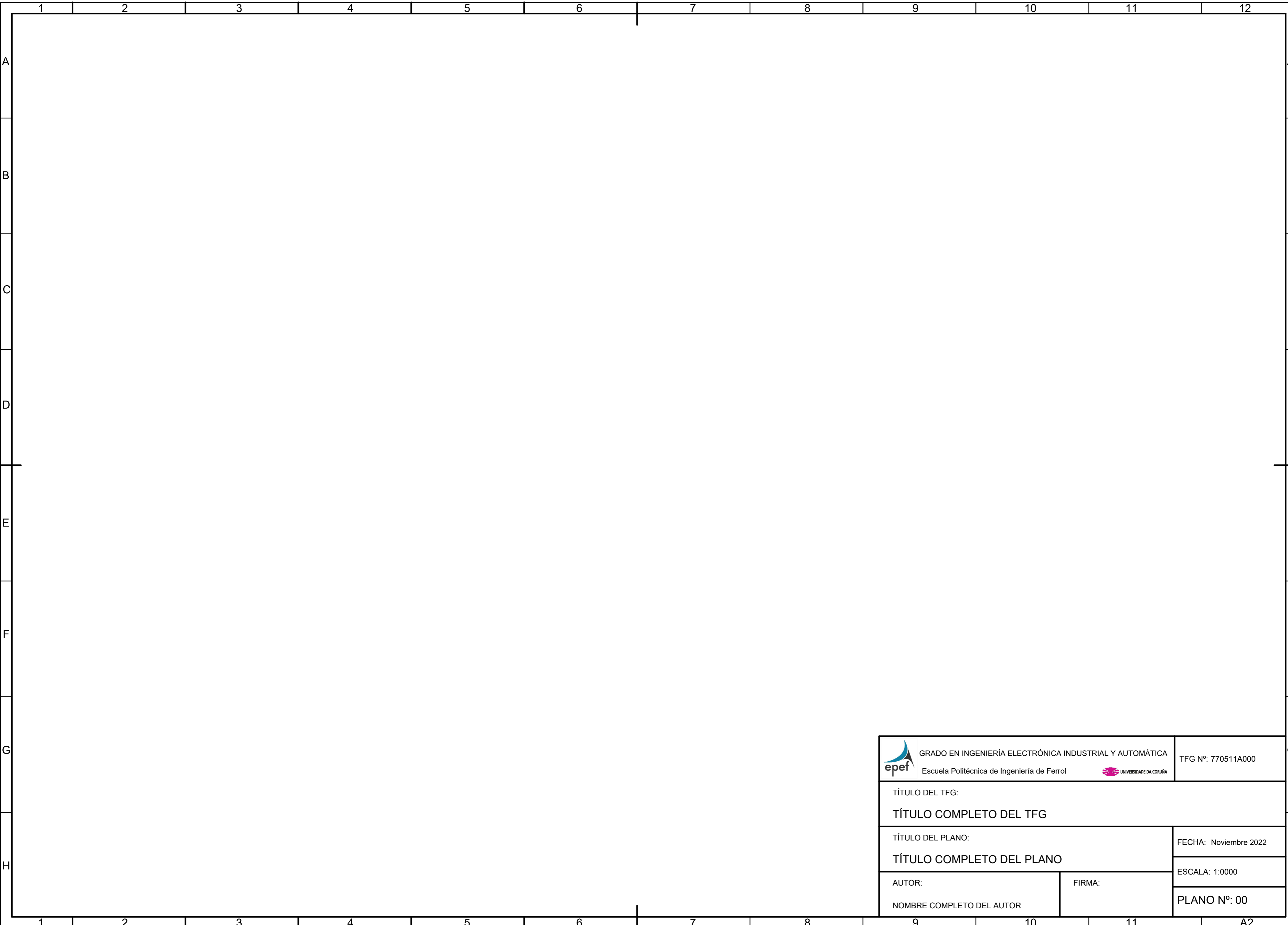
FIRMA:

NOMBRE COMPLETO DEL AUTOR

PLANO Nº: 00

A4

1	2	3	4	5	6	7	8	
A							A	
B							B	
C							C	
D							D	
E	 GRADO EN INGENIERÍA ELECTRÓNICA INDUSTRIAL Y AUTOMÁTICA Escuela Politécnica de Ingeniería de Ferrol					 UNIVERSIDADE DA CORUÑA		TFG Nº: 770511A000
F	TÍTULO DEL TFG: TÍTULO COMPLETO DEL TFG							
	TÍTULO DEL PLANO: TÍTULO COMPLETO DEL PLANO					FECHA: Noviembre 2022		
	AUTOR:				FIRMA:		ESCALA: 1:0000	
	NOMBRE COMPLETO DEL AUTOR						PLANO Nº: 00	
1	2	3	4	5	6	7	A3	



 GRADO EN INGENIERÍA ELECTRÓNICA INDUSTRIAL Y AUTOMÁTICA Escuela Politécnica de Ingeniería de Ferrol		 UNIVERSIDADE DA CORUÑA	TFG Nº: 770511A000
TÍTULO DEL TFG:			
TÍTULO COMPLETO DEL TFG			
TÍTULO DEL PLANO:			FECHA: Noviembre 2022
TÍTULO COMPLETO DEL PLANO			ESCALA: 1:0000
AUTOR:		FIRMA:	
NOMBRE COMPLETO DEL AUTOR		PLANO Nº: 00	

1 2 3 4 5 6 7 8 9 10 11 12

A
B
C
D
E
F
G
H

A
B
C
D
E
F
G
H

1 2 3 4 5 6 7 8 9 10 11 A2

Título Título del Trabajo Final

Pliego de condiciones

Peticionario Escuela Politécnica de Ingeniería de Ferrol
Rúa Mendizábal, s/n, Campus de Esteiro,
15403, Ferrol

Fecha Noviembre de 2022

Autor El Alumno

Fdo. Nombre del autor del Trabajo

Índice Pliego de condiciones

7.1	Contenido	45
-----	---------------------	----

El Pliego de Condiciones es uno de los documentos que constituyen el Trabajo Final y tiene como misión establecer las condiciones técnicas, económicas, administrativas, facultativas y legales para que el objeto del Trabajo Final pueda materializarse en las condiciones especificadas, evitando posibles interpretaciones diferentes de las deseadas. Su contenido y extensión queda a criterio de su autor y en función del tipo de Trabajo Final.

7.1. Contenido

El Pliego de Condiciones se debe iniciar con un índice que haga referencia a cada uno de los capítulos y apartados que lo componen, con el fin de facilitar su utilización. Debe contener:

- a) Descripción de las obras, productos, instalaciones o servicios.
- b) Las especificaciones de los materiales y elementos constitutivos del objeto del Trabajo Final, incluyendo:
 - Un listado completo de los mismos.
 - Las calidades mínimas a exigir para cada uno de los elementos constitutivos del Trabajo Final, indicando la norma (si existe) que contemple el material solicitado.
 - Las pruebas y ensayos a que deben someterse, especificando:
 - La norma según la cual se van a realizar.
 - Las condiciones de realización.
 - Los resultados mínimos a obtener.
- c) Ejecución de las obras, productos, instalaciones o servicios.
- d) La reglamentación y la normativa aplicables incluyendo las recomendaciones o normas de no obligado cumplimiento que, sin ser preceptivas, se consideran de necesaria aplicación al Trabajo Final a criterio del autor.
- e) Aspectos del contrato que se refieran directamente al Trabajo Final y que pudieran afectar a su objeto, ya sea en su fase de materialización o en su fase de funcionamiento.

Debe incluir:

- Documentación base para la contratación de su materialización. Los trabajos a realizar deben quedar definidos en:
 - Los Planos
 - Las Mediciones
 - La Memoria
 - Las especificaciones mencionadas en el epígrafe b).
- Limitaciones en los suministros, que especifiquen claramente dónde empieza y dónde termina la responsabilidad del suministro y montaje.
- Criterios de medición, valoración y abono.
- Criterios para las modificaciones del proyecto original, especificando el procedimiento a seguir para las mismas, su aceptación y cómo deben quedar reflejadas en la documentación final.
- Pruebas y ensayos, especificando cuales y en qué condiciones deben someterse los suministros según lo indicado en el apartado b).
- Garantía de los suministros, indicando el alcance, duración y limitaciones.
- Garantía de funcionamiento, indicando el alcance, duración y limitaciones.

Título Título del Trabajo Final

Mediciones

Peticionario Escuela Politécnica de Ingeniería de Ferrol
Rúa Mendizábal, s/n, Campus de Esteiro,
15403, Ferrol

Fecha Noviembre de 2022

Autor El Alumno

Fdo. Nombre del autor del Trabajo

Índice Mediciones

8.1	Contenido	49
-----	---------------------	----

El documento Mediciones es uno de los documentos del Trabajo Final y tiene como misión definir y determinar las unidades de cada partida o unidad de obra que configuran la totalidad del producto, obra, edificio, instalación y servicios objeto del Trabajo Final, basándose en la información contenida en el documento Planos. Debe incluir el número de unidades y definir las características, modelos, tipos y dimensiones de cada partida de obra o elemento objeto del Trabajo Final. Preferentemente se debe utilizar el sistema internacional de medidas conforme a la Norma UNE 82100 (partes 0 a 13). Se debe utilizar el concepto de partida alzada en aquellas unidades de obra en que no sea posible desglosar, en forma razonable, el detalle de las mismas.

8.1. Contenido

El documento Mediciones se debe iniciar con un índice que hará referencia a cada uno de los capítulos y apartados que lo componen, con el fin de facilitar su utilización. Debe contener un listado completo de las partidas de obra que configuran la totalidad del Trabajo Final. Se debe subdividir en distintos apartados o subapartados, correspondientes a las partes más significativas del objeto del Trabajo Final. Debe servir de base para la realización del Presupuesto.

Título Título del Trabajo Final

Presupuesto

Peticionario Escuela Politécnica de Ingeniería de Ferrol
Rúa Mendizábal, s/n, Campus de Esteiro,
15403, Ferrol

Fecha Noviembre de 2022

Autor El Alumno

Fdo. Nombre del autor del Trabajo

Índice Presupuesto

9.1	Contenido	53
-----	---------------------	----

El Presupuesto es uno de los documentos del Trabajo Final y tiene como misión determinar el coste económico, en unidades monetarias, de la ejecución material del objeto del Trabajo Final especificando las partidas ejecutadas por contrata y/o por administración. Se debe basar en el documento de Mediciones y seguir su misma ordenación.

9.1. Contenido

El Presupuesto se debe iniciar con un índice que hará referencia a cada uno de los capítulos y apartados que lo componen, con el fin de facilitar su utilización. El Presupuesto debe contener:

- Un cuadro de precios unitarios de materiales, mano de obra y elementos auxiliares que componen las partidas o unidades de obra.
- Un cuadro de precios unitarios de las unidades de obra, de acuerdo con el Estado de Mediciones y con la descomposición correspondiente de materiales, mano de obra y elementos auxiliares.
- El Presupuesto propiamente dicho que contenga la valoración económica global, desglosada y ordenada según el documento de Mediciones.

El Presupuesto debe establecer el alcance de los precios, indicando claramente si incluyen o no conceptos tales como:

- Gastos generales y beneficio industrial.
- Impuestos, tasas y otras contribuciones.
- Seguros.
- Costes de certificación y visado.
- Permisos y licencias.
- Cualquier otro concepto que influya en el coste final de materialización del objeto del Trabajo Final.

