



Trabajo final de grado

**GRADO DE INFORMÁTICA**

Facultat de Matemàtiques  
Universitat de Barcelona

---

**Visualización de redes inteligentes**

---

**Autor: Martin Azpillaga Aldalur**

Director: Dr. Jesús Cerquides  
Realitzat a: IIIA  
(nom del departament)

Barcelona, 5 de enero de 2016

# Abstract

Example abstract.

# Resumen

Resumen

# Índice

<b>1. El entorno(1)</b>	<b>1</b>
1.1. La humanidad necesita energía eléctrica . . . . .	1
1.2. La idea de los Smart Grids . . . . .	1
1.3. Problemas: ¿Como se tradea la energía? . . . . .	1
<b>2. Definiendo el problema(2)</b>	<b>2</b>
2.1. El problema . . . . .	2
2.2. Partes del problema: Creación de grafo, CEAP, Simulación, Visualización, Reports . . . . .	2
2.3. Buscando tecnologías adecuadas . . . . .	2
2.4. A partir de ahora . . . . .	2
<b>3. CEAP(5)</b>	<b>3</b>
3.1. Se me presentó este proyecto. . . . .	3
3.2. Formalización en participant, ILV, PLV, Link. . . . .	3
3.3. RadPro. . . . .	3
3.4. Limitaciones y ampliaciones posibles. . . . .	3
<b>4. Simulación(8)</b>	<b>3</b>
4.1. Filosofía de java. POO, encapsulación, interfaces. . . . .	3
4.2. Input/Output. . . . .	3
4.3. Diagramas. . . . .	3
4.4. Modelado de una casa . . . . .	3
4.5. Battery, . . . . .	3
4.6. Generator, . . . . .	3
4.7. Appliance, . . . . .	3
4.8. Bid. . . . .	3
<b>5. Visualización(5)</b>	<b>3</b>
5.1. Filosofía de Unity. Componentes/no. . . . .	3
5.2. Menú dinámico. . . . .	3
5.3. Animaciones. . . . .	3
<b>6. Reports(2)</b>	<b>3</b>
6.1. Aún por decidir. . . . .	3

<b>7. Creación de la ciudad(5)</b>	<b>3</b>
7.1. Algoritmos: Steiner tree, algoritmo RTT, algoritmo nearest. . . . .	3
7.2. Ampliar a Open Street Map, CityEngine. . . . .	3
<b>8. Conclusiones(1)</b>	<b>3</b>

# **1. El entorno(1)**

## **1.1. La humanidad necesita energía eléctrica**

Hay gran interés por parte de gobiernos en invertir en investigar maneras más eficientes de tratar con la energía.

## **1.2. La idea de los Smart Grids**

Auge de energías renovables personales. Mencionar artículos donde se tratan smart grids.

## **1.3. Problemas: ¿Como se tradea la energía?**

Quien controla todo el flujo, cuando se hacen los intercambios, como se hacen los intercambios, como pueden interferir los usuarios en estos intercambios. Mencionar reglamentos de otros países y estado actual de España.

## **2. Definiendo el problema(2)**

### **2.1. El problema**

Explicar qué se intenta resolver

### **2.2. Partes del problema: Creación de grafo, CEAP, Simulación, Visualización, Reports**

Explicar por que existe cada parte y por que está diferenciado del resto.

### **2.3. Buscando tecnologías adecuadas**

Explicar programas adecuados para cada parte así como posibles distintas maneras de implementar cada aspecto a grandes rasgos

### **2.4. A partir de ahora**

Iremos explicando cada parte desde el núcleo (CEAP) hasta el exterior (Visualización)

### 3. CEAP(5)

- 3.1. Se me presentó este proyecto.
- 3.2. Formalización en participant, ILV, PLV, Link.
- 3.3. RadPro.
- 3.4. Limitaciones y ampliaciones posibles.

### 4. Simulación(8)

- 4.1. Filosofía de java. POO, encapsulación, interfaces.
- 4.2. Input/Output.
- 4.3. Diagramas.
- 4.4. Modelado de una casa
- 4.5. Battery,
- 4.6. Generator,
- 4.7. Appliance,
- 4.8. Bid.

### 5. Visualización(5)

- 5.1. Filosofía de Unity. Componentes/no.
- 5.2. Menú dinámico.
- 5.3. Animaciones.

### 6. Reports(2)

- 6.1. Aún por decidir.

### 7. Creación de la ciudad(5)

- 7.1. Algoritmos: Steiner tree<sub>3</sub>, algoritmo RTT, algoritmo nearest.
- 7.2. Ampliar a Open Street Map, CityEngine.

## Referencias

- [1] Batut, C.; Belabas, K.; Bernardi, D.; Cohen, H.; Olivier, M.: User's guide to *PARI-GP*,  
`pari.math.u-bordeaux.fr/pub/pari/manuals/2.3.3/users.pdf`, 2000.