

# CREACIÓ AUTOMÀTICA DE GRÀFICS ESTADÍSTICS A PARTIR DE DADES DE SOROLL D'EUROPA AMB TECNOLOGIES OPEN SOURCE EN EL VISOR NOISE MAP VIEWER DE L'ETC-LUSI

PROJECTE FINAL  
MÀSTER EN TECNOLOGIES DE LA INFORMACIÓ GEOGRÀFICA  
11<sup>a</sup> EDICIÓ  
Universitat Autònoma de Barcelona. Departament de Geografia

Autora: Montserrat Oller Garcia

Tutor LIGIT: Ignacio Ferrero Beato

Tutors ETC-LUSI: Walter Simonazzi i César Martínez



# Continguts:

1. Introducció
2. Anàlisi
3. Disseny del software
4. Desenvolupament
5. Implementació
6. Proves
7. Resultats
8. Conclusions

### Presentació :

- ▶ El projecte s'ha desenvolupat en el Centre Temàtic Europeu d'Usos del Sòl i Informació Espacial (ETC-LUSI).
- ▶ S'emmarca dins el projecte *Noise Map Viewer* per Europa, desenvolupat per l'ETC-LUSI per a l'Agència Europea de Medi Ambient (AEMA).
- ▶ Neix de la necessitat de mostrar les dades de soroll d'Europa de manera que puguin arribar al major nombre de persones possible.
- ▶ Ve determinat per la Comissió Europea que té una Directiva sobre Soroll Ambiental, la qual sol·licita mapes de soroll segons la població exposada a aquest.
- ▶ Aquesta Directiva pertany a l'AEMA i té com a objectiu principal oferir informació sòlida sobre el medi ambient, per això demana als països membres, de manera obligatòria, unes dades de soroll determinades.

## El visor:

- ▶ El visor pretén mostrar de la manera més clara possible les principals dades geogràfiques i estadístiques reportades pels diferents països.
- ▶ Les dades vénen donades a nivell de país.
- ▶ Se centra en les estadístiques de soroll a les que està exposada la població en les aglomeracions i fora d'elles.

Principals fonts de soroll estudiades
Aeroports
Carreteres
Ferrocarrils
Indústria (només dins les aglomeracions)

Franja horària i bandes de soroll (dB)	
Durant el dia: <i>Lden (day, evening, night)</i>	>55 >65 >75
Durant la nit: <i>Lnight (night)</i>	>50 >60 >70

### Objectius:

- ▶ Creació de gràfics estadístics de soroll d'Europa de forma automàtica amb tecnologies *Open Source* dins el visor *Noise Map Viewer* (<http://noise.eionet.europa.eu/index.html>).
- ▶ Fer una anàlisi en profunditat del sistema d'informació, com funcionava i quines eren les necessitats de l'ETC-LUSI a l'hora de mostrar les dades estadístiques amb el gràfic corresponent.
- ▶ Estudiar les diferents tecnologies i llibreries existents.
- ▶ Substituir la part que mostrava les estadístiques de forma manual (mitjançant imatges JPEG).
- ▶ Integrar el procés que havia de generar els gràfics estadístics automàticament (*servlet*) a la web, de manera que no afectés el treball realitzat fins aleshores.

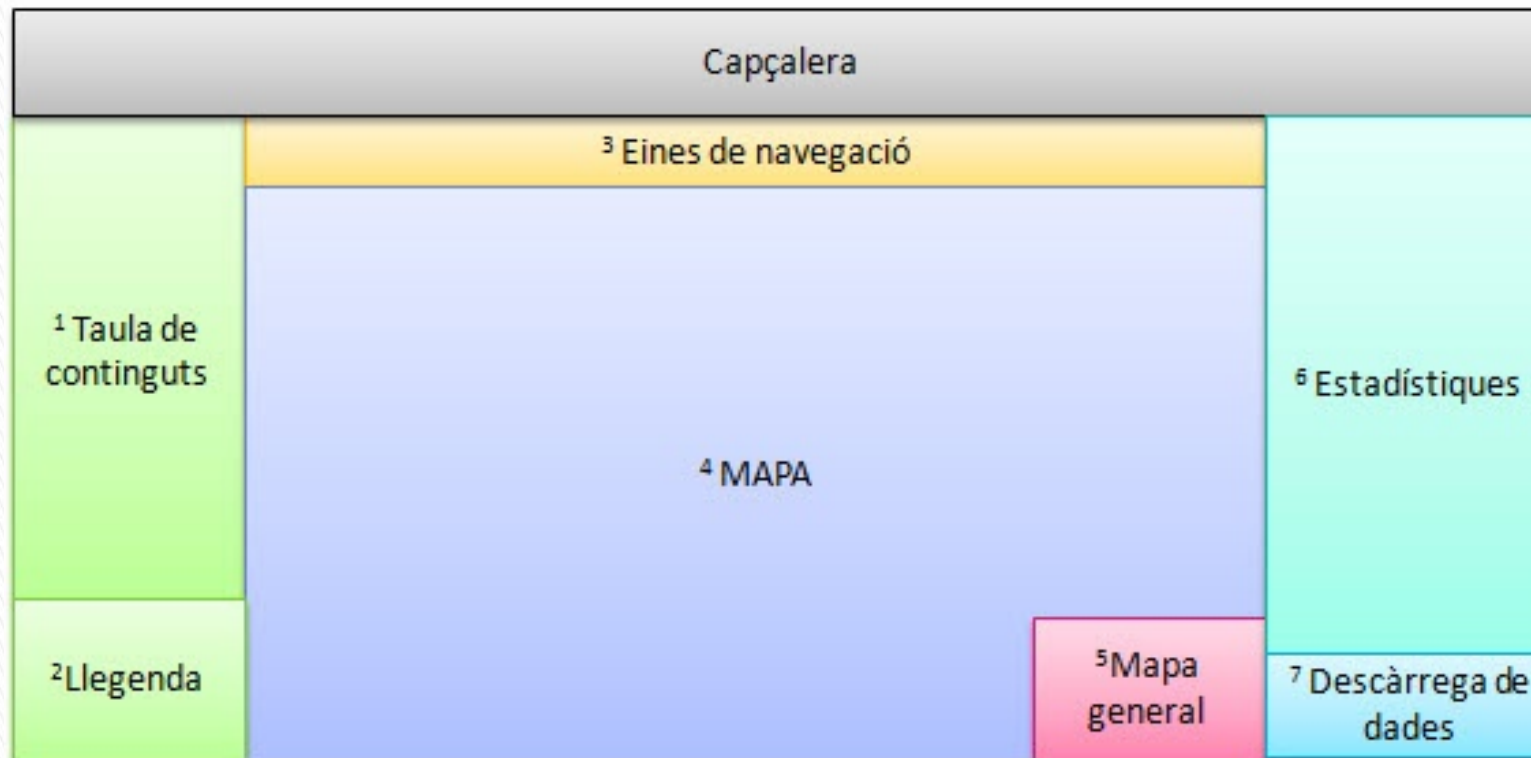
### Metodologia:

- ▶ La metodologia a seguir s'ha basat en **Mètrica versió 3** (Metodologia de Planificació, Desenvolupament i Manteniment de Sistemes d'Informació), promoguda pel Ministeri d'Administracions Públiques del Govern Espanyol.
- ▶ S'ha adaptat a les necessitats del present projecte.
- ▶ Dóna les pautes a seguir en tots aquells projectes de desenvolupament de sistemes d'informació orientats a objectes o estructurats.
- ▶ D'aquesta manera podem definir les quatre grans fases del projecte:
  1. Anàlisi
  2. Disseny del software
  3. Desenvolupament
  4. Implementació

## Definició del sistema:

1. Base de dades alfanumèrica (Access)
2. Base de dades geogràfica

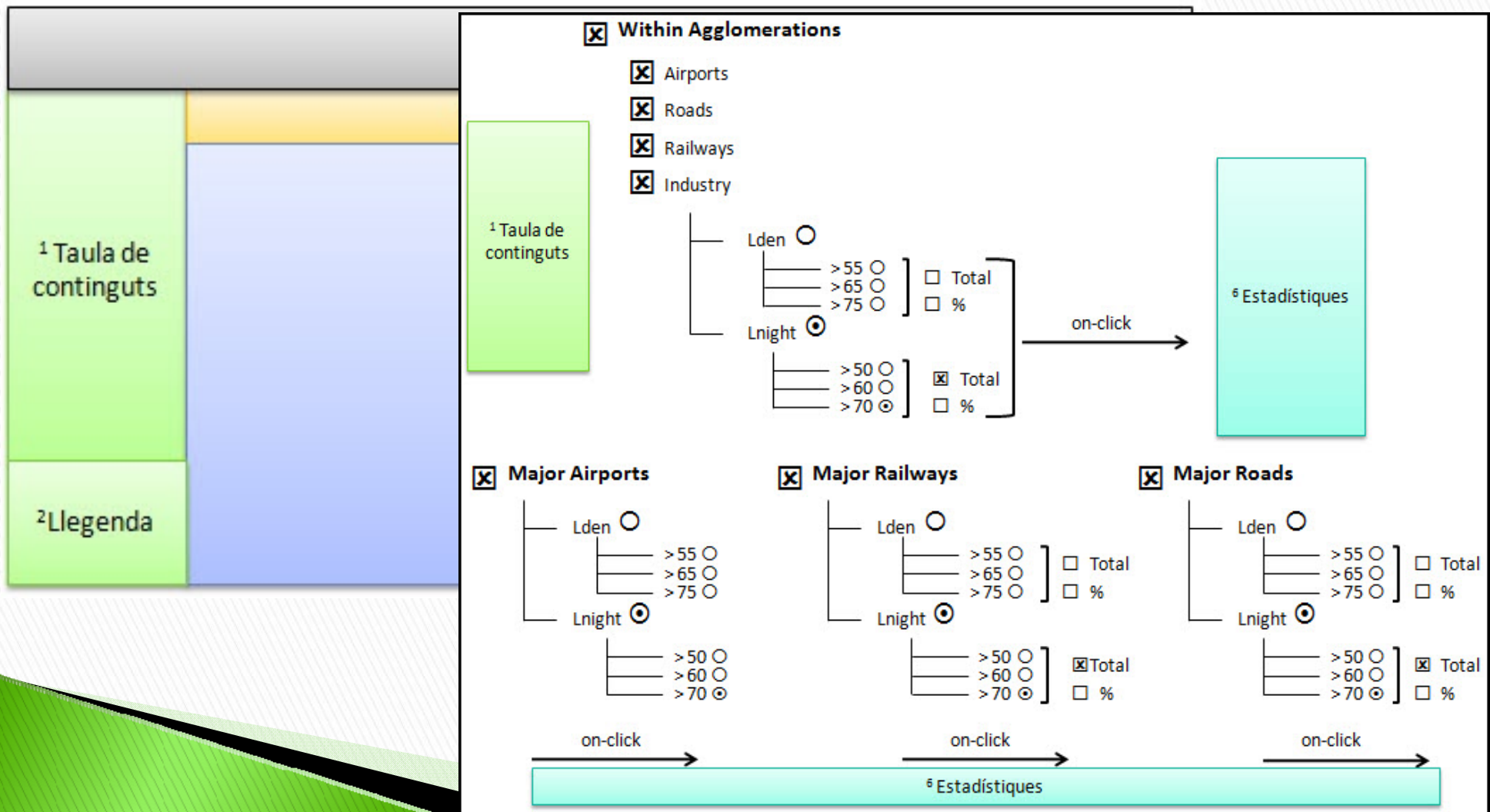
Geodatabase



## Definició del sistema:

1. Base de dades alfanumèrica (Access)
2. Base de dades geogràfica

Geodatabase





## Definició del sistema:

1. Base de dades alfanumèrica (Access)
2. Base de dades geogràfica

Geodatabase

**Noise Observation and Information Service for Europe - NOISE**

**Contents**

- Original View
- Within agglomerations
  - Airports
    - L<sub>den</sub>
      - > 55dB
        - Total
        - Percentage (%)
      - > 65dB
        - Total
        - Percentage (%)
      - > 75dB
        - Total
        - Percentage (%)
    - L<sub>night</sub>
      - > 50dB
        - Total
        - Percentage (%)
      - > 60dB
        - Total
        - Percentage (%)
      - > 70dB
        - Total
        - Percentage (%)
  - Roads
  - Railways
  - Industry
  - Major Airports
  - Major Railways
  - Major Roads

**Legend**

- Countries with data outside coverage
- EU countries
- EEA member countries
- Sea
- Airports
- Agglomerations

**Noise exposure map**

Zoom to country ... More detailed information ... 3778024.40247°E 3415294.96863°N

**Statistics**

A noise source should be selected in order to display noise exposure data reported by the MS.

Number of noise sources

Noise Source on the left panel

55-59 60-64 65-69 70-74 >75

Download Noise exposure data

Click the following [link](#) to download:

- Summary of noise exposure data in WORD - EXCEL format.
- Complete database as reported by Member States in WORD - ACCESS format.
- Quick guidance to deal with the complete database.

Noise viewers at national and local scale

Links

Glossary of technical terms

Total

%

Total

%

European Topic Centre  
Land Use and Spatial Information

F. A. O. Legal Notice Privacy Policy Disclaimer Contact Help

European Environment Agency

1 Taula conting

2 Lleger

## Identificació de l'entorn tecnològic:

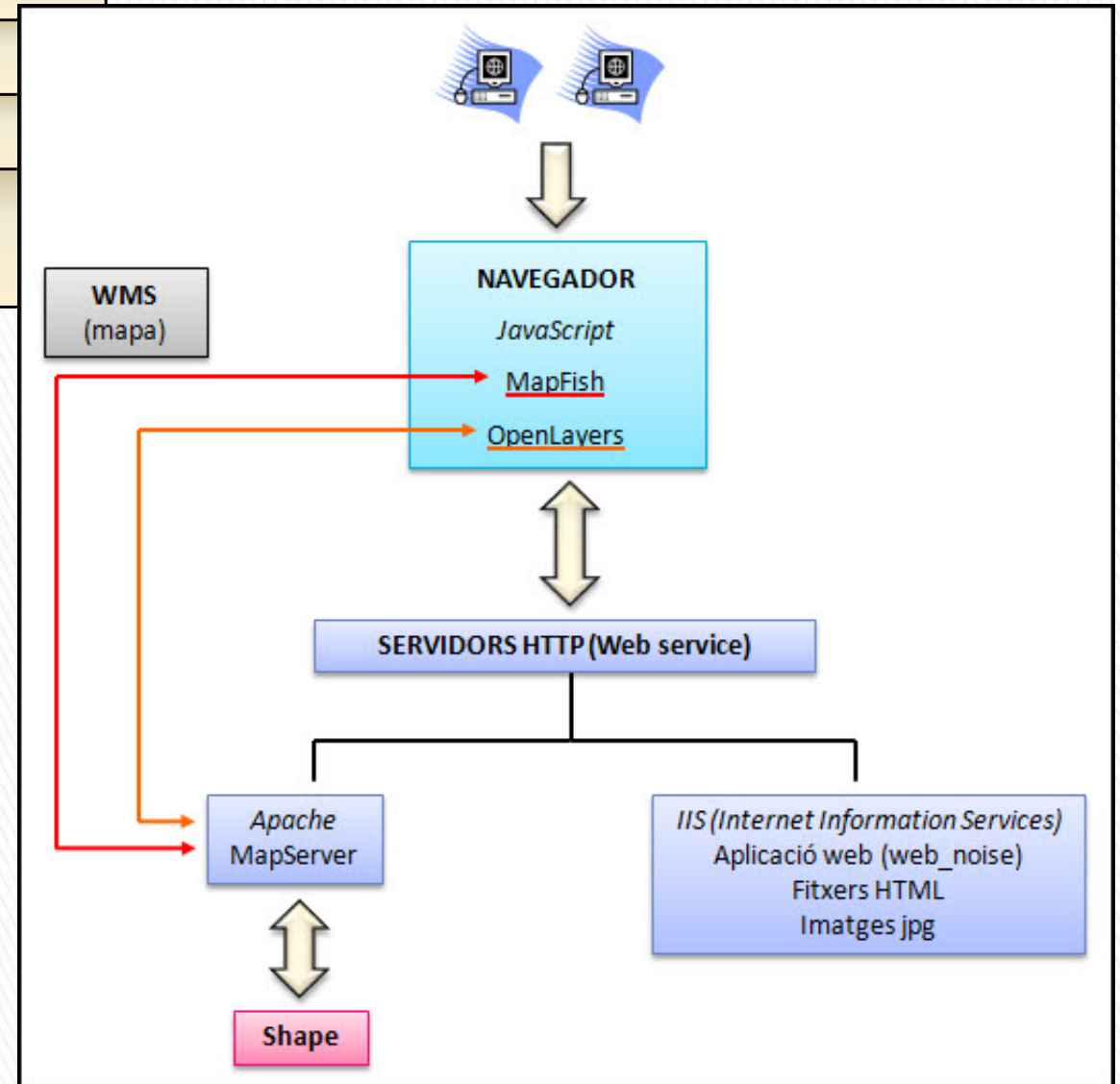
Sistema Operatiu: Windows 2003 Server

MapServer: Apache

MapFish, OpenLayers → JavaScript

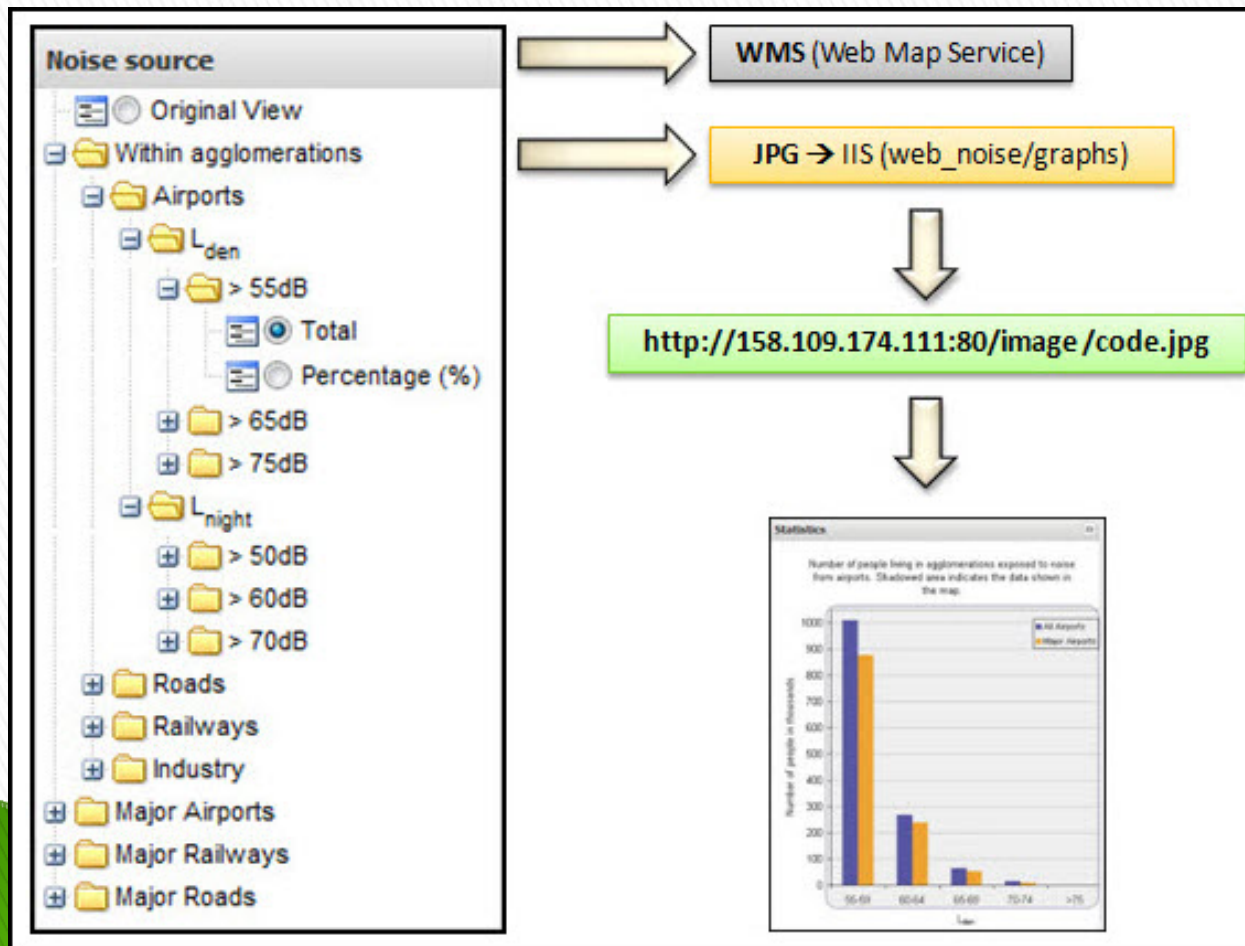
IIS (Internet Information Services):  
web\_noise, JPEG, fitxers HTML/CSS

Arquitectura inicial del sistema



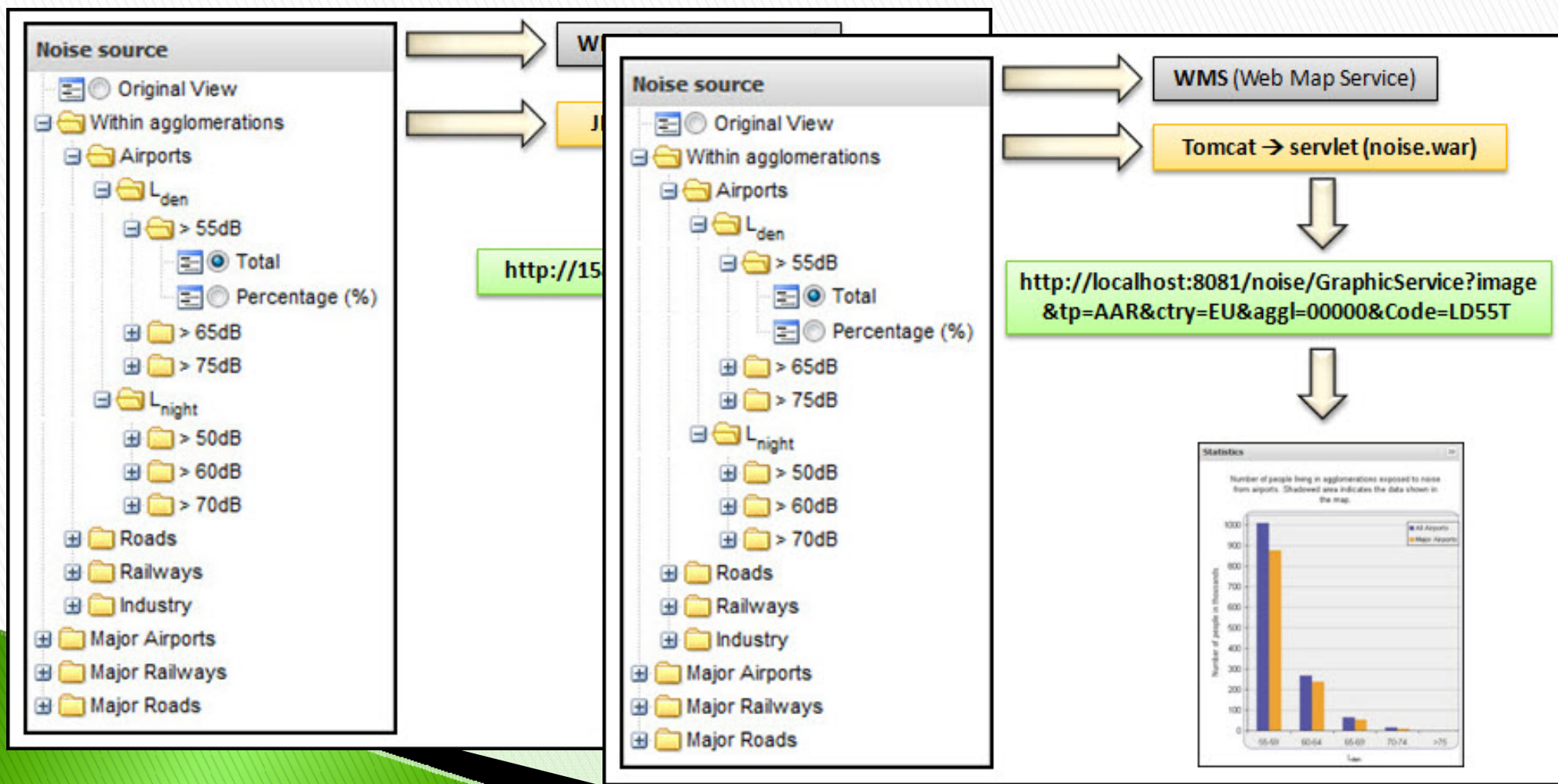
## Establiment de requisits funcionals inicials i finals:

- ▶ Ens donen el comportament intern del software tenint en compte els diferents casos d'ús, els quals ens proporcionen un o més escenaris que ens indiquen com hauria d'interactuar el sistema amb l'usuari per obtenir un objectiu específic.



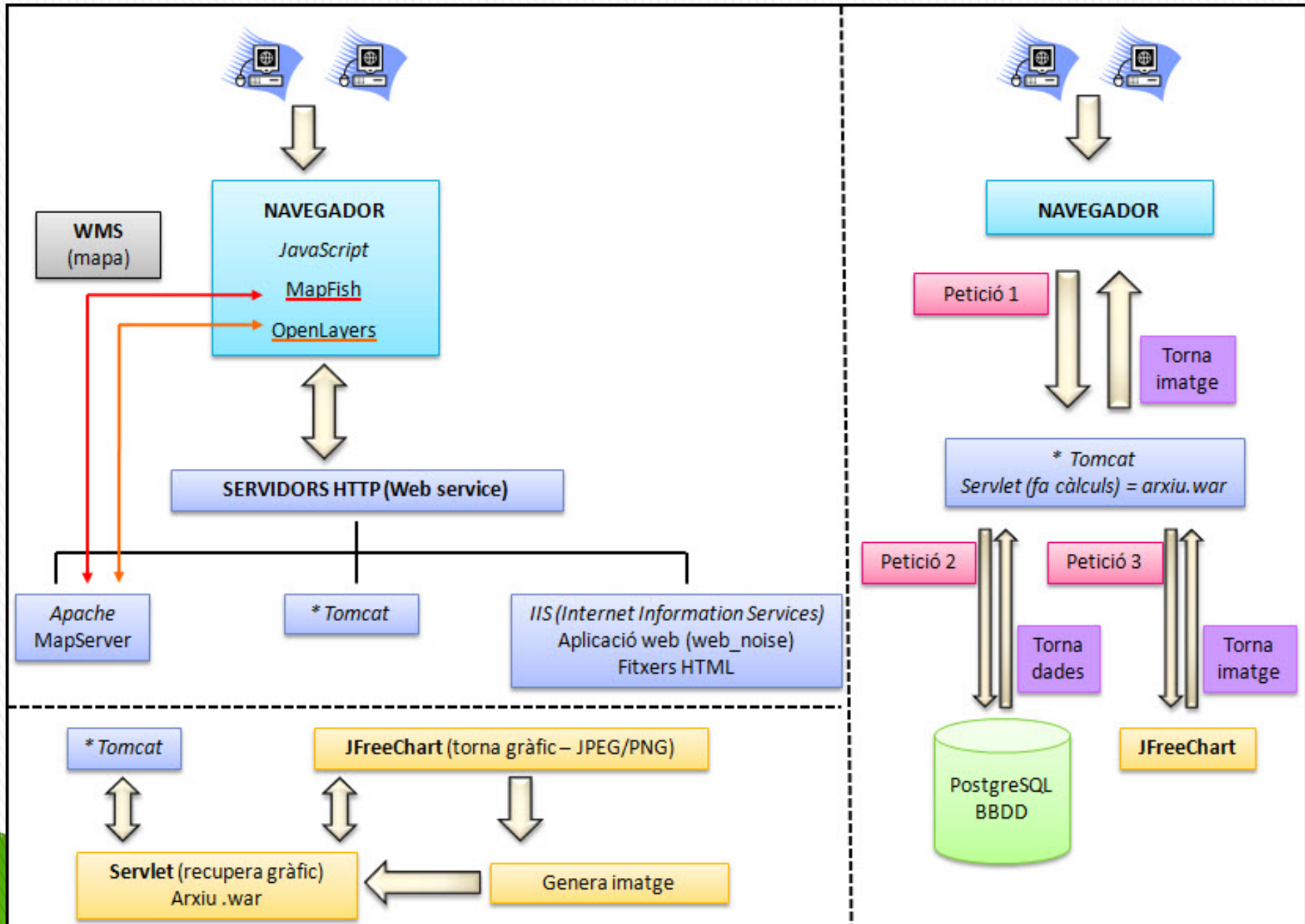
## Establiment de requisits funcionals inicials i finals:

- ▶ Ens donen el comportament intern del software tenint en compte els diferents casos d'ús, els quals ens proporcionen un o més escenaris que ens indiquen com hauria d'interactuar el sistema amb l'usuari per obtenir un objectiu específic.



## Establiment de requisits tecnològics:

- ▶ S'ha hagut de tenir en compte que el visor s'havia desenvolupat amb MapFish.
- ▶ S'ha realitzat un estudi de les diferents llibreries existents i s'ha analitzat quines d'elles eren compatibles amb aquest.
- ▶ Finalment s'ha escollit la llibreria JFreeChart, llibreria Java, totalment lliure, que permet mostrar gràfics de qualitat professional en les aplicacions on s'utilitza.
- ▶ De la gran gamma de gràfics que ens ofereix aquesta llibreria, ens haurem de fixar ens els gràfics d'una o més barres, tant verticals com horitzontals.
- ▶ Tenint tot això en compte, i sabent com es comportarà cada un dels elements, podem obtenir la nova arquitectura del sistema:



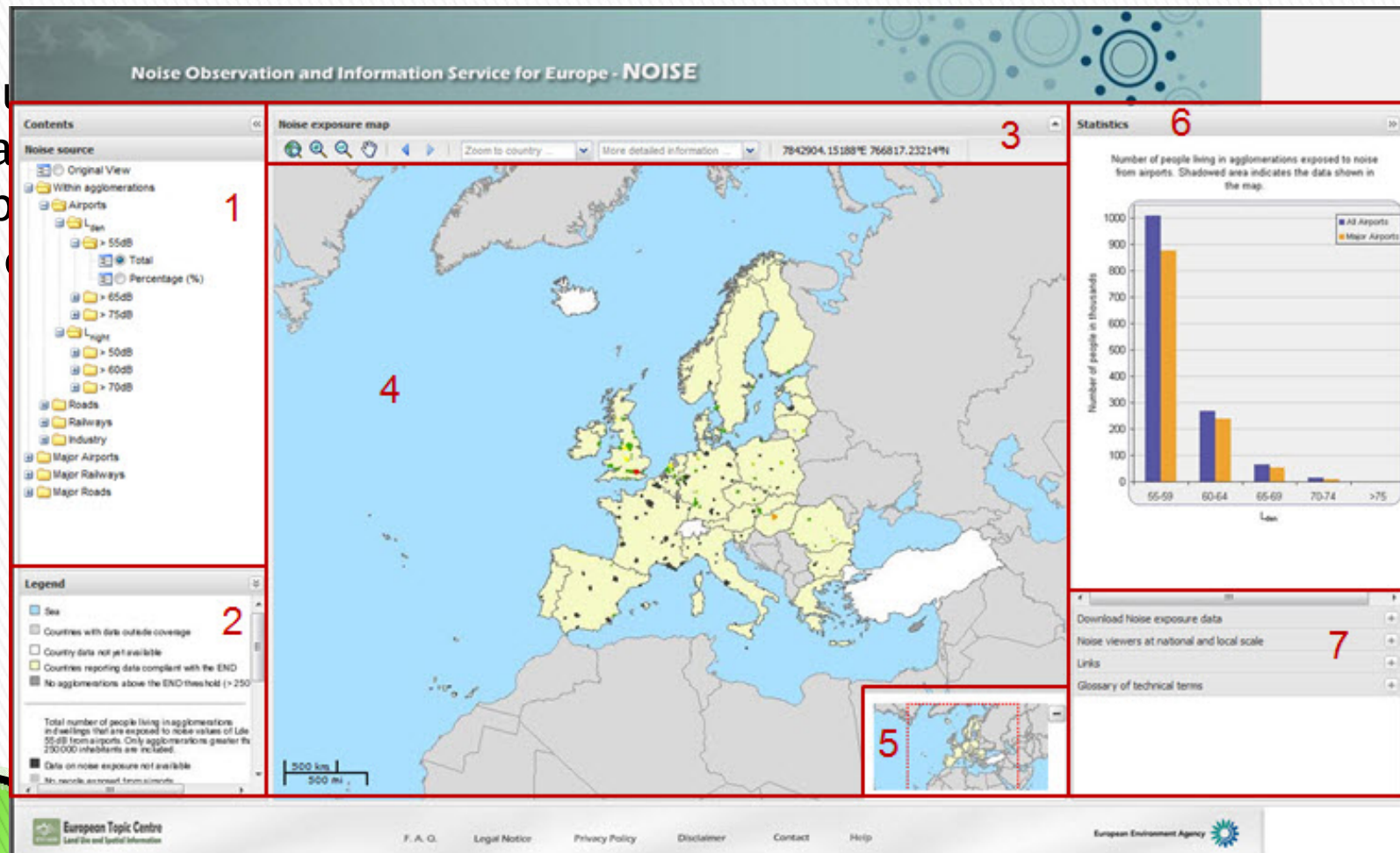
## Definició de les interfícies d'usuari :

- ▶ Definir el format i contingut de les interfícies de pantalla especificant el seu comportament dinàmic quan l'usuari realitza una acció sobre elles.
- ▶ La única diferència que existeix entre la interfície d'usuari inicial i final, és la manera en que es mostra el gràfic estadístic. Això és imperceptible per a l'usuari que no nota cap canvi en el funcionament del visor, ja que el canvi només afecta la part interna del sistema.

## Definició de les interfícies d'usuari :

- ▶ Definir el format i contingut de les interfícies de pantalla especificant el seu comportament dinàmic quan l'usuari realitza una acció sobre elles.

- ▶ La finalitat principal de la interfície és el



al i és el rna

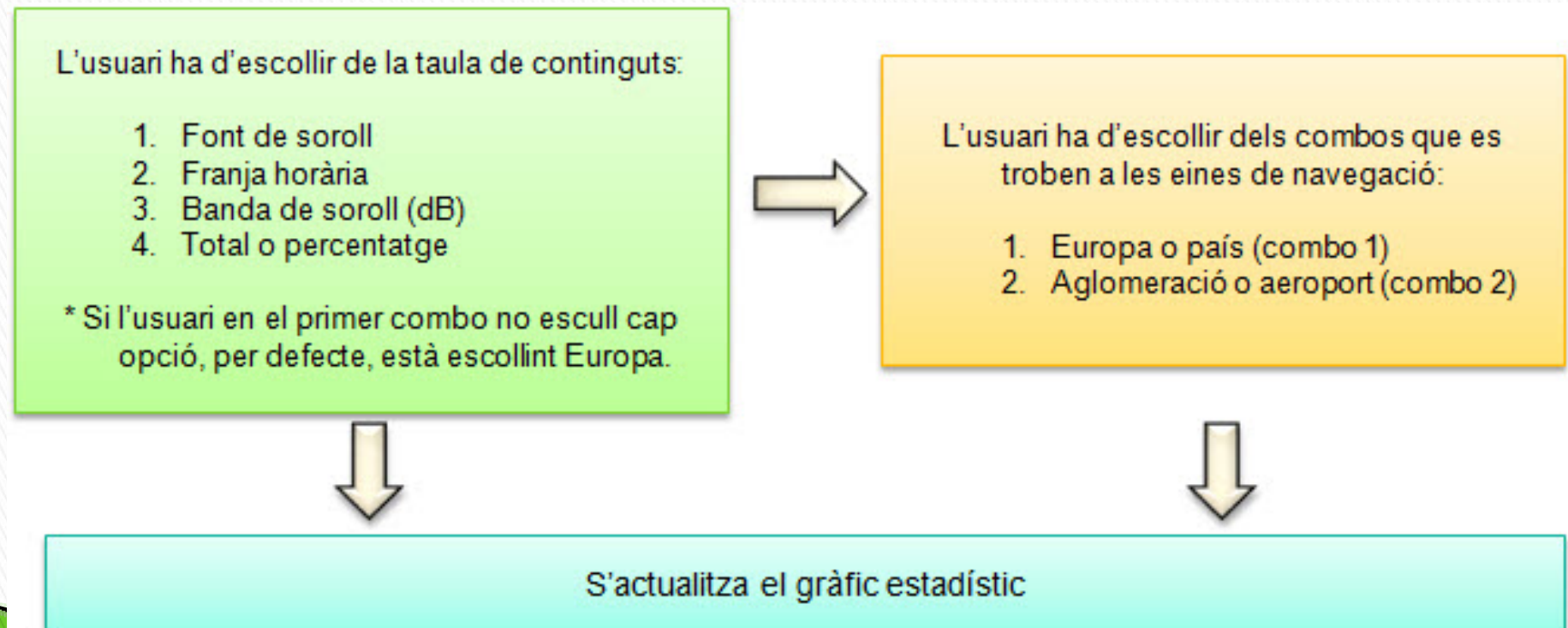


## Casos d'ús:

- ▶ L'especificació dels casos d'ús és obligatòria en aquells projectes que estan orientats a objectes.
- ▶ Sovint s'han d'especificar múltiples casos d'ús que ens acabin de definir el sistema.
- ▶ S'han hagut d'especificar quines opcions pot escollir l'usuari:
- ▶ Seran molt útils a l'hora de començar a programar, doncs ens facilita un esquema dels mètodes que hem de crear.
- ▶ Serviran per fer la tasca més fàcil a l'hora de crear consultes SQL (*Structured Query Language*) per fer la petició de les dades requerides a la base de dades.

## Casos d'ús:

- ▶ L'especificació dels casos d'ús és obligatòria en aquells projectes que estan orientats a objectes.
- ▶ Sovint s'han d'especificar múltiples casos d'ús que ens acabin de definir el sistema.
- ▶ S'han hagut d'especificar quines opcions pot escollir l'usuari:



Anàlisi

Disseny del software

Desenvolupament

Implementació

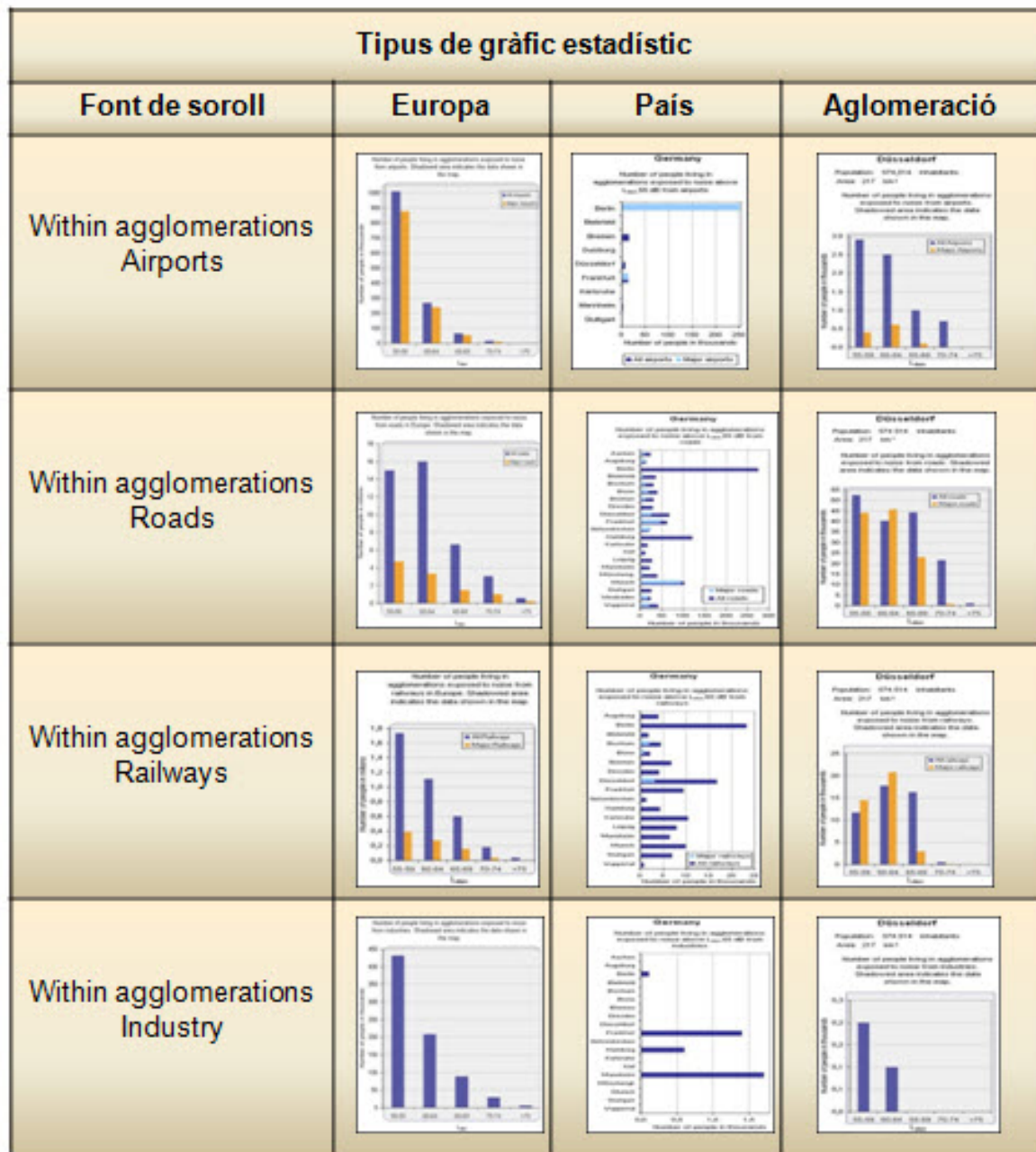
Font de soroll	Europa	País	Aglomeració	Aeroport	Total	Percentatge	Tràfic
Within agglomerations Airports	X	X	X	-	X	X	-
Within agglomerations Roads	X	X	X	-	X	X	-
Within agglomerations Railways	X	X	X	-	X	X	-
Within agglomerations Industry	X	X	X	-	X	X	-
Major Airports	X	X	-	X	X	-	X
Major Railways	X	X	-	-	X	X (Relatiu: Hab/km)	-
Major Roads	X	X	-	-	X	X (Relatiu: Hab/km)	-

Franja horària i banda de soroll (dB)

Lden	Categories	Lnight	Categories
>55	55-59 60-64	>50	50-54 55-59
>65	65-69 70-74	>60	60-64 65-69
>75	> 75	>70	>70

Font de soroll	Europa	País	Aglomeració
Within agglomerations Airports	X	X	X
Within agglomerations Roads	X	X	X
Within agglomerations Railways	X	X	X
Within agglomerations Industry	X	X	X
Major Airports	X	X	-
Major Railways	X	X	-
Major Roads	X	X	-

Franja horària i banda de soroll (dB)			
Lden	Categories	Lnight	Categories
>55	55-59 60-64	>50	50-54 55-59
>65	65-69 70-74	>60	60-64 65-69
>75	> 75	>70	>70



# Anàlisi

# Disseny del software

# Desenvolupament

# Implementació

Font de soroll	Europa	País	Aglomeració	Tipus de gràfic estadístic			
Font de soroll	Europa	País	Aglomeració	Font de soroll	Europa	País	Aglomeració
Within agglomerations Airports	X	X	X	Within agglomerations			
Within agglomerations Roads	X	X	X				
Within agglomerations							

Tipus de gràfic estadístic			
Font de soroll	Europa	País	Aglomeració
Within agglomerations Airports			
Within agglomerations Roads			
Within agglomerations Railways			
Within agglomerations Industry			

Tipus de gràfic estadístic			
Font de soroll	Europa	País	Aeroport / Aglomeració
Major Airports			
Major Railways			<p>At this scale of visualization, only noise exposure Within Agglomerations and noise exposure due to Major Airports are displayed.</p> <p>In order to see the noise exposure data outside agglomerations (due to Major Roads or Major Railways), either the Member State or Europe option should be selected.</p>
Major Roads			



Anàlisi

Disseny del software

Desenvolupament

Implementació

## Programari:



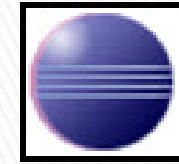
Access



Excel



FireFox



Eclipse



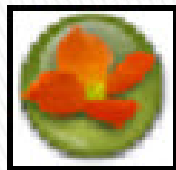
Apache Tomcat 6.x



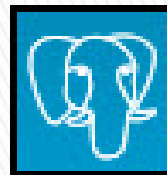
MapServer – MS4W  
Apache



JFreeChart 1.0.13  
JCommon 1.0.16



Pentaho Kettle 3.2.0  
Spoon



PostgreSQL 8.4  
pgAdminIII 1.10



Macromedia Dreamweaver 8

## Servidors:

### Servidor web HTTP – Apache:

- Paquet MapServer – MS4W.
- Entorn que s'ha utilitzat per construir el visor *Noise Map Viewer*.
- S'instal·la la carpeta *web\_noise*, que conté la pàgina web.

### Servidor web contenidor de *servlets* – Tomcat:

- El procés que genera els gràfics estadístics és un *servlet* (és un objecte Java que respon a peticions HTTP servint pàgines dinàmiques).

## Eclipse:

- Entorn de desenvolupament integrat de codi obert per desenvolupar aplicacions.
- Inclou eines de desenvolupament de Java.
- Fa ús d'un espai de treball propi (*workspace*).

## JFreeChart:

- Llibreria Java, totalment lliure.
- Permet crear múltiples tipus de gràfics.
- Utilitza una altra llibreria que també ha de ser instal·lada: *Jcommon*.
- També s'han d'instal·lar els *javadocs* (documentació API del codi font Java).

## Base de dades:

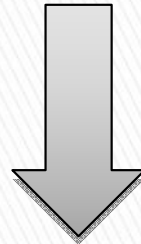
### PostgreSQL:

- Sistema de gestió de base de dades relacional orientada a objectes.
- Software lliure i de classe empresarial.
- Gran quantitat de dades i nombre d'usuaris en un mateix instant.



### pgAdminIII:

- Plataforma de desenvolupament i administració de PostgreSQL.
- Codi obert.
- Desenvolupament de bases de dades complexes.



### Pentaho Kettle – Spoon:

- Interfície gràfica d'usuari que permet automatitzar el procés de passar les dades d'una base de dades a l'altra.
- Necessita una connexió ODBC per poder transformar les taules d'entrada (Access/Excel) a unes noves de sortida (PostgreSQL).



## Desenvolupament del procés:

- ▶ El *servlet* que genera els gràfics s'ha fet en el llenguatge de programació Java (orientat a objectes).
- ▶ Es crea la classe que crearà el gràfics (*GraphicService*).
- ▶ Es genera la connexió a la nostra base de dades via JDBC (*Java*

Codis tp			
Codi	Consulta	Codi	Consulta
AAR	Aeroports dins les aglomeracions	MAR	Aeroports principals fora de les aglomeracions
ARD	Carreteres dins les aglomeracions	MRL	Ferrocarrils principals fora de les aglomeracions
ARL	Ferrocarrils dins les aglomeracions	MRD	Carreteres principals fora de les aglomeracions
AIN	Indústria dins les aglomeracions		

## Desenvolupament del procés:

- ▶ El *servlet* que genera els gràfics s'ha fet en el llenguatge de programació Java (orientat a objectes).
- ▶ Es crea la classe que crearà el gràfics (*GraphicService*).
- ▶ Es genera la connexió a la nostra base de dades via JDBC (*Java*

Codis tp					
Codi	Consulta	Codis Code			
		Codi	Consulta	Codi	Consulta
AAR	Aeroports dins les aglomeracions				
ARD	Carreteres dins les aglomeracions	LD	Dia	55 o 65 o 75	Bandes de soroll de dia
ARL	Ferrocarrils dins les aglomeracions	LN	Nit	50 o 60 o 70	Bandes de soroll de nit
AIN	Indústria dins les aglomeracions	T	Total		
		R	Percentatge		

## Implementació del *servlet* al Tomcat:

Exportació del *servlet* a arxiu WAR (Arxiu d'Aplicació Web)

S'instal·la dins del Tomcat

S'estableix la variable d'entorn JRE\_HOME (Java/jre6)

Modificació de l'arxiu server.xml (port diferent al que utilitza Apache)

Iniciem Tomcat

Descomprimeix arxiu WAR

El *servlet* ja funciona

## Implementació del *servlet* al MapFish :

- ▶ S'ha treballat amb els arxius compilats amb el programa Dreamweaver.
- ▶ El llenguatge de programació ha estat JavaScript, que és un llenguatge que s'utilitza per accedir a objectes en aplicacions.
- ▶ Es creen les noves variables necessàries per donar el codi de les imatges que es corresponen amb el codi Java.

```
var graphservlet='<img src="http://158.109.49.37:8081/noise/GraphicService?image';  
var tp= "&tp=""; var ctry= "&ctry=""; var agglcd= "&aggl=""; var code2= "&Code="";
```

- ▶ Per últim:

Canvi del codi  
que controla els  
combos (país i  
aglomeració)

Canvi del codi  
que controla el  
*LayerTree*

Els gràfics dins el  
visor ja funcionen  
correctament

## Proves:

### 1. Java

- ▶ Les proves del *servlet* s'han anat realitzant durant tot el procés per veure si els canvis que s'anaven fent en el codi (*Eclipse*) ens donaven els resultats que volíem obtenir.
- ▶ Es canviava la ruta del *GraphicService* des del Firefox.



- ▶ El mode *debug* de l'*Eclipse*: per veure el codi amb més detall, i sobretot per veure els errors.

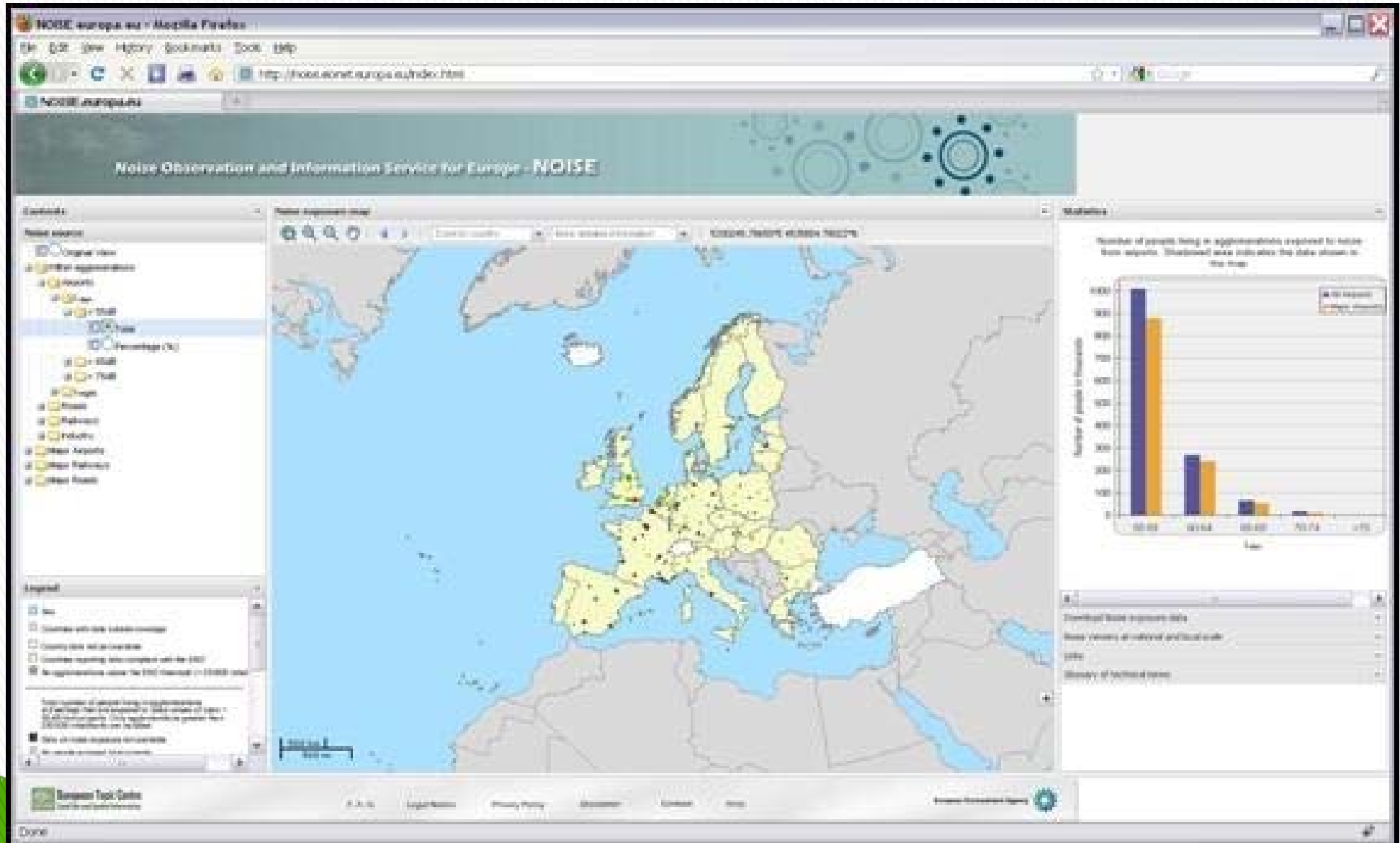
### 2. JavaScript

- ▶ Com estem en un medi de desenvolupament i no de producció, les proves s'han realitzat contra la pròpia màquina i des d'una altra màquina de l'ETC-LUSI.

Proves

Resultats

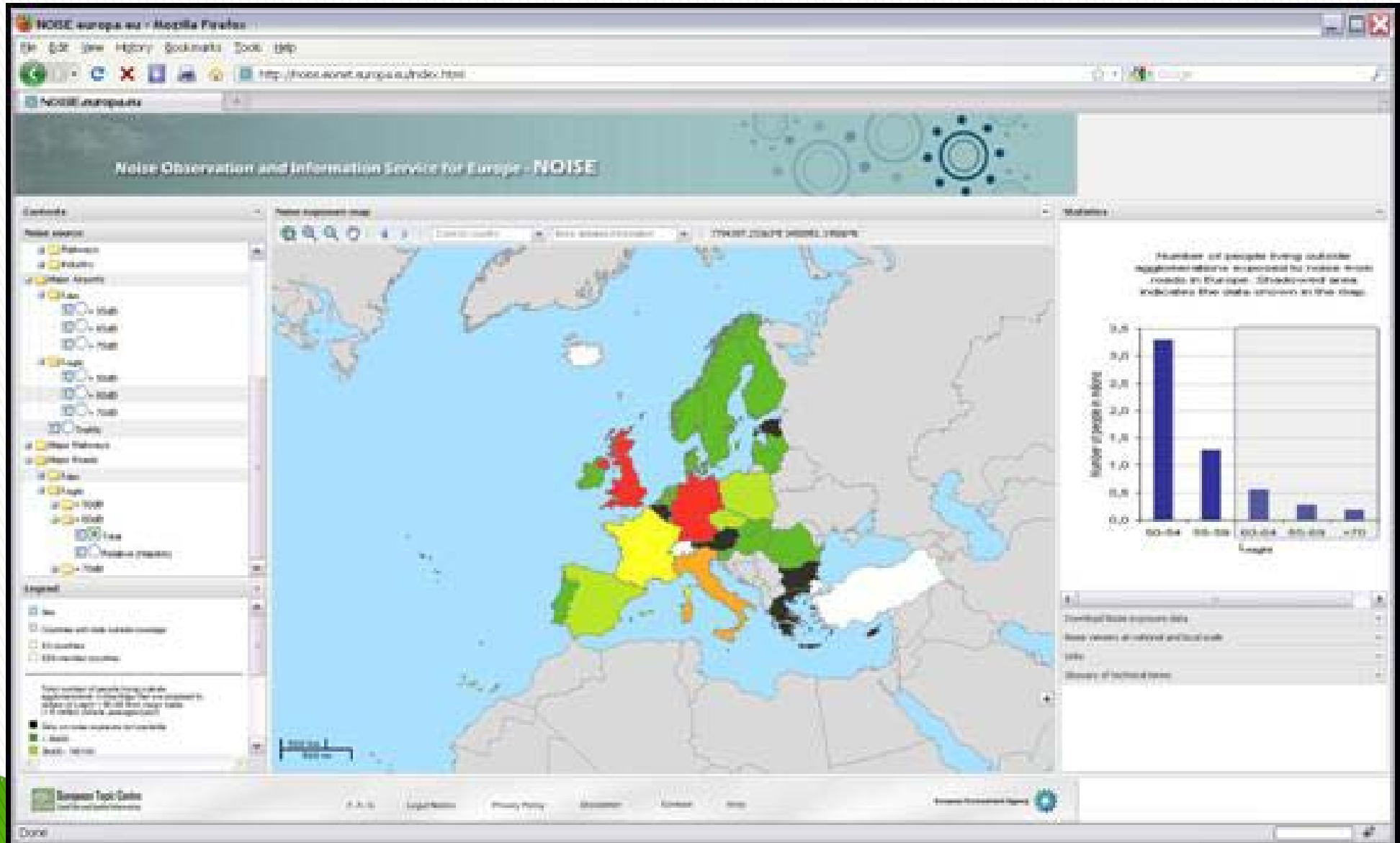
## Gràfics estadístics inicials:



Proves

Resultats

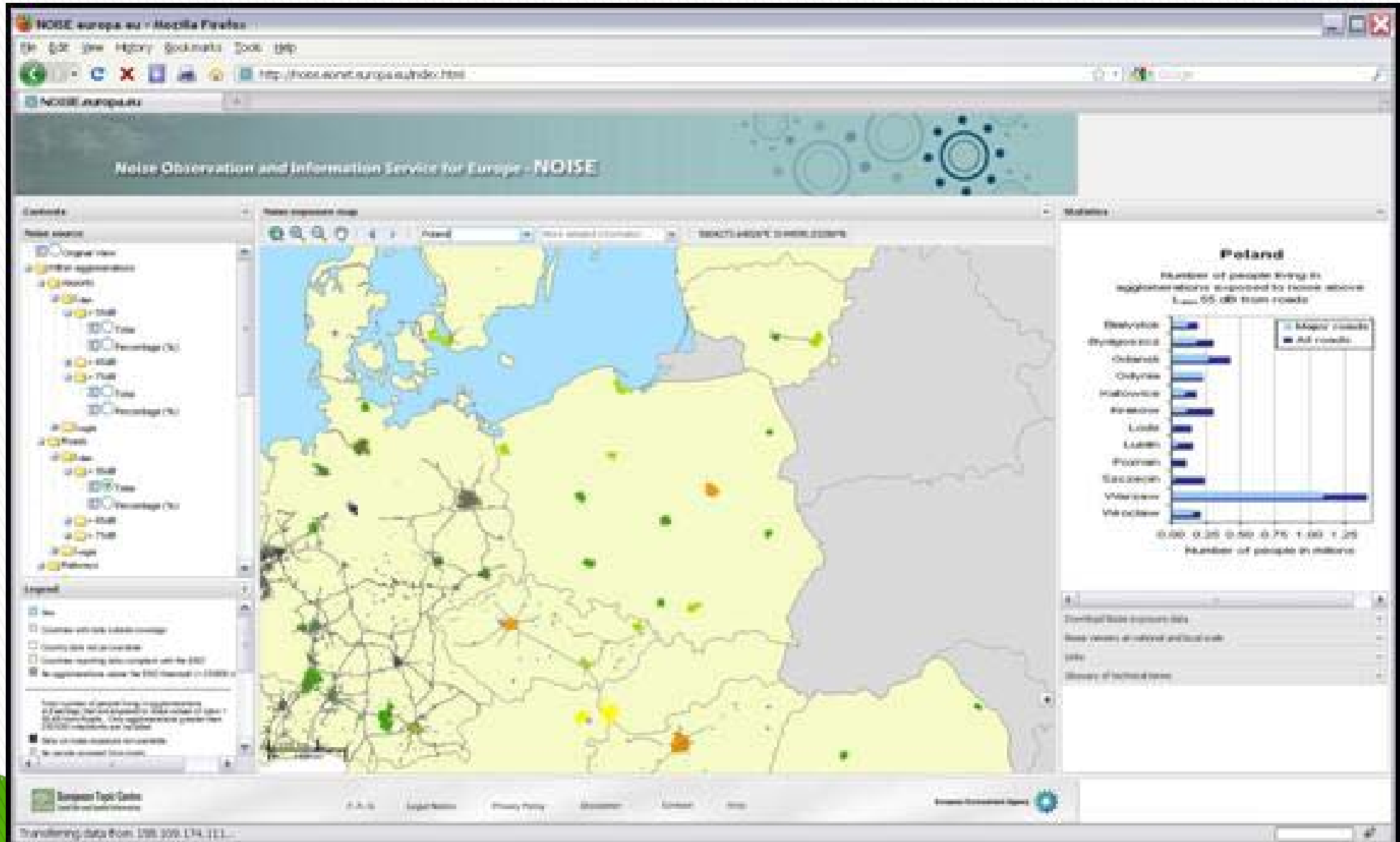
## Gràfics estadístics inicials:



Proves

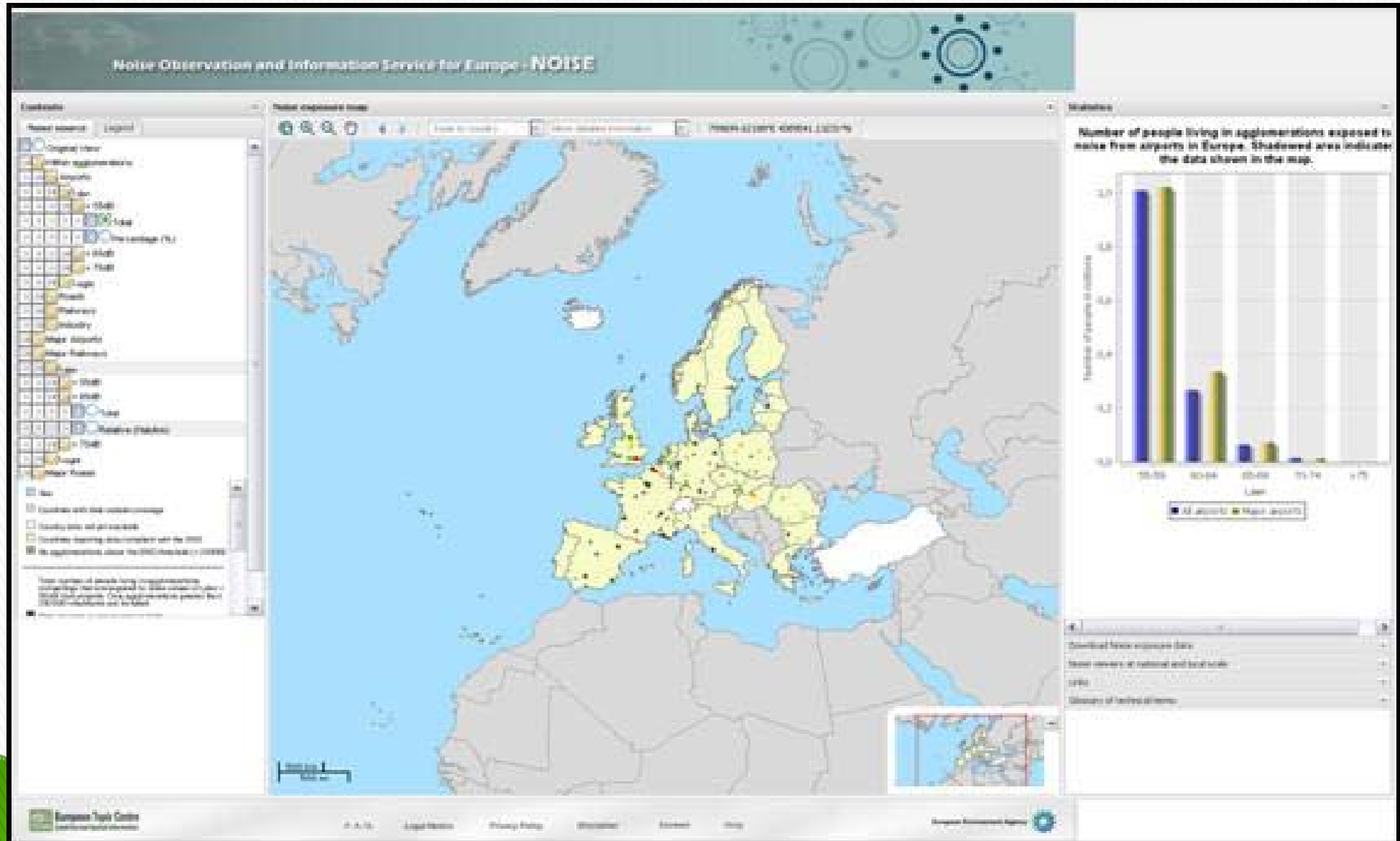
Resultats

## Gràfics estadístics inicials:





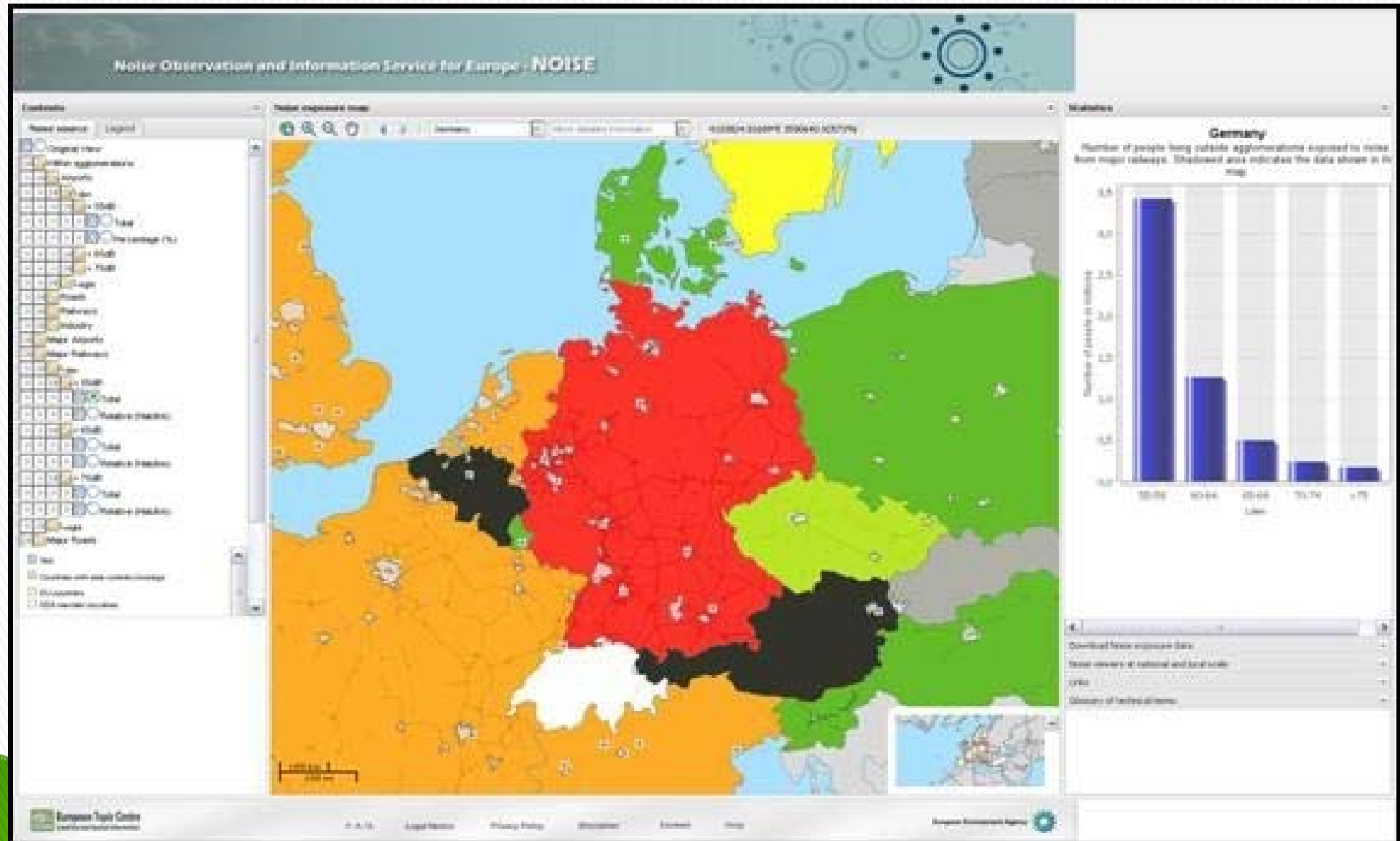
## Gràfics estadístics finals:



Proves

Resultats

## Gràfics estadístics finals:



Proves

Resultats

## Gràfics estadístics finals:



### Conclusions:

- ▶ Importància de conèixer els **objectius** de l'ETC-LUSI pel que respecta el mòdul que havia de generar els gràfics.
- ▶ La **fase d'anàlisi** ha estat clau per entendre el funcionament del visor i ha marcat els passos que s'haurien de seguir posteriorment.
- ▶ Els **casos d'ús** ha estat força complex de realitzar ja que calia tenir molt clar quines eren totes les peticions que podia fer un usuari a la web i quin tipus de gràfic havia d'obtenir.
- ▶ La fase de **desenvolupament** ha estat la més llarga:
  - a) És aquí on s'ha programat el procés que genera els gràfics estadístics de forma automàtica.
  - b) Funcionament i paper de cada un dels diferents servidors.
  - c) Escollir la base de dades més adequada per a un funcionament ràpid, sòlid i amb suport per a un gran nombre de dades dins d'un servidor.

## Conclusions

- ▶ La implementació del *servlet* ha estat la part més ràpida i senzilla. S'ha vist la importància de treballar des d'un bon principi amb els mateixos codis d'imatge tant en el codi Java com JavaScript.
- ▶ Suposarà un gran estalvi de temps i recursos a l'hora de crear els gràfics que a l'acabar el present projecte es generen automàticament.
- ▶ Suposarà una manera molt més ràpida de poder actualitzar les dades a la web.

# Moltes gràcies!

MÀSTER EN TECNOLOGIES DE LA INFORMACIÓ GEOGRÀFICA

11<sup>a</sup> EDICIÓ

Montserrat Oller Garcia