Automated Daily Mortality Surveillance Systems
Integration of Data Collection, Data Analysis and Reporting System Components

Tiago Rafael Santos Chaves

Thesis to obtain the Master of Science Degree in

Biomedical Engineering

Supervisors: Professor Mário Jorge Costa Gaspar da Silva
Doctor Baltazar Emanuel Guerreiro Nunes Bravo Nunes

Examination Committee

Chairperson: Professor Ana Luísa Nobre Fred
Supervisor: Professor Mário Jorge Costa Gaspar da Silva
Members of the Committee: Professor Paulo Jorge da Silva Nogueira

July 2015
Life can only be understood backwards; but it must be lived forwards.

Søren Kierkegaard
Acknowledgments

Firstly, I would like to express my thanks to Professor Mário Silva for all his availability to provide guidance and support through all my doubts. His vast knowledge taught me many new things for which I am very grateful.

Secondly, I would want to thank Doctor Baltazar Nunes, for all the useful information and knowledge that were crucial for the development of this thesis.

To the Epidemiology Department of the National Health Institute Dr. Ricardo Jorge staff, in particular Susana Silva, I express my many thanks for all the help, availability in clearing my doubts and the way I was welcomed in the Department. This helped me to overcome a lot of the challenges that I faced.

To my dearest colleagues David Machado, Hugo Paixão, João Ramalhinho and Ricardo Filipe, I would like to say how grateful I am for sharing this long journey with you, and may another journey follow.

To my long date friends from primary and high school, I express my many thanks for being there in the right moments and for all the support and inspiration.

To my family, I love you and could not complete this journey without you.
Abstract

The daily mortality surveillance system (VDM) monitors the observed all-cause mortality in Portugal. VDM is developed by the National Health Institute Dr. Ricardo Jorge (INSA). The system monitors the effects that disease outbreaks, extreme weather conditions or other health-related events have on the Portuguese mortality, by detecting and estimating the excess deaths associated with such events. The workflow of the system includes data collection, analysis and report generation activities. The resulting statistics and reports are disseminated to multiple authorities, at the national and European level.

This dissertation presents the development of a new version of VDM, with new added features that substantially optimize it and enable more effective data analysis. Additions include the calculation of the expected mortality baselines for each data stratification, both on daily and weekly aggregations, the implementation of a nowcasting approach to predict upcoming death notifications and an automatic alert system for excess deaths. In the remodeled VDM system, public health practitioners now have a more user-friendly interface for generation and visualization of data for each stratification against the respective baseline, within a user defined time period. Report generation is now performed automatically, producing documents ready to be sent to the authorities requesting the mortality reports. The new version of the VDM software was user-validated and is now in production at INSA.

Keywords

Mortality Surveillance, Surveillance System, Public Health Informatics
Resumo

O sistema de vigilância diária da mortalidade (VDM) monitoriza a mortalidade por todas as causas observada em Portugal. O VDM é desenvolvido pelo Instituto Nacional Dr. Ricardo Jorge (INSA). O sistema monitoriza os efeitos que surtos epidémicos, condições meteorológicas extremas ou outros eventos relacionados com a saúde têm na mortalidade Portuguesa, ao detectar e estimar o excesso de óbitos associado a estes. O processamento realizado pelo sistema inclui as componentes de colecção de dados, análise e geração de relatórios. As estatísticas e relatórios obtidos são dissemnados por múltiplas autoridades, a nível nacional e europeu.

Esta dissertacção apresenta o desenvolvimento de uma nova versão do VDM, com funcionalidades adicionais que o melhoram de forma substancial e permitem uma análise de dados mais eficaz. As novas funções incluem o cálculo de linhas de base para a mortalidade esperada para cada estratificação de dados, tanto em agregações diárias como semanais, a implementação de um método de nowcasting para prever e corrigir o número de óbitos ainda não notificados e um módulo de geração automática de alertas. No sistema VDM remodelado, os profissionais de saúde pública têm agora uma interface mais amigável para a geração e visualização de dados de cada estratificação contra a respectiva linha de base, dentro de um período de tempo definido pelo utilizador. A geração de relatórios é agora realizada automaticamente, sendo produzidos documentos prontos a serem enviados às autoridades que os requisitam. A nova versão do software do VDM foi validada pelos utilizadores e está agora em produção no INSA.

Palavras Chave

Vigilância da Mortalidade, Sistema de Vigilância, Informática da Saúde Pública
Contents

1 Introduction 1
   1.1 Motivation ....................................................... 2
   1.2 Goals ............................................................ 3
   1.3 Results ........................................................... 3
   1.4 Methodology ..................................................... 4
   1.5 Dissertation Outline ......................................... 5

2 State of the Art 7
   2.1 Public Health ................................................... 8
   2.2 Surveillance Systems ......................................... 9
   2.3 Mortality Monitoring ......................................... 10
   2.4 European Monitoring of Excess Mortality .................. 11
   2.5 Automated Daily Mortality Surveillance Systems around the World 12
   2.6 Automated Daily Mortality Surveillance Systems in Portugal ........................................... 16
   2.7 Summary .......................................................... 17

3 VDM - Daily Mortality Surveillance System 19
   3.1 VDM Development ............................................... 20
   3.2 System Components ............................................ 22
      3.2.1 Data Collection ........................................... 22
      3.2.2 Data Analysis ............................................. 23
      3.2.3 Reporting ................................................ 24
   3.3 System Architecture ........................................... 25
   3.4 Front-End Application ......................................... 26
      3.4.1 Base Software .............................................. 26
      3.4.2 Information Architecture ................................. 26
      3.4.3 Information Presented ................................... 26
   3.5 Database Model ................................................. 27
   3.6 Conclusions and Summary .................................... 29

4 Proposed Solution and Implementation 31
   4.1 New VDM System ............................................... 32
List of Figures

1.1 Conceptualization of the new VDM version and system components. 4

2.1 Canadian mortality surveillance system screenshot. 16

3.1 Mortality data flow used by the VDM in 2003. 21
3.2 Mortality data flow used by the VDM in 2007. 22
3.3 Daily workflow of VDM-2007. 23
3.4 System deployment architecture of the VDM-2007. 25
3.5 Screenshot of ICAROweb navigation bar. 26
3.6 ICAROweb information architecture. 27

4.1 New VDM system deployment architecture. 34
4.2 Screenshot of VDMweb navigation bar and its modules. 34
4.3 VDMweb information architecture. 35
4.4 (a) Screenshots of the daily mortality records file upload button. (b) Screenshot of the daily mortality records file upload button with an error. 39
4.5 Data ingestion flow, from the front-end into the VDM database. 40
4.6 Daily Mortality Surveillance and Weekly Mortality Surveillance web pages buttons. 41
4.7 Overview of all the parameter controls in Daily Mortality Surveillance and Weekly Mortality Surveillance. 42
4.8 First page of the dynamic table displayed in the Daily page, by ARS region. 43
4.9 Reporting web pages buttons. 46
4.10 (a) Screenshot of the plot of total daily deaths in the Daily web page of the Reporting module. (b) Screenshot of the plot of total weekly deaths in the Weekly web page of the Reporting module. 47
4.12 Screenshot of the parameters in the Bulletin page in the Reporting module, to generate the daily bulletin. 50
4.13 Alerts web pages buttons. 52
4.14 Table of alerts for total weekly data. 53
4.15 Example of the rules used in the VDMweb alert system. 55
4.16 (a) Nowcasting web pages buttons. (b) Screenshot of the daily plot of observed deaths and nowcasts in Nowcasting. 56
4.17 *Data Management* web pages buttons. ........................................ 58

4.18 User-defined parameters for the generation of daily baselines in the *Data Management*. 59

5.1 (a) Annual comparison plot of the number of deaths over the last 5 years. (b) Annual comparison plot of 2015 and years with high mortality values in January. ............... 66

5.2 (a) Plot of weekly number of deaths between 2013-05-29 and 2015-05-25. (b) Plot of weekly number of deaths between 2014-09-28 and 2015-05-10. ............... 67

5.3 Table of weekly alerts for total data between 2014-12-15 and 2015-05-10. ................ 67

5.4 (a) Comparison table of weekly alerts by ARS regions between 2014-12-15 and 2015-05-10. (b) Comparison table of weekly alerts by NUTS II regions between 2014-12-15 and 2015-05-10. ................ 69

5.5 (a) Comparison table of weekly alerts by age groups between 2014-12-15 and 2015-05-10. (b) Comparison table of weekly alerts by gender between 2014-12-15 and 2015-05-10. ................ 70

5.6 Plots of weekly number of deaths by gender during the influenza epidemic. ............. 71

5.7 Plots of weekly number of deaths by ARSs during the influenza epidemic. ............... 72

5.8 Plots of weekly number of deaths by the eldest age groups during the influenza epidemic 73

5.9 Weekly *Customized* table with absolute values for different strata, between 2014-12-15 and 2015-05-10. ....................................................... 74

A.1 Table of numerical information provided in the previous VDM version .................. A-2

A.2 Page 1 of the previous daily bulletin disseminated by INSA .......................... A-3

A.3 Page 2 of the previous daily bulletin disseminated by INSA .......................... A-4

B.1 Table of numerical information using average values as the comparison reference. . . B-2

B.2 Page 1 of the daily bulletin disseminated by INSA ........................................ B-3

B.3 Page 2 of the daily bulletin disseminated by INSA ........................................ B-4

B.4 Page 3 of the daily bulletin disseminated by INSA ........................................ B-5

B.5 Page 4 of the daily bulletin disseminated by INSA ........................................ B-6

B.6 Page 5 of the daily bulletin disseminated by INSA ........................................ B-7

C.1 Plot of daily number of deaths 2011-01-01 and 2013-05-25 .......................... C-2

C.2 Plot of daily number of deaths during the 2013 heat-wave. .......................... C-2

C.3 Table of alerts for total daily deaths between 2013-06-25 and 2013-07-21. ............... C-3

C.4 Comparison table of alerts for daily data by ARS regions between 2013-06-25 and 2013-07-21. ....................................................... C-3

C.5 Comparison table of alerts for daily data by NUTS regions between 2013-06-25 and 2013-07-21. ....................................................... C-4

C.6 Comparison table of alerts for daily data by age groups between 2013-06-25 and 2013-07-21. ....................................................... C-4

# List of Tables

2.1 Activation year of European mortality surveillance systems and their oldest historical data. .................................................. 12

2.2 Mode and frequency of data collection of European mortality surveillance systems. .................................................. 13

2.3 Mode, frequency of data dissemination and period of aggregation for disseminated data of European mortality surveillance systems. .................................................. 13

2.4 Variables collected in the mortality records file, by European mortality surveillance systems. .................................................. 13

3.1 Database table Deaths and description of its fields. .................................................. 28

4.1 New database table Regions and description of its fields. .................................................. 36

4.2 New database view vwDeaths and description of its fields. .................................................. 37

4.3 New database view vwWeekly and description of its fields. .................................................. 38
Abbreviations

ANPC  National Authority of Civil Protection - Autoridade Nacional de Protecção Civil
ARSC  Regional Administration of Health - Administração Regional de Saúde
CRO  Civil Registry Office
CSV  Comma Separated Value
DEP  Epidemiology Department - Departamento de Epidemiologia
DGS  Directorate-General of Health - Direcção-Geral da Saúde
EuroMOMO European Monitoring of Excess Mortality for Public Health Action
ICARO Importance of Heat and its Repercussion on Mortality - Importância do Calor: Repercussão nos Óbitos
ILI  Influenza-like Illness
IGFEJ  Institute of Financial Management and Equipment of Justice - Instituto de Gestão Financeira e Equipamentos de Justiça
INSA  National Health Institute Dr. Ricardo Jorge - Instituto Nacional de Saúde Dr. Ricardo Jorge
IPMA  Portuguese Sea and Atmosphere Institute - Instituto Português do Mar e Atmosfera
IRN  Institute of Registries and Notaries - Instituto dos Registos e Notariado
ISO  International Organization for Standardization
SIRIC Integrated System for Civil Registries and Identification - Sistema Integrado do Registo e Identificação Civil
SP  Stored Procedure
UI  User Interface
VDM  Daily Mortality Surveillance - Vigilância Diária da Mortalidade
XML  eXtensible Markup Language
WHO  World Health Organization
1. Introduction

Contents

1.1 Motivation ................................................................. 2
1.2 Goals ................................................................. 3
1.3 Results ................................................................. 3
1.4 Methodology ............................................................ 4
1.5 Dissertation Outline .................................................... 5
1.1 Motivation

Throughout history mortality data has been used to measure the burden of diseases and, ultimately, to assess the potential impacts of future epidemics. Being one of the oldest disease registries, mortality surveillance plays an important role in public health. It provides data to support the planning of public health programmes and regional and national health policies. This enables the implementation and evaluation of public health actions, aiming to mitigate the impacts of future health-related events leading to deaths. Therefore, it is important to analyse these data in order to provide adequate and timely epidemiologic information to decision makers through surveillance systems. To serve these purposes, the National Health Institute Dr. Ricardo Jorge (INSA) developed the Daily Mortality Surveillance (VDM) system, which tracks all-cause mortality. The system receives information on a daily basis, with a national coverage, and disseminates a daily bulletin to multiple entities, the State Secretary of the Ministry of Health, the Directorate-General of Health (DGS), the National Authority of Civil Protection (ANPC), the Regional Administration of Health (ARS) and the Portuguese Sea and Atmosphere Institute (IPMA). Besides, it also participates in the European Monitoring of Excess Mortality for Public Health Action (EuroMOMO) project.

VDM is used to detect and estimate the magnitude of deaths that are caused by different health-related events, such as influenza epidemics and extreme weather conditions. The system is therefore an important tool for public health action. VDM is organized in three major system components:

1. Data collection, consisting of harvesting data and storage in a database;
2. Data analysis, performed through charts and a statistical algorithm to determine the expected mortality for weekly aggregated data. Daily and weekly data are analysed by different stratifications: age groups, regions and gender;
3. Reporting, related to the production and dissemination of reports with monitoring outcomes;

However, there were a few requirements, identified through recent information requests from public entities, that the system couldn’t provide. For instance, the ARS were interested in getting accurate information about their areas of influence and the system was assigning NUTS regions only. Besides, the workflow of the system, within the INSA intra-network, was not very efficient. To public health practitioners that work with the system in a daily basis it was not very user-friendly, requiring many operator supervised data exchanges from one software module to another to perform different tasks. This makes the system patchy and not fully automated, especially regarding the generation of reports that required multiple interventions. There was a web application (ICAROweb) implemented to ease the data exchange between users and the VDM database, but it was not harnessed to the fullest, given that it only serves as an intermediary between the database records and the analysis/reporting tools.

With the wider availability of new software frameworks, it should be possible to group these multiple components into one, improving the system by also providing new and better features. Therefore, remodelling the VDM front-end so that it automatically integrates all the components would bring

http://www.euromomo.eu
benefits both to the user and to the system efficiency point-of-view. The data analysis could be complemented with new additional features, such as baselines for all daily and weekly stratifications, the integration of a daily nowcasting algorithm to correct the number of deaths that had yet to be notified, and a new automatic alert system designed for all daily and weekly stratifications. Such features should result on an improvement of the system by adding robustness to the information, now more detailed, reducing the resources needed and providing timely and faster information.

1.2 Goals

The main goal of this project was to develop a new integrated version of all the VDM system components: data collection, data analysis and reporting. This includes the following:

1. Data ingestion through the front-end application;
2. Data visualization of all stratifications, regions by ARS and user-defined age groups;
3. Generation of baselines for all stratifications, daily and weekly, and display of dynamic charts;
4. Integration of a nowcasting method;
5. Implementation of an automatic alert system;
6. Automatic generation of reports;

In short, the work entailed remodelling VDM into a more complete, user-friendly, intuitive and customized tool that INSA officers could use to retrieve data faster and obtain more comprehensive information.

1.3 Results

INSA now has the new VDM software deployed in its intra-network. All system functions are performed automatically through the application environment, speeding up the daily monitoring process carried out by INSA public health experts. I also revised and updated the VDM database scheme so that it can handle all the new implemented features.

Data ingestion into the database is now performed through the VDM front-end. In this new application of the VDM system (VDMweb), data analysis is carried out by an R\(^2\) software environment, which is an open-source software platform for statistical data analysis, implemented within the system. This environment runs scripts developed to implement the generation of baselines, nowcasting values and alerts. As INSA’s public health experts are used to the R functional language and the scripts are independent from the VDM environment, these scripts can be developed in an R environment and updated by the experts, to be loaded to the VDM system with direct outcomes in the front-end data display. Regarding reports, the intended feature was successfully implemented and these are now automatically generated. VDM and its three major system components, alongside the data visualization, are depicted in Figure 1.1.

\(^{2}\)http://www.r-project.org
1.4 Methodology

The methodology adopted for the development of this thesis has followed the activities listed below:

1. Bibliography review and thorough analysis of the VDM system. This activity was carried out to obtain information about mortality monitoring and international systems implemented for that purpose. The VDM analysis was based on bibliography review and personal interaction with the system and discussion with its daily users in INSA.

2. Replication of the VDM database structure and application in the development environment. Prior to the development phase the VDM system was replicated in my personal environment, in which I created simulated data to test the system and better understand its software infrastructure.

3. Development of the new system. This development was based on an agile methodology of software development [1], in which new components were added incrementally to the existing system. Requirements were first gathered by meeting with INSA public health practitioners and then the development proceeded in cycles of four stages:
   (a) Software design, to implement the identified requirements for the component;
   (b) Modification of the VDM database to implement new features;
   (c) Coding of the new components;
   (d) End-user validation. Each implemented module was reviewed by an end user. At the end of each cycle, if user feedback demanded revision or additional improvements the cycle was repeated before moving to the next component;

4. Evaluation and deployment of the final version at INSA. The application was subject to an end-user testing phase in a shared development environment with the previous version. Later it was deployed into the exploitation environment, replacing the old version which is now discontinued.
1.5 Dissertation Outline

This dissertation is divided in 6 chapters:

- Chapter 2 introduces mortality monitoring systems for public health and similar international systems designed for that purpose;
- Chapter 3 reviews the previous VDM system, detailing its features;
- Chapter 4 presents the software architecture and design of the new version of VDM;
- Chapter 5 highlights the new capabilities and uses it to showcase the recent January influenza epidemic;
- Chapter 6 summarizes the work done, states the final conclusions and provides an outlook for future work.
# State of the Art

## Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Public Health</td>
<td>8</td>
</tr>
<tr>
<td>2.2</td>
<td>Surveillance Systems</td>
<td>9</td>
</tr>
<tr>
<td>2.3</td>
<td>Mortality Monitoring</td>
<td>10</td>
</tr>
<tr>
<td>2.4</td>
<td>European Monitoring of Excess Mortality</td>
<td>11</td>
</tr>
<tr>
<td>2.5</td>
<td>Automated Daily Mortality Surveillance Systems around the World</td>
<td>12</td>
</tr>
<tr>
<td>2.6</td>
<td>Automated Daily Mortality Surveillance Systems in Portugal</td>
<td>16</td>
</tr>
<tr>
<td>2.7</td>
<td>Summary</td>
<td>17</td>
</tr>
</tbody>
</table>
This chapter reviews the mortality surveillance systems implemented nowadays and their role in the public health. Section 2.1 defines the concept of public health and its main indicators. Section 2.2 presents the main features and purposes of surveillance systems, in general, and the main types of surveillance performed in public health. Section 2.3 points to the importance of mortality monitoring. Section 2.4 presents the European mortality monitoring project, the EuroMOMO. Section 2.5 reviews international mortality surveillance systems, implemented nowadays, along with their main features. Section 2.6 presents the other daily mortality surveillance system operating in Portugal, the eVM. The final section summarizes the chapter.

2.1 Public Health

"Public health refers to all organized measures (whether public or private) to prevent disease, promote health, and prolong life among the population as a whole." - WHO

Public health is related to the health status assessment of a population. Its intervention is focused on improving both the quality and extension of life by reducing the disease and treating other physical and mental conditions. It targets communities rather than individuals, hence why it is seen as a collective effort to make the communities healthier. Besides, it does not rely on single actions as consequences of certain knowledge, but rather on a combination of continuously developed sciences and social approaches. These approaches differ from clinical medicine by not focusing on the treatment of the individual, but instead on the prevention and actions in different social and environmental determinants of diseases [2, 3].

The definition of public health has evolved through time, as well as its practice with the successive science breakthroughs, but its problems remained the same. However, there was not a consensual definitive list of public health functions. To answer that, the concept "essential public health functions" (EPHFs) was created and refined with the objective of obtaining an internationally consensus on public health core features. This was conducted by an international Delphi study in 1997 after a recommendation from the World Health Organization (WHO) Executive Board to work on this matter [4]. This study resulted in an EPHFs categorization and the definition of its core functions with a high level of consensus ordered by a ranked prioritization, being the top 10 the following ones:

1. Immunization.
2. Monitoring of morbidity and mortality.
3. Disease outbreak control.
4. Disease surveillance.
5. Promotion of community involvement in health.
6. Monitoring the determinants of health.
7. Production and protection of, and access to, safe water.
8. Control of food quality and safety.

10. Evaluation of the effectiveness of promotion, prevention and service programs.

Nowadays, to assess the quality of the public health there are numerous global health indicators that can be evaluated. As they are measurable, they can be used to estimate the impact of a health problem, or even to provide information to decision makers regarding their health interventions by evaluating their effects on the population. The WHO mentions these indicators in its yearly report “World Health Statistics”, available online on its website, from which I highlight life expectancy and mortality, directly related to this project [5].

2.2 Surveillance Systems

Surveillance systems consist on the ongoing systematic and continuous collection, analysis and interpretation of health-related data needed to plan, implement and evaluate the public health practice, and then disseminating those data to the authorities responsible for making the decisions that affect the prevention and control of both diseases and injury [6,7]. By doing so, surveillance is an important tool to either provide information upon which the decision makers can rely on to decide interventions or to get feedback information about the effects of those intervention strategies [8]. Besides that, it can also serve as a warning system to detect outbreaks or other hazardous events, which can lead to immediate public health actions, and support multiple research hypotheses, by figuring out possible correlations between some abnormal observed indicators [6,9]. Furthermore, it can also be used to prioritize population age groups or geographical areas to target interventions, if its data stratification is processed at that level. At the European level, the NUTS classification (Nomenclature des Unités Territoriales Statistiques - Nomenclature of Territorial Units for Statistics) is commonly used. NUTS is a nomenclature designed to divide the geographical territory of each of the UE members, in single and coherent regions to better compile European statistics [10]. It consists of a hierarchical system with three levels, I, II and III, where I corresponds to the largest division (major socio-economic regions) and III the smallest. The frequency of data collection of these systems varies depending on what their public health objective is. For instance, monitoring epidemics of acute diseases requires systems capable of providing rapid information, while on the opposite side, the surveillance of chronic diseases can be done once a year, or maybe more [8]. When monitoring by week, the most commonly used standard in the surveillance systems is the one defined by the International Organization for Standardization (ISO) week system, part of the ISO 8601 international standard. So, put together, surveillance systems are determined by how frequently and what type of information is expected from them.

Surveillance can either be passive, when the data is collected from reports sent to a central institution and then obtained by the surveillance system, or active, when data is collected from regular direct contacts with health care providers or population to collect data [8]. Passive surveillance is usually related to reports of deaths, disease registries, hospital records and physician billing systems.
As it is relatively cheap to maintain, many countries have implemented this type of systems to report outbreaks. On the other hand, active systems refer to sentinel systems, serial health surveys and database linkages, which can become expensive if for instance a lot of health surveys with high sample size and direct measures like health examination surveys are realized [11].

One of the most important features in the public health surveillance systems is timeliness. This is related to the time period between the event occurrence and the intervention carried out in response to the event, so it depends on how much time each step lasts on the surveillance system. The system response celerity depends on the type of health-related event being monitored, while in the determination of timeliness it must be taken into account what is being analysed: for instance, in acute or infectious diseases, the time interval generally starts around the date of exposure, while in chronic diseases it is more useful looking to the elapsed time since the diagnosis. Shortening this time is an important matter and a recurrent problem all over the world in surveillance systems, given the delays that usually occur in the notification of events, which depend on many factors. These factors include whether the date is a weekday or not, holidays, the patients recognition of symptoms, the attending physician's diagnosis or even the submission of a laboratory test, among others. The increasing use of electronic data collection and web-based systems promotes timeliness, but it is important to include nowcasting methods in these systems to predict the number of events in the present, near past or near future still to be notified, taking into account the already collected but incomplete information from the surveillance systems [7].

2.3 Mortality Monitoring

Mortality is the final event of the most serious outcome of a disease, hence it is one of the most important indicators of health. The epidemiological value of mortality monitoring is crucial when it comes to providing health information to support effective health planning and action. Mortality surveillance systems, as history indicates, are one of the first public health surveillance systems. Today, they are established within most nations and can be used to cross information between them. Therefore, developing continental networks for this type of surveillance is viable and important since data pooling increases the ability to detect disease trends which are not restricted by borders and can rapidly propagate (e.g. pandemic influenza, AIDS, SARS) [12].

Distribution of mortality through time is regarded as seasonal, mostly due to the effects of influenza epidemics (reaching peaks more often in the winter months) among the elderly and other vulnerable groups of the population. Therefore, monitoring it during epidemics provides timely information to decision makers on whether and when to increase the use of antiviral and vaccines [13]. Besides influenza, monitoring mortality can also be used to detect excess peaks related to environmental conditions, for instance heat waves, cold snaps and the release of biological agents, like several mortality systems across Portugal [14], France [15], UK [16] and the United States [17] have proved its worth. The timeliness of mortality monitoring is vital to guide public health measures, for instance focusing interventions on vulnerable groups. However, there are a lot of parameters and data stratification
that can be done, and each country may present information differently from each other, so the risk of European countries sharing incompatible information is high and that hampers the need to cross information and delineate a broader public health approach. Thus, planning a uniform approach to assess mortality during major health crisis affecting several continental regions benefits a timely mapping of the impact of health threats [18].

2.4 European Monitoring of Excess Mortality

The European Monitoring of Excess Mortality project (EuroMOMO), is an European initiative for pooling mortality data of the excess number of deaths. EuroMOMO was implemented in 2008, in the department of epidemiology at the Statens Serum Institut (SSI) in Copenhagen, originally funded by the European Union Health and later by the European Centre for Disease Prevention and Control (ECDC) [19]. It received information from up to 16 European countries, and after proving its usefulness during the 2009/2010 influenza H1N1 pandemic, it has remained active since then. Nowadays, it receives data from up to 19 European countries or regions of countries. EuroMOMO uses all-cause mortality because it is a strong uniform indicator that avoids issues related to delays, due to the registration of the cause of death, and uncertainties regarding under-reporting of influenza and other infections as the cause of death [13].

Presently, EuroMOMO releases a weekly all-cause mortality bulletin with its analysis, consisting of the observed number of deaths, expected number of deaths (baseline), the deviation from the baseline to detect the excess deaths, in the form of a standardized variation around the baseline (z-score) and the number of deaths corrected for the delay in data transmission. To generate the bulletin, the coordinating team at SSI receives information from its partners (mostly the countries’ national health institutes). The information must meet minimum requirements, which involve knowledge of the number of observed deaths, the baseline, a weekly periodicity and the ability to disaggregate data by region [20]. The information is exchanged on a weekly basis every Thursday.

The algorithm used to calculate the baseline of the expected weekly number of deaths uses a generalized linear model (GLM) of the Poisson regression family, accounting for over-dispersion, using a trend and two sine components. The input data to apply this algorithm must contain historical information from the last three to five years. This way, the model is fitted to this historical period with the prediction of deaths. Then, the algorithm forecasts predictions for a defined time period, not taking into account events associated with death peaks which would influence the baseline estimation [20, 21]. Users can also make small changes to the algorithm, according to the characteristics of the mortality time series to study, especially in some data stratifications with low counts of deaths.

The delay corresponds to the week difference between the week of a death and the week that the death registry was received at the partner institutes. To correct the delay on data transmission, another EuroMOMO algorithm is also computed on a weekly basis, now using the entire valid historical period of the latest update on each national mortality data. This algorithm is based on a binomial regression to model the proportion of deaths already registered in a certain week, according to the
number of days that the administration offices were closed. After that, the real number of deaths of the week in question is modelled with the proportion model, the number of deaths reported and a trend using another GLM of the Poisson family. Then the "corrected" numbers of deaths are predicted for the weeks requiring the respective correction for delays, providing more plausible analysis and faster detection of mortality variation [20].

2.5 Automated Daily Mortality Surveillance Systems around the World

EuroMOMO, as responsible for the European mortality monitoring network, is in close contact with European countries and their own mortality surveillance systems, having done in 2009 an inventory of such systems. This inventory was the result of a 2008 survey about existing and planned mortality surveillance systems within 32 countries [21]. Its results showed that seven of them had functioning mortality surveillance systems (Belgium, France, Germany, Italy, Portugal, Spain and Switzerland), with 2 of them having two systems (France and Italy). Several other countries also had this type of system, but it was either on a pilot (Denmark, Germany, Hungary, Ireland, the Netherlands and Scotland) or a planning phase (Greece, Sweden and the United Kingdom) [21]. It is worth noticing that all of the already established mortality surveillance systems are located in Western Europe. In the Eastern part only the Hungarian was reported, still in pilot phase. Regarding the current status of the systems in planning phase, I did not find recent information about them. However, in the 2009 inventory, it was already expected that the Greek system would become operational in 2009. As for the Swedish system, it was reported in 2006, before the survey, the availability of an all-cause mortality surveillance, used as a warning system. The system used threshold algorithms, similar to the Poisson regression algorithm used by EuroMOMO, to calculate the expected mortality baseline [21][22].

The year on which each active system was activated and the oldest historical data that the system stores, are presented in Table 2.1.

<table>
<thead>
<tr>
<th>System</th>
<th>Activation year</th>
<th>Oldest data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Belgium</td>
<td>2005</td>
<td>1985</td>
</tr>
<tr>
<td>France-I</td>
<td>2004</td>
<td>NA</td>
</tr>
<tr>
<td>France-II</td>
<td>2008</td>
<td>NA</td>
</tr>
<tr>
<td>Germany</td>
<td>2007</td>
<td>2006</td>
</tr>
<tr>
<td>Italy-I</td>
<td>2004</td>
<td>1995</td>
</tr>
<tr>
<td>Italy-II</td>
<td>2005</td>
<td>2003</td>
</tr>
<tr>
<td>Spain</td>
<td>2004</td>
<td>1981</td>
</tr>
<tr>
<td>Switzerland</td>
<td>2006</td>
<td>1969</td>
</tr>
</tbody>
</table>

Table 2.1: Activation year of European mortality surveillance systems and their oldest historical data.
disseminated data. Table 2.4 groups all the variables that are sent to each system, in the mortality records file. Next, the systems mentioned in the tables are briefly described.

<table>
<thead>
<tr>
<th>System</th>
<th>Mode</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>Belgium</td>
<td>E-mail</td>
<td>Weekly</td>
</tr>
<tr>
<td>France-I</td>
<td>Internet</td>
<td>Daily</td>
</tr>
<tr>
<td>France-II</td>
<td>Web portal</td>
<td>Daily, in real-time: time of death + 4 hours</td>
</tr>
<tr>
<td>Germany</td>
<td>Downloaded files submitted by the Office for Statistics</td>
<td>Weekly</td>
</tr>
<tr>
<td>Italy-I</td>
<td>E-mail, fax</td>
<td>Daily</td>
</tr>
<tr>
<td>Italy-II</td>
<td>E-mail</td>
<td>Daily</td>
</tr>
<tr>
<td>Spain</td>
<td>E-mail</td>
<td>Daily</td>
</tr>
<tr>
<td>Switzerland</td>
<td>Electronic data transfer</td>
<td>Daily</td>
</tr>
</tbody>
</table>

**Table 2.2:** Mode and frequency of data collection of European mortality surveillance systems.

<table>
<thead>
<tr>
<th>System</th>
<th>Mode</th>
<th>Frequency</th>
<th>Data Aggregation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Belgium</td>
<td>Public website</td>
<td>Weekly</td>
<td>Daily</td>
</tr>
<tr>
<td>France-I</td>
<td>Restricted website, e-mail, hard copy</td>
<td>NA</td>
<td>Weekly</td>
</tr>
<tr>
<td>France-II</td>
<td>E-mail, hard copy</td>
<td>NA</td>
<td>Weekly (daily if necessary)</td>
</tr>
<tr>
<td>Germany</td>
<td>NA</td>
<td>NA</td>
<td>Daily, weekly</td>
</tr>
<tr>
<td>Italy-I</td>
<td>E-mail, hard copy</td>
<td>NA</td>
<td>Monthly</td>
</tr>
<tr>
<td>Italy-II</td>
<td>Public website</td>
<td>3 months, annual report</td>
<td>Monthly</td>
</tr>
<tr>
<td>Spain</td>
<td>E-mail</td>
<td>Daily report, final summary report</td>
<td>Daily</td>
</tr>
<tr>
<td>Switzerland</td>
<td>Public website, hard copy</td>
<td>Yearly</td>
<td>Weekly, monthly</td>
</tr>
</tbody>
</table>

**Table 2.3:** Mode, frequency of data dissemination and period of aggregation for disseminated data of European mortality surveillance systems.

<table>
<thead>
<tr>
<th>System</th>
<th>Sex</th>
<th>Age</th>
<th>Age group</th>
<th>Marital status</th>
<th>Date of birth</th>
<th>Date of death</th>
<th>Site of death</th>
<th>Place of death</th>
<th>Residence</th>
<th>Nationality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Belgium</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>France-I</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>France-II</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Germany</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Italy-I</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Italy-II</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Spain</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Switzerland</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td>X</td>
</tr>
</tbody>
</table>

**Table 2.4:** Variables collected in the mortality records file, by European mortality surveillance systems.

The automated mortality surveillance in Belgium, BE-MOMO, aims for the early detection of outbreaks and mortality peaks in a timely manner, using baselines and setting thresholds to different mortality time series, by gender, age group, day and week [23]. These are calculated using the Farrington algorithm, a powerful statistical algorithm, provided in the R package “surveillance” using weeks in the ISO format. This general algorithm was initially designed as a robust way of routinely monitor the weekly reports of infections of several pathogens by predicting the observed number of...
counts based on historical data and then fitting it using an over dispersed Poisson GLM [24]. For recent dates, the information is incomplete due to the natural delays that occur in the notification of deaths, so to maintain the timeliness of the monitoring system the final number of deaths are estimated by extrapolating already observed number of deaths using the median of data available after the same number of registration days during the previous year [23]. The results of the BE-MOMO are disseminated through a public website, containing charts and a tabulation for excess deaths.

In France, there are two systems, the Surveillance de la Mortalité (Mortality Surveillance) - “France-I”, and the Surveillance de la Mortalité par Cause (Mortality by Cause Surveillance) - “France-II”. France-I was implemented in 2004 to identify as soon as possible early variations of mortality trends and to emit the respective alerts, while the France-II, in 2008, aims to shorten the time delay of medical causes of death availability.

The German mortality surveillance system is both funded and managed by a public health institute. From the geographical point of view, it covers regions at the NUTS I level. I found no recent information about the system population coverage after searching in the Internet, but at the time of the euroMOMO survey the system only covered the State of Hesse, corresponding to 7% of the German population. Relatively to the timeliness of the system, it was reported back then that this system had the longest median time between the date of death and the date that it was received by the surveillance system, corresponding to 10 days (the average of all the systems studied was three days) [21].

Similarly to France, Italy also has two mortality surveillance systems, the Sistema Nazionale di Sorveglianza Rapida della Mortalità (Quick National Mortality Monitoring System) - “Italy-I” - and the Sorveglianza Epidemiologica Rapida della Mortalità nelle Città Capoluogo di Regione/Provincia Autonoma (Rapid Epidemiological Surveillance of Mortality in City Capital of the Region / Autonomous Province) - “Italy-II”. Italy-I focuses on the detection of increases in mortality associated to heat-waves, mostly during summers, thus it evaluates Heat Health Watch Warning (HHWW) systems. Italy-II describes mortality in all age classes and is more general - does not focus only in heat-waves. Both receive data covering the capital cities of Italy’s 21 Regions and Autonomous Provinces. To analyse data, baselines are created by mathematical models with several variables taken into account. The results of these analysis are then sent to the appropriate institutions every 3 months or in an annual report, by e-mail and hard copy from the first system and in a public website from the second one, aggregating monthly data [21].

The Spanish system, called MOMO: Monitorizacion de la Mortalidad Diaria (MOMO: Daily Mortality Surveillance), is similar to the ones presented above. It is maintained by the Centro Nacional de Epidemiologìa del Instituto de Salud Carlos III (National Epidemiology Center of the Institute of Health Carlos III) [21, 25]. The system mostly relates excess mortality with high temperatures, reason why it collects the maximum and minimum temperature. The data collected covers the country at NUTS II and NUTS III level [21]. Its analysis are stratified by age group (<65 years, 65-74 years, ≥75 years) gender, age group and gender combined and regions including each autonomous community and each provincial capital whose municipalities were computerized [23]. Besides, the system uses a CUSUM modification and Kriging regression algorithms to calculate the expected mortality
and consequently analyse and detect changes in the mortality \[21\]. From this algorithm, the system has designed an alert system that consists of 3 status \[25\]:

- Continued situational excess, where an alert signal is continuously generated if 2 of the last 4 days register an observed mortality higher than the 99% upper confidence limit of expected mortality for those 4 days.
- Continued mortality excess, when the CUSUM threshold is exceeded in the 90th percentile of consecutive days and the observed mortality exceeds more than 3 standard deviations from the expected mortality.
- Severe mortality excess, when the observed values exceed the 95% upper confidence limit of the expected mortality.

The Swiss system, Überwachung der Sterblichkeit (Exzessmortalität) - Monitoring of Mortality (Excess Mortality) does analysis based on influenza data and climate data like maximum and minimum temperatures, producing absolute values of mortality \[21\].

Another system, the Scottish automated mortality surveillance system, uses a correction factor to correct the reporting of deaths delay based on the empirical cumulative distribution of delays, which is grouped in 4 categories: weekdays, Saturdays, Sundays and public holidays. To analyse data, the system uses two different statistical models for calculating expected mortality, one using an extended Serfling model that consists of both sine and cosine terms to capture the annual seasonality of mortality, and the other relying on a Generalised Additive Model, which corrects the flaw of the Serfling model in predicting mortality in the winter period by allowing a more flexible season pattern that includes the winter peaks \[26\].

Regarding the actual end-user applications of the systems, there is not information available about the countries systems mentioned above. An automated mortality surveillance system in South-Eastern Ontario, Canada (the Mortality Surveillance System - MSS) consists of a Java web-based application connected to an Oracle database. The system automates the data processing, classification and statistical analysis. By using the application, the public health practitioner spends about 30 minutes in analysing data. The process flow consists on receiving the municipal mortality record, transferring it to the Oracle database in a secure server by an IP-restricted data transfer via the Secure File Transfer Protocol (SFTP). Then, already within the boundaries of the health institute where the surveillance system is implemented, personal identifiers in the records are removed to maintain privacy and the mortality syndrome is classified, according to ICD-10 terminology. After getting the reference statistical values, the system uses an alert detection algorithm, that generates an alert when the mortality threshold is exceeded. A browser-based interface allows the user to plot the data of a certain syndrome on a daily or monthly basis within a definable range against an historical baseline and the CUSUM signal, like the screenshot in Figure 2.1 of the graphical interface in a standard web browser shows \[27\].

In China another automated web-based surveillance system for outbreak detection and rapid response was implemented nationwide in 2008, the China Infectious Disease Automated-alarm and
Response System (CIDARS) developed by the Chinese Center for Disease Control and Prevention (CCDC). This system, which deals with mortality by cause of death, greatly improved data analysis, excess detection, data dissemination and communication between the major health institutes all over the country, by shortening the interval of days between case diagnosis and case reporting in one day in average, providing faster answers to the different syndrome outbreaks [28].

2.6 Automated Daily Mortality Surveillance Systems in Portugal

In Portugal, there are currently two parallel but independent mortality surveillance systems. e-Mortality Surveillance (eVM), the most recent, started its activity in October 1, 2014, hosted on a public website by DGS (Ministry of Health) [29]. The other one, Daily Mortality Surveillance (VDM), is a system initially designed in 2003, managed by INSA, that articulates with euroMOMO and feeds the health dashboard provided by DGS, besides working with several other Portuguese authorities. VDM is explained in detail in the next chapter, since it is the subject of this project. While VDM is an all-cause mortality system, eVM reports deaths by accidental or not-accidental cause (discriminating some causes) and is updated in real time. It uses information from the Information System of Death Certifications (SICO), implemented with the objective of de-materializing the notification of deaths by doctors, transforming it into an electronic communication tool to health authorities. SICO is accessed by a web portal in DGS website, and then an encoding team at DGS maps the causes of death to ICD-10, a medical terminology. Furthermore, SICO also provides anonymised data to the National
Statistical Institute of Portugal (INE) for national mortality statistics purposes, besides transmitting the certification of deaths to the Institute of Registries and Notaries (IRN) and Civil Registry Office (CRO), who also transmit to SICO the certification of deaths issued by them [30].

The SICO operation was decreed by law - dispatch n° 13788/2013 from the State Secretary of the Ministry of Health, which stated that starting from 00h on January 1 of 2014, every death should be notified on the system, accessed on a web portal via the DGS website. Due to this mandate, the timeliness of the eVM is more effective than in VDM, which receives one file compiling all the deaths recorded on the day before by the CROs, on a daily basis. All of the data analysis concerning the generation of expected mortality baselines and correlation of the mortality trends with events like influenza and extreme temperatures, as well as daily bulletins sent to the appropriate entities and weekly data to the euroMOMO project, is performed at INSA. Until April 2015, eVM had no historical data prior to January 2014, it only stored information since the launch of the SICO system. However eVM is still in development and, at the moment of the writing of this thesis, it stores information from the last six years, displaying in its public website multiple charts with different data stratifications: total daily mortality, mortality by age groups (<1, 1-4, 5-14, 15-24, 25-34, 35-44, 45-54, 55-64 and ≥65 years) and mortality by cause (natural cause, Subject to investigation and external cause, which groups traffic accident, work accident, suicide, homicide, other type of accident, ignored). The time series of the charts displayed in eVM covers one month period, and in the total daily mortality plot, with no stratification, there is an overlay of different series. Each of these series correspond to each year with mortality data stored in the system, enabling the comparison between such years. Besides the charts, eVM also provides a monthly table with the number of the week and the dates of the month of those weeks (in ISO standard), next to the total number of deaths verified in each week, divided by deceased with <28 and ≥28 days of age [29].

2.7 Summary

When trying to detect outbreaks in a timely manner, it is not only the timeliness of the data collection that matters, but also the effectiveness of analysing a great volume of data and disseminating the results in a fast way to the adequate health authorities, so they have time to plan interventions and act based on them. That urges the implementation of automated systems all over the world, a theme that is gaining importance.

The information provided by mortality surveillance systems is similar among the systems implemented all over the World, as proven by the systems used by the EuroMOMO country partners, described in this chapter. It is the way that the information is treated and analysed that is different. All of the mortality surveillance systems here described are fairly recent, with the oldest one being in fact VDM, one of the two Portuguese systems. The other system is managed by DGS, eVM, and it is hosted in its public website, with mortality information that can be accessed by the general public.

The next chapter describes the VDM system, the subject of this thesis, and how it operated.
3

VDM - Daily Mortality Surveillance System

Contents

3.1 VDM Development .................................................. 20
3.2 System Components ............................................... 22
3.3 System Architecture ............................................... 25
3.4 Front-End Application ............................................. 26
3.5 Database Model .................................................... 27
3.6 Conclusions and Summary ....................................... 29
This chapter describes the VDM system, managed by DEP at INSA. Section 3.1 describes the origin and development of the VDM system. Section 3.2 reviews the major system components of the VDM, and how the VDM-2007, the most VDM version in production before this thesis, operates. Section 3.3 describes the VDM system deployment architecture at INSA. Section 3.4 presents the VDM-2007 front-end, ICAROweb, and the information it provided. Section 3.5 reviews the most important features of the VDM database. Lastly, Section 3.6 points some limitations of the VDM-2007 and closes the chapter.

3.1 VDM Development

Public health actions require confirmation of the impacts on the population (e.g. excess mortality) of health-related events. Normally, when this is known it is already too late, given the event has already taken its course. To deal with this limitation, it is mandatory to have timely predictions of the impacts to allow implementation of public measures and policies. For this purpose, a pilot version of the VDM system was designed in 2003, to evaluate the effects of the heatwave that occurred at the time in Portugal. The system has been active since 2004, with a new version in 2007. The content of this section is based on the article written by the INSA staff involved in the creation of the system [31].

In Portugal heat-waves represent an important threat. Since 1999 a Portuguese Heat Health Watch Warning (HHWW) known as the Importance of Heat and its Repercussion on Mortality (ICARO) system, has been operating, developed in cooperation with IPMA, DGS and ANPC. It is based on statistical models that predict increases in mortality due to high temperatures resulting in the creation of an ICARO-index, where 0 means absence of risk and positive values represent risk of mortality associated with extreme and continued heat [32]. The 2003 VDM version was created to support these predictions. It collected information about the number of daily death registrations from a total of 31 CROs representing capital cities and districts in Portugal mainland, which accounted for 40% of the Portuguese mortality. VDM proved to be useful by confirming the excess mortality predicted by ICARO, so the development of VDM has followed.

Given the success of the test version, the original VDM system was launched in 2004, expanding the number of CROs to 67. The collected data consisted only of the number of deaths registered in each CRO for individuals aged 75 or more, which was sent by each CRO to INSA’s Epidemiology Department (DEP) either at the end of the day or in the morning after. However, in this early VDM version only a few offices were able to send information by e-mail, with the majority of the data being sent by fax and telephone. It functioned on weekdays during the summers and, given its success, at the end of the summer of 2005 the project was expanded to cover the full year. Back then, the mortality information from DEP was disseminated within the daily ICARO bulletin sent to the Ministry of Health, DGS and ANPC, as shown in Figure 3.1.

Regarding features, baselines were created for further data analysis, based on mortality information of the previous year. Delayed information was a problem, due to weekends and mostly to
holidays, so in 2006 a mathematical model was tested to consider the delays, however despite the seemingly accurate results it did not have all the required statistical properties. Although its proven functionality and usefulness, the system had a major limitation by using the daily number of death registrations instead of the daily number of deaths, which made it heavily dependent on the weekday, since deaths occurring in weekends and holidays are registered later. Besides, the detection and calculation of excess mortality was not straightforward.

In 2006 the Ministry of Justice implemented the Integrated System for Civil Registries and Identification (SIRIC), taking advantage of the informatization of the CROs, enabling the generation of a Portuguese national database. SIRIC connected the already computerised CROs and collected their data in a centralised way. The information included in SIRIC was obtained from IRN, which is responsible for the CROs.

With the creation of SIRIC, an automated VDM system started to be delineated. In mid-2006 IRN requested INSA to test an automated version of the VDM, by making use of a daily exchange between INSA and ITIJ of data collected by SIRIC. These data were exchanged via e-mail on a daily basis, including weekend days or holidays. It contained the following variables: date of death, age, gender and geographical code for location of death registration.

SIRIC connects all CROs since 2007, when its implementation process in national territory was concluded. Since 2010 it also covers 67 Portuguese consulate posts. This provides more infor-
3.2 System Components

The VDM system is implemented within INSA’s intranet and it has a front-end application, which in VDM-2007 is named ICAROweb, designed later, in early 2013. This application is accessed only by a restricted number of the Department of Epidemiology (DEP) staff. Besides ICAROweb, several Microsoft Office tools are used in order to complete the daily workflow of the system, for data analysis and reporting. The workflow is shown in Figure 3.3 which covers the system components described next.

3.2.1 Data Collection

Data collection by the VDM system consists on gathering of mortality records, obtained as sets of records of death from IGFEJ. Mortality records are encoded in eXtensible Markup Language (XML).
files. Once this file is obtained by INSA's professionals, they proceed to its data integration in the VDM database, a process named data ingestion. The variables contained in the daily mortality records file are listed next:

- Date of death;
- Date of registration;
- CRO code;
- CRO description;
- Geographical code of site of death;
- Date of birth;
- Gender;

After receiving the mortality records file, the responsible public health practitioner saves the file in a shared folder within INSA's intranet. Then an ASP.NET windows application named “ITIJxml2csv.exe” reads the file structure and integrates its information in the database. When this ingestion tool is called, it searches for XML files in a shared folder and its execution is scheduled by Windows Task Scheduler, on weekdays at 9:00 and from then every 10 minutes until 12:00. After the ingestion in the database, the web application ICAROweb, accessed by common browsers, displayed its data. In here, data could be exported to a CSV file, which enabled the 2 subsequent components.

### 3.2.2 Data Analysis

Data analysis corresponds to the analysis of the mortality absolute values obtained by the VDM system, aiming to detect and quantify mortality peaks and to identify, as soon as possible, when they occur. It is performed through daily and weekly plots for different stratifications: age groups, regions and gender.

In the VDM-2007, after the Comma Separated Value (CSV) file export from ICAROweb, containing all the records of death in the VDM database not grouped or stratified yet, the file is imported to a local MS Access database in the user computer. However, sometimes the exported data is directly used in analysis that don't require the data treatment of MS Access. In MS Access, the absolute values
of number of deaths are grouped by day and stratified by age groups (0-14, 15-24, 25-44, 45-64, 65-74, ≥75 years), regions by NUTS II (Norte, Lisboa, Centro, Alentejo, Algarve, Madeira and Açores), Portugal and Portugal Mainland, and gender. To assess the excess mortality by regions, the average of the observed values over the past 4 years is calculated for the past 15 days. These average values are displayed in a table, along with the observed values for each region (see Figure A.1 in Appendix A) in the respective current dates. Since the table of the example provided in the Appendix corresponds to a Monday, the previous two days - Sunday and Saturday - have 0 records. This is an example of the relation between the delay in the notification of deaths and the weekdays, particularly weekends. As CROs are closed during the latter, it is very rare to receive records on Monday about deaths that occurred in weekends since there are no records of it yet. The values of the last five days are displayed in a grey colour since this is the number of days considered to have values yet to be updated, due to the delay in the notification of deaths. The mortality variation in the numerical table in VDM-2007 is indicated by small arrows below the observed values: two upward pointing red arrows if they are greater than the corresponding average value by two standard deviations, one blue arrow if they are greater only by one standard deviation, and a blue arrow pointing downwards if the values are below it. A black square means that there is no difference between the values.

Regarding weekly data analysis, the same daily data stratifications is done, which was already provided by ICAROweb. So, in this case, data stratification is processed solely the database server. The expected mortality is independently calculated, by adjusting cyclic regression models for time series of mortality using the R library flubase [34]. These models are adjusted to a time series and result on a mortality prediction considered as the mortality baseline or expected mortality, plus a 95% upper confidence limit of the baseline. The model used to generate the baseline adjusts to an historical data of the observed mortality, excluding the periods where excess mortality events were observed, mainly heat-waves and influenza epidemics. The heat-waves events are determined by the ICARO program, which groups daily information about observed and predicted temperatures provided by IPMA. The events are identified when two consecutive days register an average temperature of 32°C or more. As for influenza events, these correspond to epidemic influenza periods reported by the National Sentinel-Network, which is a sentinel surveillance system composed of general practitioners physicians. This network provides an estimate of weekly influenza incidence rate. When this estimate is above the 95% upper confidence limit of its baseline on a set of consecutive weeks, an event is defined.

The comparison between observed and expected mortality to determine the excesses in VDM-2007 is mostly ad-hoc, in periods of known high influenza activity, to quantify its impacts.

### 3.2.3 Reporting

Reporting corresponds to the generation of reports and its dispatch to the appropriate entities. After the import and automatic data treatment in MS Access in the VDM-2007, the plots and the numerical table are automatically updated in a MS Excel whose spread sheets are connected to the user local database. These plots consist of the time series of total mortality, regions of Portugal.
Mainland and an annual time series of a mortality comparison of the past 5 years including the current one (see last two pages of Appendix A and ). After the updates of the plots, the INSA professional saves them and the numerical table in two separate PDF files and uploads them to a cloud, whose URLs are posteriorly forwarded, within a daily e-mail, to the State Secretary of the Ministry of Health, DGS, ANPC, ARS and IPMA.

### 3.3 System Architecture

The VDM-2007 system architecture in INSA’s local network consists of two independent servers: one database cluster server with MS-SQL Server 2008 R2 and one application server with the IIS 7.0 web server, .NET framework and the operative system Windows Server 2008 R2 installed. SQL Server 2008 R2 is a relational database management system responsible for managing multiple databases of the various INSA’s programs, and the IIS 7 is a web server developed by Microsoft for servers with its operating systems (OS) installed. Figure 3.4 sums up the communication between these and the client, in this case the DEP staff with access permissions to the web application. Although IIS 7.0 is mainly a web server, its integrated .NET framework enables it to host .NET applications such as ICAROweb. Hence it can also be called of an application server, which by definition runs applications at server-side dispensing it from the client, besides handling HTTP requests. In this network flow, the client posts a page request via HTTP in the User Interface (UI) of the VDM-2007 front-end into the application server, which processes the request and passes parameters to queries in the SQL Server where data is stored. As a response, the SQL Server provides the queries results to the ICAROweb application, so that they are bound to its UI and displayed to the client.

The client and the application server are connected using the TCP/IP network protocol (Transmission Control Protocol/Internet Protocol), which consists of several communication protocols layered on top of each other (Application, Transport, Internet and Network Interface) [35]. The highest in the hierarchy corresponds to the application layer, e.g. HTTP. It deals with more abstract data enabling the transfer of web pages between both ends, while the layers below it deal with less abstract data. For instance, the transport layer relies on TCP and its responsible of sequencing the stream of data packets sent in both directions of the data flow.
3.4 Front-End Application

3.4.1 Base Software

Regarding software, the VDM front-end application was ICAROweb, an ASP.NET web application accessible by browsers. Active Server Pages (ASP) is a basic structure of libraries to facilitate the processing of script languages at the server side in order to create dynamic content. ASP.NET is an open source server-side web application framework that uses ASP and is built on the Common Language Runtime (CLR) based on Microsoft’s .NET framework, hence ASP.NET code is written in any supported .NET language (C#, Visual Basic .NET, F#, ...). ICAROweb was coded using the ASP.NET Web Forms programming model, combining HTML, server controls and server code to allow the user to enter data that is sent to the server for processing.

Web Forms pages consist of two components. The first is the markup code file (the .aspx), which contains the visual portion of the page, including UI elements like buttons and text boxes among others. The second components is the code-behind page, where the logic of the web page is programmed and all of the events triggered by the UI elements are defined, for instance what clicking a button does [36]. The markup language used in ICAROweb is XHTML 1.0 and the code-behind pages are written in C#.

3.4.2 Information Architecture

As I previously stated, in VDM-2007 ICAROweb is used as an intermediary between the SQL Server that manages the VDM database and the public health practitioner. Hence it comprises a small number of features, grouped in 2 modules. These modules are displayed in the screenshot of the application navigation bar interface, in Figure 3.5.

The dynamic web pages displaying VDM data are accessed by the navigation bar. The information architecture of the application, i.e. how the pages are connected, is schemed in Figure 3.6.

3.4.3 Information Presented

The information provided by ICAROweb consists in the absolute values of mortality aggregated by ISO week and all of the individual records with the addition of their NUTS III region codes. These are assigned in the database to ease its subsequent mapping when analysing data by region.

In the Home module of ICAROweb there is a description of the ICARO program instead of the VDM, from the time that the original VDM was created. The module Daily Mortality Surveillance groups different pages containing GridView controls with the mortality data, exportable to CSV files.
These GridView controls are one example of the UI controls used in the ASP.NET Web Forms and, although they are graphic controls, they present data in a tabular view. In the Daily Records page, which displays the individual records, the information consists of:

- Death ID;
- Date of death;
- Date of registration;
- Geographical code;
- NUTS III code;
- Date of birth;
- Age;
- Gender;

Weekly stratified data is provided in more detail in the other pages to ease the analysis of the correlation between mortality and influenza incidence rate, which is analysed by week. EuroMOMO, as the name indicates, displays data to be sent to the EuroMOMO project, consisting of the date of death, date of registration and age. The set of Weekly pages provide the weekly data to be analysed by DEP and to be included in its weekly influenza bulletin. All of them provide the same basic information: first day of the week, last day of the week and the number of the week. Then, depending on the stratification, each page provides the number of deaths that occurred in each week by stratum and the total of them. The total of deaths in each stratification could differ due to incomplete information in the records, for instance lacking information about the region of death or the date of birth.

### 3.5 Database Model

All of the data obtained and provided by the VDM system is stored in a database within the MS SQL Server. Communication between this server and the application server is constant and each
time a web page is presented in an ASP.NET application, a .NET class of SQL connection is used to connect to the respective SQL database and execute a Stored Procedure (SP). SPs are a collection of SQL commands that access a relational database system, very similar to user-defined functions with the exception that they can’t be invoked within SQL statements, but instead with a CALL statement. They are pre-defined and stored in the database, hence reducing traffic in web applications given that the SQL Commands are executed directly in the server. Its implementation language in MS SQL Server is Transact-SQL (T-SQL), which allows SPs to define variables, receive parameters to be used as variables, return results, implement IF and WHILE loops, among others. To sum up, these procedures basically encapsulate the application business logic, dispensing programmers of programming that logic elsewhere.

The VDM-2007 database, named **ICAROvigilanciaMortalidade** (ICAROMortalitySurveillance), consists of 12 tables, from which I highlight the main table called **Deaths**, Table 3.1, 3 views and 9 SPs. **Deaths** store mortality information and it underlies every analysis that is made. Most of its columns coincide with the ones provided in the mortality records file, however one column, **Age**, has the datatype **Computed** since it is created on-the-fly as records are added to the table. To do that, a user-defined function determines the age of the deceased whose record is added to the table, given his date of death and date of birth.

Records are inserted in table **Deaths** using a stored procedure each time the ingestion tool is called to handle the mortality records files. A staging table, to where the ingestion tool sends data, is used to check for restrictions in the records. These consist of unique indexes implemented in the table **Deaths** to prevent the repetition of records.

Another table concerns the number of successfully inserted records from each integrated mortality records file, identified by the Record ID as a primary key. This ID is defined as an identity column, present in Table 3.1 to identify the belonging file of each record of death. The rest of the tables are regions related, defined to obtain a mapping between the geographical codes in the records and the NUTS regions. The assignment of the NUTS III code, provided in ICAROweb **Daily** page, is performed in a view considering the geographical code of six characters. In this code, the first two numbers are related to Districts, the second two to Municipalities and the last two to Parishes, enabling

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
<th>Data type</th>
</tr>
</thead>
<tbody>
<tr>
<td>idObito (PK)</td>
<td>Death ID</td>
<td>int</td>
</tr>
<tr>
<td>dataObito</td>
<td>Date of death</td>
<td>date</td>
</tr>
<tr>
<td>dataRegisto</td>
<td>Date of registration</td>
<td>date</td>
</tr>
<tr>
<td>codConservatoria</td>
<td>CRO code</td>
<td>int</td>
</tr>
<tr>
<td>descConservatoria</td>
<td>CRO description</td>
<td>varchar(255)</td>
</tr>
<tr>
<td>codGeografico</td>
<td>Geographical code of region of death</td>
<td>varchar(10)</td>
</tr>
<tr>
<td>dataNascimento</td>
<td>Date of Birth</td>
<td>date</td>
</tr>
<tr>
<td>Idade</td>
<td>Age</td>
<td>Computed, int</td>
</tr>
<tr>
<td>Sexo</td>
<td>Gender</td>
<td>varchar(1)</td>
</tr>
<tr>
<td>idRegistro</td>
<td>Record ID</td>
<td>int</td>
</tr>
</tbody>
</table>

*Table 3.1: Database table **Deaths** and description of its fields.*
different degrees of geographical depth. On the other hand, NUTS III codes consist of five numerical characters that correspond to the join of NUTS I (one number) number, and NUTS II (two numbers) plus two numbers. So, the geographical codes are mapped to NUTS III using a table containing the correspondence between the four numbers of Municipalities (or three if there is a leading zero - it is read as a number and not a string) and all the NUTS codes. This implicates the division of the geographical code by 100, to get a result with the same length of the Municipalities codes.

The SPs prepare data to be displayed at ICAROweb, using the COUNT command to aggregate data by different stratifications (e.g. by NUTS II regions). To aggregate data by week, each SP uses 2 variable tables. One table is used to store the first and last days of each week with records in the database, the other one counts records by the respective data stratification within multiple time periods of 7 days. Then a LEFT JOIN of these two, linking the first days of the week of both tables, is performed on the former to obtain a complete table with death counts by ISO week. These SPs receive as parameters the initial date and final date of the mortality time period to be provided at ICAROweb, limiting the SQL queries to dates of death between the two. Since data display is done on GridViews and these have a defined maximum number of rows and paging capability, to populate them the result of the SPs is sent to the web application and displayed in there each Gridview page size at a time.

3.6 Conclusions and Summary

The VDM system, thoroughly described through this chapter, has been functional and successfully used at INSA for years. However, from the user point of view, the workflow is extensive and it is not very user friendly. This is mostly due to the wait for the ingestion tool, to integrate the mortality records file, and the software exchanges to generate the daily bulletin.

Besides, VDM-2007 does not stratify data by ARSs, whose regions do not map completely the NUTS II regions. This represents a comprehensible problem since it is of more ARSs interest to get accurate statistics about mortality in their areas of influence. Moreover, recently there has been an addition of alphanumeric region codes that are not covered by VDM-2007. Still regarding geographical codes, sometimes these are incompletely registered (four numbers or less), for instance when CROs don’t know the exact location of death. This results in codes that in VDM-2007 are not assigned to any region.

Regarding analysis, statistical methods are only used in weekly data, such as the independent generation of the expected mortality baseline and the ad-hoc identification of alerts when needed. To answer to several governmental entities requests, involving crossing stratifications, other layers of analysis are required, which are not automatically implemented in the VDM system. Relatively to the usual delay of the notification of deaths, a recent DEP project developed a nowcasting algorithm to correct the number of observed deaths, which is not integrated into the VDM system yet. The problem of these delays is that they result in abrupt falls in the time series plot on the day of the notification of the mortality records file, as the first plot of the daily bulletin in the Appendix A shows. Since the
algorithm is fully tested and has even been presented in international meetings, the VDM system can benefit from its fully integration, in order to earlier detect the impacts of mortality.

The system reporting is not directly operated from the VDM front-end. Besides, the plots represented in the bulletin do not contemplate disaggregation by gender. This dissemination of gender stratification was recently demanded by a governmental dispatch.

The next chapter presents the proposed approach to handle the issues just discussed above.
## Proposed Solution and Implementation

### Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1</td>
<td>New VDM System</td>
<td>32</td>
</tr>
<tr>
<td>4.2</td>
<td>Modules I/II: Home and Records</td>
<td>38</td>
</tr>
<tr>
<td>4.3</td>
<td>Modules III/IV: Daily and Weekly Mortality Surveillance</td>
<td>40</td>
</tr>
<tr>
<td>4.4</td>
<td>Module V: Reporting</td>
<td>45</td>
</tr>
<tr>
<td>4.5</td>
<td>Module VI: Alerts</td>
<td>52</td>
</tr>
<tr>
<td>4.6</td>
<td>Module VII: Nowcasting</td>
<td>55</td>
</tr>
<tr>
<td>4.7</td>
<td>Module VIII: Data Management</td>
<td>58</td>
</tr>
<tr>
<td>4.8</td>
<td>Summary</td>
<td>60</td>
</tr>
</tbody>
</table>
This chapter describes the proposed solution of this project. Section 4.1 describes the main features of the proposed VDM system, as well as its technical features. The following sections are related to each module implemented in the new front-end of the system, VDMweb. Each one of these sections describe the respective module(s) UI implementation, the interaction with the database and, in the modules using it, the interaction with the R statistical environment.

4.1 New VDM System

4.1.1 Main Features

The solution developed in this dissertation entailed the reformulation of the ICAROweb front-end, making it an optimised and more robust application with data ingestion, analysis, visualization and reporting features. All these system components are available from the new application, named VDMweb. New analysis features are implemented in an R software environment and in the VDM database management system. The implementation involved redesigning the front-end, to provide users with new added features, modifying the database, to store and provide all the new information produced and required for data analysis and reporting, and developing the R environment (some of the scripts were provided by INSA). The main new functions implemented in VDM are:

- Data ingestion of the mortality records file through VDMweb;
- Creation of a new regions table in the VDM database, with correspondences between geographical codes and ARS and NUTS regions and codes. This table is editable through VDMweb;
- New daily and weekly data stratification features:
  - Age groups of 75-84 and ≥85 years and option for a dynamic user-defined group;
  - ARS regions;
  - User-customized grouping of strata (e.g. regions plus age groups) in the tables displayed;
  - User-parametrized stratification (e.g. region by gender by age group);
- Automatic generation and integration in VDM database of daily and weekly baselines for each data stratification:
  - Baseline;
  - 95% lower and upper confidence limits;
  - 99% lower and upper confidence limits;
- Online visualisation of dynamic daily and weekly mortality variation charts within a user-defined time period for daily and weekly data. Besides total values, these charts display data of the following stratifications, with its baselines:
  - Age Groups (0-14, 15-24, 25-44, 45-64, 65-74, 75-84 and ≥85 years);
  - Region by ARS or NUTS II (Norte, Centro, LVT, Alentejo, Algarve, Madeira and Açores);
  - Gender (Male and Female);
  - User-parametrized stratification;
• Annual mortality comparison plot, up to 5 user-defined years;
• Display of numerical data of the last 15 days and comparison of their mortality values to one of the following reference values, selected by user:
  – Average of the last 4 years;
  – Mortality baseline;
  – 95% upper confidence limit of the baseline;
• Automatic generation of daily bulletin with new plots to include the next stratifications:
  – Age groups;
  – ARS or NUTS II regions, including Açores and Madeira;
  – Gender;
• User-defined time periods and years to display in the bulletin charts. Baselines are also selected by the user to be displayed in the respective charts or not;
• Automatic generation of mortality alerts for each daily and weekly data stratification. Additionally, a comparison map between all strata of each stratification was also implemented;
• Integration of a nowcasting method, implemented in R. Plot display of the method results with observed, nowcasting and baseline values, and display of its alerts;

4.1.2 System Architecture

Since the purpose of the VDMweb was to replace ICAROweb, the system architecture of VDM was practically unmodified (Figure 4.1). This was an important implementation decision to INSA, because a critical requirement for the solution was that it should not require any additional hardware, as well as it should ease the new system maintenance by the Department of Informatics. The only difference lied within the application server. While with ICAROweb the application server only hosted the ASP.NET application and everything else was performed outside of it, VDMweb produces outputs in the form of CSV files that are used by the R environment hosted in the application server to analyse data. The R scripts, in turn, produce CSV outputs to be read by the front-end and integrated back into the VDM database. Additionally, as part of the data integration centralization, the ingestion tool "ITIJxml2sql.exe", for the ingestion of records into the VDM database, is also hosted in the application server. The access to the database server is secured by SQL connections, consisting of trusted connections (Windows authentication) from the ASP.NET user session in the application server.

4.1.3 VDMweb

4.1.3.A Development Tools and Base Software

VDMweb, the new VDM front-end, was developed using MS Visual Studio 2013 Community version, which is a free full-featured Integrated Development Environment (IDE) consisting of a code editor, a compiler, a debugger and a Graphical User Interface (GUI) builder. This IDE contains an embedded IIS Express local web server, which is used to debug the application. The .NET framework
version used was 4.5, representing an update in the previous framework being used in ICAROweb. The code behind pages for VDMweb were also implemented in C#, as its predecessor ICAROweb. However, to enable the implementation of the automatic generation and display of the daily bulletin, the MS Report Viewer version 11.0.0 control was installed, and a .NET reference to generate a .ZIP file was also added to the list of references used, the Ionic.zip. The markup language remained the same as ICAROweb, XHTML 1.0. Regarding data analysis, VDMweb makes use of the R programming language, a system used for statistical computation and graphics. The version used in the development of the VDMweb was 3.1.2 and its setup comes with a tool “RScript.exe” that is used to execute R scripts without the need of opening an R terminal and can be used as a shell script.

4.1.3.B Information Architecture

To support access to the new features, new links were added to the user interface of the VDMweb navigation bar (see screenshot in Figure 4.2). Each entry in the navigation bar, called modules, groups multiple web pages that are responsible for a set of related functions in VDMweb. The VDMweb information architecture is shown in Figure 4.3

The web pages display dynamic data retrieved from the SQL Server, as the user requests. Some of the pages also involve interactions with R scripts to analyse data, which is performed in the application server. These interactions are reviewed in the following sections by module, together with their implementation and UI. The content of each one of the modules is briefly described next:

- **I. Home** is where data ingestion is operated;
- **II. Records** displays details on the successfully integrated mortality records files;
- **III. Daily Mortality Surveillance** displays multiple daily data in tables;
- **IV. Weekly Mortality Surveillance** displays multiple weekly data in tables;
- **V. Reporting** includes the data visualization of daily and weekly plots for all stratifications, and the generation of the daily reports;
• **VI. Alerts** refers to the alert system implemented in daily and weekly aggregations;

• **VII. Nowcasting** refers to the nowcasting method integrated in VDMweb, displaying its daily plot and alerts;

• **VIII. Data Management** consists of pages where the user can update data used in data analysis, such as the regions and the baselines (daily and weekly).

**Figure 4.3:** VDMweb information architecture.
4.1.4 Extended Database Model

The implementation of the new VDM syste required an update to the model of the VDM database, renamed to *DailyMortalitySurveillance*. The base table, *Deaths*, remains the same as before (see Section 3.5), since it stores the basic information used by all SPs, as well as the SPs used to insert the daily records of death into the table and the *Records* table. The added tables mostly store the baselines, alerts and nowcasting values and the mortality events to be excluded in the generation of baselines, all of which are discussed below in the respective module section in this chapter. However, a new table, *Regions* (Table 4.1), is of important notice, since it is the basis for the new method of stratifying data by region - ARS and NUTS. It is a robust and complete table that groups every correspondence between geographical codes and the ARS and NUTS (I, II and III) regions, among other information. This was possible since, recently, it was agreed between DEP and IRN that a standard definition for geographical codes would be used, always with 6 numbers, regardless the region not being known. In the *Regions*, the recently created alphanumeric codes are already introduced. However, when implementing this table and migrating the old data that contained geographical codes with less than 6 numbers, every code without a length of 6 would not be assigned to any region, resulting in a loss of information. To overcome this, codes with less than 6 numbers (when CROs only had partial knowledge of the District, Municipality and Parish of the region of death) that unequivocally correspond to a known ARS or NUTS and are not ambiguous, were converted into 6-digit codes and added to the table, alongside its correspondences to the ARS and NUTS regions. This enabled the establishment of correspondences between regions and all the geographical codes.

The view *vwDeaths*, which already existed in the previous database version, was remodelled to handle the code regions limitations, by using the new table *Regions*. Due to the data migration issues of the old geographical codes, with less than 6 numbers, within this view every geographical code is normalized to 6 numbers, prior to mapping to ARS/NUTS using the table *Regions*. To do this, conditions were created for every length that each geographical code could have. Codes with length of 1 or 3 are padded with 0’s to their left, to have a code with 2 or 4 digits. After that, more 0’s

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
<th>Data type</th>
</tr>
</thead>
<tbody>
<tr>
<td>DD</td>
<td>District code</td>
<td>varchar(2)</td>
</tr>
<tr>
<td>CC</td>
<td>Municipality</td>
<td>varchar(2)</td>
</tr>
<tr>
<td>PFF</td>
<td>Parish code</td>
<td>varchar(2)</td>
</tr>
<tr>
<td>codRegiao (PK)</td>
<td>Geographical code</td>
<td>varchar(6)</td>
</tr>
<tr>
<td>RegiaoSaude</td>
<td>ARS description</td>
<td>varchar(15)</td>
</tr>
<tr>
<td>nut1</td>
<td>NUTS I description</td>
<td>varchar(40)</td>
</tr>
<tr>
<td>nut2</td>
<td>NUTS II description</td>
<td>varchar(40)</td>
</tr>
<tr>
<td>nut3</td>
<td>NUTS III description</td>
<td>varchar(255)</td>
</tr>
<tr>
<td>descDistritoIlha</td>
<td>District/Island description</td>
<td>varchar(100)</td>
</tr>
<tr>
<td>descMunicipio</td>
<td>Municipality description</td>
<td>varchar(50)</td>
</tr>
<tr>
<td>descFreguesia</td>
<td>Parish description</td>
<td>varchar(255)</td>
</tr>
<tr>
<td>Freg2013</td>
<td>Parish created in 2013</td>
<td>varchar(4)</td>
</tr>
</tbody>
</table>

Table 4.1: New database table *Regions* and description of its fields.
are padded to their right until obtaining a 6-digit number (for instance, the code “9” would become “090000” and “121” would be “012100”). Codes with lengths of 2 or 4 followed the same logic, but they only need to be padded with 0’s at their right. On the contrary, for codes with 5 digits, a “0” is added to their left. After setting these geographical codes, a direct lookup on the table *Regions* obtains combinations with ARS and NUTS. The *vwDeaths* data fields are presented in Table 4.2.

Another new relevant view, *vwWeekly*, groups every stratification by week, is depicted in Table 4.3. It dispensed the definition of the two variable tables for each weekly stratification SP (described in Section 3.5). Hence, using this view was more intuitive and avoided rewriting the same T-SQL statements repeatedly in multiple SPs. *vwWeekly* is based on nested select statements with the same purpose as the two tables (one to define the first and last days of the week and the other to count data for the respective week). The view queries the *vwDeaths* to count data and group it by week.
<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
<th>Data type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inicio</td>
<td>First day of the week</td>
<td>date</td>
</tr>
<tr>
<td>Fim</td>
<td>Last day of the week</td>
<td>date</td>
</tr>
<tr>
<td>Semana</td>
<td>ISO week number</td>
<td>int</td>
</tr>
<tr>
<td>Obitos</td>
<td>Total number of deaths</td>
<td>int</td>
</tr>
<tr>
<td>codConservatoria</td>
<td>Year of the ISO week</td>
<td>int</td>
</tr>
<tr>
<td>Norte</td>
<td>Death counts in Norte - ARS</td>
<td>int</td>
</tr>
<tr>
<td>Centro</td>
<td>Death counts in Centro - ARS</td>
<td>int</td>
</tr>
<tr>
<td>LVT</td>
<td>Death counts in Lisboa e Vale do Tejo - ARS</td>
<td>int</td>
</tr>
<tr>
<td>Alentejo</td>
<td>Death counts in Alentejo - ARS</td>
<td>int</td>
</tr>
<tr>
<td>Algarve</td>
<td>Death counts in Algarve - ARS</td>
<td>int</td>
</tr>
<tr>
<td>Madeira</td>
<td>Death counts in Madeira - ARS</td>
<td>int</td>
</tr>
<tr>
<td>Açores</td>
<td>Death counts in Açores - ARS</td>
<td>int</td>
</tr>
<tr>
<td>Norte_nut</td>
<td>Death counts in Norte - NUTS</td>
<td>int</td>
</tr>
<tr>
<td>Centro_nut</td>
<td>Death counts in Centro - NUTS</td>
<td>int</td>
</tr>
<tr>
<td>Lisboa_nut</td>
<td>Death counts in Lisboa - NUTS</td>
<td>int</td>
</tr>
<tr>
<td>Alentejo_nut</td>
<td>Death counts in Alentejo - NUTS</td>
<td>int</td>
</tr>
<tr>
<td>Algarve_nut</td>
<td>Death counts in Algarve - NUTS</td>
<td>int</td>
</tr>
<tr>
<td>Madeira_nut</td>
<td>Death counts in Madeira - NUTS</td>
<td>int</td>
</tr>
<tr>
<td>Açores_nut</td>
<td>Death counts in Açores - NUTS</td>
<td>int</td>
</tr>
<tr>
<td>M</td>
<td>Death counts in Males</td>
<td>int</td>
</tr>
<tr>
<td>F</td>
<td>Death counts in Females</td>
<td>int</td>
</tr>
<tr>
<td>i0</td>
<td>Death counts in the 0-14 years Age Group</td>
<td>int</td>
</tr>
<tr>
<td>i15</td>
<td>Death counts in the 15-24 years Age Group</td>
<td>int</td>
</tr>
<tr>
<td>i25</td>
<td>Death counts in the 25-44 years Age Group</td>
<td>int</td>
</tr>
<tr>
<td>i45</td>
<td>Death counts in the 45-64 years Age Group</td>
<td>int</td>
</tr>
<tr>
<td>i65</td>
<td>Death counts in the 65-74 years Age Group</td>
<td>int</td>
</tr>
<tr>
<td>i75</td>
<td>Death counts in the 75-84 years Age Group</td>
<td>int</td>
</tr>
<tr>
<td>i85</td>
<td>Death counts in the ( \geq 85 ) years Age Group</td>
<td>int</td>
</tr>
</tbody>
</table>

Table 4.3: New database view \textit{vwWeekly} and description of its fields.

The added SPs take into account the different requests and input/outputs of the VDMweb web page in which they are called. The SP referred in Section 3.5 used to insert the records of death into the \textit{Deaths} table, was not modified, since this method was not changed.

4.2 Modules I/II: Home and Records

4.2.1 User Interface

The \textit{Home} module displays the default web page of VDMweb. Its content is a static display of text corresponding to a description of the VDM system and a brief summary about each one of VDMweb modules. On the left side of the page, there is an ASP.NET file upload control, as presented in Figure 4.4a. Beneath the upload button there is a label displaying a text message about the status of the upload process. This corresponds to the upload of the mortality records file into the application server and subsequent ingestion of its records in the VDM database. Clicking to chose the file to be uploaded opens a Windows explorer window in which the user browses to the mortality records file and selects it. Then, when clicking the upload button, an event defined in the C# code behind page is triggered.
First, this event checks if there is a file in the upload file control, then it checks whether it is a file with the ".xml" extension. If these conditions are met, the file is uploaded to the server and saved in a target folder where all the previously uploaded XML files are stored. After saving the file, the ingestion tool "ITIJxml2sql.exe" is called using the .NET Process.Start method. It is important to notice that these mortality records files are all named differently, with the date of its registry in its name. For instance, "Deaths20150515.xml" corresponds to a file compiled by IGFEJ on the day 15-05-2015 and sent to INSA on the day after (unless it was compiled on a day between Friday and Sunday, in that case it would only be sent at Monday).

Figure 4.4a also presents the label displayed when the daily mortality records file is successfully uploaded and its records ingested. However, if it does not meet one of the conditions previously referred, the label is updated in red colour with what is failing in the upload and the data ingestion process is stopped. Figure 4.4b shows an example of when there is an equally named XML file in the target folder of the already uploaded files.

In Records module, which also consists of only one page, a GridView with alternating coloured rows displays data related to the integration of the daily mortality records files. Using the .NET SqlConnection class, that takes as input a ConnectionString indicating the name of the data source to connect to (database server name) and the SQL Server instance name, an SqlCommand is defined in C# and sent to the SQL server, consisting of the execution of a SP. This communication and posterior execution is performed using the .NET SqlDataAdapter class, which represents the SQL Command and the connection properties and serves as a bridge between a DataSet - an in-memory cache of data - and the database by filling the former with the results from the SQL Command. The DataSet is then used as the DataSource of the GridView in the markup page, displayed in the browser to the end-user. As this is only a display of file related details (name of the file, number of read records, number of inserted records, number of failed records and date of integration in the database), only...
4.2.2 Interaction with SQL Server

The configuration file of the data ingestion tool sets up the communication with the SQL Server instance, with the VDM database, from the Home module. Furthermore, it also defines the correspondence between each XML record node and the corresponding column (a record in the mortality records XML file comprises one node for each record variable) in the table to where each record is inserted. This table is a staging table called DeathsQuarantine and it is used in a SP that is executed by the ingestion tool, designed to insert the records into the Deaths permanent table in the VDM database. This SP uses a cursor to go through each row of the staging table and insert it into the table Deaths, referred in Section 3.5 using a TRY CATCH method. Errors identified in the CATCH are stored in a Log Event table, for instance when there is a violation of the unique table index, implemented to avoid duplication of records. However, these rows from DeathsQuarantine do not include the idRecord column, i.e., the id of the mortality records file from where each record is from. To select this field, the database table Records displayed in the VDMweb Records module is also populated in this SP. Before using the cursor, the Records table is initially filled with the id of the mortality records file, in the field idRecord (an identity column), the name of the file and its total number of records plus the current date and time. This enables the assignment of the idRecord to each record in the table Deaths as the cursor is used, by using an INNER JOIN of the staging table and the Records on the name of the file field, since it is also present in the Deaths table. After the conclusion of each file ingestion, the number of inserted records in the table Records is updated and data in the staging table is deleted, so it can be used in the next data ingestion cycle. The data ingestion flow is summarized in Figure 4.5. The SP used to display data in the respective page GridView, is a simple query to select the table Records plus a field with the difference between the total records read and the records successfully inserted in the table Deaths.

4.3 Modules III/IV: Daily and Weekly Mortality Surveillance

4.3.1 User Interface

Next to Home and Records in the navigation bar, there are two modules that display mortality absolute values in GridViews, which are essentially tables. These modules, Daily Mortality Surveillance and Weekly Mortality Surveillance, have the same UI, hence they have the same markup language and the same functionalities, differing only in the type of data aggregation, daily or weekly (ISO week).
There is a web page to display each data stratification performed - age groups, region and gender. Besides these pages with the pre-defined stratifications, two more web pages were created for both modules, the Customized and Parametrized. Customized provides a table grouping the fields of total counts of user-selected strata from each one of the three stratifications (e.g. total deaths in a specific region plus total deaths of a certain age group by day or week, depending on the module). This feature was implemented in order to provide a data file with custom aggregated fields of different stratifications, to be sent to entities that require that specific information. The second, Parametrized, is a more specific query, where the user selects one parameter from each data stratification that are combined to get the resulting death count (e.g. total male deaths in the region Norte within an user-defined age group). All of these web pages are accessed by buttons, presented in Figure 4.6.

Once inside these dynamic pages, the user can select the time series of the data to be displayed, inserting the from date and the to date in the corresponding text-boxes. The former is by default 2007-01-01, which corresponds to the oldest historical date of information present in VDM database, while the latter is the current date. An HTML calendar control, that the user can browse through when clicking on the text-boxes to select days, is also implemented. Depending on the web page, more parameters are defined by the user, which are sent to SQL Server so that the proper queries are performed. These parameters are selected using ASP.NET DropDownList controls, customized CheckBox trees, implemented using Cascade Style Sheets (CSS) and overlayed check-boxes, or text-boxes with RegularExpressionValidator expressions, to restrict input values to ones that match the pattern specified by a defined regular expression. Each one of these controls provide options that are associated to values so that the server can interpret the option selected. The controls are presented in Figure 4.7 and summarized in the following list:

- DropDownList cover the selectable options for the Region data stratification - ARS or NUTS - in Daily, Daily by Region, Daily Customized, Weekly, Weekly by Region and Weekly Customized. In Daily Parametrized and Weekly Parametrized the DropDownList control is used for every
data stratification to narrow the query by selecting one condition for each;

- **CheckBox** trees provide the options to select what columns to display in the table displayed in Daily Customized and Weekly Customized;

- Input text-boxes, in Daily by Age Group, Daily Customized, Weekly by Age Group and Weekly Customized correspond to the minimum and maximum age that the user can search for a certain age group that is not present in the pre-defined groups. The **RegularExpressionValidator** used validates input values that consist of only positive integers ("\d+"), displaying an error message if it is violated and not submitting the form. Besides this custom age group definition, text-boxes are used for inserting dates in every web page.

Data are displayed in paged GridViews, whose columns correspond to the day and the count of deaths of each stratum, in Daily Mortality Surveillance, or the first/last days of the week, the week number and the count of deaths of each stratum in Weekly Mortality Surveillance. The first page of Daily Mortality Surveillance, Daily, provides a table with the information similar to what is listed in Sub-Section 3.4.3 with the addition of a column corresponding to the ARS region of the death. The data binding to the GridViews is implemented in the code behind page, after setting up SQL connections using the .NET DataAdapter class and filling a dataset with data from the SQL SP. An example of the GridView provided in the Daily page is presented in Figure 4.8. All the other GridViews in the multiple VDMweb pages by stratification have the same UI of this one, only its content changes, accordingly.

Every table can be downloaded by the user to its computer, by clicking the respective Export button above each GridView in VDMweb. The event associated to clicking the button resorts to a Response object, which is used to send outputs to the client. In the C# file, first a Response.Clear is performed, to erase any buffered HTML output, then the Response.Buffer is set to “true”, which
implicates that the server will not send the response to the client until every server scripts have been processed. Then the file output is prepared, by using *Response.AddHeader* that sets the HTML header name to a value that will correspond to the name of the file. In the case of the region stratification, an IF statement is used to check what region value is selected in the *DropDownList* and, according to it, add "ars" or "nuts" to the end of the file name so that both are differentiated when downloaded. The content encoding is also defined as UTF-8, as well as the content type, defined as *application/text*. To write data to the output file, a *Response.Output.Write* is used. In this class, a .NET *StringBuilder* class is firstly initiated containing a string corresponding to the rows in the dataset of the *GridView* with its values separated by commas and lines by \\
\n. This string is constructed by a C# function developed for that purpose. The file is finally written with the .csv extension (defined in the *Response.AddHeader*) and then a *Response.End* terminates the process and *Response.Flush* sends the file to the client.

### 4.3.2 Interaction with SQL Server

Data exchange between the ASP.NET application and the VDM database in the SQL Server is performed using the .NET *SqlConnection* class and another *DataAdapter*. The SPs involved in these two modules are the most complex, due to the amount of user customization that is implemented, especially in the *Customized* and *Parametrized* pages. All of the pages have many user-selected values that are passed to the VDM database when defining the *DataAdapter* class. After initiating an instance of the class, all parameters are transmitted using *SelectCommand.Parameters.AddWithValue (parametername,value)*, where the value is retrieved from the text-boxes, *DropDownList* and *CheckBox* trees, along with the page parameter corresponding to the *GridView* navigation page number. The SQL data type of each parameter is redefined inside each SP when declaring the variables.

![Figure 4.8](image.png)

*Figure 4.8: First page of the dynamic table displayed in the Daily page, by ARS region.*
corresponding to the parameters that are required in order to be executed.

For each web page consisting of one of the three data stratifications, the SPs are simple. The SPs used in the weekly pages were not changed, so they remain the same as the previous VDM version, only with the addition of the new stratification features. In the daily pages SPs, a sub-query selects the day of death and the counts of a certain condition using the SQL function \texttt{COUNT()}, grouped by day of death (e.g. for the age groups stratification SP, counts of deaths for each age group are performed, resulting in multiple columns one for each group). Then an SQL Dataset is filled with the results of the main query statement, in which the subquery is implemented, selecting row numbers using the \texttt{ROW\_NUMBER()} function, the day of death as varchar(10) and each count. The conversion to varchar(10) is done so it is displayed in the VDMweb gridview in the form of "aaaa-mm-dd", otherwise it would also display the time ("aaaa-mm-dd hh:mm:ss") since the .NET would read it as having a date time data type. After setting up the dataset, the result of the SP is retrieved selecting rows with a row number within the number of rows in the page that the GridView is currently on (e.g. assuming a GridView with a maximum of 50 rows, if the navigation page is 2 then the records selected would correspond to records between the 51th and 100th rows in the SQL dataset). The web page corresponding to the stratification by age groups is the only one assigned to 2 SPs, given that there is the option of the user defining the age group by introducing both minimum and maximum ages. An IF statement is coded in the code behind page to call one of the two SP, which depends on whether the user requested the pre-determined age groups or if he defined one. If only the value for minimum age is introduced, then the age group for which data is analysed corresponds to all ages higher than the one inserted, the opposite happens if only the maximum value is introduced.

The \textbf{Parametrized} and \textbf{Customized} SPs required the implementation of conditional queries and a dynamic construction of the query, respectively.

In the \textbf{Parametrized} query each user-selected value, using one of the controls in the upper half of Figure 4.7 is sent to the SQL Server. Variables are assigned to these values in the SQL Server SP and according to them, the query to be executed is selected (the SP groups all possible queries) using nested IF statements to select the query that is suitable to the selected values. First, an IF statement checks whether the regions to search are ARS or NUTS. Then, the next IF checks if the selected region stratum corresponds to a specified region, Portugal Mainland (corresponding to all the records with known regions in the Mainland) or Portugal (corresponding to all Portuguese records). After this, another IF statement is implemented to select which gender is to be taken into account.

If no age values are inserted by the user, the queries are defined using the conditions known until this point (ARS/NUTS, region, gender and date interval) using the appropriate WHERE conditions: for instance, in the case of Portugal Mainland and Male conditions the query would count the number of deaths where the region of death would correspond to all ARS or NUTS regions in Portugal Mainland, of the male sex and between the days defined by the user. If an age value is added, then another similar SP is used, with the difference that it has another layer of conditions in the WHERE statement by adding the minimum and maximum age to the count of the number of deaths. These queries target the view \texttt{vwDeaths} for the daily stratification, and use the two variable tables method for the weekly
data, being that the first table is defined before all the IF statements (since it just defines the limiting
dates of each week), and within them the second table is calculated.

In the Customized SP, used to enhance the customization and to reduce work resources in select-
ing multiple columns from different stratifications into one table, the values are sent to the SQL Server
as variables of the data type BIT. If the corresponding check-box is checked, then it has the value “1”;
otherwise it is “0”. Then in the SP a variable @sql of the data type nvarchar(MAX) is used, MAX is to
ensure that it can store all the query string that will be dynamically built. According to the ARS/NUTS
filter and the BIT value of each parameter, a field to be selected (corresponding to a column stratum)
is appended to the @sql query string. After going through all parameters and adding all strata to be
searched whose BIT value is defined as “1”, the query is performed similarly to the one done in the
previous SPs by stratification, using the COUNT statement. However, in this SP a first query counts
every possible stratum and selects them into a temporary SQL table alongside its ROW_NUMBER
for the same purpose that was explained in the previous paragraph. This query is appended to the
query string @sql constructed until this point, resulting in a select statement of certain fields from the
temporary table with the condition of which rows are to be selected, according to their row number.
Since the @sql query variable is a string it, can’t be executed as a regular query, so the SQL Server
system’s stored procedure sp.executeSQL.

The calculation of the values of the column “Total Regiões”, selected in Figure 4.7, are imple-
mented by the C# code. They correspond to the sum of each value of the regions selected by the
user (excluding the global ones Portugal and PtMainland), so in the SQL Server they correspond to
selecting a NULL column. The view vwDeaths and the 2 variable tables are used in the respective
Daily and Weekly Customized SPs, respectively.

4.4 Module V: Reporting

4.4.1 User Interface

The Reporting module groups data visualization, daily and weekly, and the generation of reports
of the VDMweb. These functions include the display of a numerical daily information table, the daily
bulletin, the annual comparison plot and a plot for each one of the data stratification performed,
including a plot for the same Parametrized search defined in the previous section. Each web page
 corresponds to one of the referred features and the pages with plots by age groups, regions and
gender display one plot per stratum. They are accessed by the buttons shown in Figure 4.9.

All the plots displayed in the Reporting web pages correspond to static images that are created
on the moment plotting the results obtained from the SPs executed in the SQL Server. These images
are displayed using the .NET Image Location method, by which the images are stored, with a pre-
defined name, in a certain folder path. VDMweb stores them inside its application project folder in
the application server. With this method, a problem could arise from the fact that two users could
be accessing the same plot (and therefore creating the same image). That would not be possible
for the second user because it would result in a error stating that the file in question was already
being used by another process - the first user. The workaround to prevent this image overridden error was to add a keyword in the end of each image file name - #SEQ(maxfiles, minutes_timeout). This keyword generates a sequence of numbers up until the maximum files defined ("Chart_000001.png", "Chart_000002.png", etc) with a specified time-out for each file in the sequence.

Additionally, all plots are customizable. The user can select which time period to observe plot and the type and interval of the X axis spacing - whether its done by days, weeks or years, and how many. For the weekly plots only the week interval is available, since the purpose is to display data by weeks. Besides the date interval and X axis customization, and due to the fact that in this new VDM version the baselines are calculated, for every data stratification the user can also chose multiple baseline series to overlay the observed values. This overlay is relevant for a comparison between observed, expected and the limits values, enabling a faster draw of conclusions. Five baselines are generated and selectable to add to the plots series: the expected mortality baseline (corresponding to the "fit" baseline), the 95% and 99% lower confidence limits and the 95% and 99% upper confidence limits. These user-selected parameters and a daily plot example with baselines, are presented in Figure 4.10a.

In the plots it is also possible to observe a tooltip that displays the day and respective number of deaths. This tooltip is displayed when hovering the mouse on each series marker, that correspond to the data points retrieved from the SQL SP used to draw the series line. Baselines are added to the plot, according to the check-boxes selected, by using if statements in the code behind page for each check-box value that was checked. For each baseline check, the respective series’ Y values are assigned to the corresponding baseline column of the dataset resulting from the SP (the X values are the same column for all series - the day of death).

Every daily plot by stratum in this module is of the same type as the presented above, and the weekly plot type is presented as an example in Figure 4.10b.

---

**Figure 4.9**: Reporting web pages buttons.

---

<table>
<thead>
<tr>
<th>Boletim</th>
<th>Diário</th>
<th>Semanal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ocorre boletim</td>
<td>Evolução diária</td>
<td>Evolução semanal</td>
</tr>
<tr>
<td>diário</td>
<td>da mortalidade</td>
<td>da mortalidade</td>
</tr>
<tr>
<td>Tabela</td>
<td>Diário por</td>
<td>Semanal por</td>
</tr>
<tr>
<td>Valores tabelados</td>
<td>Grupo Etário</td>
<td>Grupo Etário</td>
</tr>
<tr>
<td>dos últimos dias</td>
<td>Evolução diária</td>
<td>Evolução semanal</td>
</tr>
<tr>
<td></td>
<td>da mortalidade</td>
<td>por grupo etário</td>
</tr>
<tr>
<td></td>
<td>por grupo etário</td>
<td></td>
</tr>
<tr>
<td>Comparação Anual</td>
<td>Diário por</td>
<td>Semanal por</td>
</tr>
<tr>
<td>Comparação</td>
<td>Região</td>
<td>Região</td>
</tr>
<tr>
<td>anual</td>
<td>Evolução diária</td>
<td>Evolução semanal</td>
</tr>
<tr>
<td></td>
<td>da mortalidade</td>
<td>da mortalidade</td>
</tr>
<tr>
<td></td>
<td>por região</td>
<td>por região</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Diário por</td>
<td>Semanal por</td>
</tr>
<tr>
<td></td>
<td>Sexo</td>
<td>Sexo</td>
</tr>
<tr>
<td></td>
<td>Evolução diária</td>
<td>Evolução semanal</td>
</tr>
<tr>
<td></td>
<td>da mortalidade</td>
<td>por sexo</td>
</tr>
<tr>
<td></td>
<td>por sexo</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Diário</td>
<td>Semanal</td>
</tr>
<tr>
<td></td>
<td>Parametrizado</td>
<td>Parametrizado</td>
</tr>
<tr>
<td></td>
<td>Evolução diária</td>
<td>Evolução semanal</td>
</tr>
<tr>
<td></td>
<td>da mortalidade</td>
<td>da mortalidade</td>
</tr>
<tr>
<td></td>
<td>com estratificação</td>
<td>com estratificação</td>
</tr>
<tr>
<td></td>
<td>específica</td>
<td>específica</td>
</tr>
</tbody>
</table>
Figure 4.10: (a) Screenshot of the plot of total daily deaths, its baselines and its user-selected parameters in the Daily web page of the Reporting module. (b) Screenshot of the plot of total weekly deaths, its baselines and its user-selected parameters in the Weekly web page of the Reporting module.

The weekly chart has, as expected, less data points - one for each week - which are identified by square markers to better distinguish them. Its tooltip message displays the number of deaths and the last day of the week, which correspond to the Y and X series values respectively. The X values correspond to the last day of the ISO week (Sundays) and not the week number (1-53) because
sending data from the SQL Server as the \textit{int} data type of the week numbers (in groups of 1-53, 1-53, 1-53, ...) would cripple the ordering of the values in the X axis, which is prevented by using values of a date type. However, when observing the weekly chart it is noticeable that the X axis labels are of the type \textit{"YYYY-W(weeknumber)"}. To accomplish that, a customized label was created by a specific C# method developed for it, since C# does not convert on its own dates to the corresponding ISO week. Contrary to the daily data, where the default initial date of the plot is of the type \textit{"YYYY-01-01"} and \textit{"YYYY"} is equal to the year of the current date minus 4 years, for weekly data this default date is exactly 2 years before the current date. This is to be concordant to the weekly mortality chart time period in the influenza bulletin. Nonetheless, as stated before, the dates are mutable parameters.

The charts by stratification are equal to the ones presented above. They have the same user-selected parameters (dates, parameters, baselines and X axis) and graphic interface, either daily or weekly. However, as there are a lot of strata, even within each stratification, in each web page there are presented multiple charts, 7 for age groups, 7 charts for regions and 2 for sex, one for each stratum. The regions also have the option to be selected by ARS or NUTS.

The Reporting component, in terms of generating the information to be sent to multiple entities, is carried out by 3 other specific web pages, corresponding to the first 3 buttons at the left in Figure 4.9. These buttons correspond to the generation of the daily bulletin, the tabular information of the last 15 days and its comparison with reference values and an annual comparison plot.

The annual comparison plot consists in an overlay of the mortality series of up to 5 years, defined by the user in text-boxes. The same interface of the previously referred charts is used, although now the X axis labels only consist of \textit{"DD-MM"}, with an interval of 1 month (1-01, 1-02, ..., 1-12). These labels are defined in the C# file by selecting which date parts of the X values to use, since these values are of the type \textit{"YYYY-MM-DD"}, as obtained from the SQL SP. The series, one for each year, have a particularity that had to be handled, associated to leap years. If one of the 5 years selected is a leap year, that will introduce the data point of the leap day into the dataset that is plotted. This resulted in a gap in the leap day of the series corresponding to years that don’t have it. To overcome this, a method was implemented in C# to check whether the leap day row has null values for all columns (years). If it has, it means that none of the 5 years have a leap day, so that row is deleted from the dataset and no leap day will be plotted. If at least one of the years has a leap day, then the leap day data point of the years with a null value will correspond to the average of the previous and after days, in order to plot a continuous line in that specific day.

The numerical table that is sent to authorities (referred in Section 3.2) is presented in the web page as a \textit{GridView} grouping all observed values for a total of 18 days (14 past plus current days, 3 after)- Additionally, now it includes the 2 gender strata, as Figure 4.11 presents. The regions are selected by the user, if ARS or NUTS, as well as what reference values to use: the average of the past 4 days, the baseline or the 95% upper confidence limit, in the respective \textit{DropDownList} controls. The variation of the observed values is now defined in font colours and it is done differently for each type of reference value selected:

- Using the average values, the observed values appear in red if they are higher than the expected
value plus one deviation pattern, green the opposite and black if they are in between.

- Using the baseline, the observed values are red if they are higher than the 95% upper confidence limit of the baseline, green if they are lower than the 95% lower confidence limit of the baseline and black if they are in between.

- Using the 95% upper confidence limit, only the red differentiation is used for when the observed value is higher.

These comparisons are only made in the last 6 to 14 days, given that there is not representative information of the last 5 days due to the delay that usually occurs in the notification of deaths - that’s also why they appear in italic. For this same reason, the “Total” column at the right only sums the values of those days. The GridView can also be exported to the client as a file with the extension “.doc” and be opened by MS Word. The steps for this export are similar to the ones implemented for the GridViews in the Mortality Surveillance modules, explained in the Sub-Section 4.3.1. However, instead of a StringBuilder, it uses a StringWriter class that is associated to an HtmlTextWriter that will write in the response file (the downloaded table) common HTML mark up code defined in the C# (title, legend of the table, etc) plus the HTML code of the gridview, since this is an UI element. Basically, a “fake” MS Word document is created, containing the HTML code of a page that can be visualized. Then it can be edited, for instance to add comments, and saved as a PDF file to be uploaded into the web cloud and embedded in the reporting e-mail sent on a weekday basis.

The daily bulletin is generated using a MS Report Viewer control. This control supports the generation of a Client Report Definition (RDLC) using its built-in processing capability file in the application.
folder, which is basically a report template that can be edited. This template contains external images (the plots to be reported), that are associated to a defined path, which was edited to be the same path as the png images saved in the application reporting folder. This way, each time a report chart is saved, the image in the report is refreshed as well and it is always updated. The paths were defined using ReportViewer parameters implemented in the code behind page of its web page, posteriorly used by the RDLC file. As the plots are basically the same that are displayed in each daily web page stratification, they use the same SPs. Therefore, a web page - the Bulletin page - is displayed first, Figure 4.12 presenting the parameters that each SP requires. When clicking the button “Ok” all SPs are executed, each chart is saved, and a redirection of the web page to the one containing the actual Report in the ReportViewer control is performed. With this redirection, variables holding the current data, the time interval of the user-selected dates and whether regions are stratified by ARS or NUTS are also sent to be redefined as ReportViewer parameters and included in the information provided in the rdlc file. The report can then be saved as an Excel, PDF or Word document. An example of the resulting report is presented in Appendix B from the day “2015-05-25”, as well as the tabular information.

4.4.2 Interaction with SQL Server

The core of this module is mainly UI related, given its high number of graphic controls. However, there is also a lot of processing in the SQL server to provide all the data. The majority of the SPs used in this module are similar to the ones in the previous section for both daily and weekly aggregations, since their functions and parameters are essentially the same. However, for each web page in Reporting there is a SP with some differences. For instance, there is no need to populate a dataset with the row number and then select partitions from it, according to a number. This is because data won’t be displayed in a limited size GridView, but in a time series chart that will use all data at once. To sort the X axis by an ascendant order in the .NET framework of the VDMweb, the dates are also not converted into the data type varchar(10) when selecting the result from the SP. Still, with the exception of the parametrized SPs and the annual comparison chart, every web page provides the possibility of plotting user-selected baselines and this requires building dynamic SQL query strings. Variables corresponding to the baselines parameters are defined as BIT (1 checked, 0 unchecked), and their
column is added to the SELECT statement as they are checked. The body of the query corresponds to the selection of the counts by the respective stratification within the user-defined time period and a FULL OUTER JOIN of it with the respective baseline table, on the field day of death. Each data stratification is associated to a baseline table. I opted for this way so that there wouldn’t exist tables with more than 100 columns given that each stratum corresponds to 5 baselines which could compromise the efficiency of the queries. Then, a specific MS SQL SP, sp_executeSQL, is used to execute the query string. For daily plots, the vwObitos is queried, but for the weekly ones, the vwWeekly is used to get the stratum counts instead of the 2 method table. This is in order to select the entire week death count if the “from” date defined in VDMweb happens to be in the middle of a week.

To generate the daily bulletin, the SPs used to create the plots correspond to the same used in the stratified web pages, given that the charts appearing the bulletin are the same ones that are presented individually in their respective page.

For the tabular information, the only variables used by the SP are the region filter (ARS or NUTS) and which references values were selected. First a one-column dataset is defined using a recursion to get the dates from -14 to +3 days around the current date. Then these dates are inserted into a temporary table (#dsdates) that is used as a holder of the dates whose values are to be calculated. Next, according to ARS or NUTS, another temporary table is defined, this one storing the counts for all strata to be displayed (#regionsARS or #regionsNUTS). The table #dsdates is used in an OUTER APPLY command to select the day of death and the corresponding reference values for each one, whether its the average or baselines (IF statements are used to check which method it is). The resulting dates and reference values are used in a FULL OUTER JOIN with the temporary table #regions (NUTS or ARS) on the field day of death. The main query then selects all the columns resulting from this FULL OUTER JOIN, consisting of 24 columns - all data strata plus the headers (selected as NULL in the SP and defined only in the C# code) of both halves of the table presented in Figure 4.11 and 18 rows - the dates in #dsdatas. The GridView presented in the dynamic web page corresponds to the transpose of these results, to ease its reading and comprehensiveness.

Regarding the annual comparison chart, all the five text-boxes parameter values are passed to the SQL Server, empty or not, as the years of interest. A sub-query selects every sub-string corresponding to the “MM-YYYY” of each day of death that exists on the table Deaths and the sum of all deaths, grouped by day of death, that occurred in the parameter year value. This results in a table with a row for each day of death (only with month and year) and five columns, one for each year parameter. Most of these fields have NULL value for the majority of rows, since each year will only have values in its days (e.g. for the day “2014-01-01”, its value in the date column is "01-01" and every year column will have NULL values except for the 2014 one). The main query of the SP selects the MAX of each year column (dispensing the NULL values) and the dates column prepended with the string "2012-" (it could have been another number corresponding to a leap year) and converted to date, grouping the results by this new column date. This is implemented so that the subsequent conversion to date data type can be made, because otherwise the SQL Server would be trying to convert a string such as "2014-02-29" to a date, which does not exist. The result is a table with 366 rows, corresponding to
all dates of a leap year, and 5 columns with the counts of death by year. In the VDMweb plot, only the month and day are displayed, so this definition of the year 2012 is merely an intermediary method to send proper data. This conversion to date, as previously referred, is required to sort the X axis values in the plot.

4.5 Module VI: Alerts

4.5.1 User Interface

The **Alerts** module consists on the implementation of the automatic alert system, which provides information about when excess peaks occur. Its web pages are summarized by their buttons on Figure 4.13. They display *GridViews* with a visual interpretation of whether there is an alert or not, by each data stratification and both in daily and weekly aggregations.

Each *GridView* displays data of each day (first and last day plus week number for the weekly pages) consisting of the observed mortality values and the values of 3 of their 5 baselines: fit, 95% and 99% upper confidence limits. These are used in the assessment of the alert status, explained in the next sub-section. A column corresponding to the alert status displays colours according to it:

- Green for non-alert status;
- Red when an alert is generated;
- Grey for the last 5 days or 1 week - depending on the type of data aggregation. These are considered the periods that don’t have representative information due to the delay in the notification of deaths. However, if during them there is already a high number of counts that result an alert, the red colour will be displayed.

This colour assignment is implemented in the C# coding. After filling a dataset with the alerts data posteriorly to connecting to the database, the value in the alert column is checked, whether it is 1 (alert), 0 (non-alert) or "NA" (the last days without representative information). Then, a colour is assigned to the alert cell in the *GridView* (see Figure 4.14), according to the list stated above. All
of the GridView are exportable, with the alert status value and not the colour to ease the numerical calculation of excess deaths, defined as the difference between the observed values and the baseline.

In the web pages with data by age group, region and gender the end-user can select in a Drop-DownList which specific stratum to visualize. Besides these alerts displays for each stratification, there are two other web page that correspond to a comparison map of the alerts, one for daily and another for weekly data. Each one consists of a GridView displaying only dates and the alerts of all strata within a certain stratification (i.e. one GridView with dates and the respective alert colours for all age groups, or regions (ARS or NUTS) or gender).

To update the alert status, in each web page there is a text-box where the user inserts the number of past days whose alert is to be updated. This starts a process flow implemented in C# that starts by executing in the SQL Server a SP that retrieves data to be exported into a folder in the application server. To read the stream of rows from the SQL database a .NET class SqlDataReader is used. A StringBuilder is used to store what is read from the SqlDataReader, using a C# function specifically implemented to generate a CSV type of string from the Reader class. Then, using this StringBuilder while the reader is reading data, a CSV file is generated by a StreamWriter that writes into a file the strings from the StringBuilder class with the character encoding 1252 - the MS default. After this, the R script of the alert system is executed using Process.Start of the “RScript.exe” that will run the intended R script. After the R process is finished (the method Process.WaitForExit is used for this), its output file with the updated alert status is inserted back into the VDM database using the SqlBulkCopy class. This class bulk loads a virtual data table in an ASP.NET application into a specific table in the SQL database. The data table is filled with values from the R script output CSV file, using another C# function implemented to parse a CSV file into a data table, and posteriorly the target database table, to insert the new values, is defined and assigned to the SqlBulkCopy class.

Figure 4.14: Table of alerts for total weekly data.

<table>
<thead>
<tr>
<th>Início</th>
<th>Fim</th>
<th>Semana</th>
<th>Óbitos</th>
<th>fit</th>
<th>upperline</th>
<th>upperline99</th>
<th>Alerta</th>
</tr>
</thead>
<tbody>
<tr>
<td>18.05.2015</td>
<td>24.05.2015</td>
<td>21</td>
<td>1992</td>
<td>1972.98000</td>
<td>2216.79400</td>
<td>2234.23900</td>
<td></td>
</tr>
</tbody>
</table>
4.5.2 Interaction with R - Automatic Alerts System

The assessment of the alert status is performed by an algorithm implemented in R. These status are defined using the Westgard rules [37], created by James Westgard as part of a multi-rule quality control procedure. The procedure uses a combination of criteria to decide whether the process being analysed is in-control or out-of-control. Although there are 5 rules, these can be selected and combined according to the process to be analysed. The rules are defined according to the average and standard deviations. The rule 1_2s is an alert rule, triggered when an observed value is above the upper control limit or below the lower control limit, defined as twice of the standard deviation. The rule 2_2s states that the process is out-of-control when two consecutive observed values are above the same upper control limit or below the lower control limit. The rule 1_3s states the same as the 2_2s but for limits that are defined as the triple of the standard deviation. The combination of the last 2 rules are the ones implemented in the R script. In this case, the limits correspond to the 95% and the 99% upper confidence limits of the mortality baseline, dispensing the lower limits since what is at stake here is determining only the periods with excess mortality.

The R scripts implemented read an input CSV file with the observed values and baselines for each data stratification. Since the request to update the alerts may come from different web pages corresponding to each stratification, there is an R script for each one (total, age group, regions and gender), so there are as much as input CSV files as scripts, the same goes for their output. In the script, the algorithm creates a column, and appends it to the input data with the respective alert status assigned for each row (the “1” for alert state, “0” or “NA” for absence of alert). They follow two rules:

- The alert status changes to “1” when two consecutive mortality counts (days or weeks) are higher the 95% upper confidence limit of the mortality baseline, and returns to the non-alert status, “0”, when two consecutive counts are lower this same limit.
- The alert status changes to “1” when one count is higher the 99%, and it is cleared to “0” when two consecutive counts are lower the 95% upper confidence limit.

For the last 5 days/1 week instead of “0” the alert status is “NA”, due to the information yet to be updated. An example of how these rules apply is presented in the Figure 4.15 where red dots correspond to an alert. These rules implicate that the status has “knowledge” of historical data, in the way that it is changed along the cycle that goes through all the counts. Therefore, for the first count analysed there is no information about whether it was already on alert or not (if it is above 95% and there is no information about its previous count being above it or not, there will not be an alert assigned to it, when in fact it could be). To overcome this, in the input CSV file there are some extra counts that are only used to determine in which alert status is the first count to be analysed.

4.5.3 Interaction with SQL Server

In this module the SQL Server is used to provide data for the automatic alert system, and straightforward SELECT queries to select the alerts. They contain the day and its alert state, for daily aggregation, and the first day of the week and its alerts, for weekly aggregation. The SP used to select the
Figure 4.15: Example of the rules used in the VDMweb alert system. In the X axis there are the counts, and in the Y the baseline and limits. The 4th count exceeds the 99% upper limit, so it displays an alert and since the next two periods are below the 95% upper limit, the alert status returns to a non-alert state. The 7th count is higher than the 95% limit, however its next count is not, so an alert is not identified. The next alert is in the 9th count, since this and the next count are above the 95% limit. This alert status is maintained until the 14th count, when finally the first of two consecutive counts is below the 95% limit. During this last alert period, the 12th count is lower than the 95% limit, however as it is required two consecutive counts lower than the limit, the alert status is not changed.

alerts consists of crossing these tables with the baselines tables and with the count of the observed values (for daily aggregation) or the view vwWeekly, resulting in the set of columns displayed in each GridView. However, as the alerts update process also involves getting information from the SQL database, there are 2 SPs for each page. One is the SP just described, the other is to provide data that is exported into the CSV file to be used by the R scripts. The latter uses a variable corresponding to the number of days/weeks whose alerts the user has requested to be updated. This implicates that the select statement will only select the top days or weeks, plus some rows to provide the background alert state referred in the previous sub-section, and its values/baselines. To insert the updated values after the execution of the R scripts, the top rows in the respective alert table are deleted. The comparison map of alerts uses a straightforward select query of each alert table, depending only in the parameters of which stratification is requested. These parameters determine which alert table is to be selected using IF statements: the total, age group, region or gender, and in the case of regions, which territorial division - ARS or NUTS.

4.6 Module VII: Nowcasting

4.6.1 User Interface

The Nowcasting module is implemented in VDMweb following a recent dissertation project [38] carried out at DEP. Its author developed an R statistical algorithm to correct the number of deaths given the usual delay in their notification, taking into account multiple criteria. This module, since it is the first time that it is automatically implemented and used on a daily basis, is considered to be in a test phase. A plot with the observed values and the nowcast values, as well as the alert GridView of the nowcasts, are displayed within this module. Nowcasting pages are accesed by the buttons in Figure 4.16a.

The chart displayed in the Daily Chart web page contains the total observed values and the nowcast values - this algorithm is implemented for the daily aggregation, therefore there is no weekly values. In this page the user can also select which baselines, of the observed values, to visualize in
Figure 4.16: (a) Nowcasting web pages buttons. (b) Screenshot of the daily plot of observed deaths and nowcasts in Nowcasting.

the plot, as presented in Figure 4.16b. It has a particularity, related to updating the nowcast values. Similarly to the Alerts, on the Nowcasting the user can click a button to update the nowcasts. Since the algorithm always updates the last 15 days and this is not a changeable parameter, there is not an user-defined number of days. The process flow for the generation of nowcasts by the R script is essentially the same as the one in the Alerts module, with a difference in the bulk loading phase: the output file containing the nowcasts of the last 15 days is parsed into a data table that is used as a parameter in an SQL SP. The alerts for the Nowcasting are performed exactly the same way as in the Alerts module, but instead of analysing the total observed values it analyses the newly determined nowcasts.

4.6.2 Interaction with R - Nowcasting Implementation

The nowcasting algorithm allows the daily estimate of deaths that have occurred but were not yet notified to the VDM system, by modelling the delay of its notification using logistic regression models. For this purpose many variables are used in the models, mostly related to the delay of the notification
of deaths, such as whether the day of registration or prediction was a holiday or on which weekday the death occurred - the thesis can be found in INSA’s online repository [38]. The R script in which this method is operated was provided to me in order to integrate it in the VDMweb data analysis, hence the generation of this module. The script is executed from the code behind page when the event associated with clicking the update button in the UI is triggered. Its input CSV file consists of 2 fields, the day of death and the day of registration. After running the algorithm the script generates an output CSV file for the last 15 days with the number of notified deaths in each and the predicted value (nowcast). The result is a continuous update of the nowcast values, which are stored in the database and displayed as in Figure 4.16b. In this Figure, the start of the nowcast series corresponds to the first day that the script was used in INSA after successfully implementing the VDMweb. The alerts R script for this module is equal to the one explained in the previous section for the Alerts module, with the same alert assessment but for the nowcasts values.

4.6.3 Interaction with SQL Server

To display the nowcasting plot, the SQL SP used is the SP used for the total daily mortality chart in the Daily web page from the Reporting module, since both charts are basically the same, with the exception of the nowcasts. In the Daily one, the resulting fields of the SP do not include the nowcast values from the nowcasting database table (whose fields are day and respective nowcast), because the parameter that enabled its selection, @Nowcast, was 0. This variable is defined with the BIT data type, and when the SP is called from a Reporting web page, it is equal to 0. When it is called from the Nowcasting module, then that parameter is set to 1 so that the nowcast series is added to the chart. The table nowcasting is updated through the process of updating the nowcast values. First, a SP is called so that the input CSV file of the R script is created by the VDMweb, consisting of the selection of the two fields that are used by the nowcasting algorithm - day of death and day of file registration. The day of file registration is obtained from the name of the mortality records file that is ingested in the database, since this has a standard name of the type “DeathsYYYYMMDD.xml”. The date “YYYYMMDD” corresponds to the day of the notification of the file by IGFEJ, usually a day before when DEP receives it (e.g. in the day 2015-01-01 the filed received by DEP would be relative to 2014-12-31 with the name “Deaths20141231.xml”), unless it is a Monday and then the file is relative to Friday and only occasionally to the weekend days. The date is retrieved from the file name by selecting sub-strings of it and rearranging them to get the standard date “YYYY-MM-DD”. The day of death field corresponds to a straightforward selection of the last 15 days with records.

To integrate the updated values into the nowcasting table, another SP is used. The .NET datatable containing the nowcasting output CSV file is passed by the VDMweb as a parameter and defined in the SQL SP as a variable table, whose data type corresponds to an user-defined table type. This table type is basically the template of the datatable with its same fields and it was implemented to easily update the table nowcasting, without deleting the records like it is done in the Alerts module. A merge of the nowcasting table with the variable table is performed in the SP, comparing each combination of day and nowcast of the nowcasting table and the new updated one. If there is no match, it means
that there is no record in the database of the new nowcast values and the new day and nowcast are added to the table. However, if there is, the database nowcast for the respective day is updated with the new value.

For the nowcasting alerts, the SP used is analogous to those in the Alerts but using the nowcasts.

4.7 Module VIII: Data Management

4.7.1 User Interface

Data Management is where the user directly manages information within the database. This module provides options to export the VDM total data, to manually update the Regions table and to generate baselines. It groups 2 pages, one for the export and update and the other one to select the parameters used to calculate the baselines used in all data analysis methods, as their access buttons in Figure 4.17 indicate.

In the Regions Update web page the user can export all information that can be acquired for each record in the database. This was a specific DEP request with the purpose of providing an alternative to the already processed data displayed in the rest of the VDMweb. Therefore this is a way of making available all the raw data. In here the user selects the time period from which he wants data. Since the application deals with very large files (reaching almost 100 MBs depending on the time period extracted) the SqlDataReader class is used. This class efficiently handles the memory used in the creation of the file by directly reading the SQL SP result, as it is being retrieved. Then a StringBuilder generates the string of the result in a CSV type and a StreamWriter writes it into the file that is saved in the server. Posteriorly, this file is transmitted to the end-user computer via web browser.

The update of the Regions table is sometimes performed when there is a change in some geographical code, so that the stratification by region is always updated. The user exports the entire Regions table, which is handled by the same way as the one previously explained. Then, in the user private environment, the table is updated with the most recent changes. This updated table is then uploaded into the VDMweb which then proceeds to its data ingestion in the database. The FileUpload control used in this upload UI has restrictions that are checked before saving the table Regions file in the server, for instance checking if the extension of the uploaded file is ".csv". After it is saved in the server, the Regions file is imported into a .NET data table by using the C# function created for that purpose. An SqlBulkCopy class is defined to complete the ingestion of the datatable into the

Figure 4.17: Data Management web pages buttons.
Figure 4.18: User-defined parameters for the generation of daily baselines in the Data Management.

The generation of baselines is handled in the other page of this module, being divided by daily and weekly baselines but using exactly the same methods and controls. Before entering in more details, it is an important reminder that for the generation of baselines the periods where mortality events were observed are excluded, so it is important to have the periods of these events stored in the database. To do so, in the baseline generation page the user can export a table with these events by day/week to check what events are missing in the database. After updating the file in his private environment, he uploads it into the database through VDMweb. Then he proceeds to the automatic generation of baselines. First, the historical input data used in the baselines R scripts is selected. Next to the text-boxes where this is defined, there is a red button that is used to generate the baselines and store their values in the database. The event triggered by this button starts by generating a CSV file containing the information used in the R script to generate the baselines. Then the R script is executed by the process "Rscript.exe". Its multiple outputs, consisting of one CSV file per stratification that contains all 5 baselines for each stratum, are imported into the respective database tables to store their values, using SqlBulkCopy classes, one for each.

Next to the generate button is the option to export all the baselines that are stored in the database. This, similarly to the other exports in this module, generates the file in the server and then sends them to the client. However, since now there are multiple files (one per stratification), they are assembled and exported in a ZIP file. Since baselines are crucial for the operation of VDMweb, the user can also manually upload each stratification baselines file, to ensure that the system always displays baselines are correct. Figure 4.18 presents the sequel of controls used in the generation of the daily baselines - for the weekly ones it is the same, but with combination of weeks and years instead of days.

4.7.2 Interaction with R - Generation of Baselines

The R scripts used in the generation of baselines were derived from the algorithm already used to calculate the weekly baseline, referred in Sub-Section 3.2.2. The original script with the algorithm for the total weekly mortality values was provided to me in order to integrate it in the new system, however the purpose was to expand its use for all stratifications. So the R script was adapted and several R scripts were created from it in order to calculate the baselines for each stratification, also expanding
its use to the daily aggregated data. The input file for the scripts contains the day (or first day of the week and week number in the weekly aggregation), the identification of whether there was an event associated with the day (1 if yes, 0 if not), and the number of observed deaths for each stratum of every stratification (total, age group, regions, gender). Each of these strata has 5 baselines assigned to it (the 99% and 95% upper and lower confidence limits and the expected mortality baseline fit), so as the algorithm is executed in the R script, the output file is separated in multiple files, by stratification (total, age groups, regions by ARS, regions by NUTS and gender).

In the case of the daily cyclic regression, the algorithm predicts up to one year after the last date of the input file. This is usually the last day of the year that has passed, for instance the input data with the time period 2007-01-01 to 2014-12-31 results in a model adjustment to dates in this interval and until 2016-01-01, one year after the to date.

For the weekly baseline the same logic is applied. The model is adjusted until one year after the last week of the file, which is usually the week 20, given that the effects of influenza are more severe between the weeks 40 and 20 of the posterior year - which is also the time period used to generate the baseline (for instance from week 40 of the year 2007 until week 20 of 2014.).

4.7.3 Interaction with SQL Server

The table of the VDM raw data exported to the client in the Data Management web page corresponds to selecting all fields of the view vwDeaths plus the day of registration of the daily mortality records file, exactly like it is implemented for the nowcasting update SP. The SP used in the export of the table Regions corresponds to a select of the entire table.

The baselines require more SQL Server processing. The .NET datatable storing the events and respective periods in VDMweb is sent as a parameter into the SQL SP, which defines it as the respective user-defined table data type (daily or weekly). Then a merge of this new table with the events table in the VDM database is performed, updating the event values that already existed and inserting the new ones. To generate the result set that is written into the file holding the historical input data to calculate the baselines, a select statement is executed to get all the death counts for each stratum using COUNT with the respective stratum criteria. The imports performed in the web page are not carried out by SPs, but instead by the class defined in the code behind page, as well as the command to delete the target table content before adding the new values to be imported.

4.8 Summary

The new VDM version has a remodelled front-end application, VDMweb. This application centralizes all the operations in the VDM daily workflow by INSA’s professionals and adds new features.

The redesigned VDM front-end has six new modules and the two modules that existed in the previous front-end were reformulated. The VDM database had several changes, in parallel with the development of the front-end application, so that all data related to the new features are also stored in the same database. In total, 22 tables and 55 SP were added to the previous VDM database.
Regarding data analysis, the interaction with the R environment involved the implementation of several R scripts. Of these scripts, INSA provided the nowcasting method script and the script being used in VDM-2007 for the baseline of total weekly data, which I adapted and extended for daily data and all of the stratifications on both aggregations.

Among the many new features of VDMweb, there are plots for each stratification with their baselines, the alert system enables a comparison between every strata and the nowcasting method was successfully implemented. The combination of these and the new stratification features (new strata and new combined strata queries) enable the display of more specific information, resulting in a powerful tool displaying the mortality information, both numerical and graphical, required by public health researchers and national authorities.

The next chapter briefly recaps the major differences between the old and new VDM system version, and a case study is performed to illustrate the potential of the new VDMweb features for mortality monitoring.
Contents

5.1 Comparison between ICAROweb and VDMweb ........................................ 64
5.2 Data Visualization - Case Study .......................................................... 65
5.3 Summary ......................................................................................... 75

Evaluation
The new VDM system was evaluated through informal interviews with its end-users (2-3), since the number of users is very small. They all stated that the new VDM system is a lot more intuitive, direct and helpful. The response time of the VDMweb was considered adequate by its users, similar to the previous VDM version, despite handling more new features. In short, the performance of VDMweb was praised by INSA professionals that use it, reason why it is already in production at INSA. This chapter starts by pointing out the main differences between the previous and the new VDM version, in Section 5.1. Section 5.2 then presents a case study of the recent January 2015 epidemic that hit Portugal, to illustrate the potential of the new VDM system, using screenshots of the VDMweb front-end features.

5.1 Comparison between ICAROweb and VDMweb

The VDMweb provides new and more powerful features than the previous version, ICAROweb.

For data ingestion, instead of waiting for the ingestion tool to be called to integrate the mortality records file, the INSA professional now only has to upload the file through VDMweb, which is integrated automatically in the VDM database. The message appearing when the file is successfully integrated, or when an error has occurred, was also praised because now the user can track the status of the integration of the mortality records file.

Regarding data analysis, in the previous VDM version it was divided by different softwares, after extracting the data (not stratified) from ICAROweb. It involved many operator-supervised data exchanges on a weekday routine, in order to observe plots, which were not generated for all data stratifications. With the new VDM version, this analysis is handled by the new system, and the plots are displayed in VDMweb, for every stratification in daily and weekly aggregations, with the addition of their baselines. Besides, the ability to select any time period to visualize on the chart and customize its X axis, all of which are selected in the UI of the web application, represents another upgrade on the system. That way, the public health practitioner can rapidly isolate periods and display it in plots to better observe small time periods that are relevant to investigate to draw conclusions.

With the introduction of the new method to map the geographical code of the region of death and the ARS/NUTS, the new VDM system accounts more deaths that were not being assigned to any region before. At the time that this change was implemented 4226 records in the entire database had a valid geographical code (i.e. not null and different than "000000"), but no NUTS III code assigned to it. Now only 923 records have no NUTS III code, which represents a decrease of 80% of the records without assigned regions. Although the size of the database turns this into a small variation on the data analysis, some days had changes in their number of deaths of up to 5 deaths. Regarding NUTS II, the number of records without a region assigned in this new VDM version is 0, as well as ARS. However, some records are assigned to an "Undetermined" region, specifically 365 for the NUTS II case and 271 for the ARS one. So with VDMweb the already small number of records not being assigned to regions are even more reduced. The remaining of the records without regions are due to the fact that sometimes CROs itself don’t have the exact location of death and therefore the noted
geographical code corresponds to a broad geographical location.

Other important addition is the automatic alert system. While in the previous VDM version, the alerts were identified *ad-hoc*, with the new VDM system the alerts are visible in coloured tables displayed in VDMweb. Besides, the nowcasting integration into the system is unprecedented. These were positively referenced when presenting the application to DEP. The fact that they are automatic and based on independent R scripts, that can be altered by DEP investigators, was also pointed out as one of the major strengths of the new VDM. This allows for the iterative and continuous development of the methods used in the scripts.

Reporting is also automatically done in VDMweb. Using ICAROweb, the daily bulletin and the numerical information were created as the result of several data exchanges between the ICAROweb, MS Access and MS Excel. With the new VDM system, the generation of reports is centralised in the VDMweb front-end. To better compare the old and the new reports, the previous and the new daily bulletin, consisting of the numerical information table and the report, are provided in Appendix A and in Appendix B respectively. The new daily report presents more data, with more stratifications with the respective baselines, contrarily to the previous report. This process automation is seen as one of the major upgrade in the system. It dispenses several steps that had to be taken on a daily basis by INSA professionals in order to generate the daily report, which were very time consuming for the public health practitioner.

5.2 Data Visualization - Case Study

To better illustrate the potential of the application, I present as case study the influenza epidemic that recently hit Portugal between January and February 2015. Since influenza events tend to be analysed by week at INSA, this study uses the application features in the weekly data aggregation. All the images displayed in this section were automatically displayed in VDMweb and were retrieved from the system with data updated until 2015-05-25. It is important to keep in mind that in all charts there is a sudden decrease in the series’ most recent week (in the daily aggregations this is equivalent to a decrease in the last 4 days) due to the delay in the notification of deaths. This results in the displaying of a lower number of deaths that were recorded, to be updated in the following days.

To start with, the annual comparison plot of the number of deaths over the last five years, confirms that this year started with an increase in the mortality observed values in its first two months, January and February (see Figure 5.1a).
Figure 5.1: (a) Annual comparison plot of the number of deaths over the last 5 years. (b) Annual comparison of the number of deaths between 2015 and the years with high number of deaths in January and February since 2007.

It is clear the high number of deaths in January of this year (thick blue series). In February, although this number gradually decreased, it was still high comparing to other years represented in the chart, with the exception of 2012 (in yellow). On a side note, the mortality impact of the heat-wave that occurred in July 2013 is also quite highlighted. When selecting the years with the highest number of deaths in January, as Figure 5.1b shows, the peaks of January 2015 still rank the highest, hence the gravity of the mortality impacts during this specific period. These years can be selected until 2007, since this is the oldest year with data in the VDM system.
Figure 5.2: (a) Plot of weekly number of deaths between 2013-05-29 and 2015-05-25 and its baseline and 95% and 99% upper confidence limits. (b) Plot of weekly number of deaths between 2014-09-28 and 2015-05-10 and its baseline and 95% and 99% upper confidence limits.

This increase in the mortality absolute values is confirmed by observing the weekly chart displayed in the VDMweb. This chart allows the visualization of mortality evolution through time by week, represented in Figure 5.2a. The selection of the time period, in the "Filter" section of the VDMweb web page, in order to cover only the winter season enables a more direct analysis and detection of the distribution of mortality through this period, as Figure 5.2b presents.

In Figure 5.2a, the peak in 2013 (between week 25 and week 30) due to the heat-wave is still
observed. This event was a more punctual event than the one responsible for the mortality excess in the winter season in the first weeks of 2015, which dragged for a long number of weeks. Besides, this difference between the two distinct events can also be concluded from the annual comparison plot, given that the mortality peak in summer 2013 resulted in an abrupt peak of excess mortality.

Regarding Figure 5.2b, comparing its observed values with the upper limits of the baseline enables the identification of the periods associated with excess mortality.

This excess can also be detected using the new automatic alert system. The alert system for the total observed values is presented in Figure 5.3. In the table provided in this figure, the excess peaks corresponding to alerts are assigned to each week.

The alert table enables the conclusion that Portugal was in an alert state (represented by red) in the period corresponding to the excess mortality recorded from week 1 to week 9 of 2015. As this table is exportable to a public health professional environment, the absolute values of excess death in each of these weeks can be calculated to quantify the gravity of the excesses. This is achieved by subtracting, to the observed values, the "fit" value that corresponds to the mortality baseline (the "upperline" and "upperline99" in the table in Figure 5.3 are its 95% and 99% upper confidence limits).

To identify which age group, region or gender stratum is more affected by the event causing the excess death that results in the alert status, the comparison map of alerts for each stratification can be used. For the case analysed in this section, these maps are presented in the following figures, one for each stratification, including the two territorial divisions - NUTS II and ARS.
Figure 5.4: (a) Comparison table of weekly alerts by ARS regions between 2014-12-15 and 2015-05-10. (b) Comparison table of weekly alerts by NUTS II regions between 2014-12-15 and 2015-05-10.
During the January 2015 influenza epidemic, the only alert difference between ARS and NUTS II regions, in Figure 5.4, was in week 52. In this week, the NUTS II Centro was in alert, while its homologous ARS region did not. This shows the usefulness of differentiating both territorial divisions.

Figure 5.5: (a) Comparison table of weekly alerts by age groups between 2014-12-15 and 2015-05-10. (b) Comparison table of weekly alerts by gender between 2014-12-15 and 2015-05-10.
In the age groups, the number of consecutive weeks with red alerts increased from the younger groups to the eldest. In fact, there were not identified alerts for groups younger than the 45-64, with the exception of an isolated alert in the first week of 2015 in the age group of 0-14 years. On the opposite side, in the \( \geq 85 \) age group, the alert status was continuous from week 1 until week 10. Observing the overall alerts for this stratification, the age groups were, in general, more affected in the beginning of the event. Towards the end of it, only the elderly were still associated with excess mortality, explained by the fact that they represent the most vulnerable group. As the influenza epidemic was battled with measures, it stopped affecting so much the less vulnerable age groups. The stratification by gender also shows what was expected, because since influenza is not gender-specific, the alerts were evenly distributed. This alert information is relevant if only one gender shows a more regular pattern of alerts, because then hypothesis about gender targeted events should be made and measures regarding it should be taken. The same logic applies for regions and age groups, if the alerts are concentrated on specific strata. To retrieve numerical information similar to Figure 5.3, VDMweb also displays one table with that information for each strata of each stratification.

Besides the alerts, the plots displayed for each data stratum can also be used to study events. They complement the alerts system by presenting a graphical variation of the mortality time series. On the other hand, the alerts complement the plots by identifying exactly which periods are associated with excess mortality. The plots of the most relevant strata in this case study (the ones with red alerts in the alerts figures), are presented next, followed by a brief analysis. These plots represent an alternative approach to determine the magnitude of the epidemic effects, mostly from observing the deviations of the observed values’ series, in light blue, and the upper baselines.

**Figure 5.6:** Plots of weekly number of deaths by gender during the influenza epidemic, their baseline and 95% and 99% upper confidence limits.
The charts of the stratification by gender in Figure 5.6 indicate that in the feminine gender the impact of the pandemic was more severe, since the observed values are more deviated from the 99% limit than the ones registered for the masculine gender. In the latter, although some are higher than the 99% upper confidence limit, the observed values are closer to this limit and they even reach a value lower than the 95% limit, in week 8. However, the alert status is not cleared since on the following week, the 9th, the values instead of decreasing again and confirm the exit from the red alert, a small increase was recorded. On the opposite side, in the feminine chart the only weeks that were lower than the 99% limit correspond to the first and last week of the entire period with excess death - when the epidemic started spreading and stopping, respectively.

Figure 5.6: Plots of weekly number of deaths by the most affected ARSs (Norte, Centro and LVT) during the influenza epidemic, their baseline and 95% and 99% upper confidence limits.
The same analysis exercise as the previous one is now performed for the region charts in Figure 5.7, which only display the most affected 3 out of the 7 regions displayed in VDMweb, by ARS. The observed values of mortality increased from the ARS LVT to the ARS Norte. In fact, the deviation of the observed values’ series and the 99% upper limit in the Norte region were higher than any other, so the influenza epidemic effects were more severely felt in this region. Norte was followed by Centro and then LVT, while Alentejo, not shown in the figure, registered a very large peak during this period in week 3. The remaining 3 regions (Algarve, Madeira and Açores) did not register such high values nor even alert states, with the exception of 2 weeks in Algarve (7 and 8) that were higher than the 95% limit but lower than the more severe 99%.

**Figure 5.8:** Plots of weekly number of deaths by the eldest age groups (65-74, 75-84 and ≥85) during the influenza epidemic, their baseline and 95% and 99% upper confidence limits.
Regarding age groups, in Figure 5.8, the elderly groups, as discussed before, were the most affected by the influenza epidemic, given their higher vulnerability. Comparing the observed mortality series with the 95% and 99% limits the conclusions are the expected ones: the excess deaths were higher in the 85+ years group, followed by the 75-84 and then 65-74.

To retrieve absolute values in the periods identified above, VDMweb provides tables in the *Mortality Surveillance* modules, one for daily and another for weekly data. Using the customized search, the public health practitioner can download a table with all the data strata necessary to analyse numerical statistics. For instance, the user can request the absolute values of all the strata represented in the plots. The weekly *Customized* search results, of the strata pictured in the plots of the previous three figures, is shown in Figure 5.9. Additionally, other searches by stratification can be performed. To get information on a certain target group that is not covered by one stratification since it crosses all the three (age groups, regions and gender), the *Parametrized* search could be used. This is a relevant tool if there is the need to search for the number of deaths in a certain region of a certain gender within an age group, providing more accurate information to support interventions.

![Tabela Customizada de Registos Semanais (ARS)](image)

**Figure 5.9:** Weekly *Customized* table with absolute values for different strata by ARS regions, between 2014-12-15 and 2015-05-10. The strata selected are the ones whose plots are displayed in this section.
In the example analysed in this section, all the information is related to a past time period. However, as all of these data are constantly updated, both the plots and alerts are always related to current events that might be happening.

For a more specific analysis, this analysis exercise can be applied to the daily aggregated data, since the implemented features are the same. When studying influenza events, daily mortality data presents a lot of variation in the observed values, given that an influenza epidemic is a prolonged event and not a focused one like a heat-wave. For that matter, during extreme weather events, analysis by days are more adequate. As all the logics used in the data analysis are the same as the ones discussed in this weekly case study, Appendix C presents the plots and alarm tables for daily data during the heat-wave in 2013, as an example of the VDMweb potential for data visualization of daily aggregated data. This heat-wave lasted from the end of June until mid-July of 2013, and it affected the most the feminine gender, Norte and Centro regions and the eldest age groups (75-84 and ≥85 years).

5.3 Summary

Overall, the new VDM was praised by the end-users due to its simplicity and automatic processing of workflows. It is an intuitive and automatic tool that centralizes all the VDM system components and enables a more efficient workflow for INSA’s public health practitioners, who praised the application and its new features.

Using the new features enables a more effective study of the effects that several health-related events have on mortality by automatically determining when excess peaks occur. This enhances the potential of the mortality surveillance operated by the VDM system to identify possible threats of epidemics or other life-threatening events and to pass that information to the appropriate authorities, as well as what groups of the Portuguese population is being more affected, either age groups, regions or gender. The analysis presented in this chapter showed the added usefulness of VDMweb. It now includes many monitoring outcomes that can be used either by the experts or the decision makers.
Conclusions and Future Work

Contents

6.1 Conclusion ................................................................. 78
6.2 Future Work ............................................................. 79
6.1 Conclusion

The main objective of this project, the design and implementation of a new version of the VDM system, was completed and the system is now deployed at INSA. The implementation covered the integration of the data ingestion, analysis and reporting components, and was user-validated and tested. The VDM system now in production at DEP/INSA, is used on a daily basis by its public health experts.

The main focus of the project was given to data analysis and visualization, a crucial requirement of the system. I intended to provide a high number of custom options so that the public health professionals at INSA that use the application could request different types of information, either to use it for further analysis or to send it to the requesting entities. This customization was successfully implemented, making use of several controls from which the users can select parameters to query for information on each stratification implemented, for both data aggregation - daily and weekly.

The user-customized information requests, the display of several charts, the nowcasting module and the automatic alert system provide new and more complete information. All these features benefit from the ASP.NET controls versatility, along with the storing and query processing of the SQL Server and the statistical computing power of the R environment. The integration of these three frameworks is key to the system, with the integration of the R environment providing the most innovative feature now available in the new system. The possibility of manipulating and changing the R scripts outside of the application server opens a new door to researchers. Since the VDMweb operation is now completely automated, the opened possibility of testing new algorithms and methods for the generation of baselines, nowcasting or alerts is rather useful. The most adequate method can be chosen iteratively, by updating the R scripts. The new VDM application allows the direct observation of the resulting changes in each of the respective modules in the VDMweb. This gives to public health experts a more manageable automatic system, allowing it to be always updated with better and more accurate analysis algorithms. By doing so, this VDM version provides precise and valuable data relevant for an extended lifetime.

The automatic reporting reduces the time and computing resources used in the generation of the daily reports. Besides, with the introduction of the baselines, these reports were also complemented with new information. This was appreciated by INSA public health practitioners, since it dispenses the exhaustive routine of manually shipping data between software modules. Regarding the reports, as new information was added, there is also an improvement of the information sent to decision makers. They can rely on more information that was acquired to take measures or implement new health policies to mitigate effects of health-related events.

To sum up, the new VDM represents an improvement to the previous, with new added and user-customized features. Its workflow is more efficient and autonomous, resulting into faster and more detailed information to be used on the detection and estimation of excess deaths. With the new system and its new features, DEP/INSA consolidates its major role in the mortality monitoring in Portugal, providing even more complete and timely information to authorities.
6.2 Future Work

One improvement to VDMweb would be the introduction of information related to the incidence rates of influenza-like illness (ILI). As ILI are a major cause of mortality, it would be interesting to automatically overlay an incidence rate series in the mortality charts displayed, with two Y axis. Similarly to ILI, the ICARO alerts could also be used in this series overlay instead of ILI, providing that option to users. This would enable the comparison of the distribution of mortality through time and the ILI incidence rates or ICARO-index, to determine if they are related or to refute their relationship. By doing so, these two causes of mortality peaks would be rapidly ruled out by observing the plots. This would let researchers focus on measures and policies for other events.

Another improvement would be the development of a framework to test, and ultimately adopt, different algorithms and methods to generate outbreak alerts. In this case, the outbreaks consist of periods associated with excess deaths. This area is well developed within public health and is currently one of the most focused areas of improvement. Statistical algorithms, such as the ones described by Farrington [39] and Stroup [40], can be tested and adopted if they present more timely results. As these are available in R packages, they can easily be implemented and tested in the application.

A third improvement would be the display of statistics alongside the tables already provided in the application. As this project aimed to provide an optimized tool to support mortality investigation, it was designed to provide numerical information, independently of the methods for the analysis of baselines, nowcasting and alerts. However, it could have embedded the automatic calculation of statistics in terms of percentage rates of the total mortality or by stratum, as well as calculating absolute values of excess deaths, among others. Implementing these features dispenses processing outside of the application and would complement VDMweb, making it an even more complete tool.

Lastly, as the parametrized queries are the result of parameters selected within each stratification, their baselines are not generated. The baselines are calculated only for the strata already determined in the system. It would be useful if they could be calculated on-the-fly for the mortality variation displayed in the parametrized query chart. This would provide more customization to analyse new different strata that are not covered by the pre-determined stratifications. To do this, an R script would have to be created and some changes in the database would have to be implemented. A session would have to be created to store the baselines values for the visualization of results.
Bibliography


[13] euroMOMO. Pooled analyses of all-cause mortality indicates low excess mortality in Europe in the winter of 2013/14, in particular amongst the elderly.


Previous VDM Version Daily Report
## Sistema de Vigilância Diária da Mortalidade

ded-feira, 25 de Maio de 2015

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Portugal</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2430</td>
<td></td>
</tr>
<tr>
<td>Portugal Continental</td>
<td>258</td>
<td>264</td>
<td>268</td>
<td>253</td>
<td>246</td>
<td>244</td>
<td>285</td>
<td>271</td>
<td>225</td>
<td>215</td>
<td>182</td>
<td>91</td>
<td>0</td>
<td>0</td>
<td>2314</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Norte</td>
<td>68</td>
<td>91</td>
<td>81</td>
<td>74</td>
<td>68</td>
<td>80</td>
<td>96</td>
<td>72</td>
<td>68</td>
<td>62</td>
<td>63</td>
<td>27</td>
<td>0</td>
<td>0</td>
<td>698</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Centro</td>
<td>88</td>
<td>66</td>
<td>79</td>
<td>59</td>
<td>63</td>
<td>64</td>
<td>74</td>
<td>80</td>
<td>61</td>
<td>64</td>
<td>52</td>
<td>30</td>
<td>0</td>
<td>0</td>
<td>634</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lisboa</td>
<td>66</td>
<td>69</td>
<td>81</td>
<td>82</td>
<td>76</td>
<td>62</td>
<td>72</td>
<td>77</td>
<td>62</td>
<td>58</td>
<td>48</td>
<td>21</td>
<td>0</td>
<td>0</td>
<td>647</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Alentejo</td>
<td>24</td>
<td>27</td>
<td>18</td>
<td>26</td>
<td>26</td>
<td>20</td>
<td>24</td>
<td>29</td>
<td>22</td>
<td>23</td>
<td>15</td>
<td>10</td>
<td>0</td>
<td>0</td>
<td>216</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Algarve</td>
<td>12</td>
<td>11</td>
<td>9</td>
<td>12</td>
<td>13</td>
<td>18</td>
<td>19</td>
<td>13</td>
<td>12</td>
<td>8</td>
<td>4</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>119</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Baselines*

<table>
<thead>
<tr>
<th>Portugal</th>
<th>272</th>
<th>276</th>
<th>288</th>
<th>274</th>
<th>276</th>
<th>275</th>
<th>275</th>
<th>264</th>
<th>246</th>
<th>257</th>
<th>266</th>
<th>264</th>
<th>257</th>
<th>247</th>
<th>272</th>
<th>263</th>
<th>265</th>
<th>277</th>
<th>272</th>
<th>2428</th>
</tr>
</thead>
<tbody>
<tr>
<td>Portugal Continental</td>
<td>254</td>
<td>257</td>
<td>268</td>
<td>255</td>
<td>253</td>
<td>260</td>
<td>240</td>
<td>230</td>
<td>237</td>
<td>247</td>
<td>240</td>
<td>239</td>
<td>230</td>
<td>251</td>
<td>241</td>
<td>240</td>
<td>259</td>
<td>251</td>
<td>2254</td>
<td></td>
</tr>
<tr>
<td>Norte</td>
<td>81</td>
<td>79</td>
<td>82</td>
<td>84</td>
<td>82</td>
<td>82</td>
<td>87</td>
<td>82</td>
<td>77</td>
<td>74</td>
<td>79</td>
<td>67</td>
<td>74</td>
<td>77</td>
<td>77</td>
<td>87</td>
<td>77</td>
<td>7167</td>
<td>718</td>
<td></td>
</tr>
<tr>
<td>Centro</td>
<td>65</td>
<td>66</td>
<td>70</td>
<td>62</td>
<td>64</td>
<td>68</td>
<td>58</td>
<td>66</td>
<td>64</td>
<td>63</td>
<td>65</td>
<td>64</td>
<td>58</td>
<td>63</td>
<td>62</td>
<td>65</td>
<td>63</td>
<td>65</td>
<td>583</td>
<td></td>
</tr>
<tr>
<td>Lisboa</td>
<td>67</td>
<td>77</td>
<td>75</td>
<td>71</td>
<td>65</td>
<td>72</td>
<td>66</td>
<td>62</td>
<td>61</td>
<td>70</td>
<td>65</td>
<td>67</td>
<td>69</td>
<td>77</td>
<td>71</td>
<td>65</td>
<td>72</td>
<td>72</td>
<td>616</td>
<td></td>
</tr>
<tr>
<td>Alentejo</td>
<td>27</td>
<td>23</td>
<td>30</td>
<td>22</td>
<td>24</td>
<td>23</td>
<td>22</td>
<td>22</td>
<td>19</td>
<td>26</td>
<td>24</td>
<td>20</td>
<td>23</td>
<td>23</td>
<td>21</td>
<td>22</td>
<td>22</td>
<td>24</td>
<td>212</td>
<td></td>
</tr>
<tr>
<td>Algarve</td>
<td>13</td>
<td>10</td>
<td>14</td>
<td>17</td>
<td>14</td>
<td>13</td>
<td>10</td>
<td>11</td>
<td>10</td>
<td>10</td>
<td>11</td>
<td>8</td>
<td>11</td>
<td>14</td>
<td>9</td>
<td>10</td>
<td>13</td>
<td>12</td>
<td>112</td>
<td></td>
</tr>
</tbody>
</table>

* média dos valores diários (mesmo dia e mês) observados em 2011, 2012, 2013 e 2014
Figure A.2: Page 1 of the previous daily bulletin disseminated by INSA, containing plots of the total daily data and the annual comparison.
Sistema de Vigilância Diária da Mortalidade
Regiões de Portugal continental
Informação entre 1-1-2011 e 24-5-2015

segunda-feira, 25 de Maio de 2015


Figure A.3: Page 2 of the previous daily bulletin disseminated by INSA, containing daily plots of the Portugal Mainland NUTS II regions.
VDMweb Daily Report
**Vigilância Diária da Mortalidade - ARS**

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Portugal</td>
<td>269</td>
<td>271</td>
<td>285</td>
<td>274</td>
<td>259</td>
<td>258</td>
<td>292</td>
<td>285</td>
<td>237</td>
<td>227</td>
<td>196</td>
<td>98</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2430</td>
</tr>
<tr>
<td>Masculino</td>
<td>150</td>
<td>134</td>
<td>141</td>
<td>156</td>
<td>133</td>
<td>125</td>
<td>139</td>
<td>140</td>
<td>117</td>
<td>109</td>
<td>102</td>
<td>50</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1235</td>
</tr>
<tr>
<td>Feminino</td>
<td>119</td>
<td>137</td>
<td>144</td>
<td>118</td>
<td>126</td>
<td>133</td>
<td>153</td>
<td>145</td>
<td>120</td>
<td>118</td>
<td>94</td>
<td>48</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1195</td>
</tr>
<tr>
<td>Portugal Continental</td>
<td>258</td>
<td>266</td>
<td>269</td>
<td>253</td>
<td>246</td>
<td>244</td>
<td>285</td>
<td>271</td>
<td>225</td>
<td>215</td>
<td>182</td>
<td>91</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2317</td>
</tr>
<tr>
<td>Norte</td>
<td>68</td>
<td>93</td>
<td>82</td>
<td>74</td>
<td>68</td>
<td>80</td>
<td>96</td>
<td>72</td>
<td>68</td>
<td>62</td>
<td>63</td>
<td>27</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>701</td>
</tr>
<tr>
<td>Centro</td>
<td>69</td>
<td>52</td>
<td>58</td>
<td>49</td>
<td>47</td>
<td>49</td>
<td>53</td>
<td>55</td>
<td>47</td>
<td>49</td>
<td>39</td>
<td>27</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>479</td>
</tr>
<tr>
<td>LVT</td>
<td>91</td>
<td>106</td>
<td>96</td>
<td>100</td>
<td>83</td>
<td>97</td>
<td>111</td>
<td>87</td>
<td>79</td>
<td>65</td>
<td>27</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>865</td>
</tr>
<tr>
<td>Alentejo</td>
<td>18</td>
<td>16</td>
<td>14</td>
<td>22</td>
<td>18</td>
<td>14</td>
<td>20</td>
<td>20</td>
<td>11</td>
<td>17</td>
<td>11</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>153</td>
</tr>
<tr>
<td>Algarve</td>
<td>12</td>
<td>11</td>
<td>9</td>
<td>12</td>
<td>13</td>
<td>18</td>
<td>19</td>
<td>13</td>
<td>12</td>
<td>8</td>
<td>4</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>119</td>
</tr>
<tr>
<td>Madeira</td>
<td>2</td>
<td>3</td>
<td>5</td>
<td>11</td>
<td>8</td>
<td>8</td>
<td>1</td>
<td>5</td>
<td>3</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>46</td>
</tr>
<tr>
<td>Açores</td>
<td>5</td>
<td>2</td>
<td>6</td>
<td>5</td>
<td>4</td>
<td>5</td>
<td>5</td>
<td>9</td>
<td>7</td>
<td>9</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>46</td>
</tr>
</tbody>
</table>

**Média dos últimos 4 anos***

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Portugal</td>
<td>272</td>
<td>276</td>
<td>288</td>
<td>274</td>
<td>276</td>
<td>275</td>
<td>264</td>
<td>246</td>
<td>257</td>
<td>266</td>
<td>264</td>
<td>257</td>
<td>247</td>
<td>272</td>
<td>263</td>
<td>265</td>
<td>277</td>
<td>272</td>
<td>2428</td>
</tr>
<tr>
<td>Masculino</td>
<td>143</td>
<td>141</td>
<td>145</td>
<td>137</td>
<td>136</td>
<td>134</td>
<td>137</td>
<td>128</td>
<td>126</td>
<td>137</td>
<td>131</td>
<td>130</td>
<td>129</td>
<td>143</td>
<td>141</td>
<td>136</td>
<td>143</td>
<td>142</td>
<td>1227</td>
</tr>
<tr>
<td>Feminino</td>
<td>128</td>
<td>134</td>
<td>142</td>
<td>137</td>
<td>139</td>
<td>141</td>
<td>126</td>
<td>118</td>
<td>130</td>
<td>128</td>
<td>132</td>
<td>127</td>
<td>118</td>
<td>128</td>
<td>122</td>
<td>128</td>
<td>133</td>
<td>130</td>
<td>1195</td>
</tr>
<tr>
<td>Portugal Continental</td>
<td>255</td>
<td>259</td>
<td>270</td>
<td>256</td>
<td>255</td>
<td>261</td>
<td>242</td>
<td>231</td>
<td>239</td>
<td>248</td>
<td>241</td>
<td>241</td>
<td>230</td>
<td>252</td>
<td>243</td>
<td>242</td>
<td>261</td>
<td>253</td>
<td>2268</td>
</tr>
<tr>
<td>Norte</td>
<td>81</td>
<td>80</td>
<td>79</td>
<td>82</td>
<td>85</td>
<td>83</td>
<td>82</td>
<td>67</td>
<td>82</td>
<td>77</td>
<td>74</td>
<td>79</td>
<td>67</td>
<td>74</td>
<td>78</td>
<td>78</td>
<td>88</td>
<td>77</td>
<td>721</td>
</tr>
<tr>
<td>Centro</td>
<td>46</td>
<td>52</td>
<td>51</td>
<td>49</td>
<td>53</td>
<td>53</td>
<td>43</td>
<td>50</td>
<td>47</td>
<td>47</td>
<td>51</td>
<td>48</td>
<td>44</td>
<td>46</td>
<td>48</td>
<td>50</td>
<td>45</td>
<td>49</td>
<td>442</td>
</tr>
<tr>
<td>LVT</td>
<td>93</td>
<td>98</td>
<td>102</td>
<td>91</td>
<td>85</td>
<td>97</td>
<td>87</td>
<td>84</td>
<td>85</td>
<td>94</td>
<td>86</td>
<td>89</td>
<td>89</td>
<td>100</td>
<td>91</td>
<td>87</td>
<td>97</td>
<td>96</td>
<td>822</td>
</tr>
<tr>
<td>Alentejo</td>
<td>20</td>
<td>18</td>
<td>23</td>
<td>16</td>
<td>19</td>
<td>13</td>
<td>17</td>
<td>17</td>
<td>14</td>
<td>19</td>
<td>17</td>
<td>15</td>
<td>19</td>
<td>17</td>
<td>15</td>
<td>16</td>
<td>17</td>
<td>17</td>
<td>157</td>
</tr>
<tr>
<td>Algarve</td>
<td>13</td>
<td>10</td>
<td>14</td>
<td>17</td>
<td>14</td>
<td>13</td>
<td>10</td>
<td>11</td>
<td>10</td>
<td>11</td>
<td>8</td>
<td>11</td>
<td>14</td>
<td>9</td>
<td>10</td>
<td>13</td>
<td>12</td>
<td>112</td>
<td></td>
</tr>
<tr>
<td>Madeira</td>
<td>5</td>
<td>6</td>
<td>5</td>
<td>8</td>
<td>8</td>
<td>6</td>
<td>7</td>
<td>5</td>
<td>8</td>
<td>7</td>
<td>5</td>
<td>3</td>
<td>6</td>
<td>7</td>
<td>7</td>
<td>5</td>
<td>6</td>
<td>58</td>
<td></td>
</tr>
<tr>
<td>Açores</td>
<td>4</td>
<td>6</td>
<td>6</td>
<td>5</td>
<td>6</td>
<td>3</td>
<td>6</td>
<td>5</td>
<td>4</td>
<td>6</td>
<td>8</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>6</td>
<td>7</td>
<td>4</td>
<td>8</td>
<td>45</td>
</tr>
</tbody>
</table>

*Vermelho: Valores acima do valor médio mais o desvio padrão; Verde: Valores abaixo do valor médio mais o desvio padrão*
Número de óbitos registados por data do óbito
Todas as conservatórias do registo civil informatizadas
(SIRIC/IRN e IGFEJ/MJ)

Comparação anual do número de óbitos registados por data do óbito
Todas as conservatórias do registo civil informatizadas
(SIRIC/IRN e IGFEJ/MJ)

O decréscimo da série deve-se ao facto de os dados dos últimos dias não estarem actualizados.

Figure B.2: Page 1 of the daily bulletin disseminated by INSA, containing plots of the total daily data and the annual comparison.
Figure B.3: Page 2 of the daily bulletin disseminated by INSA, containing daily plots of the Portugal Mainland ARS regions.
Figure B.4: Page 3 of the daily bulletin disseminated by INSA, containing daily plots of Açores, Madeira and gender.
Figure B.5: Page 4 of the daily bulletin disseminated by INSA, containing daily plots of age groups from 0 to 64 years.
Figure B.6: Page 5 of the daily bulletin disseminated by INSA, containing daily plots of age groups older than 65 years.
Data Visualization of the 2013 Heat-Wave Effects on Mortality
**Figure C.1:** Plot of daily number of deaths between 2011-01-01 and 2013-05-25 and its baselines: the baseline, the 95% and 99% upper and lower confidence limits.

**Figure C.2:** Plot of daily number of deaths between 2013-06-01 and 2013-09-01 and its baselines: the baseline, 95% and 99% upper confidence limits.
Figure C.3: Table of alerts for total daily deaths between 2013-06-25 and 2013-07-21.

<table>
<thead>
<tr>
<th>dataObito</th>
<th>Obitos</th>
<th>Fit</th>
<th>upperline</th>
<th>upperline99</th>
<th>Alerta</th>
</tr>
</thead>
<tbody>
<tr>
<td>21-07-2013</td>
<td>248</td>
<td>236,811786</td>
<td>281,84670</td>
<td>296,01600</td>
<td>Green</td>
</tr>
<tr>
<td>20-07-2013</td>
<td>263</td>
<td>236,90756</td>
<td>281,93640</td>
<td>296,10570</td>
<td>Green</td>
</tr>
<tr>
<td>19-07-2013</td>
<td>247</td>
<td>237,01110</td>
<td>282,04010</td>
<td>296,20930</td>
<td>Green</td>
</tr>
<tr>
<td>18-07-2013</td>
<td>297</td>
<td>237,12880</td>
<td>282,15770</td>
<td>296,32700</td>
<td>Green</td>
</tr>
<tr>
<td>17-07-2013</td>
<td>254</td>
<td>237,20540</td>
<td>282,29330</td>
<td>296,45960</td>
<td>Green</td>
</tr>
<tr>
<td>16-07-2013</td>
<td>258</td>
<td>237,40500</td>
<td>282,43490</td>
<td>296,60400</td>
<td>Green</td>
</tr>
<tr>
<td>15-07-2013</td>
<td>280</td>
<td>237,56520</td>
<td>282,50410</td>
<td>296,70340</td>
<td>Green</td>
</tr>
<tr>
<td>14-07-2013</td>
<td>287</td>
<td>237,73840</td>
<td>282,76730</td>
<td>296,93650</td>
<td>Green</td>
</tr>
<tr>
<td>13-07-2013</td>
<td>312</td>
<td>237,92330</td>
<td>282,95410</td>
<td>297,12340</td>
<td>Green</td>
</tr>
<tr>
<td>12-07-2013</td>
<td>303</td>
<td>238,12590</td>
<td>283,15470</td>
<td>297,22990</td>
<td>Green</td>
</tr>
<tr>
<td>11-07-2013</td>
<td>378</td>
<td>238,34020</td>
<td>283,36890</td>
<td>297,53810</td>
<td>Green</td>
</tr>
<tr>
<td>10-07-2013</td>
<td>401</td>
<td>238,56000</td>
<td>283,59670</td>
<td>297,70550</td>
<td>Red</td>
</tr>
<tr>
<td>09-07-2013</td>
<td>430</td>
<td>238,80060</td>
<td>283,83830</td>
<td>298,03720</td>
<td>Red</td>
</tr>
<tr>
<td>08-07-2013</td>
<td>503</td>
<td>239,05410</td>
<td>284,09270</td>
<td>298,26180</td>
<td>Red</td>
</tr>
<tr>
<td>07-07-2013</td>
<td>424</td>
<td>239,33220</td>
<td>284,30070</td>
<td>298,52990</td>
<td>Red</td>
</tr>
<tr>
<td>06-07-2013</td>
<td>374</td>
<td>239,51320</td>
<td>284,54200</td>
<td>298,81110</td>
<td>Red</td>
</tr>
<tr>
<td>05-07-2013</td>
<td>319</td>
<td>239,90810</td>
<td>284,93650</td>
<td>299,04130</td>
<td>Red</td>
</tr>
<tr>
<td>04-07-2013</td>
<td>319</td>
<td>240,21570</td>
<td>285,24410</td>
<td>299,31290</td>
<td>Red</td>
</tr>
<tr>
<td>03-07-2013</td>
<td>292</td>
<td>240,53630</td>
<td>285,56460</td>
<td>300,02370</td>
<td>Red</td>
</tr>
<tr>
<td>02-07-2013</td>
<td>317</td>
<td>240,86590</td>
<td>285,89910</td>
<td>300,32710</td>
<td>Red</td>
</tr>
<tr>
<td>01-07-2013</td>
<td>331</td>
<td>241,21020</td>
<td>286,24490</td>
<td>300,41390</td>
<td>Red</td>
</tr>
<tr>
<td>30-06-2013</td>
<td>314</td>
<td>241,57520</td>
<td>286,60330</td>
<td>300,77220</td>
<td>Red</td>
</tr>
<tr>
<td>29-06-2013</td>
<td>311</td>
<td>241,94660</td>
<td>286,97460</td>
<td>301,14380</td>
<td>Red</td>
</tr>
<tr>
<td>28-06-2013</td>
<td>318</td>
<td>242,39050</td>
<td>287,35860</td>
<td>301,52700</td>
<td>Red</td>
</tr>
<tr>
<td>27-06-2013</td>
<td>272</td>
<td>242,72370</td>
<td>287,73520</td>
<td>301,92410</td>
<td>Red</td>
</tr>
<tr>
<td>26-06-2013</td>
<td>283</td>
<td>243,13600</td>
<td>288,16380</td>
<td>302,32360</td>
<td>Red</td>
</tr>
<tr>
<td>25-06-2013</td>
<td>276</td>
<td>243,59800</td>
<td>288,58450</td>
<td>302,75310</td>
<td>Red</td>
</tr>
</tbody>
</table>

Figure C.4: Comparison table of alerts for daily data by ARS regions between 2013-06-25 and 2013-07-21.
Figure C.5: Comparison table of alerts for daily data by NUTS regions between 2013-06-25 and 2013-07-21.

Figure C.6: Comparison table of alerts for daily data by age groups between 2013-06-25 and 2013-07-21.
Figure C.7: Comparison table of alerts for daily data by gender between 2013-06-25 and 2013-07-21.

Figure C.8: Plots of daily number of deaths by gender, their baseline and 95% and 99% upper confidence limits between 2013-06-01 and 2013-09-01.
Figure C.9: Plots of daily number of deaths in ARS Norte, Centro and LVT, their baseline and 95% and 99% upper confidence limits between 2013-06-01 and 2013-09-01.
Figure C.10: Plots of daily number of deaths by 65-74, 75-84 and ≥85, their baseline and 95% and 99% upper confidence limits between 2013-06-01 and 2013-09-01.
### Consulta Diária Customizada (ARS)

Escolha os parâmetros a visualizar na tabela:

<table>
<thead>
<tr>
<th>Grupos etários</th>
<th>Regiões</th>
<th>Sexo</th>
</tr>
</thead>
<tbody>
<tr>
<td>dataObito</td>
<td>65-74</td>
<td>75-84</td>
</tr>
<tr>
<td>21-07-2013</td>
<td>44</td>
<td>82</td>
</tr>
<tr>
<td>20-07-2013</td>
<td>42</td>
<td>91</td>
</tr>
<tr>
<td>19-07-2013</td>
<td>36</td>
<td>73</td>
</tr>
<tr>
<td>18-07-2013</td>
<td>42</td>
<td>112</td>
</tr>
<tr>
<td>17-07-2013</td>
<td>42</td>
<td>78</td>
</tr>
<tr>
<td>16-07-2013</td>
<td>36</td>
<td>88</td>
</tr>
<tr>
<td>15-07-2013</td>
<td>47</td>
<td>90</td>
</tr>
<tr>
<td>14-07-2013</td>
<td>47</td>
<td>91</td>
</tr>
<tr>
<td>13-07-2013</td>
<td>45</td>
<td>95</td>
</tr>
<tr>
<td>12-07-2013</td>
<td>31</td>
<td>96</td>
</tr>
<tr>
<td>11-07-2013</td>
<td>37</td>
<td>129</td>
</tr>
<tr>
<td>10-07-2013</td>
<td>54</td>
<td>151</td>
</tr>
<tr>
<td>09-07-2013</td>
<td>70</td>
<td>141</td>
</tr>
<tr>
<td>08-07-2013</td>
<td>53</td>
<td>161</td>
</tr>
<tr>
<td>07-07-2013</td>
<td>66</td>
<td>140</td>
</tr>
<tr>
<td>06-07-2013</td>
<td>44</td>
<td>125</td>
</tr>
<tr>
<td>05-07-2013</td>
<td>26</td>
<td>111</td>
</tr>
<tr>
<td>04-07-2013</td>
<td>53</td>
<td>98</td>
</tr>
<tr>
<td>03-07-2013</td>
<td>40</td>
<td>94</td>
</tr>
<tr>
<td>02-07-2013</td>
<td>44</td>
<td>105</td>
</tr>
<tr>
<td>01-07-2013</td>
<td>41</td>
<td>98</td>
</tr>
<tr>
<td>30-06-2013</td>
<td>49</td>
<td>103</td>
</tr>
<tr>
<td>29-06-2013</td>
<td>34</td>
<td>96</td>
</tr>
<tr>
<td>28-06-2013</td>
<td>38</td>
<td>115</td>
</tr>
<tr>
<td>27-06-2013</td>
<td>36</td>
<td>97</td>
</tr>
<tr>
<td>26-06-2013</td>
<td>34</td>
<td>104</td>
</tr>
<tr>
<td>25-06-2013</td>
<td>32</td>
<td>100</td>
</tr>
</tbody>
</table>

**Figure C.11:** Daily *Customized* table with absolute values for different strata by ARS regions, between 2013-06-25 and 2013-07-21. The strata selected are the ones whose plots are previously displayed.