‘







| Data de Recepção |  |
| --- | --- |
| Responsável |  |
| Avaliação |  |
| Observações |  |





# Summary

<<The summary aims to succinctly describe the work carried out. It should contain a short introduction, followed by a brief description of the work carried out and ending with a summary indication of its final status. It should not exceed 400 words.>>

Application Area: <<Work Area Identification. For example: Design and architecture of Database Systems.>>

Keywords: <<Set of keywords that will allow you to reference domains of knowledge, technologies, strategies, etc., directly or indirectly referred to in the report. For example: Relational Databases, Index Management, JAVA, Communication Protocols.>>

# Index

[**Summary 1**](#_heading=)

[**Index 2**](#_heading=)

[**Figure indexes 3**](#_heading=)

[**Tables indexes 4**](#_heading=)

[**1. Definition of the system 1**](#_heading=h.gjdgxs)

[1.1 Application context 1](#_heading=h.29bi3ty193pd)

[1.2 Motivation and work objectives 2](#_heading=h.g1eq6n35cm44)

[1.3 Feasibility Analysis of the Process 3](#_heading=h.l8v3aihfmkuj)

[1.4 Resources and work team 5](#_heading=h.jjqhax64npr0)

[1.5 Project execution plan 6](#_heading=h.s51koey5dne8)

[**2. Requirements Gathering and Analysis 7**](#_heading=h.mp6ynpwhcxn3)

[2.1 Requirements Gathering and Analysis Method 7](#_heading=h.4rjxwiqgub5h)

[2.2 Organization of Requirements 8](#_heading=h.kmcndx6gzlnx)

[2.2.1 Description Requirements 8](#_heading=h.wsi5kzlg6uue)

[2.2.2 Manipulation Requirements 10](#_heading=h.stwvu9enxg98)

[2.2.3 Control Requirements 11](#_heading=h.2uw53h8uk35j)

[2.3 Analysis and General Validation of Requirements 11](#_heading=h.t2um8738788n)

[**3. Conceptual Modeling 13**](#_heading=h.oowrttnrxu9)

[3.1 Presentation of the Modelling Approach Carried Out 13](#_heading=h.z3bmc6bw5evn)

[3.2 Identification and Characterization of Entities 13](#_heading=h.cou69xfoac40)

[3.3 Identification and Characterization of Relationships 15](#_heading=h.dlst8cykwtb)

[3.4 Identification and Characterization of the Attributes of Entities and Relationships. 17](#_heading=h.de28zmasc9vs)

[3.5 Presentation and Explanation of the ER Diagram 20](#_heading=h.wunu9pc6v9yi)

[**4. Logical Modeling 23**](#_heading=h.pq8j97vog3ub)

[4.1 Construction and Validation of the Logical Data Model 23](#_heading=h.oa61kswlex10)

[4.2 Presentation and Explanation of the Logical Model Produced 24](#_heading=h.1dhkbzvocnzp)

[4.3 Data Normalization 26](#_heading=h.gkd5m0tnlf8i)

[4.4 Model Validation with User Interrogations 26](#_heading=h.prt97tci1pow)

[**5. Conclusions and Future Work 29**](#_heading=h.q9d9owgr9nno)

[**6. Bibliography 32**](#_heading=h.y4aqoos0u4t2)

# Figure indexes

[Figure 1.1: Gantt-schedule 6](#_heading=h.uyt73lw85gxu)

[Figure 3.1: ER-diagram 22](#_heading=h.4otrlc3l111g)

[Figure 4.1: Logical model 26](#_heading=h.vbo9ixn54nys)

# Tables indexes

Table 2.1: Description requirements 9

Table 2.2: Manipulation requirements 11

Table 2.3: Control requirements 11

Table 3.1: Identification and characterization of entities 15

Table 3.2: Identification and characterization of relationships 15

Table 3.3: Identification and characterization of the attributes of Employee 18

Table 3.4: Identification and characterization of the attributes of Department 19

Table 3.5: Identification and characterization of the attributes of Case 19

Table 3.6: Identification and characterization of the attributes of Suspect 20

Table 3.7: Identification and characterization of the attributes of Report 21

# Definition of the system

## 1.1 Application context

Based in Braga, in the North of Portugal, Erasmus Digital Solutions is a company specializing in the utilization of databases to store and sort data. In the same city, the police station of the city has in recent years suffered from an increased criminal activity. More specifically, in the year of 2023, there were a total of 440 burglaries and 120 robberies, which, to put in context, corresponds to a 0.28% chance of becoming a victim of one of these crimes. This is an increase of 8.5% compared to the previous year.

As of today, the Braga Police Station is storing all of their information manually, creating a lot of paperwork for their employees, along with major difficulties when navigating in the documents. As un-organized information about crimes not only complicates the work in terms of finding the correct documentation, but also leads to the loss of some important details of investigations. There have also been cases where some papers have gone missing, which is a serious security issue for the citizens of Braga. The Police chief believes that the large number of unsolved crimes correlates with the way of storing the data, since it is difficult to recognize links between different cases. Therefore, the Police department has explored a number of different options to facilitate the data storage, but without any luck. Until one day when they stumbled upon the company “Erasmus Digital Solutions” and reached out for help. Hence, Erasmus Digital Solutions was hired to help the Police station of Braga to store and utilize information better.

## 1.2 Motivation and work objectives

To solve the ongoing issue with file storage and allocation, the police station in Braga needs to implement a database. This digitalization could help them to better achieve their goals with lowering the crime rates as mentioned earlier, and overall improving and structuring the work at the station. The goals of the database is to address the problems mentioned in the context, and therefore the database aims to achieve the following:

* Improve the ability to collect and store information about crimes.
* Coordinate the gathered material in a way that makes the navigation process in old data more accessible.
* Allow for analysis of data collected in past cases.
* Create a search path for specific cases, and filter search results based on different criterias such as; dates, locations, crime types, staff involved etc.
* Have a specific structure of access to information among the colleagues at the police station, all members of the database should not access all data.
* Improve the data security, making it more difficult to lose important data connected to past crimes.
* Improve the overall quality of work.

## 1.3 Feasibility Analysis of the Process

At the request of Police Chief Executive Officer Mr. Belo, EDS performed a project feasibility analysis to identify time and resource savings before implementing a database management system (DBMS) at the station. This analysis aimed to ensure the project's scope and direction aligned with the department's needs. Through collaboration and consistent communication, the engineers concluded that the DBMS would not only generate cost savings up to 400 000€ a year for the police station (and thereby freeing up funds for other budgetary areas) but also ease workflows for officers. Furthermore, the implementation is expected to improve overall department performance by facilitating access to a centralized database of files.

As earlier mentioned the DBMS aims to improve the overall quality of work and save resources for the police station. Hence, the digitization process must not only fulfill the needs of the station, but it should also be user friendly. Consistently working to develop a highly organized DBMS not only guarantees that it will be useful, but also makes it more user friendly. To serve the specific needs of the Police station, EDS will need to have an open communication with Braga Police Station and be open for feedback to make a useful DBMS that is tailored to the needs of the station. Considering that the scope of the project is multifaceted and contains a lot of elements, such as data about staff members, cases, places, dates, crime type, responsible department of the station, perpetrator (if any) etc, building a DBMS from the ground up will take a lot of resources (which are presented in a later section in this paper).

EDS have access to all knowledge, technical equipment and software needed to implement the database management system. They also have a direct and fast connection with the Police Chief Executive Officer at the police station, paving the way for a seamless cooperation and understanding the customer’s needs. Although this project may seem costly for the Police Station at first glance, EDS believes that the police station will save the exact same amount of money within the first year of the finished implementation, on the savings that the implementation will bring. The project is by all means possible for EDS to conduct, but considering the challenges the project group might face, it might be a good idea to plan for some extra time and resources, in order to also be able to follow up on requests and feedback from the police station.

## 1.4 Resources and work team

To fulfill this project and implement the database at the Braga Police department, some resources will be needed. Firstly, there is a need for human resources among which the database development team - Erasmus Digital Systems - is one of them. This team consists of five members willing to dedicate themselves in the most serious manner to complete this mission. Besides the dedication of each member, they also need to be conducting good team work within the group, really focusing on the communication between themselves.

Another key aspect is the relationship with the Braga Police department. The dialogue between the team at the Police and Erasmus Digital System’s team needs to be of good quality for the database to become perfectly adapted for the department’s needs. Moreover, there is also a need for some physical resources, which include a digital scanner, five computers, one server and a database management system, namely MySQL. The digital scanner is necessary since the present physical files at the police department must be digitalized and inserted in the system. The five computers and the server are to be used within the team implementing the database, one computer for each team member. Lastly MySQL will be used to construct and implement the database. Beyond these resources it will also be necessary to use external support regarding the legal questions. Storing information about specific details will need to be reviewed.

## 1.5 Project execution plan

In order to organize the project plan the development team structured the Gantt-schedule illustrated below. The schedule contains all necessary steps in the initial process of creating and implementing the DBMS at the Braga Police Department, and the team aims to follow the time frames for each activity in the process.

#### Figure 1.1: Gantt-schedule

The project was divided into 6 parts to facilitate the work and give a clearer view of the workload. The parts included are System Definition, Requirements Gathering and Analysis, Conceptual Modeling, Logical Modeling, Conclusion and Future Work and Preparation for Presentation - Part 1. Regarding the first part - System Definition - the time allocated was a bit longer than for the following parts. This was due to a wish of having all team members integrated in the project from the start, spending a great amount of time discussing and sharing ideas within the group. When the application context and motivation and objectives parts were to be finished, the plan was to split the rest of the system definition between the team members to do the work more efficiently.

# 2. Requirements Gathering and Analysis

## 2.1 Requirements Gathering and Analysis Method

In the process of working on the project it was decided to collect requirements, as well as their subsequent organization and validation. Therefore, we had to follow a multi-pronged approach to gather requirements for the Braga Police Station DBMS. Hence, we ensure a comprehensive understanding of station employees needs and possible future vulnerabilities, which are needed to be avoided.

**• Interview station employees:** Meetings were held between the team of of Erasmus Digital Solutions (IT-consultants, PM manager, developers and Police Chief Executive Officer (Mr. Belo) and officers from various department to identify the various operational processes that occur during the work of the police, to form a technical task, establish the project`s deadlines and resources.

**• Analysis of documentation:** For the clarification of the current data structure, existing police reports, case files, and any documentation related to data management practices were reviewed.

**• Observation:** Observation of the processes that take place within the police station (daily tasks and data management activities), with particular attention to the current workflows and data flow (documented in the activity diagram).This helps identify areas for improvement and potential bottlenecks in the system.

**•Surveys:**Understanding client needs is the most important point. We gathered the info from our client through conducted surveys and questionnaires among officers and planned to do “user stories” research based on interactions with the future system, ensuring user-centric design.

The requirements gathering process for the Braga Police Station DBMS employed a multi-method approach, strategically chosen to establish a well-rounded description of the future system. This comprehensive approach ensured a deep understanding of officers' needs and potential vulnerabilities.

## 2.2 Organization of Requirements

Explanation of the process carried out and exposition of the requirements organization adopted. Enumeration and organization of the requirements gathered, according to their categorization (description, manipulation and control). Characterization of the requirements taking into account the following elements: type, number, date, description of the requirement, source of information and analyst.

### 2.2.1 Description Requirements

| **Number** | **Description** | **Area** | **Reviewer** | **Source** |
| --- | --- | --- | --- | --- |
| 1 | Each employee will have their full names, employee ID, national ID, starting date, e-mail, phone number, date of birth, gender and home address stored in the system. | Employees | Erasmus Digital Solutions | Braga police station |
| 2 | If an employee is a field officer, their employee ID will be the same as their official badge number. | Employees | Erasmus Digital Solutions | Braga police station |
| 3 | Each employee is either a field officer, administrator or an investigator.. | Employees | Erasmus Digital Solutions | Braga police station |
| 4 | Each case is defined by their lead officer, date of occurrence, location, type of crime and victim. | Case information | Erasmus Digital Solutions | Braga police station |
| 5 | Each case should have its unique ID which is chosen by some specifics of the case. | Case information | Erasmus Digital Solutions | Braga police station |
| 6 | Cases can be ongoing, on hold or closed | Case information | Erasmus Digital Solutions | Braga police station |
| 7 | Closed cases can either be solved or unsolved. | Case information | Erasmus Digital Solutions | Braga police station |
| 8 | Statue of limitation for cases are 20 years. | Case information | Erasmus Digital Solutions | Braga police station |
| 9 | Prescripted cases can either be ongoing or closed. | Case information | Erasmus Digital Solutions | Braga police station |
| 10 | If a suspect is not a Portuguese citizen, their passport ID will be used instead of the national ID. | Suspects | Erasmus Digital Solutions | Braga police station |
| 11 | Suspects are associated with at least one case. | Suspects | Erasmus Digital Solutions | Braga police station |
| 12 | The system will have three main departments; field officers, administrative workers and investigators. | Department | Erasmus Digital Solutions | Braga police station |
| 13 | Each department will have their own unique ID, name, and list of cases | Department | Erasmus Digital Solutions | Braga police station |
| 14 | Each report has their own Report ID. | Report | Erasmus Digital Solutions | Braga police station |
| 15 | Reports have an author, reporter (person calling in about an issue), a date of the call or report, Citizen ID and phone number of the person reporting into the police station. | Report | Erasmus Digital Solution | Braga police station |
| 17 | For each report information such as reporter information, employee that received the report and report date will be stored. | Report | Erasmus Digital Solution | Braga police station |

Table 2.1: Description Requirements

### 2.2.2 Manipulation Requirements

| **Number** | **Manipulation** | **Area** | **Reviewer** | **Source** |
| --- | --- | --- | --- | --- |
| **1** | It must be possible to view all the officers that worked in a particular case. | Employees/Cases | Erasmus Digital Solutions | Braga police station |
| **2** | It must be possible to view all the cases a police officer has worked on | Employees / Cases | Erasmus Digital Solutions | Braga police station |
| **3** | Ongoing, on hold and closed cases should be able to be viewed seperately. | Cases | Erasmus Digital Solutions | Braga police station |
| **4** | It must be possible to add or remove police officers in the system | Employees | Erasmus Digital Solutions | Braga police station |
| **5** | It must be possible to add or remove suspects in the system | Employees | Erasmus Digital Solutions | Braga police station |
| **6** | It must be possible to change suspects between cases. | Suspects / Cases | Erasmus Digital Solutions | Braga police station |
| **7** | It must be possible to change police officers between cases. | Employees / Cases | Erasmus Digital Solutions | Braga police station |
| **8** | It must be possible to view all the officers of a department. | Department | Erasmus Digital Solutions | Braga police station |
| **9** | It must be possible to change the lead officer of a case. | Cases | Erasmus Digital Solutions | Braga police station |
| **10** | It must be possible to view all the suspects of a case. | Suspects | Erasmus Digital Solutions | Braga police station |
| **11** | All ongoing cases should be able to be viewed chronologically | Cases | Erasmus Digital Solutions | Braga police station |
| **12** | All suspects should be able to viewed alphabetically | Suspects | Erasmus Digital Solutions | Braga police station |
| **13** | It should be possible to view the total number of cases a suspect has been associated with. | Suspects | Erasmus Digital Solutions | Braga police station |
| **14** | It should be possible to list all the cases that happened in a location. | Cases | Erasmus Digital Solutions | Braga police station |
| **15** | It should be possible to view all information about a report. | Reports | Erasmus Digital Solutions | Braga police station |
| **16** | It should be possible to open a case from a report. | Reports | Erasmus Digital Solutions | Braga police station |
| **17** | It should not be possible to open a case without a report. | Reports | Erasmus Digital Solutions | Braga police station |

Table 2.2. – Manipulation requirements

### 2.2.3 Control Requirements

| **Number** | **Control** | **Area** | **Reviewer** | **Source** |
| --- | --- | --- | --- | --- |
| **1** | Only the administrators should be able to add or remove officers to the station. | Employees | Erasmus Digital Solutions | Braga police station |
| **2** | Only the head of the station should be able to assign cases to officers. | Employees | Erasmus Digital Solutions | Braga police station |
| **4** | The database system should be accessible at all times. | Time frame | Erasmus Digital Solutions | Braga police station |
| **5** | The system should only be accessible from the Braga Police Station. | Location | Erasmus Digital Solutions | Braga police station |
| **6** | Police officers should only have access to the cases they are associated with. | Employees | Erasmus Digital Solutions | Braga police station |
| **7** | Investigators should be able to view all cases. | Employees | Erasmus Digital Solutions | Braga police station |
| **8** | Any employee can add information about a report | Employees | Erasmus Digital Solutions | Braga police station |
| **9** | Any employee can open a case. | Employees | Erasmus Digital Solutions | Braga police station |

Table 2.3: Control requirements

## 2.3 Analysis and General Validation of Requirements

After planning and defining the requirements for the DBMS, the requirements also need to be validated. By organizing meetings between the Erasmus Digital Solutions and the Braga Police Stations the requirements could be reviewed by both parties to ensure that they were properly defined. After consolidating with the chief officer of the station, the accessibility of the system was discussed. The chief officer at Braga Police Station wished to not have the system available between midnight, 00:00, until the first working hours, 05:00. Therefore, the initial thought of having the system available at all times was dismissed.

# 3. Conceptual Modeling

## 3.1 Presentation of the Modelling Approach Carried Out

With requirements delineated, we transition to the first level of modeling by gathering pertinent data. This step involves collecting information about the entities, attributes, and relationships that constitute the system. By meticulously documenting these data elements, we mitigate the risk of overlooking crucial aspects and ensure alignment with the system's objectives.

To visually represent the gathered data, we employ conceptual modeling diagrams. These diagrams serve as blueprints, encapsulating the types of entities required by the system, along with their respective attributes and relationships. Utilizing standardized notation, such as Entity-Relationship Diagrams (ERDs), ensures clarity and consistency in communication. Next, we proceed to develop conceptual schemes for each identified usage view. Usage views encapsulate distinct perspectives or functionalities of the system, guiding the organization and structuring of data. Each conceptual scheme delineates the entities, attributes, and relationships pertinent to a specific view, facilitating a targeted and comprehensive understanding of the data requirements.Upon completion of individual conceptual schemes, we undertake the task of reconciling them into a unified global conceptual scheme. This reconciliation process involves harmonizing overlapping entities, attributes, and relationships across different usage views, ensuring coherence and consistency in the overall conceptual model.

Throughout the conceptual modeling process, we employ industry-standard notation and specialized tools to streamline and enhance our modeling efforts. The Entity-Relationship Model (ERM) serves as the primary notation framework, providing a standardized vocabulary for representing entities, attributes, and relationships. Additionally, we leverage sophisticated modeling tools to facilitate diagram creation, optimizing efficiency and accuracy in the modeling process.

## 3.2 Identification and Characterization of Entities

| **Designation** | **Description** | **Synonyms** | **Occurrences** | **Attributes** |
| --- | --- | --- | --- | --- |
| Employee | This entity represents any working personnel of the police station. | Employee | Each employee is registered in the database by the administrators. | Each employee has their own employee ID. They are identified by their name, citizen ID, starting date, address, department ID, gender, date of birth, phone number and their rank. |
| Report | This entity represents the information that was received in the Braga police station as a starting point to open the case. | Report | Each report is registered to the system by a police officer. | Each report has its own report ID and the Id of the Author. It is identified by the name of the reporter, their ID and a phone number as well as the date of the reporting. |
| Suspect | Entity that represents the suspect connected to each case ID through the relation “committed”. The entity contains information about the suspect such as the gender, citizen ID and name. | Suspect | Each suspect is registered to the system by an officer. | Each suspect is identified through gender, citizen ID, date of birth, name, address and phone number. |
| Department | Entity that defines the field of work of each employee. | Agency | Departments are registered to the database by administrators. | Each department is associated with a unique department ID, a department name, e-mail and phone numberl. |
| Cases | This entity contains the important information about the case. | Case | Each can be added by administrator or officers. | Each case is associated with a unique case ID, date of occurrence, type of crime and location. |

Table 3.1 Identification and characterization of entities

## 3.3 Identification and Characterization of Relationships

| **Entity** | **Multiplicity** | **Relationship** | **Multiplicity** | **Entity** |
| --- | --- | --- | --- | --- |
| Employee | N | works\_at | 1 | Department |
| Employee | N | investigates | M | Case |
| Employee | 1 | reports | N | Report |
| Suspect | N | committed | M | Cases |
| Report | 1 | case\_opened | 1 | Case |

Table 3.2: Identification and characterization of relationships

**Employee - Department**

* Relationship: Each employee is connected to a department.
* Description: Between the two entities employee and department there is a singular relationship “works\_at” that connects them.
* Cardinality: Employees N - Department 1. In the Braga Police station the work team is divided into different departments. Hence each employee has its dedicated department, but each department can have multiple employees.¨
* Attributes: The relationship does not have attributes.

**Employee - Case**

* Relationship: Employee can be connected to a case.
* Description: Between the two entities Employee and Case there is a singular relationship “investigates” that connects them.
* Cardinality: Employees N - Cases M. Each employee can be connected to one or multiple cases at the same time, and each case can be connected to one or multiple employees.
* Attributes: The relationship has one attribute called “date” which defines when the case is assigned to the employee.

**Employee - Report**

* Relationship: Employees can be connected to a report.
* Description: Between the two entities employee and report there is a singular relationship “reported to” that connects them.
* Cardinality: Employees 1 - Report N. Each employee can be connected to one or multiple reports at the same time, and each report can be connected to only one employee.
* Attributes: The relationship does not have any attributes.

**Suspect - Case**

* Relationship: Each suspect is connected to a case.
* Description: Between the two entities Case and Suspect there is a relationship “Committed” that connects them.
* Cardinality: Suspects N - Cases M. In the Braga Police station every suspect is associated with one or more cases and in every case can be one or more suspects.
* Attributes: The relationship does not have any attributes.

**Report - Case**

* Relationship: Each report is connected to a case.
* Description: Between the two entities Report and Case there is a relationship “case\_opened” that connects them.
* Cardinality: Report 1 - Cases 1. In the Braga Police station every received call (report) can open a case.
* Attributes: The relationship does not have any attributes.

## 3.4 Identification and Characterization of the Attributes of Entities and Relationships.

| **Entity** | **Attribute** | | **Description** | **Data type** | **Null** | **Composite** | **Multiple valuation** | **Derived** | **Primary key** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Employee | Employee ID | | Unique number | VARCHAR(30) | No | No | No | No | Yes |
| Department ID | | ID of the employee’’s department | VARCHAR(30) | No | No | No | No | No |
| Date of Birth | | Date of birth | DATE | No | No | No | No | No |
| Gender | | Gender | VARCHAR(30) | No | No | No | No | No |
| Address | Street name | Address | VARCHAR(50) | No | No | No | No | No |
| Street number | VARCHAR(50) | No | No | No | No | No |
| Name | | Name | VARCHAR(100) | No | No | No | No | No |
| Citizen ID | | Unique number | VARCHAR(10) | No | No | No | No | No |
| Starting date | | Starting date | DATE | No | No | No | No | No |
| Phone number | | Phone number | VARCHAR(12) | No | No | No | No | No |
| E-Mail | | E-Mail | VARCHAR(100) | No | No | No | No | No |

Table 3.3 Identification and characterization of the attributes of Employee

| **Entity** | **Attribute** | **Description** | **Data type** | **Null** | **Composite** | **Multiple valuation** | **Derived** | **Primary key** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Department | Department ID | Unique number | VARCHAR(30) | No | No | No | No | Yes |
| Department name | Name | VARCHAR(100) | No | No | No | No | No |
| E-mail | E-mail | VARCHAR(100) | No | No | No | No | No |
| Phone number | Phone number | VARCHAR(12) | No | No | No | No | No |

Table 3.4: Identification and characterization of the attributes of Department

| **Entity** | **Attribute** | **Description** | **Data type** | **Null** | **Composite** | **Multiple valuation** | **Derived** | **Primary key** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Case | Case ID | Based on specifics of the crime | VARCHAR(30) | No | No | No | No | Yes |
| Date of occurence | Date of crime | DATE | No | No | No | No | No |
| Type of crime | Unique identifier | VARCHAR(100) | No | No | No | No | No |
| Description | Short description of the case | VARCHAR(1000) | Yes | No | No | No | No |
| Location | Place of crime | VARCHAR(100) | No | No | No | No | No |
| Status |  | VARCHAR(10) | No | No | No | No | No |

Table 3.5 Identification and characterization of the attributes of Case

| **Entity** | **Attribute** | | **Description** | **Data type** | **Null** | **Composite** | **Multiple valuation** | **Derived** | **Primary key** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Suspect | Citizen ID | | Unique number | VARCHAR(30) | No | No | No | No | Yes |
| Gender | | Gender | VARCHAR(30) | No | No | No | No | No |
| Date of birth | | Date of birth of the suspect | DATE | No | No | No | No | No |
| Name | | Full name | VARCHAR(100) | No | No | No | No | No |
| Phone number | | Phone number | VARCHAR(30) | No | No | No | No | No |
| Address | Street name | Address | VARCHAR(50) | No | No | No | No | No |
| Street number | VARCHAR(50) | No | No | No | No | No |

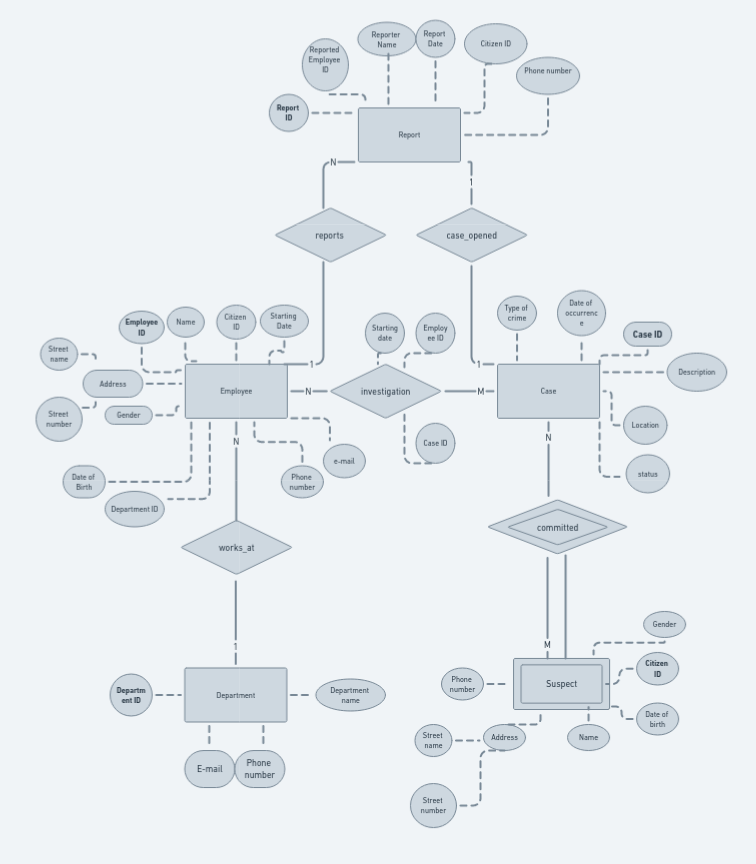
Table 3.6: Identification and characterization of the attributes of Suspect

| **Entity** | **Attribute** | **Description** | **Data type** | **Null** | **Composite** | **Multiple valuation** | **Derived** | **Primary key** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Report | Report ID | Unique number | VARCHAR(30) | No | No | No | No | Yes |
| Repórter | The full name of the person who made a “call” | VARCHAR(100) | No | No | No | No | No |
| Citizen Id | Unique number | VARCHAR(100) | No | No | No | No | No |
| Report date | Date of receiving a call | DATE | No | No | No | No | No |
| Author | An employee unique ID | VARCHAR(30) | No | No | No | No | No |
| Phone number | Phone number | VARCHAR(30) | No | No | No | No | No |

Table 3.7: Identification and characterization of the attributes of Report

## 3.5 Presentation and Explanation of the ER Diagram

The ER diagram below shows the entities and their relationships with other entities and their attributes. Each entity has its own defined properties and ID numbers used to identify it.



#### Figure 3.1: ER-diagram

Firstly, the key attributes are written with bold text. Our main entities are five: "Case", "Employee", "Department", "Suspect" and "Report", which you can see in the form of rectangles in the diagram. To explain the relationship between these, let's first assume that we have a "Case". There should also be at least one "Employee" to investigate this case. To keep track of the date of each case, the relationship “investigation” between Case and Employee has an attribute “start date” so that it is customized based on the date this employee takes over the case. This employee works in a specific department and should investigate the case. The case can be opened when the Braga police station worker receives a signal about the crime and documents this report for the future need of the investigated case. Not always a report leads to the opening of the case.

Additionally, as you can see from the diagram, the suspect is a weak entity, since it does not exist without a case. It should be underlined that there will not be a suspect unless there is a case, represented by the double lines between the suspect and the relationship “committed” is double lined, indicating total participation

We also have a relationship between the department and the employee since every employee is associated with a department. With this modeling, it is possible to understand the diagram we have built better and the relationships we have established. Thus, we can move on to the logic modeling stage.

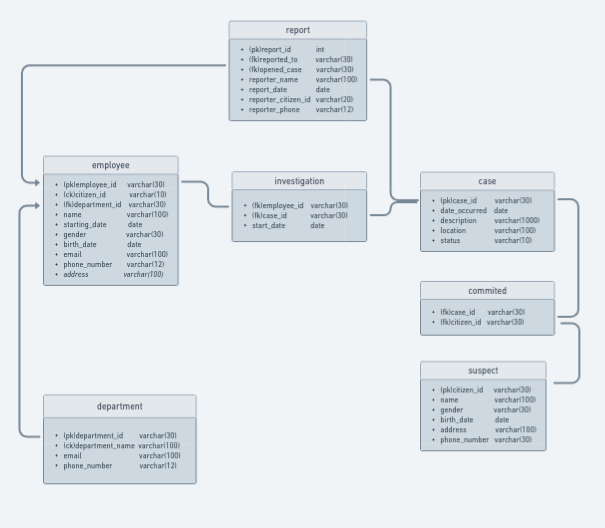
# 4. Logical Modeling

## 4.1 Construction and Validation of the Logical Data Model

When the conceptual model is finalized, a logical data model can be created based on that. The logical model is a more simplified version of the conceptual one since it mainly depends on the entities and their relationships to each other. All of this has already been defined in the conceptual model which makes the logical modeling easily constructed. The logical model consists of a number of tables that each have their own attributes called keys. The keys could either be primary or foreign, where a primary key is a unique identifier for the table while the foreign keys are fields that refer to a primary key in another table. To construct these attributes, one can follow the following set of regulations:

* Each table has one primary key attribute, and could also consist of one or many foreign key attributes.
* Each entity has its own table listing its attributes. Moreover, it could also consist of foreign keys which are attributes originating from other entities. This is to enable the establishment of relationships between different entities. So for each entity (Report, Employee, Case, Department and Suspect) there will be a table in the logical modeling.
* In binary 1:N relationships, a single record in one table can be linked to many records in another table. We handle this by adding a foreign key to the "many" table. This foreign key references the primary key of the "one" table, creating the connection. This is the case for the following relationships:
  + Employee (N) works at one Department (1)
  + Employee (1) creates one or multiple reports (N)
* In binary 1:1 relationships, the relationships can be modeled using a single table. This is because each entity has a unique connection to the other. However, in practice, it's often recommended to normalize the data and separate them into distinct tables. This avoids redundancy and simplifies maintenance. This is the case of for the following relationship:
  + One Report (1) is linked to (1) case.
* Many-to-many (N:M) relationships involve many entities of one type connecting to many entities of another type. These cannot be directly represented in a single table with foreign keys.To handle N:M relationships one can create a Junction Table: a new table with the purpose of linking the two related tables together. This junction table will have two foreign keys: One foreign key referencing the primary key of the first entity type (N table) and another foreign key referencing the primary key of the second entity type (M table). This junction table essentially creates a "many-to-many" association between the original tables.
* When one attribute is composite, like for example a “contacts” attribute that has both phone number and email address to a unique identifier, then this attribute needs to have its own table. In this model the Employee Address and the Suspects Address is of that kind. To solve this conflict, while keeping in mind the normalization steps, instead of using a composite attribute we use a single address attribute that has both informations.

## 4.2 Presentation and Explanation of the Logical Model Produced



#### Figure 4.1: Logical model

The logical model we built consists of nine tables and 5 of them are our entities. In this model; ***pk*** stands for primary key, ***ck*** stands for candidate key and ***fk*** stands for foreign key. Our attributes can be in varchar, int or date form. Birthdays, occurring date, report date and start date are in date form. Report id is in int form and all the other attributes are in varchar form.

1. **Case:** When a case occurs, a special case id is created for this case. For this reason, this entity’s primary key is case id. It doesn’t have any foreign keys. Its attributes are occuring date, description and location.
2. **Employee:** This entity’s primary key is employee id. It has one foreign key which is department id because of the relationship with the department entity. It has one ***ck*** which is the citizen id and it corresponds to the citizen numbers of the people working at the police station. Name, starting date, gender, birthday, email, phone number and address are the other attributes that this entity has.
3. **Department:** This entity has one primary key and it is the department id. Its ***ck*** is department name because there are different departments and they all have their own unique names. Email and phone number are the other attributes that the Department entity has.
4. **Suspect:** This entity’s primary key is citizen id which corresponds to the citizen id numbers of the possible culprits. Name, gender, birthday, address and phone number of suspects are the other attributes of this entity.
5. **Report:** Since every report has their own id number, this entity’s primary key is report id. It has two foreign keys: The first one is reported to, which is coming from the employee entity corresponding to the employee to whom it is reported. The second one is the opened case, which is coming from the case entity. Reporter’s name, citizen id, phone number and the report date are the other attributes of this entity.
6. **Investigation:** This table does not have a primary key. It has two foreign keys which are employee id (coming from employee entity) and case id (coming from case entity) because it represents the relation between case and employee. It also has one attribute and that is the start date.
7. **Committed:** This table does not have a primary key. It has two foreign keys: the first one is case id which is coming from the case entity and the second one is citizen id which is coming from the suspect entity. Because this table represents the relation between case and suspect. It does not have any other attributes.
8. **Address:** This table has 2 entities: street number and street name. It does not have any primary key or foreign key. We have two of this table and one of them is the employee's and the other one is the suspect's address.

## 4.3 Data Normalization

It is possible to conclude that the database adheres to the principles of the First- Second- and Third Normal Form.. The First Normal Form highlights that all of the data attributes must be atomic and that there should be no repeating groups in the tables, which is ensured through examining the data types and domains of each attribute. The Second Normal Form highlights that all the non-key attributes (attributes not part of the primary key) must depend on the entire primary key, not just a part of it. This eliminates redundancy and ensures data integrity.

Also the Second Normal Form presupposes that the requirements of the First Normal Form are met as well. In all tables the primary key is the sole determinant of all non-key attributes. This means every non-key attribute relies on the entire primary key, not just a part of it. The Third Normal Form builds upon the First- and the Second Normal Form together with the focus on no transitive dependencies. It eliminates situations where a non-key attribute depends on another non-key attribute, which in turn depends on the entire primary key. This indirect reliance creates a “transitive dependency” and can lead to data inconsistencies. Since the model does not contain any non-prime attributes that depend on other non-prime attributes the Third Normal Form is respected as well.

## 4.4 Model Validation with User Interrogations

Presentation of 4 expressions in Relational Algebra that represent queries previously stated in the set of manipulation requirements established above.

1. **It must be possible to view all officers that worked in a particular case.**

The list of officers working in each case is viewable through the “investigating” relations, which has employees as a foreign key as attributes.

π*employee*\_id, *employee*\_name(σ *investigation . case\_id = ‘selected\_case\_id’* (employee ⋈investigation))

1. **It must be possible to view all the cases a police officer has worked on**

All cases have their unique Case ID as primary keys that have relations to one or many Employee ID’s, through “investigation” .

π*case\_id, description*(σ*investigation.employee\_id = ‘selected\_officer\_id’*(investigation⋈case))

1. **It must be possible to view all officers of a specified department.**

The list of officers working in a department is viewable through the “works at” relations, which has employees as a foreign key as attributes.

π*employee*\_id, *employee*\_name(σ*employee.department\_id = department.department\_id*(employee x department))

1. **All ongoing cases should be able to be viewed chronologically.**

The cases that are still being investigated should be able to be viewed.

π*case*\_id, *description*(σ*status* = "ongoing" *case)*

# 5. Physical Implementation

## 5.1. Presenting and Explaining the Implemented Database

>> Implementation of the logical schema previously developed in MySQL. Presentation of the implementation process carried out, exposing and justifying how you created each table of the logical scheme. Presentation of the physical scheme produced.

## 5.2. Creating database users

>> Presentation and characterization of the users (one or two examples) of the database. Creation of the usage profiles of the users in the database system, defining, for each of them, their permissions.

## 5.3. Populating the database

>> Presentation and description of the various settlement actions carried out. The populating of the database should be carried out in two ways: 1) inserting the data into the database directly with INSERT statements; 2) use a small program (developed in a language of your choice) that accesses the implemented database and populates one or two tables with "half a dozen" records. Briefly explain the design and implementation of the program used.

## 5.4. Calculating database storage space (initial and presenting an annual growth rate)

>> Calculate the initial size of the database (1 record per table) and then indicate its annual growth rate, assuming values that you think are appropriate for your application case. You should pay attention to the data type of the various attributes of the tables and the space that MySQL reserves for each of them. I suggest the use of a spreadsheet to do this topic.

## 5.5. Defining and characterizing usage views in SQL

>> Develop a set of views (VIEW), which you consider necessary or useful for improving database querying performance and security.

## 5.6. Translating user queries to SQL

>> Presentation of "half a dozen" SQL queries, which represent the questions previously presented when validating the constructed logical schema.

## 5.7. Indexing Database system

>> Presentation of possible reasons for defining a set of indexes on one or more tables in the database. Creation and explanation of the indicated indexes.

## 5.8 Implementing stored procedures, functions and triggers

>> Development of a procedure, a function and a trigger that work with some data objects created in the database. Present the reason for its creation and explain how it works. One of the examples created should include transactions.

# 6. Conclusions and Future Work

# 7. Bibliography

* Connolly, T., Begg, C., Database Systems: A Practical Approachto Design, Implementation, and Management, AddisonWesley, Global Edition, 26 Sep 2014
* The lectures provided in the given course