



universität
wien

PRAKTIKUM 2

INFORMATION SECURITY POLICY REPOSITORY AND INFORMATION SYSTEM

Verfasser

Kleinl Lukas

Betreuer

Univ.-Prof. Dipl.-Ing. Dr. Dr. Gerald Quirchmayr

Wien, 30.07.2024

Fachrichtung:

Informatik

Betreuerin / Betreuer:

Univ.-Prof. Dipl.-Ing. Dr. Dr. Gerald Quirchmayr

Contents

1	Introduction.....	5
1.1	Scope of this work.....	5
2	State of the Art	6
2.1	Legal background.....	6
2.1.1	GDPR	6
2.1.2	Relevant Institutions	7
2.2	Information Security Policy	8
2.2.1	Information Security Policy Lifecycle (Tukiyeze, 2010)	8
2.2.2	Analyzing and enforcing policies	9
2.2.3	Legal Ontologies	9
2.2.4	Tools	9
3	Methodology	10
4	Use cases and Requirements	11
4.1.1	Scenario.....	11
4.1.2	Cases.....	11
5	Architecture.....	11
6	Implementation	12
6.1	Installation and usage	12
6.2	MongoDB.....	14
6.2.1	Stored information about company controls	14
6.2.2	Atlas Vector search	14
6.3	Okta – Authentication	15
6.4	Neo4j & GDPRtEXT.....	16
6.5	Services	16
6.5.1	Authorization and routing	16
6.5.2	LLM information – Generative and MongoDB Atlas Vector Search.....	17
6.5.3	Company controls manager	18
6.5.4	GDPR Exploration with Graph	20

7	Conclusion and future work	20
8	Bibliography	21

Figure 1 - Information Security Policy Lifecycle from Tukiyeze (2010).....	8
Figure 2 - Spiralmodel (Boehm, 1988)	10
Figure 3 - Base architecture of the solution	12
Figure 4- Setup Vector Search Index.....	15
Figure 5 - Configuration for Vector search	15
Figure 6 - Finished setup of Vector search	15
Figure 7 - Setup of Vectorsearch in Python	15
Figure 8 - Sample user Okta.....	16
Figure 9 - Sample roles Okta.....	16
Figure 10 - Login page provided by Okta.....	17
Figure 11 - Response if the given role does not support the functionality	17
Figure 12 - Generative AI response - Example 1.....	18
Figure 13 - Generative AI response - Example 2.....	18
Figure 14- Vector search response - Example 1	18
Figure 15 - Vector search response - Example 2.....	18
Figure 16 - Home view to the company controls management	19
Figure 17- Create new company control	19
Figure 18 - View with options for the company control.....	19
Figure 19 - Sample query to extract information from Neo4j.....	20
Figure 20 - Graph based GDPR exploration	20

1 Introduction

In today's rapidly evolving digital landscape, ensuring robust information security practices has become a necessity for organizations of all sizes and industries. However the huge volume and complexity of different types of information security policies is a problem the organizations are faced with. One of these challenges is processing of personal data. As it is stated in the charter of Fundamental Rights of the European Union §8(1): „Everyone has the right to the protection of personal data concerning him or her.“ This is especially important since the European Union released the GDPR (General Data Processing Regulation) in 2016, which got active in 2018. With this regulation the European Union wants to improve the individual power over their own data and setup regulations and standards for organizations. The content of the GDPR is divided into 11 chapters, which ranges from the data subject, over the duties of the data processors to imposing administrative fines. Within the regulation, fines for organization which are not complying, are defined. (Article 83 GDPR) Therefore the organizations need to make a huge effort to adhere to these regulations and setup policies to act accordingly. Since companies find it very difficult to keep track of all the policies and take appropriate actions, a solution is needed that allows them to display the relevant passages and suggest necessary next steps.

Based on the previously outlined challenges, this paper presents a solution for the storage of policies that facilitates user access to relevant sections of those policies. The proposed system is applicable to various use cases, including auditing, editing, and usage of policies for incident management. Furthermore the proposed solution adheres to the information security policy life cycle. The underlying storage model is based on an object oriented database concept. For searching the policy repository different search approaches are considered and compared.

1.1 Scope of this work

Based on the previously outlined importance of the work, following research questions need to be answered:

- How can relevant GDPR content (policies, guidelines, ...) be represented, organized and analyzed?
- How can ontologies be incorporated for a standard approach → Client search?
- → Cross referencing
- How could a architecture of such a system look like?
- How can such a system be implemented?

Objectives:

1. Evaluate existing legal ontologies (DAPRECO, ODRL, DPV, ...)
2. Develop an application which showcases relevant information in regards to the GDPR
3. Develop an application which enables maintaining policies according to the information security policy lifecycle

4. Create a user-friendly interface that allows data controllers to maintain policies
5. Create a user-friendly interface that allows users to access relevant information according to their needs

Deliverables:

1. Prototype of a system, which showcases relevant information for the given user and gives the possibility to maintain policies according to the information policy lifecycle
2. User documentation and tutorials for navigating and utilizing the features
3. Quality assurance reports outlining testing procedures and results.

Constraints:

1. In scope of this work is only a prototype to showcase the solution

2 State of the Art

2.1 Legal background

2.1.1 GDPR

The General Data Protection Regulation (GDPR) is a comprehensive data protection law enacted by the European Union (EU) in 2018. It aims to safeguard the privacy and personal data of EU citizens by regulating how businesses and organizations handle and process this data. GDPR mandates strict requirements for obtaining consent, providing transparency in data processing practices, and ensuring individuals' rights over their personal information, such as the right to access, rectify, and erase their data. It also imposes significant penalties for non-compliance, including fines of up to €20 million or 4% of global annual revenue, whichever is higher. (GDPR, 2016)

Scope and Definitions

The GDPR applies to any organization that processes personal data of individuals residing in the EU, regardless of where the organization is based. It defines various terms such as "personal data," "data controller," and "data processor." (GDPR, 2016)

Principles of Data Protection

GDPR outlines fundamental principles for processing personal data lawfully, fairly, and transparently. These principles include purpose limitation, data minimization, accuracy, storage limitation, integrity, and confidentiality. (GDPR, 2016)

Rights of Data Subjects

GDPR grants individuals several rights regarding their personal data. These rights include the right to access their data, the right to rectify inaccurate information, the right to erasure (or

"right to be forgotten"), the right to data portability, the right to restrict processing, and the right to object to processing. (GDPR, 2016)

Lawfulness of Data Processing

GDPR specifies lawful bases for processing personal data, such as consent, contract performance, legal obligation, vital interests, public task, and legitimate interests pursued by the data controller or a third party. (GDPR, 2016)

Consent Requirements

GDPR sets strict standards for obtaining valid consent from individuals for processing their personal data. Consent must be freely given, specific, informed, and unambiguous, and individuals have the right to withdraw consent at any time. (GDPR, 2016)

Data Breach Notification

GDPR mandates organizations to notify relevant supervisory authorities and affected individuals of data breaches without undue delay, particularly if the breach poses a risk to individuals' rights and freedoms. (GDPR, 2016)

Data Protection Impact Assessments (DPIAs)

Organizations are required to conduct DPIAs for processing activities that are likely to result in high risks to individuals' rights and freedoms. DPIAs help identify and mitigate risks associated with data processing. (GDPR, 2016)

Data Protection Officers (DPOs)

Some organizations are required to appoint a Data Protection Officer responsible for overseeing GDPR compliance, advising on data protection matters, and serving as a point of contact for supervisory authorities and data subjects. (GDPR, 2016)

Cross-Border Data Transfers

GDPR restricts the transfer of personal data outside the EU to countries or organizations that do not provide an adequate level of data protection unless appropriate safeguards are in place. (GDPR, 2016)

2.1.2 Relevant Institutions

CNIL is the French data protection authority. It is an independent administrative regulatory body responsible for ensuring that the processing of personal data in France complies with data protection laws, including the General Data Protection Regulation (GDPR).

BSI The Federal Office for Information Security is the federal cybersecurity authority and a promoter of secure digitization in Germany.

2.2 Information Security Policy

2.2.1 Information Security Policy Lifecycle (Tukiyeze, 2010)

The information security policy life cycle is a well-planned and continuous process that organizations must follow to ensure that their policies are up-to-date and supportive of their security principles. The life cycle consists of four phases:

1. Planning and risk assessment: This phase involves identifying the organization's security goals, assessing its security infrastructure, and determining the necessary steps for policy development.

2. Development: In this phase, policies are created based on the information gathered during the planning phase. The policies must be reviewed and approved by stakeholders to ensure they are comprehensive and supportive of the organization's security principles.

3 and 4. Implementation and Maintenance: Once the policies are developed, they need to be implemented and maintained through continuous monitoring and review. This phase involves reviewing the security infrastructure, identifying new threats, and incorporating changes into the policies through the maintenance phase.

Repeating the policy development life cycle is necessary for policy changes, as organizations must continuously review and update their policies to ensure they remain effective and supportive of their security principles. By following this lifecycle approach, organizations can ensure that their information security policies are mature, operationally entrenched, and comply with new laws and technological advancements.

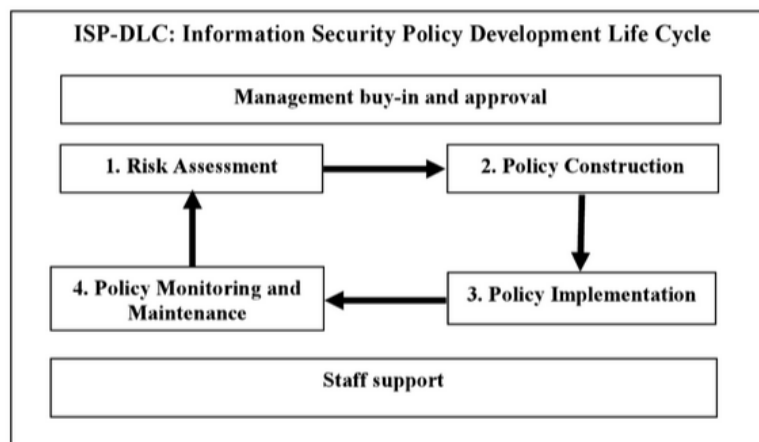


Figure 1 - Information Security Policy Lifecycle from Tukiyeze (2010)

2.2.2 Analyzing and enforcing policies

PrivGuard's core component, PrivAnalyzer, is a static analyzer designed to ensure compliance of analysis programs with privacy policies. Unlike previous methods relying on access control or manual verification, PrivAnalyzer utilizes abstract interpretation, offering provable soundness for some properties without revealing data content. It examines programs and policies, not data, making it suitable for general-purpose programming languages with complex features like control flow and loops. PrivAnalyzer is implemented in Python, making it accessible to analysts without the need to learn new languages. (Wang, 2022)

RuleKeeper a GDPR-aware policy enforcement system for web frameworks, which allows to express data protection policies, ensure transparency by clearly displaying policies to users, and is easy to maintain by separating policy enforcement from application code. Focus

2.2.3 Legal Ontologies

GDPRtEXT (GDPR text extension) is the semantification of GDPR into a linked data resource and ontology. It enables referencing articles and concepts/terms within the GDPR using RDF/RDFS/OWL

PrOnto: Legal Ontology for Modelling GDPR Concepts and Norms (Palmirani, 2018)

Dapreco Knowledge Base is a repository of rules written in LegalRuleML, an XML formalism designed to be a standard for representing the semantic and logical content of legal documents. The rules represent the provisions of the General Data Protection Regulation (GDPR), the new Regulation that is significantly affecting the digital market in the European Union and beyond. The DAPRECO knowledge base builds upon the Privacy Ontology (PrOnto) (Palmirani et al., 2018c), which provides a model for the legal concepts involved in the GDPR, by adding a further layer of constraints in the form of if-then rules, referring either to standard first order logic implications or to deontic statements.

Data Privacy Vocabulary (DPV) enables expressing machine-readable metadata about the use and processing of personal data based on legislative requirements such as the General Data Protection Regulation [[GDPR](#)]. This document describes the DPV specification along with its data model.

The Open Digital Rights Language (ODRL) is a policy expression language that provides a flexible and interoperable information model, vocabulary, and encoding mechanisms for representing statements about the usage of content and services. The ODRL Information Model describes the underlying concepts, entities, and relationships that form the foundational basis for the semantics of the ODRL policies.

2.2.4 Tools

PIA Software

The PIA tool was built on three core principles:

1. **User-friendly interface:** The tool offers a straightforward interface to conduct Privacy Impact Assessments (PIAs). It guides users through each step of the assessment process and provides visualizations to understand risks easily.
2. **Legal and technical knowledge base:** It integrates legal requirements and technical considerations into the assessment process. The tool includes GDPR regulations, PIA guides, and relevant security guidelines from organizations like CNIL to ensure compliance.
3. **Modularity:** The tool is customizable to fit specific needs or business sectors. Users can adapt the tool's contents, create PIA models, and modify the source code under a free license to add features or integrate it into existing organizational tools.

3 Methodology

The spiral model, developed by Barry W. Boehm in the 1980s, is one of the first iterative and incremental process models in software engineering. It serves to minimize risks through continuous testing after all phases. Its defining feature is the visual representation of a spiral shape in a rectangular coordinate system, which becomes larger with each cycle. Risk management, which offers an early evaluation of problems to avoid project hurdles, is also a key point. (Boehm, 1988) The developed solution was built on the principles of Barry Boehm.

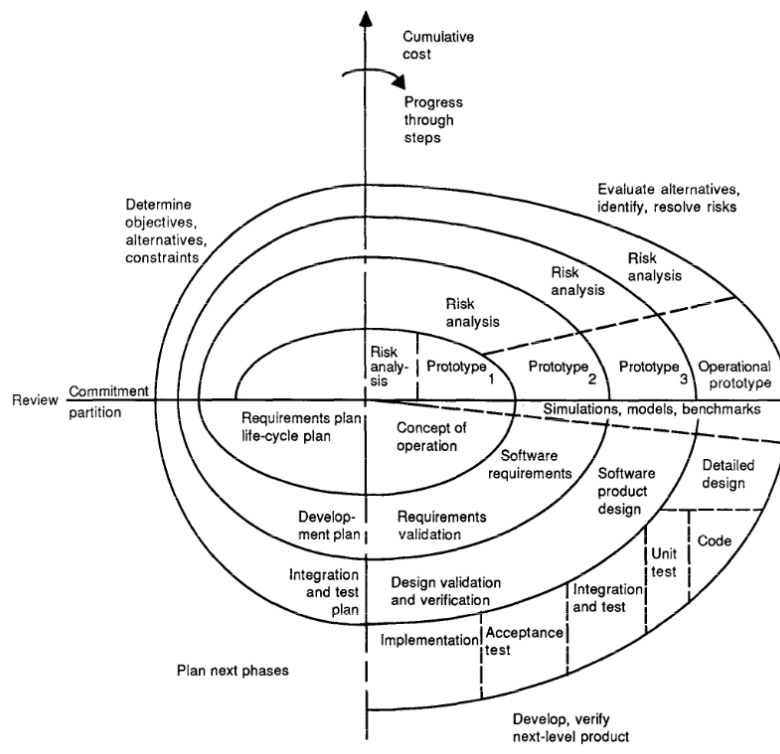


Figure 2 - Spiralmodel (Boehm, 1988)

4 Use cases and Requirements

4.1.1 Scenario

RECPLAST GmbH produces and sells around 400 different plastic products made from recycled materials, for example construction elements such as round and board profiles, fences, flower tubs or waste garbage cans, partly in large series for end customers and partly specifically for individual business customers. The order volume, the frequency of orders and the customers vary: There are a few regular and major customers and numerous individual customers. The company's total annual turnover is around 50 million euros with a profit of around one million euros. (BSI – Recplast)

The company wants to help their employees gather information about relevant information about GDPR relevant data and policies. Therefore an application should be developed which is able to help the employees get information and enables the permitted users to maintain the policies (create, edit, delete)

4.1.2 Cases

Compliance officers and policy creators

Use Case 1: Conducting periodic or continuous audits to assess compliance with internal policies and GDPR.

Use Case 2: Users can create new policies or update existing policies within the application.

Use Case 3: Users can propose revisions to existing policies based on changes in regulations, organizational requirements, or feedback from stakeholders.

Business employees

Use Case 4: Employees interact with the policy editing system to integrate compliance requirements seamlessly into their daily work

Use Case 5: Employees conduct reviews with a focus on ensuring compliance with internal policies and GDPR.

Use Case 6: Employees want to access relevant information for the given situation

Examples for policies:

Consent Management Policy, Data Breach Notification Policy, Data Retention Policy, Access Control Policy

5 Architecture

API

Authorization Endpoint: Define roles for role based access

Information Endpoint: Endpoint for getting relevant information

This endpoint is split in 2: GDPR exploration and recommendation with a large language model.

Policy Maintenance Endpoint: Enables the user to maintain GDPR relevant policies according to the information security lifecycle

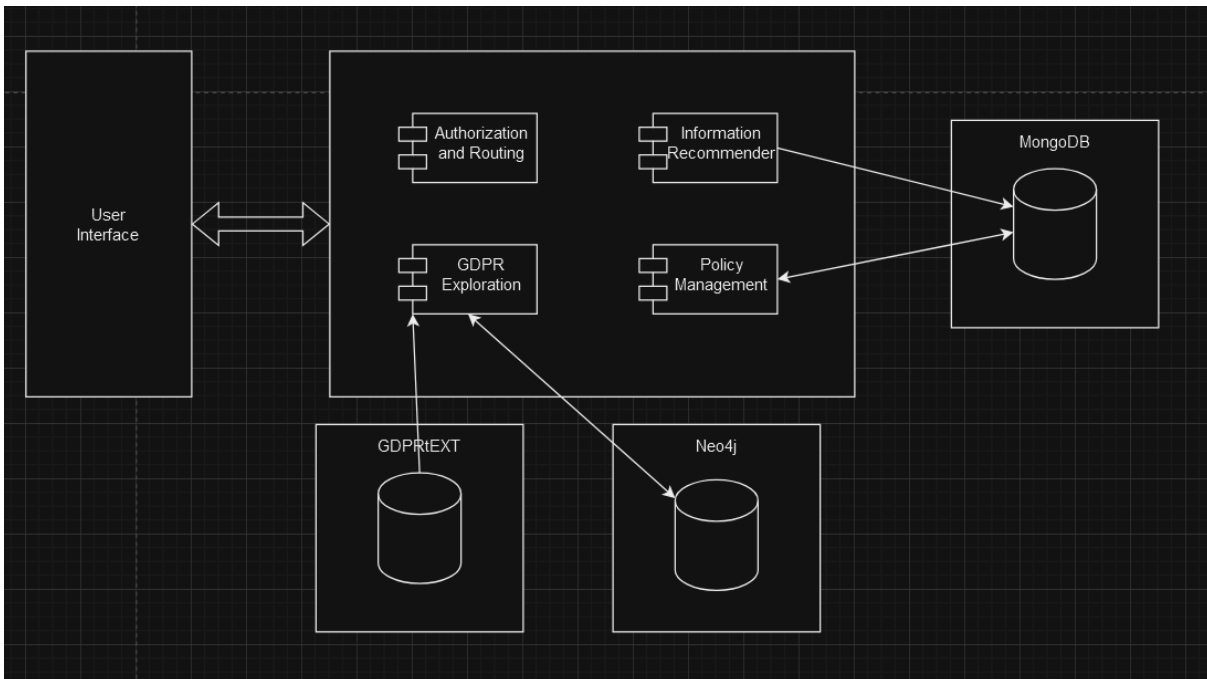


Figure 3 - Base architecture of the solution

6 Implementation

6.1 Installation and usage

The app is build based on 4 Flask packages. In each package the requirements are provided. Install these either locally or in a virtual environment. Furthermore follwing services need to be installed or setup:

- MongoDB with Atlas Vector Search
- Neo4j instance
- Ollama or another LLM (needs to be configured)
- Okta for authorization

For the configuration details an .env needs to be set up with following relevant information:

Recommender Package

AURA_DB_URI = "bolt://localhost:xxxx" - normally 7687

AURA_DB_USERNAME = "neo4j"

AURA_DB_PWD = "anypassword"

Policy Manager package

MongoDB_URI =

mongodb+srv://lukaskleinl:[xxx@cluster.xxxx.mongodb.net](#)?retryWrites=true&w=majority&appName=Cluster - Looks something like this if setup in cloud

If setup locally the correct URI needs to be provided

Compliance Authorization package - information can be found in OKTA

URL for Okta: <https://manage.auth0.com/dashboard/eu/dev-lpidbb8cgvdmuria/>

AUTH0_CLIENT_ID=xxxx

AUTH0_CLIENT_SECRET=xxx

AUTH0_DOMAIN=dev-lpidbb8cgvdmuria.eu.auth0.com

APP_SECRET_KEY=ALongRandomlyGeneratedString

URL_Recommender = <http://127.0.0.1:2001/>

URL_Company_Control_Store = <http://127.0.0.1:2002/>

URL_Chat = <http://127.0.0.1:2003/>

Compliance LLM Recommender package

Here a Open API Key was used for the embedding for the vector search. Another embedding like Ollama can also be used. An Open API Key is not needed in this case. The key can be found in the webpage after login to OpenAI

OPENAI_API_KEY = xxxx

MongoDB_URI =

mongodb+srv://lukaskleinl:[xxx@cluster.xxxx.mongodb.net](#)?retryWrites=true&w=majority&appName=Cluster

Afterwards run follwing commands with the given ports:

Authorization and routing - flask run --host=0.0.0.0 --port=2000

GDPR exploration - flask run --host=0.0.0.0 --port=2001

Company Controls Manager - flask run --host=0.0.0.0 --port=2002

LLM recommender - flask run --host=0.0.0.0 --port=2003

Steps

1. Clone the repository:
2. Navigate to the project directory:
3. Install requirements

4. Create .env files
5. Run flask apps with given ports

Usage

Start up all services and open your localhost at 2000. The authorization and routing package running on port 2000 is the starting point for all operations

6.2 MongoDB

6.2.1 Stored information about company controls

```
_id: ObjectId('66a861eebb053c111d09f95f')
title: "test"
description: ""
file: Object
  name: "file"
  file: ObjectId('66a861edbb053c111d09f95d')
  version: 0
old_files: Array (empty)
timestamp: 2024-07-30T05:45:50.694+00:00
```

6.2.2 Atlas Vector search

The atlas vector search enables the analysis of a large text corpus. This is done via a semantic comparison of the created embedding model. With this approach it is easily and quickly possible to analyse huge amounts of data.

To setup the Atlas Vector search the MongoDB cloud was used. Depending on which embedding is used the variable index has to be adjusted. For the similarity MongoDB offers 3 options: Euclidean, Cosine and DotProduct to measure the distance between the nodes. For the embedding 2 different embeddings were compared, namely the OpenAI embedding and the Ollama embedding. The OpenAI embedding offered better results after investigation and was therefore further used.

Create a Vector Search Index

1 Configuration Method

2 JSON Editor

3 Review

Configuration Method

Select how you would like to build and customize your search index. You can also create, edit, and manage search indexes using the [Atlas API](#).

NOTE

At this time, search indexes cannot be created for time series collections.

Atlas Search [Learn more](#)

Visual Editor

Learn about index definitions in a more guided experience.

JSON Editor

Edit the raw index definition with an embedded JSON editor.

Atlas Vector Search [Learn more](#)

JSON Editor

Create a vector search index definition with an embedded JSON editor.

Figure 4- Setup Vector Search Index

Database and Collection

Search for database or collection ...

company_controls

Chat_Search

fs.chunks

fs.files

guideline

policy

Information_Security_Store

Index Name

vector_index

```

1  {
2    "fields": [
3      {
4        "type": "vector",
5        "path": "embedding",
6        "numDimensions": 1536,
7        "similarity": "cosine"
8      }
9    ]
10  }

```

Back

Cancel

Next

Figure 5 - Configuration for Vector search

company_controls.Chat_Search

Indexes Used: 2 of 3.

Name	Index Type	Index Fields	Status	Size	Documents	Actions
vector_index	vectorSearch	embedding	ACTIVE View status details	Primary Node: 3,264MB	Primary Node: 552 (100%) indexed of 552 total	QUERY ...

Figure 6 - Finished setup of Vector search

```

vectorstore = MongoDBAtlasVectorSearch.from_documents(documents=page_loader,
                                                       embedding=OpenAIEmbeddings(disallowed_special=()),
                                                       #OpenAI 1536 indices, ollamaEmbeddings(),(4096)
                                                       collection=new_collection, index_name="vector_index")

```

Figure 7 - Setup of Vectorssearch in Python

6.3 Okta – Authentication

„Auth0 is a flexible, drop-in solution to add authentication and authorization services to your applications. Your team and organization can avoid the cost, time, and risk that come with building your own solution to authenticate and authorize users.“ (Okta Documentation)

The implementation of Oauth2 offers user maintenance and also role based access for the application. The users are routed to the Okta service to login and redirected back to the service after a successful login. In the figures below some sample users and roles of the app are shown.

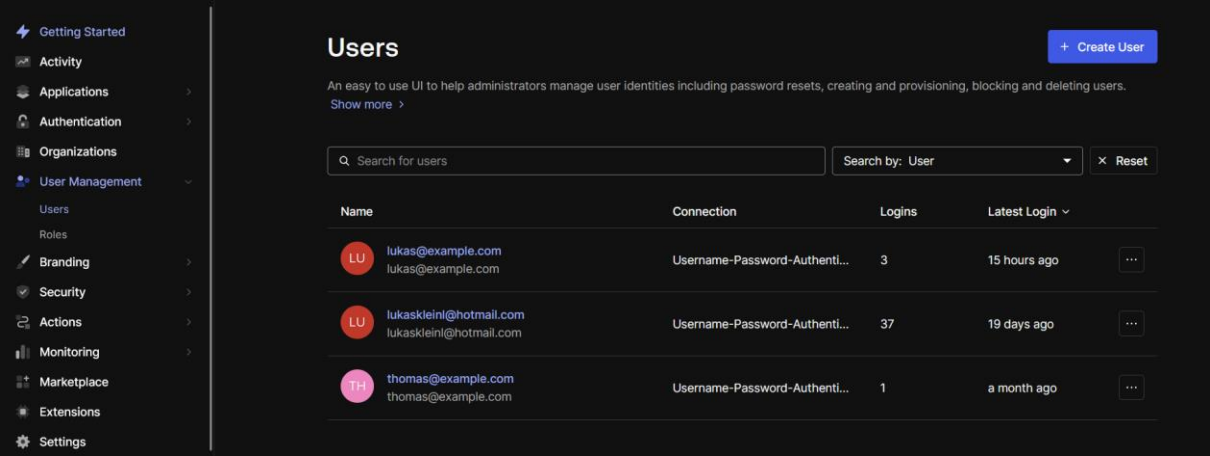


Figure 8 - Sample user Okta

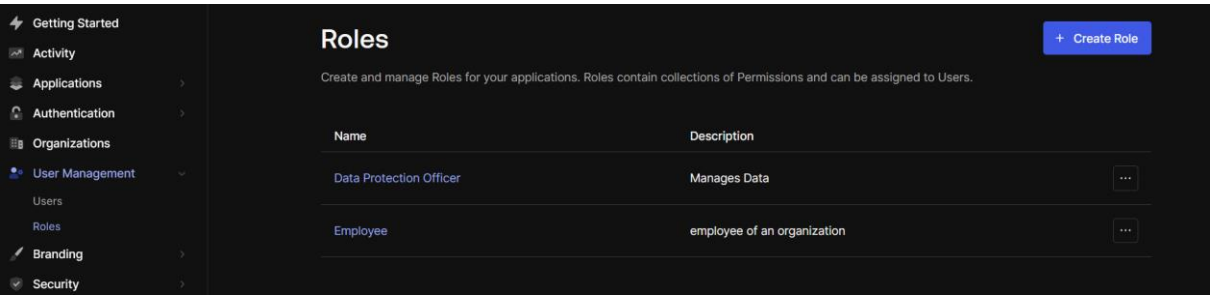
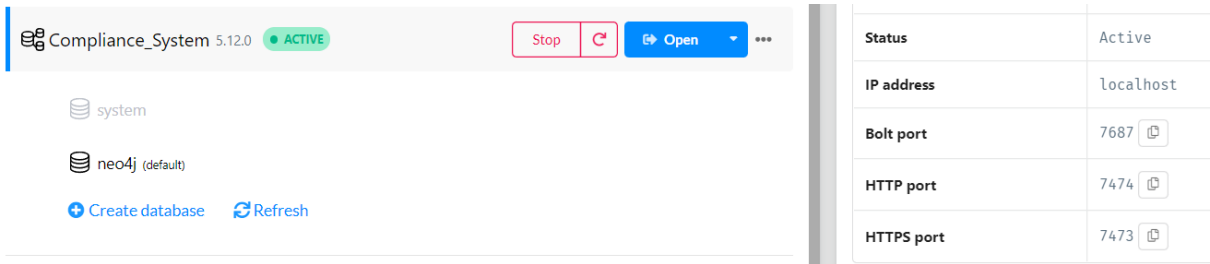


Figure 9 - Sample roles Okta

6.4 Neo4j & GDPRtEXT

Neo4j is a graph-based database which enables the user to use vector search on their constructed knowledge Graph. In the developed solution Neo4j was used together with GDPRtEXT.



6.5 Services

6.5.1 Authorization and routing

The authorization service is not only used for authorization but is also used as a gateway to access the other services. The service is redirecting and getting information from the services after checking if the

user has the correct permissions and if the user is logged in. If the user is not permitted to use a request the Error 403 – Forbidden is shown.

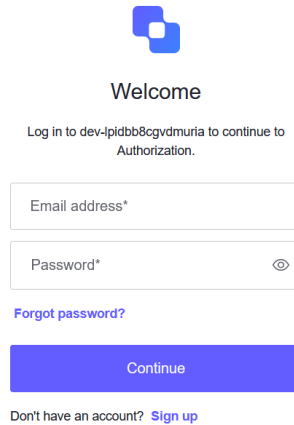


Figure 10 - Login page provided by Okta

Forbidden

You don't have the permission to access the requested resource. It is either read-protected or not readable by the server.

Figure 11 - Response if the given role does not support the functionality

6.5.2 LLM information – Generative and MongoDB Atlas Vector Search

6.5.2.1 Response with generative AI based on Ollama

When using the service “chat” to get a generative AI following prompt is used:

“Use the following context to answer the question.

Only use the knowledge provided by the documents.

Do not answer any questions out of the provided knowledge.

If you don't know the answer, just say that you don't know, don't try to make up an answer.

{context}

Question: {question}

Helpful Answer:“

This prompt should ensure that hallucinating is, which is a common problem of generative AI is kept to a minimum and it is only giving an answer, if it is clear based on the analyzed information and documents.

In the following figures 12 and 13 and extract from responses can be seen.

Query:

data breach handling

Result:

I don't know the answer to your question. According to the provided documents, there is no specific information on how to handle a data breach.

Figure 12 - Generative AI response - Example 1

Query:

How to handle a data breach?

Result:

The Data Breach Policy and Procedures document provides guidance on how to handle a data breach. The lead investigator should look at the type of breach,

Figure 13 - Generative AI response - Example 2

6.5.2.2 Response based on MongoDB Atlas Vector Search

Compared to the generative response the response from the vector search is not hallucinating. Based on the embeddings the data is compared and the part of documentation which is fitting the best is played back to the user.

Query:

Data breach handling

Result:

Policy Identifier : Data Breach Policy and Procedures

Page 10 of 12 The appointed lead should keep an ongoing log and clear report detailing t

Figure 14- Vector search response - Example 1

Query:

How to handle a data breach?

Result:

Policy Identifier : Data Breach Policy and Procedures

Page 9 of 12 The lead investigator should look at: -

Ⓜ Type of breach: A Data Breach may include any unauthorised or accidental

Figure 15 - Vector search response - Example 2

6.5.3 Company controls manager

The company controls manager enables the user to create, edit and delete company controls, namely policies and guidelines.

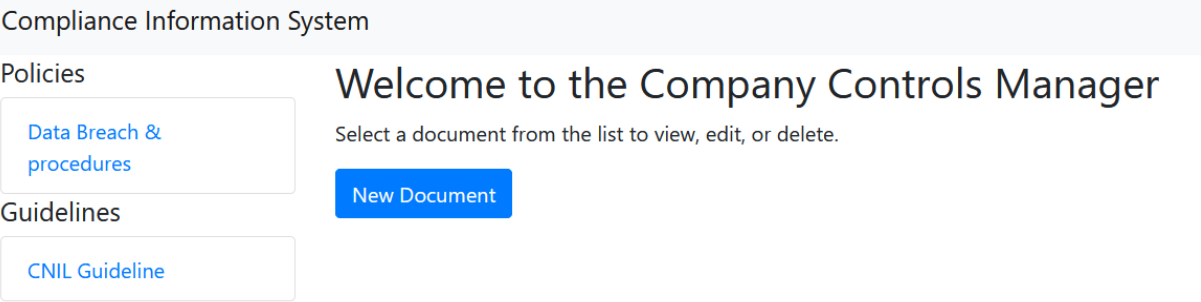


Figure 16 - Home view to the company controls management

The 'New Document' form contains two radio buttons: 'GUIDELINE' (selected) and 'POLICY'. It has input fields for 'Title' and 'Description'. Below these is a 'File' section with a file explorer button labeled 'Durchsuchen...' and the text 'Keine Datei ausgewählt.'. At the bottom are two buttons: 'Save' (blue) and 'Back to Overview' (grey).

Figure 17- Create new company control

Detailed information of the guideline

Title

CNIL Guideline

Description

guideline for developers for GDPR conform software

Latest document

Name	Version	File	Last Updated
file	3	66a846d9f501f93a28d8f576	2024-07-29 12:44:07.967000

Versions	File
0	66a77277f501f93a28d8f55c
1	66a778b1f501f93a28d8f56a
2	66a778c4f501f93a28d8f56b

Edit Delete Back

Figure 18 - View with options for the company control

6.5.4 GDPR Exploration with Graph

This service is mainly interacting with Neo4j and providing the information to the user. It offers selection by relevant categories and a search on the given ID. When clicking on the nodes the information about each node is shown. The information is extracted from Neo4j with so called Cypher queries.

```
query = f"""
MATCH (n)-[r]->(m)
WHERE id(n) = {node_id}
RETURN n, r, m
"""
```

Figure 19 - Sample query to extract information from Neo4j

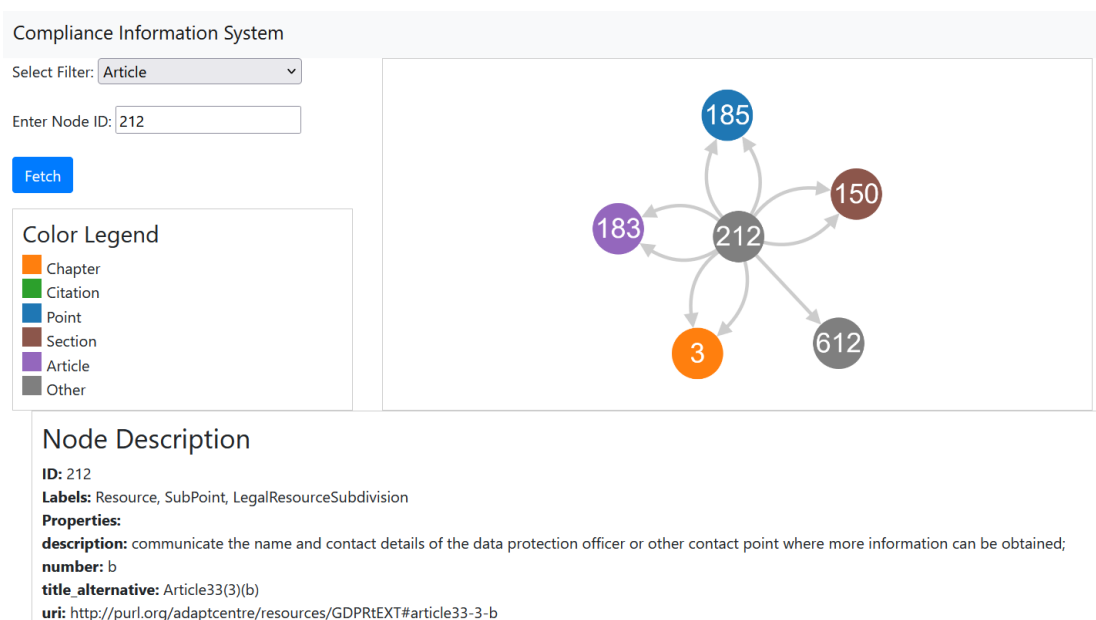


Figure 20 - Graph based GDPR exploration

7 Conclusion and future work

In the development of an information security policy repository different possibilities were analysed. Huge mappings done based on developed ontologies, namely GDPRtEXT offer a comprehensive and fast way to analyse information and knowledge on a certain domain. The build-up of a knowledge graph is a lot of work and needs constant maintenance, especially after changes to the legislation. To further use it also for policies it needs standardization in the use of policies. Due to sheer amount of policies and the huge challenge it is very difficult to do that.

The approach used in this implementation is make use of large language models to help analyse the huge amount of data, based on given documents. With this approach it is possible to neglect the

standardization of each document and therefore limit the amount companies would need to spend on building up standardization throughout all policies and departments.

With the additional help of a chatbot / information bot which has the ability to scan through documents offers employees the possibility to get a good overview and information of relevant topics.

For future work it can be analysed if and how a knowledge graph could be built up with the help of LLMs to enhance the experience of getting relevant information.

8 Bibliography

M. Alam and M. U. Bokhari, "Information Security Policy Architecture," International Conference on Computational Intelligence and Multimedia Applications (ICCIMA 2007), Sivakasi, India, 2007, pp. 120-122, doi: 10.1109/ICCIMA.2007.275.

Boehm: A Spiral Model of Software Development and Enhancement. In: IEEE Computer. Vol. 21, Aug. 5, Mai 1988, S. 61–72.

BSI – Arbeitsbeispiel Recplast: url: https://www.bsi.bund.de/DE/Themen/Unternehmen-und-Organisationen/Standards-und-Zertifizierung/IT-Grundschutz/IT-Grundschutz-Kompendium/Hilfsmittel-und-Anwenderbeitraege/Recplast/recplast_node.html accessed on 30.07.2024

Esteves, Beatriz and Rodríguez-Doncel, Víctor. ‘Analysis of Ontologies and Policy Languages to Represent Information Flows in GDPR’. 1 Jan. 2022 : 1 – 35.

European Union, Charter of Fundamental Rights of the European Union, 2012/C 326/02, 26 October 2012, <https://www.refworld.org/legal/agreements/eu/2012/en/13901> [accessed 20 March 2024]

Ferreira, Brito, Santos and Santos, "RuleKeeper: GDPR-Aware Personal Data Compliance for Web Frameworks," 2023 *IEEE Symposium on Security and Privacy (SP)*, San Francisco, CA, USA, 2023, pp. 2817-2834, doi: 10.1109/SP46215.2023.10179395.

Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design Science in Information Systems Research. *MIS Quarterly*, 28(1), 75–105. doi: 10.2307/25148625

Kenneth J. Knapp, R. Franklin Morris, Thomas E. Marshall, Terry Anthony Byrd, Information security policy: An organizational-level process model, *Computers & Security*, Volume 28, Issue 7, 2009, Pages 493-508, ISSN 0167-4048, <https://doi.org/10.1016/j.cose.2009.07.001>.

Livio Robaldo, Cesare Bartolini, and Gabriele Lenzini. 2020. [The DAPRECO Knowledge Base: Representing the GDPR in LegalRuleML](#). In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 5688–5697, Marseille, France. European Language Resources Association.

Nader Sohrabi Safa, Rossouw Von Solms, Steven Furnell, Information security policy compliance model in organizations, *Computers & Security*, Volume 56, 2016, Pages 70-82, ISSN 0167-4048, <https://doi.org/10.1016/j.cose.2015.10.006>.

MongoDB Cloud – URL: <https://cloud.mongodb.com> accessed 30.07.2024

Neo4j – URL: <https://neo4j.com/> accessed 30.07.2024

Okta Auth0 – URL: <https://auth0.com/docs/get-started/auth0-overview> accessed 30.07.2024

Paananen, Lapke, Siponen, State of the art in information security policy development, Computers & Security, Volume 88, 2020, 101608, ISSN 0167-4048, <https://doi.org/10.1016/j.cose.2019.101608>.

Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation) (Text with EEA relevance) (OJ L 119 04.05.2016, p. 1, ELI: <http://data.europa.eu/eli/reg/2016/679/oj>)

Palmirani, M., Martoni, M., Rossi, A., Bartolini, C., & Robaldo, L. (2018). Legal Ontology for Modelling GDPR Concepts and Norms. International Conference on Legal Knowledge and Information Systems.

Tuyikeze, Tite & Pottas, Dalenca. (2010). An information security policy development life cycle. 165-176.