Eberhard Karls Universität Tübingen
Mathematisch-Naturwissenschaftliche Fakultät
Wilhelm-Schickard-Institut für Informatik

Master Thesis Mediainformatics

Development of a portal for patient search in personalized medicine

Peter Placzek

March 28, 2024

Reviewers

Prof. Dr.-Ing. Oliver Kohlbacher
(Bioinformatics)
Wilhelm-Schickard-Institut für Informatik
Universität Tübingen

Prof. Dr. Thomas Walter
(Informatics)
Wilhelm-Schickard-Institut für Informatik
Universität Tübingen

Placzek, Peter:
Development of a portal for patient search in personalized medicine
Master Thesis Mediainformatics
Eberhard Karls Universität Tübingen
Thesis period: 01.10.2023-01.04.2024

Abstract

This master's thesis deals with developing and evaluating a portal for patient search in the context of **personalized medicine (PM)**. The topic of PM is introduced at the beginning, and the importance of patient search portals for medical practice is emphasized.
In the second chapter, the context and framework conditions for the work are explained in more detail. The significance of PM is discussed, and its connection to centers for PM and the **German Network for Personalized Medicine (DNPM)** is elaborated. Furthermore, legal and ethical aspects, such as data protection regulations and measures for anonymization and pseudonymization, are discussed.
The third chapter deals with the methods and materials used to develop the portal. It describes the data models used and the development process using various programming languages, libraries, and frameworks.
The fourth chapter explains the portal's visual realization. It discusses the portal's functionalities and presents various areas, such as the administration, **Molecular Tumor Board (MTB)**, and **Rare Diseases (RDs)** areas. The objectives and concepts behind the visual implementation and the evaluation of the implementation in the context of the MTB use case are also discussed.
The fifth chapter deals with user evaluation and the results of the user survey. Here, the evaluation of the graphical representation of patient information and the discussion of the survey results and their implications for portal development are covered.
Chapter six examines the implementation and results of the user evaluation in detail. It also discusses the solutions developed, their limitations, and potential weaknesses. An outlook on the portal's possible improvements and future developments is also provided.
The conclusion summarizes the most important findings and results of the master's thesis. It draws conclusions on the fulfillment of the research objectives and provides an outlook on future research directions and applications of the patient search portal in medical practice.

Zusammenfassung

Diese Masterarbeit befasst sich mit der Entwicklung und Evaluierung eines Portals für die Patientensuche im Kontext der personalisierten Medizin. Zu Beginn wird in das Thema der personalisierten Medizin eingeführt und die Bedeutung von Patientensuchportalen für die medizinische Praxis hervorgehoben.
Im zweiten Kapitel werden der Kontext und die Rahmenbedingungen für die Arbeit näher erläutert. Es wird auf die Bedeutung der personalisierten Medizin eingegangen und die Verbindung zu Zentren für personalisierte Medizin und dem Deutschen Netzwerk für personalisierte Medizin herausgearbeitet. Des Weiteren werden rechtliche und ethische Aspekte, wie Datenschutzbestimmungen und Maßnahmen zur Anonymisierung und Pseudonymisierung, erörtert.
Das dritte Kapitel beschäftigt sich mit den Methoden und Materialien, die zur Entwicklung des Portals verwendet wurden. Es beschreibt die verwendeten Datenmodelle und den Entwicklungsprozess unter Verwendung verschiedener Programmiersprachen, Bibliotheken und Frameworks.
Das vierte Kapitel erläutert die visuelle Umsetzung des Portals. Es erörtert die Funktionalitäten des Portals und stellt verschiedene Bereiche vor, wie z. B. die Bereiche Administration, MTB und RD. Die Ziele und Konzepte hinter der visuellen Umsetzung und die Evaluation der Umsetzung im Kontext des Anwendungsfalls MTB werden ebenfalls diskutiert.
Das fünfte Kapitel befasst sich mit der Nutzerevaluation und den Ergebnissen der Nutzerbefragung. Hier werden die Bewertung der grafischen Darstellung von Patienteninformationen und die Diskussion der Umfrageergebnisse und deren Implikationen für die Portalentwicklung behandelt.
In Kapitel sechs werden die Umsetzung und die Ergebnisse der Nutzerevaluation ausführlich diskutiert. Außerdem werden die entwickelten Lösungen, deren Grenzen und mögliche Schwachstellen diskutiert. Ein Ausblick auf mögliche Verbesserungen und zukünftige Entwicklungen des Portals wird ebenfalls gegeben.
Das Fazit fasst die wichtigsten Erkenntnisse und Ergebnisse der Masterarbeit zusammen. Es werden Rückschlüsse auf die Erfüllung der Forschungsziele gezogen und ein Ausblick auf zukünftige Forschungsrichtungen und Anwendungen des Patientensuchportals in der medizinischen Praxis gegeben.

Acknowledgements

The portal was developed for the DNPM:DIP project, which the German Federal Ministry of Education and Research generously funds. During my work, I had the privilege of attending the weekly meetings of the DNPM working group, for which I would like to thank everyone involved for the inspiring collaboration. In particular, I would like to thank Prof. Dr.-Ing. Oliver Kohlbacher and Prof. Dr. Walter for their support and for providing the framework for this work. Special thanks are due to Dr. Lucien Clin, who supervised me during my Master's thesis. His expert guidance, constructive feedback, and stimulating discussions were instrumental in supporting and inspiring me to develop this portal.

Contents

List of Figures **ix**

List of Tables **xi**

List of Abbreviations **xiii**

1 Introduction **1**

1.1 Motivation of the work 1

2 Background **3**

2.1 Personalized Medicine (PM) 3

2.2 Center for Personalized Medicine (ZPM) 5

2.3 German Network for Personalized Medicine (DNPM) 5

2.4 Use cases 5

2.4.1 Rare Diseases (RDs) 6

2.4.2 Molecular Tumor Boards (MTBs) 8

2.5 Projects 9

2.5.1 bwHealthCloud (BWHC) 9

2.5.2 DNPM:DIP 9

2.6 General Data Protection Regulation 11

2.7 Data Anonymization and Pseudonymization 11

2.8 Data Processing 12

2.9 Data Sharing 12

2.9.1 Centralized 13

2.9.2 Decentralized (site-level meta-analysis) 14
2.9.3 Decentralized (federated learning) 15
2.10 Clinical Decision Support System 15
2.11 Requirement engineering 17
2.11.1 Introduction 17
2.11.2 Requirement Gathering 17
2.11.3 Requirement Implementation/development 19

3 Methods and Material **21**
3.1 Data Model(s) 21
3.1.1 RD 22
3.1.2 MTB 23
3.2 Requirement analysis 24
3.3 Development & Deployment 24
3.3.1 Languages 24
3.3.2 Libraries & Frameworks 26
3.3.3 Versioning 27
3.3.4 Release-Management 27
3.3.5 Deployment 28
3.4 Architecture 28
3.4.1 Resource-Server 28
3.4.2 Auth-Server 29
3.4.3 Portal 29
3.5 Infrastructure 30
3.6 Implementation 31
3.6.1 General 31
3.6.2 Packages 31

4 Results - Realization **35**
4.1 General 35

4.1.1 Layout 35
4.1.2 Home 36
4.1.3 Login 37
4.2 Admin 37
4.2.1 Identity Provider(s) 37
4.2.2 User(s) 38
4.2.3 Role(s) 38
4.3 RD 38
4.3.1 The Search 38
4.3.2 Query 40
4.3.3 Query Summary 40
4.3.4 Query Patients 41
4.3.5 Query Patient 43
4.4 MTB 44
4.4.1 The Search 44
4.4.2 Query 45
4.4.3 Query Summary 46
4.4.4 Query Patients 50
4.4.5 Query Patient 50

5 Results - Evaluation 53
5.1 Setup 53
5.2 Result 53
5.2.1 Search view 54
5.2.2 Result view 55
5.2.3 General 56
5.2.4 System Usability Scale (SUS) 57

6 Discussion and Outlook 59
6.1 Discussion 59
6.2 Outlook 60

Bibliography **61**

List of Figures

2.1 Individual Patient [GS18] 4
2.2 Dutch MTB model [KGL+21] 10
2.3 Overview of the three main data-sharing models [SRTP+21] . . 14
2.4 Diagram of key interactions [SPB+20] 16

3.1 DNPM:DIP - Architecture 28
3.2 DNPM:DIP - Infrastructure 30

4.1 Portal - Home Page 36
4.2 Portal - Login Page 37
4.3 Portal - RDs Search Page 39
4.4 Portal - RDs Query Summary 40
4.5 Portal - RDs Query Summary Diagnostics 41
4.6 Portal - RDs Query Patients 41
4.7 Portal - RDs Query Patient 43
4.8 Portal - RDs Query Patient Diagnostics 43
4.9 Portal - MTB Search Page 44
4.10 Portal - MTB Query Summary 46
4.11 Portal - MTB Query Summary Diagnostics 47
4.12 Portal - MTB Query Summary Medication #1 47
4.13 Portal - MTB Query Summary Medication #2 48
4.14 Portal - MTB Query Summary Overall Survival 48
4.15 Portal - MTB Query Summary Progression-free Survival 49

4.16 Portal - MTB Query Patient 51
4.17 Portal - MTB Query Patient Treatment 52

List of Tables

2.1 RDs with the highest estimated prevalence 7

5.1 Results - Evaluation: Search View 54
5.2 Results - Evaluation: Result view 55
5.3 Results - Evaluation: General 56
5.4 Results - Evaluation: SUS 57

List of Abbreviations

PM	Personalized medicine
DNPM	German Network for Personalized Medicine
MTB	Molecular Tumor Board
RD	Rare Disease
GDPR	General Data Protection Regulation
ZPM	Center for Personalized Medicine
CDSS	Clinical Decision Support System
NGS	Next Generation Sequencing
ICD	International Classification of Diseases
BWHC	bwHealthCloud
DIP	DATA integration Platform
EHR	electronic health records
RE	Requirements Engineering
ETL	Extraction-transformation-loading
DW	Data Warehouse
WHO	World Health Organization
HGNC	HUGO Gene Nomenclature Committee
JS	JavaScript
TS	TypeScript
HTML	HyperText Markup Language
CSS	Cascading Style Sheets
SEMVER	semantic versioning
API	Application Programming Interface
UI	Uniform Resource Locator

Chapter 1

Introduction

PM is playing an increasingly important role in our society. This development is driven by the growing knowledge of individual differences in the response to drugs, as well as the decoding of the human genome, which has provided a deeper insight into the genetic basis of diseases [CV15].
PM enables treatments that are tailored to the individual's genetic predisposition, thereby increasing the effectiveness of the treatment. For example, in the context of oncology, PM seeks to address unexplained drug resistance, the genomic heterogeneity of tumors, inadequate means of monitoring tumor response and recurrence, and limited knowledge about the use of drug combinations. Thus, it enriches and modifies the successful approach to oncology without replacing it, while providing a strong framework to accelerate the adoption of precision medicine in other medical fields [CV15].
PM not only helps with treatment, but also enables the early detection of diseases through genetic markers, which allows timely intervention [HC10].
The problem is that each cancer is highly individualized and that RDs seldom occur in the local population, making it difficult to find a treatment plan, let alone put together a cohort for clinical trials. In addition, the **General Data Protection Regulation (GDPR)** in Europe places strict requirements on the collection, storage and processing of personal data, including genomic information. This makes it difficult or almost impossible to collect data centrally.

1.1 Motivation of the work

The thesis aims to develop a portal for the targeted search of patients in PM. This portal will enable researchers to efficiently identify patients with certain characteristics or criteria. This will optimize case preparation by deriving evidence-based treatment recommendations based on the treatment histories of similar patients. Case preparation in this context refers to the process by which physicians collect and analyze data and information about a particular

patient to make an informed decision about their treatment.
In addition, the portal is intended to be a valuable resource for scientific research by enabling researchers to analyze data in defined patient cohorts and gain new insights. Finally, identifying potential patient cohorts supports the effective planning of studies by enabling the targeted identification and recruitment of potential participant groups.
The portal will be developed using state-of-the-art technologies to ensure maximum security, scalability, and extensibility. This will guarantee that it meets current requirements and can cope with future developments. The aim of the portal is to integrate each use case individually into an overarching application, to ensure a clear separation between the use cases during development.

Chapter 2

Background

This chapter provides an overview of the background to deepen the understanding of the developed portal for patient search in PM (section 2.1). The importance of PM is explained, and its connection to the ZPMs (section 2.2) and the DNPM (section 2.3) is shown. The various use cases of the portal are then highlighted, and a similar project (section 2.5.1) that pursues comparable goals is outlined. The DNPM:DIP (section 2.5.2) platform, into which the portal will be integrated, is also briefly presented. Data protection regulations such as the GDPR (section 2.6) and measures for data anonymization and pseudonymization (section 2.7) are discussed. The importance of a **Clinical Decision Support System (CDSS)** (section 2.10) is also explained. Finally, the role of requirement engineering as a basis for portal development is emphasized.

2.1 Personalized Medicine (PM)

PM, a groundbreaking approach in healthcare, is about tailoring medical interventions to the individual characteristics of each patient, taking genetic information into account. This innovative paradigm aims to go beyond the traditional one-size-fits-all model and offers customized treatments based on a deeper understanding of an individual's genetic predisposition [SGK19].
A cornerstone for this was laid by "Sanger" sequencing, which was successfully used for the first time in 2003 after an enormous effort of more than ten years to decode the DNA sequence of the first human genome. **Next Generation Sequencing (NGS)**, a rapid, relatively inexpensive, large-scale DNA sequencing technology, superseded this method [SGK19].
In addition to DNA sequencing, technologies such as proteomics, imaging protocols, and wireless health monitoring devices, which have revealed significant inter-individual differences in disease processes, have also made a significant contribution [GS18].

One of the main advantages of PM is its ability to introduce targeted therapies. By tailoring treatment plans to each patient's unique genetic composition, this approach offers a more precise and effective intervention than previously possible [CV15].
While this innovative paradigm shift holds great potential, it also brings several challenges, particularly regarding regulatory approval for everyday use. Another problem is that only some people in the healthcare system - such as doctors, healthcare executives, insurance companies, and patients - fully accept PM. The biggest challenge is to show that PM works better than conventional methods. Some personalized therapies, such as specific cancer treatments or drugs for some genetic issues, can be rather expensive [GS18].
PM, despite some challenges, can therefore improve patient care. Nonetheless, it should be emphasized that while PM has its roots in the results of genetic studies, it is widely recognized that other factors (environmental influences, developmental phenomena, epigenetic changes, and behaviors) must also be considered when determining the optimal way to treat an individual patient [GS18] (Fig. 2.1).

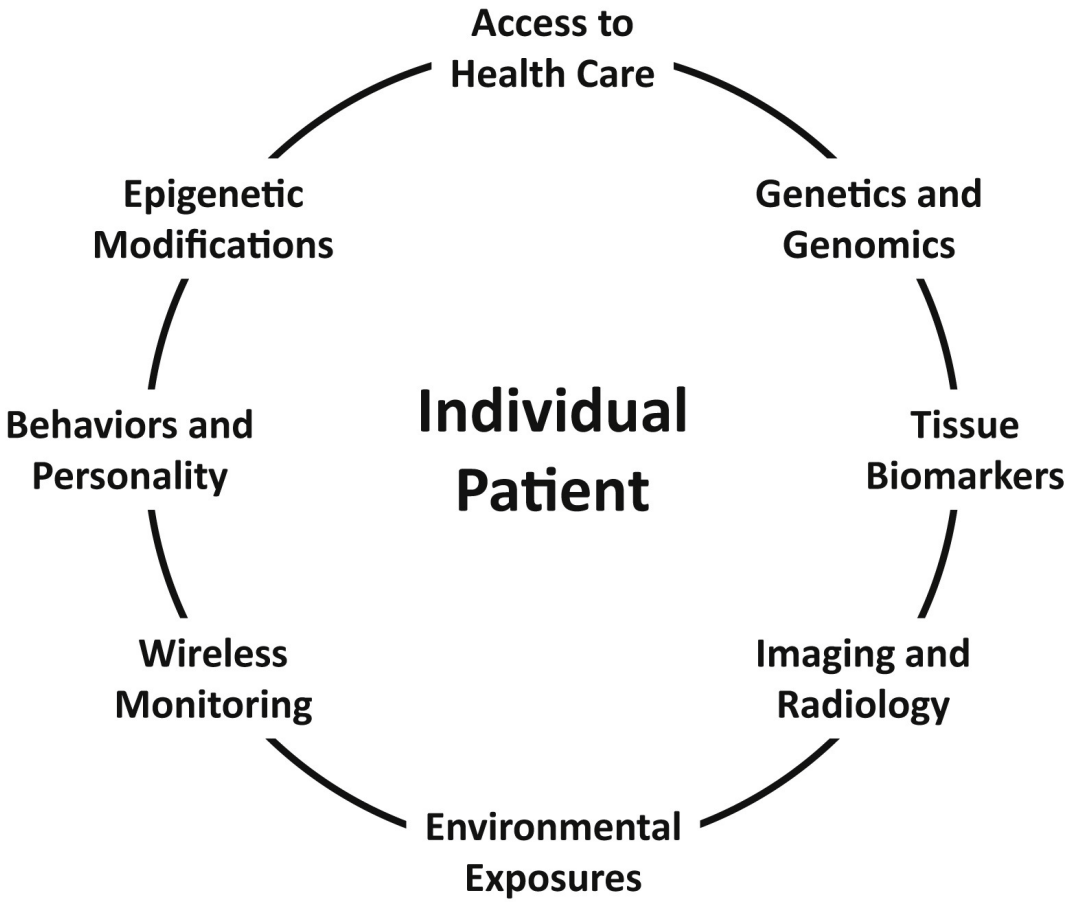

Figure 2.1: Individual Patient [GS18]

2.2 Center for Personalized Medicine (ZPM)

The major challenge to PM is to pool knowledge and integrate it into clinical practice. The Baden-Württemberg Ministry of Social Affairs and Integration tackled this task in 2019 by establishing four ZPMs at the university hospitals in Freiburg, Heidelberg, Tübingen, and Ulm. Various institutions from different fields like healthcare, research, technology, and basic scientific research work together in these ZPMs to make collected patient data usable for clinical purposes [Gra23] [zpm].
These centers have set themselves the task of identifying links between individual characteristics, such as genetic changes in tumors, and tailored treatment options based on comprehensive research findings. Oncology is the first case of PM that is realized in these centers. MTBs, interdisciplinary case conferences play a central role in developing therapeutic strategies based on clinical data, including imaging, biopsies, and high-throughput data [Gra23].
Patients with similar biomarkers for the case under consideration are sought when preparing an MTB conference. The information on the treatment results within such a group of comparable patients (a cohort) serves as evidence for the treatment proposal for the respective patient. However, due to the uniqueness of each cancer case, the number of patients within a cohort is usually relatively small [Gra23].

2.3 German Network for Personalized Medicine (DNPM)

The DNPM is a network of 26 university hospitals in Germany. The aim is to jointly improve the medical care and quality of life of people affected by an advanced or rare disease throughout Germany. To this end, ZPMs are being established at all universities, which together with the existing ZPMs in Freiburg, Heidelberg, Tübingen and Ulm form the DNPM. The aim of the network is to provide patients throughout Germany with transparent and standardized access to PM [dnpb].

2.4 Use cases

The portal being developed as part of the research work will initially cover two use cases. These use cases are RDs and MTB, which are described in more detail below.

2.4.1 Rare Diseases (RDs)

RDs affect only a limited number of people. In the European Union, up to one in 2000 people are affected; in the USA, it affects around one in 1250 people. The number of diseases that fit this definition is large, exceeding 5.000, according to the World Health Organization. Consequently, the potential number of patients affected by RDs is estimated at around 30 million in Europe and 25 million in North America. However, it remains a major challenge to accurately determine the burden of RDs, as epidemiological data is unavailable for most of these diseases [SHDA08].

RDs represent a significant public health challenge and a complex problem for the medical community. These diseases, known as "health orphans", have been neglected for many years. The pharmaceutical industry also neglected the development of therapies for RDs until the Orphan Drug Act was passed in the USA in 1983, which coined the term "orphan drugs". In its 1989 report, the US government's National Commission on Orphan Disease raised public awareness of the difficulties faced by patients with RDs. The Commission's multi-stakeholder hearings shed light on patient care issues, including limited information about RDs, challenges in funding research, disadvantages in providing adequate health insurance and medical cost coverage, and limited availability of effective treatments [SHDA08].

The globally used **International Classification of Diseases (ICD)** is proving impractical for RDs as it creates barriers to reliable patient registration and makes assessing the economic and social impact difficult. To solve this problem, the European Task Force on RDs and the World Health Organization developed ICD-10 to create a unified coding system. Although national or international registries for some diseases are established and maintained by different bodies, a universally recognized coding system remains elusive [SHDA08].

The effort to determine the prevalence of RDs carried out by the European Organization for RDs (Eurordis) and Orphanet with the support of the European Commission has shown that there are problems. One study provided estimates of the prevalence of various RDs. However, it highlighted the need for more reliable data, the poor consistency between sources of information, and the poor methodological quality of epidemiological studies. In addition, there are few facilities for biochemical or genetic testing for RDs [SHDA08] [orp]. Table 2.1 shows the RDs with the highest estimated prevalence per 100.000 people.

	Estimated prevalence (per 100.000)
Brugada syndrome	50
Erythropoietic protoporphyria	50
Guillain-Barré syndrome	47
Familial melanoma	46
Autism, genetic types	45
Tetralogy of Fallot	45
Scleroderma	42
Great vessels transposition	32,5
Focal dystonia	30
Marfan's syndrome	30
Non-Hodgkin malignant lymphoma	30
Retinitis pigmentosa	27.5
Gelineau's disease	26
Multiple myeloma	26
α1 antitrypsin deficiency	25
Congenital diaphragmatic hernia	25
Juvenile idiopathic arthritis	25
Neurofibromatosis type 1	25

Table 2.1: RDs with the highest estimated prevalence

2.4.2 Molecular Tumor Boards (MTBs)

The integration of NGS into clinical practice has ushered in a new era of precision oncology. In cases where standard therapies prove ineffective, cancer patients can be subjected to comprehensive molecular analysis using NGS to identify potential genome-based therapeutic targets. In 316 ongoing/planned clinical trials, patients are selected on the basis of molecular NGS profiling. However, the systematic interpretation of the extensive NGS data remains a challenge [LLMS20].
Due to rapid technological progress and decreasing sequencing costs, comprehensive tumor sequencing, including whole-exome and whole-genome sequencing, is expected to become the standard in cancer treatment. However, the increasing complexity of genetic information poses a challenge for clinicians, and the gap between clinical knowledge and genetic potential in cancer treatment is widening [VdVVHVL$^+$17].
MTBs, originally introduced at the University of Michigan and now widely used, are critical to close the growing gap between clinical practice and technologic potential in cancer care and have proven to be a solution.
MTBs are interdisciplinary teams that are revolutionizing cancer treatment. These teams, made up of oncologists, bioinformaticians, geneticists and others, meet regularly to analyze complex genetic data using next-generation sequencing. Unlike traditional tumor boards, MTBs involve experts from outside the clinical setting, fostering a broad range of expertise. Their main task is to formulate targeted treatment recommendations based on genetic findings and to explore experimental treatments that go beyond standard oncology practice [MSMW18].
In examining the global state of MTBs, a systematic review-based approach was used to provide a scientific basis for MTB-related questions. Summarizing data from 40 selected studies, a total of 6303 cases of MTBs were discussed, of which 1107 (17.6%) received an indication for molecular-based therapies. The main objective is to identify therapeutic strategies based on genetic analysis for patients who do not respond to standard therapies. Recommendations should take into account not only molecular changes but also patient factors such as performance status and comorbidities. MTB reports should include various parameters to ensure a comprehensive overview [LLMS20].
However, a survey shows that less than 50% of hospitals and only 5% of non-academic hospitals have such access. Furthermore, existing MTBs have significant differences in terms of composition, tasks, tools and workflows, leading to potential differences in quality of care and hindering data sharing [VdVVHVL$^+$17].
In terms of tumor types treated in the MTBs, a review of cases where standard therapy had failed revealed that of 440 cases with clear tumor histology, a significant proportion was due to rare tumors (21.4 %), followed by breast can-

cer (20%), brain tumors (15.5%), gynecological tumors (14.1%), lung cancer (7.3%) and colorectal tumors (6.4%). Rare malignancies in particular, which account for almost a quarter of all cancers, are an important target for the discussion on MTB. NGS is recommended as an integral part of histological diagnosis in malignancies for which there are no standardized therapeutic options [LLMS20].
A possible workflow for MTBs is shown in the Dutch model (see 2.2), which involves a collaborative process between external and MTB-associated physicians/oncologists and Clinical Molecular Specialist Practitioners (CSMP)/pathologists. The diagram (see 2.2) outlines the responsibilities where all parties can ask molecularly oriented questions. The MTB-associated oncologist prepares the clinical case, while the CSMP and the pathologist jointly characterize the molecular profile. During the MTB meeting, a diagnostic and/or therapeutic recommendation is formulated, communicated to the applicant, entered into the electronic medical record and registered on site. This recommendation helps the applicant to select (molecular) tests or a (targeted) therapy. For biomarkers with germline and somatic implications, the presence of a clinical geneticist is recommended, and the presence of a bioinformatician is recommended for discussions on whole-exome or whole-genome sequencing results [KGL+21].

2.5 Projects

2.5.1 bwHealthCloud (BWHC)

The **bwHealthCloud (BWHC)** is a state-wide infrastructure for data exchange in Baden-Württemberg. It brings together the databases of all four ZPM locations in Baden-Württenberg [bwh].
Since 2020, work has been underway to design a secure data integration and management platform [bwh].
It supports the networking of practitioners and researchers in different institutions and removes barriers between the individual databases. The aim is to create a broad evidence-based knowledge base on the one hand and to ensure mutual quality assurance of treatment in the ZPMs on the other. The BWHC is not a cloud in the narrower sense; instead, it allows users to search for treatment options in the other centers' databases [bwh].

2.5.2 DNPM:DIP

In the context of DNPM, a **data integration platform (DIP)** and the associated IT infrastructure are to be established. Data from molecularly diagnosed patients will form the basis for deriving new options for action and evaluating

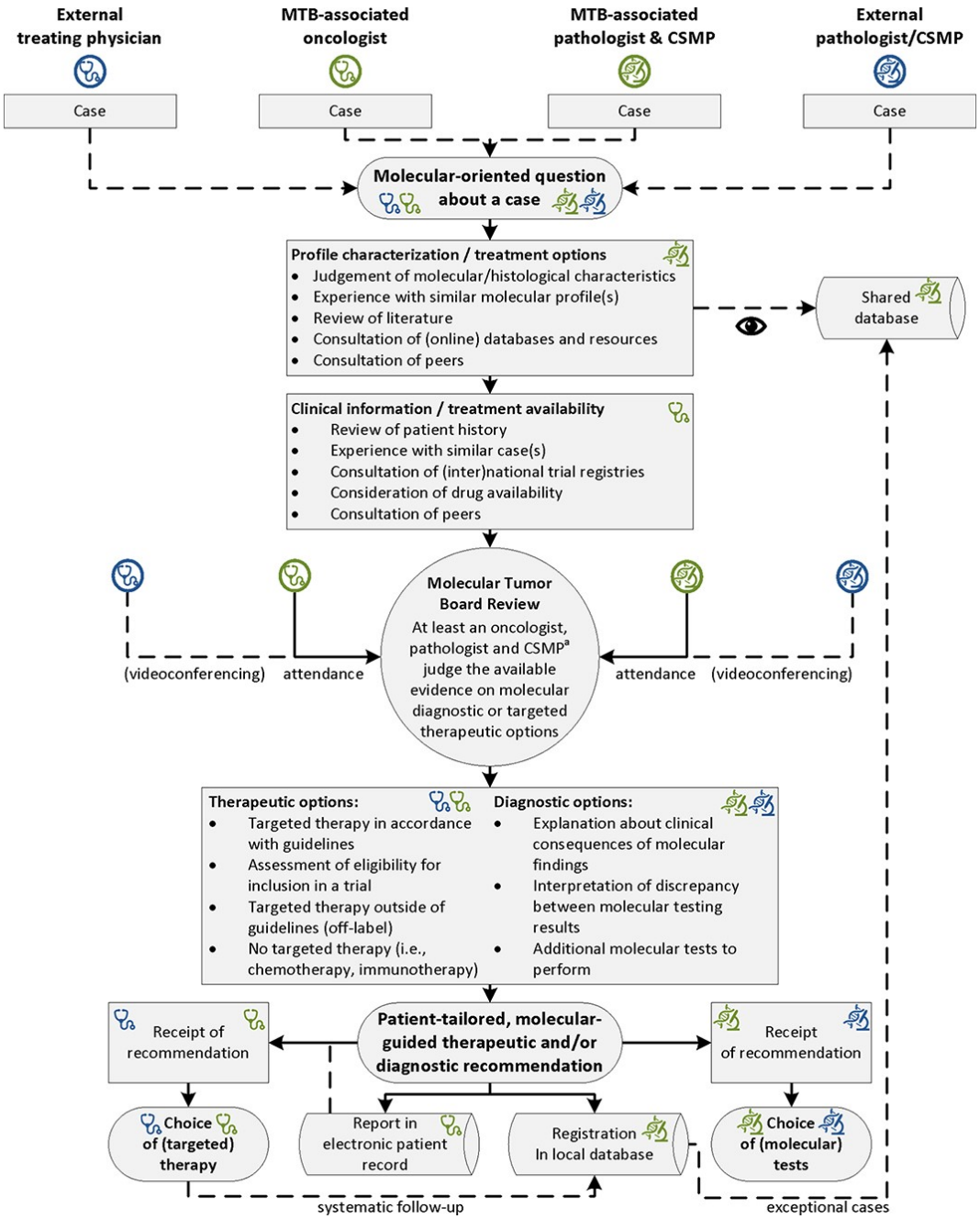

Figure 2.2: Dutch MTB model [KGL+21]

individual therapy approaches [dnpa].
A broad knowledge base will be created based on the pseudonymized data from patients at the ZPM. The evaluation of the data will make it possible, for example, to draw conclusions from the experimental treatment of patients with individual treatment trials, such as why they were particularly successful with certain mutations. These findings will be made available to the public

through publications [dnpa].
Harmonized core data sets of the ZPM patients are stored decentrally on local data nodes, and data queries are carried out via a web front end created as part of this master's thesis.
Data for individual scientific evaluations can also be requested via a release process from the local data nodes (Data Use & Access Committees) [dnpa].
In contrast to the BWHC, the DIP will cover not only the MTB application but also RDs. In the future, applications such as the Molecular Inflammation Board (MIB) will be integrated. The patient portal developed in the context of this work, part of the decentralized platform, is used for collaboration and care coordination by doctors, specialists, and researchers. Visualizations within the portal accelerate the decision-making process by showing patterns and correlations in patient data. This acceleration facilitates the timely initiation of appropriate treatment strategies. In addition, the portal supports the customization of treatment plans based on individual patient characteristics that align with PM principles.

2.6 General Data Protection Regulation

The GDPR, enacted in May 2018, aims to harmonize and unify legal regulations across the EU, fostering innovation while enhancing individuals' privacy rights. The GDPR introduces rules to provide EU citizens with more control over their personal data and governs the protection and processing of such data [KKRB+20].
It has significantly impacted data anonymization and pseudonymization (see 2.7), data processing (see 2.8), and data sharing (see 2.9) practices.

2.7 Data Anonymization and Pseudonymization

The GDPR defines personal data as data relating to an identifiable natural person and considers pseudonymized data, where identifiers are removed, as personal data. Anonymized data, where individuals are no longer identifiable, falls outside GDPR provisions. Two conflicting approaches determine this classification: the absolute approach, where any theoretical chance of reidentification makes anonymized data personal, and the relative approach, where anonymized data ceases to be personal if reidentification is reasonably unlikely. Some EU member states, like France, follow the absolute approach, while others, like Ireland and Germany, adopt the relative approach, requiring reasonable efforts for reidentification. The European Data Protection Board favors the relative approach at the supranational level, emphasizing the importance

of the likelihood of reidentification in determining the status of anonymized data under GDPR [SRTP+21].

2.8 Data Processing

The GDPR provisions apply to data controllers, including health care and research institutions, requiring them to ensure lawful, proportionate, and rights-protecting processing of personal data [SRTP+21].
Encryption, treated as equivalent to stripping identifiers, is emphasized for data security. Data controllers must consider the state of the art and associated risks when adopting security measures. The GDPR emphasizes data minimization for scientific purposes, promoting the use of nonpersonal, pseudonymized, or aggregate data instead of directly identifying data. Data transfer, especially outside the EU, requires adequate privacy protection, organizational safeguards, and consideration of joint controllership or controller–processor arrangements. Rights of data subjects can be exercised against each controller in joint controllership, and processing contracts in controller-processor relationships must define key processing aspects and ensure confidentiality and security [SRTP+21].

2.9 Data Sharing

Sharing medical data across multiple sites, nationally or internationally, is crucial in medical research. The challenge is to design data sharing in a way that preserves the privacy of the individual and the utility of the data [SRTP+21].
However, despite the obvious overcoming of barriers to research, differences between ethical and legal requirements remain in all areas of law. Various organizational strategies have been developed to address these issues, particularly for international academic consortia [SRTP+21].
One example is the International Cancer Genome Consortium, endeavoring to collect cancer genomes in a cloud environment. Conflicts between US and European Union data protection laws have prevented the establishment of an international cloud as part of the Pancancer Analysis of Whole Genomes Project. Within the EU, there are potential differences in regulating the processing of health-related personal data. The Clinical Trials Regulation and the EU GDPR require different standards of consent for processing health-related data depending on whether or not this data is collected as part of a clinical trial protocol. The consequence of this difference is that data collected for a specific purpose, such as a trial protocol, may not be made available for a secondary research purpose unless appropriate consent has

been obtained [SRTP+21].
Consent to processing personal data is essential when the data is collected directly from the data subjects. However, for consent to have a valid legal basis, the following conditions must be met: (1) freely given; (2) specific; (3) informed; and (4) unambiguous [KKRB+20].

1 Consent must be a genuine choice by the individual and the individual should have control over the information provided [KKRB+20].

2 Consent is specific to the purpose of processing and cannot be used for other purposes. It is linked to the principle of purpose limitation [KKRB+20].

3 The data subject must be given adequate information to make an informed decision. This includes information about the identity of the controller, the type of data collected, the right to withdraw consent, possible automated decision-making and the risks associated with the transfer of data outside the European Economic Area [KKRB+20].

4 Consent must be given in a clear, unambiguous form that requires a precise declaration and a confirming action. This does not necessarily have to be an opt-in field, but can also be a signature or verbal confirmation. Written consent is preferable if this is possible to prove validity [KKRB+20].

Finally, it may be impossible to share data between institutions or jurisdictions due to study restrictions. Although EU data protection law reforms have been proposed, contractual and technical measures are currently the best available solutions [SRTP+21]. Irrespective of regulations and the European GDPR, the most important models (see 2.3) for exchanging medical data are compared below. The second decentralized approach (site-level meta-analysis) is used in this thesis project.

2.9.1 Centralized

The centralized model involves medical institutions sharing patient data individually in a single repository hosted by a trusted vendor. While this model facilitates data access for analytics and reduces infrastructure costs, it also poses challenges regarding data privacy, particularly in different jurisdictions. The central location where the data is hosted becomes a potential single point of failure and requires the trust of all participating locations. Traditional anonymization techniques such as k-anonymity aim to protect privacy but may need to be improved due to increasing re-identification attacks and the

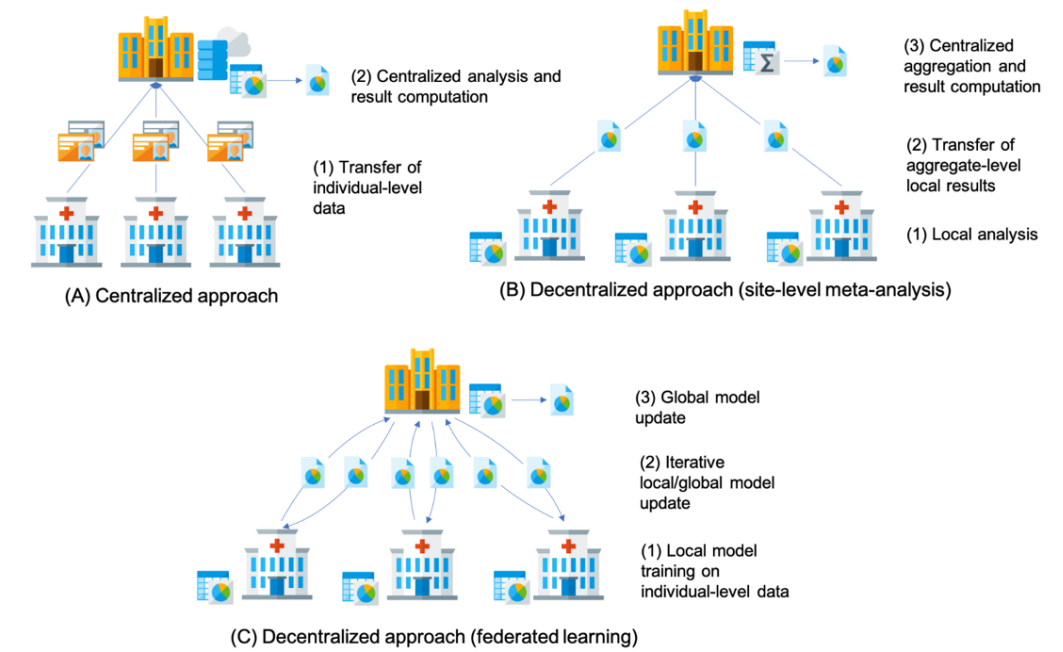

Figure 2.3: Overview of the three main data-sharing models [SRTP+21]

increasing dimensionality of data. As a result, researchers often resort to pseudonymization techniques and legal measures, leading to administrative complexity and potential risks of re-identification. While the combination of traditional pseudonymization and governance can meet legal standards, more is needed to ensure anonymization by the GDPR [SRTP+21].

2.9.2 Decentralized (site-level meta-analysis)

The decentralized data-sharing model allows medical sites to retain control over individual-level patient data, avoiding the need for physical transfer. Each site conducts statistical analysis on local datasets, and aggregate-level results are sent to a trusted site for meta-analysis. While reducing the risk of reidentification, aggregate-level data may still pose privacy concerns, especially for subpopulations. To counter inference attacks, clinical sites can apply obfuscation techniques, adding statistical noise to local statistics before transfer. However, this compromises utility, as noise levels impact the reliability of aggregated results in a meta-analysis. Additionally, the approach lacks flexibility, requiring coordination among sites before analysis, and may be affected by cross-study heterogeneity, impacting result accuracy [SRTP+21].

2.9.3 Decentralized (federated learning)

In the decentralized data-sharing model, medical sites maintain control over individual-level patient data, conducting statistical analyses locally and sending aggregate-level results to a trusted site for meta-analysis. While this model reduces reidentification risks, concerns about privacy, particularly for subpopulations, may persist. Clinical sites can use obfuscation techniques to address inference attacks, introducing statistical noise to local statistics before transfer. However, this compromises utility, affecting the reliability of aggregated results in a meta-analysis. The approach lacks flexibility, relies on coordination among sites before analysis, and may be susceptible to cross-study heterogeneity, impacting result accuracy [SRTP+21].

2.10 Clinical Decision Support System

A CDSS is software designed to help healthcare providers make informed decisions by integrating clinical knowledge, patient information, and other health data. CDSSs aim to improve medical decision-making by presenting patient-specific assessments or recommendations to physicians. They can be knowledge-based, using rules created based on expert medical knowledge, or non-knowledge-based, using artificial intelligence (AI), machine learning (ML), or statistical pattern recognition [SPB+20]. In this thesis, a knowledge-based CDSS is realized and implemented.
The knowledge-based and non-knowledge-based approach is illustrated in the following diagram.

CDSSs have evolved since the 1970s from stand-alone systems to integrating **electronic health records (EHR)** and computerized provider order entry (CPOE) systems. They play a critical role in patient safety by reducing medication errors, informing physicians of drug interactions, and automating the administration of medications. CDSSs also support clinical management by improving adherence to guidelines, managing patients in research or treatment protocols, and facilitating cost containment through clinical interventions [SPB+20].
In addition to patient safety and clinical management, CDSSs contribute to administrative functions by supporting clinical and diagnostic coding, procedure and test ordering, and patient triage. They help with diagnostic support by assisting with clinical diagnosis, imaging, laboratory testing, and interpretation. In addition, the functionality of CDSS has expanded to include patient decision support integrated with personal health records (PHRs) to promote shared decision-making between patients and providers [SPB+20].
The benefits of CDSS include improved patient safety, adherence to guidelines,

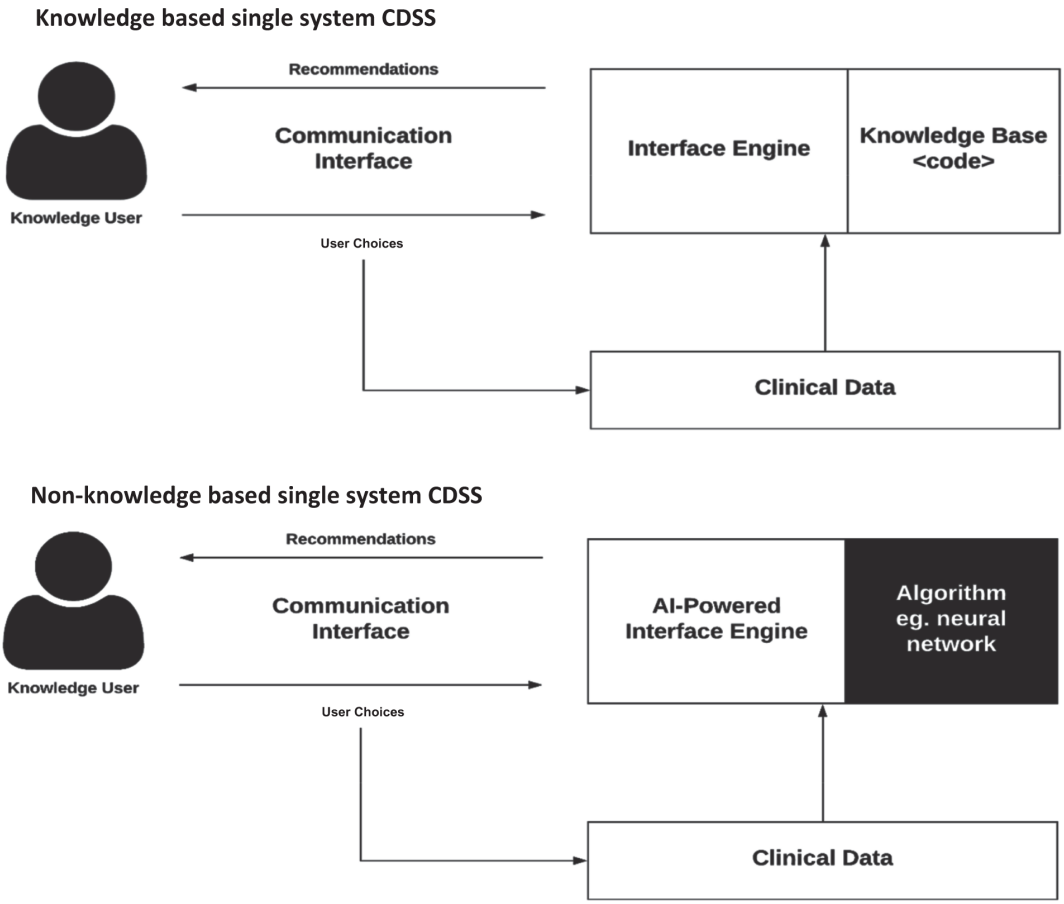

Figure 2.4: Diagram of key interactions [SPB+20]

cost containment, and improved diagnostic capabilities. However, the implementation of CDSS also presents some pitfalls and challenges. These include fragmented workflows, fatigue, impact on user skills, reliance on computer literacy, system and content maintenance, operational issues related to poor data quality, lack of transportability and interoperability, and financial challenges [SPB+20].

The adoption of CDSS has been encouraged by government legislation in the U.S., and several countries worldwide have made progress in adopting EHRs and national health records. Despite the challenges, CDSSs hold significant potential for improving healthcare delivery and decision-making, and efforts are ongoing to overcome their limitations and increase their effectiveness [SPB+20].

2.11 Requirement engineering

2.11.1 Introduction

Requirements Engineering (RE) is integral to the software development lifecycle because the foundation for developing successful software depends on first understanding the requirements. Requirements engineering involves a series of processes to capture requirements according to the needs and requirements of the users and stakeholders of the software product [uRKR13].
The goal is to gather high-quality requirements, analyze and document them, and then implement them in the software code appropriately to achieve the desired functionality and meet the users' needs [uRKR13].
Requirements engineering is a structured approach in which RE activities span the entire system and software development lifecycle. The RE process is iterative and aims to develop a high-quality product. Since requirements engineering is the cornerstone for all subsequent software development work, all key stakeholders must adhere to the exact requirements engineering process [uRKR13]. There are two broader categories of RE processes:

- Requirements Gathering (capturing, analyzing, specifying and validating requirements) [uRKR13]
- Requirements Implementation (executing the requirements in the software development activities) [uRKR13]

2.11.2 Requirement Gathering

In the requirements gathering phase, the effective RE process model consists of four phases [uRKR13]:

- Requirements elicitation and development [uRKR13]
- Requirements documentation [uRKR13]
- Validation and verification of requirements [uRKR13]
- Requirements management and planning [uRKR13]

The methods the requirements engineer chooses during requirements elicitation usually depend on available resources, time constraints, and the type of information sought [uRKR13].
There are five categories of elicitation techniques:

- **Traditional techniques** are generic data collection techniques that include questionnaires, surveys, interviews, task analysis, domain analysis, and introspection [uRKR13].
- **Cognitive techniques** refer to the knowledge/requirement elicitation used to collect and prioritize requirements. Some cognitive techniques include repertory grids, card sorting, laddering, and protocol analysis [uRKR13].
- **Group elicitation techniques** aim to understand better requirements through the involvement of teams or groups of software engineers. Group work, brainstorming, Joint Application Development (JAD) requirements workshops, and protocol analysis are some group elicitation techniques [uRKR13].
- **Prototyping** is also categorized as a modern elicitation technique used for elicitation when requirements are unclear, or urgent stakeholder feedback is required to proceed [uRKR13].
- **Contextual techniques** include ethnography, conversation analysis, and observation/social analysis, which are alternatives to traditional cognitive techniques [uRKR13].

User- & Job-Stories

User- & Job-stories are requirements-gathering techniques for understanding and documenting the needs, goals, expectations, and constraints of users, customers, and other stakeholders [LvdKD+18].
With the rise of agile software development, user stories have become increasingly popular. User stories often focus on the functionality and actions of a user and are usually structured as follows [LvdKD+18]:

"As a $<$ *persona* $>$, I want $<$ *function* $>$, so that (I can) $<$ *expected outcome* $>$."

They aim to focus on the problem space rather than the solution space. However, user stories are often formulated with a specific solution in mind, partly because some authors use them to describe functions [LvdKD+18].
Faced with this problem in the development of innovative software products, Alan Klement has introduced a new paradigm for requirements formulation called job stories, which relies on the following template [LvdKD+18]:

"When $<$ *situation* $>$, I want (to) $<$ *motivation* $>$, so that (I can) $<$ *expected outcome* $>$."

Job stories are based on the ideas of the Jobs-to-be-Done (JTBD) method of Christensen's "Disruptive Innovation Theory", a collection of principles that help to discover and understand the interactions between customers, their motivations, and the products they use [LvdKD+18].
Thus, they emphasize the motivational and situational contexts that drive user behavior rather than focusing exclusively on features and actions.

2.11.3 Requirement Implementation/development

Requirements engineering involves multiple activities in requirements development; therefore, no single process is available that is sufficient for all activities; however, a reasonable requirements engineering process model can be defined by intuition and background knowledge of the system under development. The RE process models are limited in number and have their specific application area [uRKR13]. The inputs and outputs of the RE process developed by Kotonia and Sommervile include the following five inputs:

- existing system information
- stakeholder needs
- organizational standards
- regulations
- domain information

It generates three outputs:

- agreed requirements
- system specification
- system models

This process is general and flexible as the requirements can only differ for all organizations, but these inputs and outputs remain fixed [uRKR13].

Chapter 3

Methods and Material

The following chapter describes the methods and materials used to develop the PM Patient Search Portal. This includes a description of the data models (section 3.1) on which the portal is based and the development process (section 3.3) using languages, libraries, and frameworks. The motivation for the chosen delivery method is also explained. The architecture (section 3.4) in which the portal was developed, and its decentralized integration with other sites are also covered.

3.1 Data Model(s)

Data models were designed in an iterative process with the various stakeholders for both the RDs and MTB modules. The basis for these data models is the information that can be obtained via the patient records.
As the CDSS is decentralized, data is obtained from different primary systems depending on the location. This data is available in a different format, which must be transferred to the schema of the data model.

Extraction-transformation-loading (ETL) tools often play an essential role here. This involves extracting and transforming data from heterogeneous operational data sources and transferring it to **data warehouses (DWs)**. DWs are complex computer systems whose main aim is to facilitate the decision-making process of knowledge workers [TLM03].

In the case of DNPM:DIP, the DW is part of the backend application and is available at every location.

Rules restrict the parameters of different patient records so that they can be compared and validated as best as possible. The set of values ideally is mapped using a coding system.

In most cases such coding systems are developed and published by the **World Health Organization (WHO)**. Some of the coding systems are domain-specific. However, coding systems are also relevant for both use cases and integrated into the framework of both data models. The following coding systems are examples of such cross-use systems.

- **ICD-10-GM** The ICD-10-GM is a German version of the "International Statistical Classification of Diseases and Related Health Problems" of the WHO, translated by the Federal Institute for Drugs and Medical Devices (BfArM). It is an adaptation to the specific requirements of the German healthcare system and is used for the coding of diagnoses in inpatient and outpatient healthcare [icda].
- **HGNC** The HGNC coding system represents a gene name and symbol (short-form abbreviation) for each known human gene. These representations must be approved in advance by the **HUGO Gene Nomenclature Committee (HGNC)** [hgn].
- **HGVS** The HGVS nomenclature is an internationally recognized standard for the description of DNA, RNA and protein sequence variants. It is used to specify variants in clinical reports and to share variants in publications and databases. The HGVS nomenclature is managed by the HGVS Variant Nomenclature Committee (HVNC) under the auspices of the Human Genome Organization (HUGO) [hgv].
- **ATC** The Anatomical Therapeutic Chemical (ATC) code is a unique code assigned to a medicine according to the organ or system it works on and how it works. The classification system is maintained by the WHO [atc].

3.1.1 RD

The MTB patient record, is structured as follows:

1. General
    - **Patient**
    - **Diagnosis (Condition)**
    - **HPO Term (Observation)**
2. Diagnostics
    - **NGS-Reports**

The following coding system is used exclusively with the RDs data model.

1. **ORPHA** The ORPHAcode, or the Orphanet Rare Disease Identifier, is a unique identifier assigned to each rare disease in the Orphanet database [orp].

3.1.2 MTB

The MTB patient record, which was defined on the basis of the Clinical Core Data Set (CDS) defined by the "MTB Harmonization" working group, is structured as follows:

1. General
    - Patient
2. Medical history
    - Diagnoses
    - Previous therapies and procedures
    - ECOG Performance Status
3. Diagnostics
    - Tumor-Specimens
    - NGS-Reports
    - Histology-Reports
    - IHC-Reports
4. Plans
    - Therapy recommendations
    - Study inclusion recommendation
5. Follow-up
    - Reimbursement claims and claim responses
    - Performed therapies
    - Response to therapy

The following coding systems, for example, are used in the context of the MTB data model.

1. **ICD-O-3** The ICD for Oncology, third edition (ICD-O-3), is an important classification for the coding of neoplasms. It was first published by the World Health Organization (WHO) in 2000 [icdb].
2. **RECIST** The Response Evaluation Criteria for Solid Tumors (RECIST) is a set of published rules that define when cancer patients tumors improve ("respond"), stay the same ("stabilize") or worsen ("progress") during treatment [rec].

3.2 Requirement analysis

Most of the requirements for DNPM:DIP were derived from previous requirements defined in the form of weekly meetings, user stories, and other methods for the existing BWHC system.
However, the user stories did not prove to be very target-oriented and efficient, as it was difficult for developers to grasp the users' problem and the context in which they touched on it. An attempt is now being made to capture these with the help of job stories, which have been described in section 2.11.2. In order to create a basis for future iterations, the following job stories were created:

1. **When I** prepare the MTB conference for a given patient, **I want to** be able to search for patients with similar characteristics in order **to** be able to define a promising therapy recommendation based on their course of therapy or therapy recommendations.
2. **When I** pursue a research question, **I want to** be able to search for patients with given characteristics in order **to** be able to statistically analyze the data from the resulting patient cohort.
3. **When I** am planning a study, **I want to** be able to search for patients with given genetic/diagnostic characteristics **to** see how many and which candidates are eligible as a study cohort.

Some of the collected requirements and the implementations derived from them are examined in this thesis's results section as an evaluation.

3.3 Development & Deployment

3.3.1 Languages

JavaScript (JS) and the superset **TypeScript (TS)** were used for all parts of the DNPM:DIP portal.

JavaScript

JS is a programming language that makes it possible to implement complex functions on websites [moz]. Initially, JS was created to "make web pages alive". JS can now be used not only in browsers, but also in other environments that have a JavaScript engine [jav]. The best-known JS engines are:

1. V8
2. SpiderMonkey
3. JavaScriptCore
4. Chakra

Basically, an engine parses the JS code, converts it to machine code and then executes the machine code [jav].
The V8 engine is used in the Chrome browser, for example. In addition to use in the browser, it is also used by Node.js, to execute JS on the server side.
This means that the file system can also be interacted with in the context of Node.JS, which is impossible in the browser.
The language core of JS is standardized by the so-called ECMA script and often has to be implemented individually depending on the engine

TypeScript

TS is a typed superset of JavaScript compiled to pure JS [tsV]. Although TS requires a considerable learning curve, it can ultimately lead to efficient and reliable code and is, therefore a valuable tool in modern web development [tsV].
In the context of this work, it was always preferred to pure JS programming where possible.

HTML

HyperText Markup Language (HTML) is the markup language used to structure web content and give it meaning, e.g. by defining paragraphs, headings and data tables or embedding images and videos in the page [moz].

CSS

Cascading Style Sheets (CSS) is a language with style rules that is used to design HTML content, e.g. to define background colors and fonts and to arrange the content in multiple columns [moz].

3.3.2 Libraries & Frameworks

Vue.js

Vue.js is a JS framework for creating user interfaces. It is based on standard HTML, CSS, and JS and offers a declarative and component-based programming model that helps to develop simple or complex user interfaces efficiently. Vue.js is a framework and ecosystem that covers most of the functions required in front-end development. Vue is designed to be flexible and incrementally customizable [vue]. Depending on the use case, Vue can be used in different ways:

1. Single-page application (SPA)
2. Server-side rendering (SSR)
3. Static page generation (SSG)
4. and much more

In the context of this thesis, Vue components and other building blocks are rendered on the server side (SSR) during the initial page request and then delivered to the client. All further interactions then take place on the client side (SPA). This is also referred to as a hybrid mode.

Nuxt

Nuxt is a higher-level Node.js web development framework for creating Vue applications. It was developed on the basis of Vue and equipped with additional features such as asynchronous data, middleware, layouts, modules, and plugins that can execute an application on the server side first and then on the client side. This means that the app is usually rendered faster than traditional server-side apps [Kok20].

Authup

Authup is an authentication and authorization system. It is designed to be easy and flexible, supporting multiple authentication strategies. It has a modular structure and consists of individual packages that can be integrated into an existing backend and frontend application. For example, it provides generic Vue components for resources such as users, roles, permissions, and more. Vue slots help display these in various ways.

3.3.3 Versioning

Versioning code and components is crucial for tracking changes in the development process and ensuring traceability. It also enables the consistent provision of software and ensures compatibility between different components and infrastructures.

Code

Git is used for versioning the code. The git commit messages follow the "conventional commits" convention, a lightweight convention on top of commit messages. The convention provides a simple set of rules for creating an explicit commit history, which makes it easier to build automated tools based on it [con]. This convention describes features, fixes, and changes in commit messages. Based on the rules, versions of individual components are incremented within the monorepos, which is how the portal is organized.

Components

For versioning the components **semantic versioning (semver)** is used. A version according to the semver specification is structured as follows: **MAJOR.MINOR.PATCH**. These parts should be incremented according to the following rules:

1. **MAJOR**: When incompatible API changes are made
2. **MINOR**: When new functionality is added in a backward compatible manner
3. **PATCH**: When backwards compatible bug fixes are made.

3.3.4 Release-Management

Release management is based on the GitHub action "release-please" from Google. This generates a pull request in which the versions of the individual components are incremented based on the commit messages since the last release. When the decision is made to publish a new release, the pull request is merged to trigger the package release as well as the Docker build and release.

3.3.5 Deployment

As already mentioned, a Docker image is created and published upon release. This is done because the platform is delivered to different locations, which ensures the consistent and reliable execution of the application in different environments. Additionally, the DNPM:DIP is not guaranteed to run as a standalone application on the target system. Therefore, it is crucial that the platform can be run in isolation, independent of other applications, without affecting their execution. Overall, deployment with Docker offers a flexible, consistent, and efficient method of deploying and managing software, which is why this technology was chosen.

3.4 Architecture

DNPM:DIP is a platform that is operated redundantly at several locations. It consists of several components. Firstly, the **portal** (section 3.4.3), which was designed and developed as part of this work. Secondly, a **resource server** (section 3.4.1) and the **auth server** (section 3.4.2).

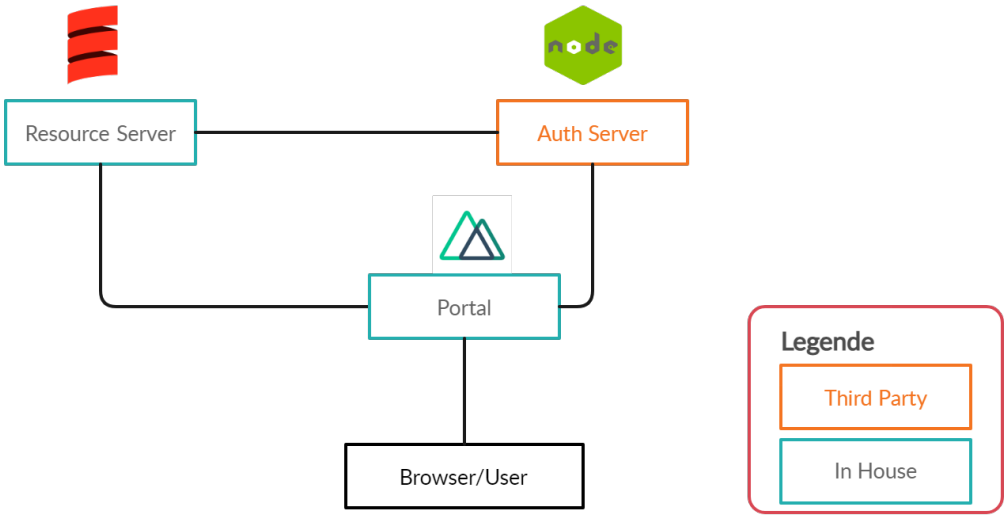

Figure 3.1: DNPM:DIP - Architecture

3.4.1 Resource-Server

The resource server comprises various components containing application logic for different usage contexts. The data validation service checks and saves the transferred data records. If data quality problems are detected, a report is

generated and saved. Otherwise, the data record is forwarded to the query component. The query/report component represents the core area by storing validated data records and providing functions for queries, reports, and query session management. The orchestration of federated queries also takes place here [LC23].
The resource server checks incoming requests from a user for authentication and authorization by the OAuth2 specifications using the auth server (section 3.4.2). In addition to the components described above, the resource server also provides other components that are not explained in detail in this paper.
The resource server is implemented according to the architectural approach of a "modular monolith", whereby the separate components are bundled within a higher-level component that acts as an **Application Programming Interface (API)** gateway. It opens the underlying functionality via corresponding REST APIs. The implementation is carried out in Scala using the Play Framework [LC23]. As the service was developed as part of the DNPM:DIP project, it is referred to as an in-house service in figure 3.1.

3.4.2 Auth-Server

The auth server is a central component of the architectural structure. The "Authup" (section 3.3.2) authentication and authorization system was selected for this crucial role. This system manages resources such as users, roles, and robot accounts. A vital functionality used in the project is verifying and validating users and robots to ensure they are authorized to access the appropriate resources. The authentication server plays a central role in the authentication and authorization process. It supports protocols such as OAuth2, OpenID Connect, and LDAP, allowing the platform to connect to different identity providers at each site.
Although the author of this work also developed the system, this was done independently of the period of this work or the DNPM:DIP project. For this reason, it is listed as a third-party service in figure 3.1.

3.4.3 Portal

Together with the auth (section 3.4.2) and resource server (section 3.4.1), the portal acts as the local node of the DNPM:DIP platform. Its tasks are to display the user interface in the browser and manage routing. To manage resources, the application interacts with the authentication and resource server via an Application Programming Interface (API).
The frontend and backend are separated in order to clearly differentiate the responsibilities between the user interface (frontend) and data processing and storage (backend). This enables independent scaling of frontend and backend

applications and the use of different technologies. This separation also helps to increase the security of the platform. Furthermore, a considerable amount of business logic is outsourced to the browser, which relieves the backend applications and the associated servers.

3.5 Infrastructure

DNPM:DIP is subject to several restrictions, including data protection. It has been stipulated that each location retains complete control over its data and that it may only be persisted locally. Although the data is pseudonymized and can be deleted, it should not be stored elsewhere. The platform must, therefore, be operated decentrally.
Thus, the components described above are available at every location. This collection of components is referred to below as a "node." These nodes are connected via the broker (see figure 3.2). The broker exists independently of the respective ZPM. A request can be made to each node via the entire network. The data is transferred from the other nodes, whereby it is only temporarily loaded into the working memory of the requesting node and aggregated.
Each node is operated in a ZPM of the respective location. Figure 3.1 shows

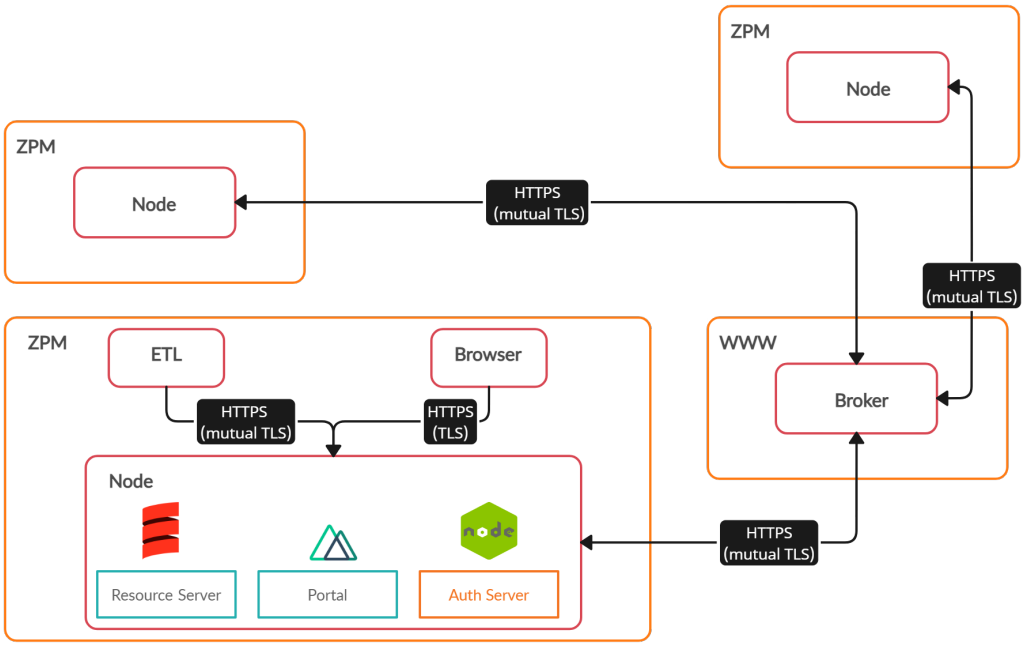

Figure 3.2: DNPM:DIP - Infrastructure

the interaction between the various parties in simplified form. Communication between the individual nodes takes place using mutual TLS encryption,

whereby mutual authentication occurs before a connection is established and data can be exchanged. This principle applies equally to the connection between the ETL pipeline and a node. Only the basic TLS protocol is used for communication between the browser and a node, whereby only the node authenticates itself to the browser.

3.6 Implementation

3.6.1 General

As already mentioned, the portal of DNPM:DIP is structured as a monorepo, which provides a central code base for different components and thus enables code reuse, common dependencies, and a consistent development environment. This leads to improved code organization, easier maintenance, and increased efficiency in development and deployment. The portal's components are described in more detail below.

1. core
2. kit
3. rd (abbreviation for rare diseases)
4. mtb (abbreviation for molecular tumor board)
5. portal

3.6.2 Packages

1. core

The "core" package is a crucial portal component and is used by all other packages. It contains practical tools that are required at various points in the application. It also contains structures of resources such as code systems, codings, and patients that are relevant for all use cases. This ensures correct and consistent use of backend content, minimizes errors during development, and facilitates debugging. The development experience is improved by features such as auto-completion, type checking, and error detection in IDEs. It also helps other developers in the project to understand the data structure between the frontend and the backend.
Each top-level resource interacted with via the REST API is abstracted by a class that maps the corresponding operations via methods. These classes use TS constructs for return values and parameters and are then merged into a primary class. The primary API client class is inserted into the API clients of

the respective use case, such as RDs or MTB, so that redundant information, such as the base **Uniform Resource Locator (URL)**, does not have to be defined multiple times. After a successful login, the token received is set as the default value for subsequent requests. This token can also be acquired from the cookie memory if it was previously acquired.
On top of that the core package provides Vue components that can be used across applications. One example is the "DCodeSystem" component, which loads and displays a specific code system of the API. As the business logic is identical in different places and only the display varies, the aim was to define this component once and not multiple times for each display method. This was achieved using the Vue "Slots" feature.
In addition to the points mentioned, tools for error handling, layout rendering, etc. are also exposed.

2. kit

The "kit" package acts as a tool for integrating use cases (e.g., RDs) into the portal. Nuxt enables the customization of almost all framework aspects using configurations and hooks. Hooks represent a concept that makes it possible to interrupt execution at discrete points in time and change the subsequent behavior. There are two different types of hooks:

1. **Build hooks** allow to interrupt the execution of code during the application's build process and change its behavior.

2. **Runtime hooks** allow to interrupt the execution of code during the application's runtime and change its behavior.

Each use case is designed as a Nuxt module executed sequentially during the project build. Modules can be used to encapsulate logic. The following concepts, for example, are realized in this way:

1. **Pages**: The page components in the "pages" folder of the respective use case are scanned and registered via a build hook under a defined path in the portal (e.g., "/mtb"). The internal mechanism of Nuxt was replicated for this, as it was not disclosed and did not fully meet the requirements.

2. **Navigation**:A plugin is created on the fly for each use case that registers top-level and side-level navigation elements in the portal navigation using runtime hooks.

In addition to modules, Nuxt offers a concept called "layers" which could have fulfilled the requirements. However using the concept, pages can only be registered under a different subpath (e.g. /mtb) with having to adapt the folder structure. The second problem was that components in the components folder were added to the global vue components register. This would have led to conflicts if components with the same name were used in different applications.

3. rd & 4. mtb

The "rd" and "mtb" packages have similar structures and are, therefore, briefly explained together. Both packages are modules inserted into the portal using the "kit" package (see section 3.6.2). During the application's runtime, they access the portal's main API client and use this as the basis for their individual API client implementations. This process has already been outlined in the description of the core package (see section 3.6.2). However, the essential part of these modules is the components. These represent either page components, if they are stored in the "pages" folder, or specific areas within a page component. They are independent and reusable units that consist of HTML, CSS and JavaScript code and map specific functions and displays within the application. A search component, for example, consists of a dozen other components.

5. portal

The portal is the only component within the monorepo that can be regarded as an independent service or application. Interaction with the portal from the user's perspective begins with a request from the user. The integrated Vue Router determines which page component is to be displayed. However, plugins and middleware are loaded before the page component and associated components are rendered. Plugins are a mechanism for integrating user-defined functions or third-party libraries into the runtime context. In the context of the portal, for example, the following plugins are relevant:

1. **APIClient**: Here, the API client of the "core" package is initialized, injected into the runtime context, and provided globally.

2. **Chart.js**: Chart types and plugins of the Chart.js library are registered in this plugin.

3. **Bootstrap**: This plugin registers Bootstrap components and plugins in the application. Bootstrap is a framework for frontend development.

In addition to plugins, the middleware concept is a vital application component. This mechanism allows for the definition of restrictions or manipulations of the request before rendering a page component. It is used for two essential tasks:

1. **Authentication & authorization**: Here, specific page components are restricted depending on the login status and the user's authorizations. These restrictions can be defined using meta information within a page component.
2. **Layout**: With this middleware, depending on the URL, the corresponding navigation element is highlighted in the top navigation, and the side navigation that depends on it is displayed. As with the page components, certain navigation elements can be hidden depending on the user's login status and authorizations.

Chapter 4

Results - Realization

In this section, the visual realization of the portal is explained, and the underlying concepts that helped meet the users' needs are outlined. First, a general portal description is provided (section 4.1). The presentation of the administration (section 4.2) area, the MTB (section 4.4) and RDs (section 4.3) areas are then examined in more detail. In addition to explaining the various views, the objectives pursued during implementation are also described. In the MTB use case context, the implementation is evaluated using a user survey in chapter 5.

4.1 General

When designing the portal, it was essential to consider various concepts to create a user-friendly and effective user interface. For this reason the design is focused on user needs and behavior. Clear structuring of the **User Interface (UI)** elements allows users to quickly navigate and find the information they need. The UI should be minimalistic and not appear cluttered to reduce visual impact and improve the user experience. A responsive design ensures the portal works well on different devices and screen sizes. A sensible color scheme, with different colors and typography, contributes to readability and visual hierarchy. Those concepts help to meet the user's needs and provide a positive user experience.

4.1.1 Layout

The portal is structured as follows: The top navigation is located at the top of the page and displays menu items that either lead directly to a specific page or display a dedicated page navigation when clicked. This makes it easy to navigate between different areas of the portal. At the bottom of the page is a minimalist footer with a copyright notice (see figure 4.2).

The layout is based on a two-column design commonly used on websites. It consists of a header, a footer, and two columns in the content area. One column is intended for the main content, while the other serves as a sidebar.
The structure of the content area extends to the header and footer, allowing the main content to be further subdivided depending on the page component. This flexibility makes it possible to adjust the number of columns for different views or use cases without the user having to recognize the subdivision at different levels.
It is also important to note that the menu elements in the top and side navigation can vary depending on the login status and authorization of the user. This helps to ensure that users can only access the functions they are authorized.

4.1.2 Home

The start page sometimes shows the registered use cases, currently only RDs and MTB. These are displayed below the graphic in the main content. Clicking on the text takes visitors to the corresponding use case. Alternatively, this is also possible via the corresponding menu element in the top navigation.

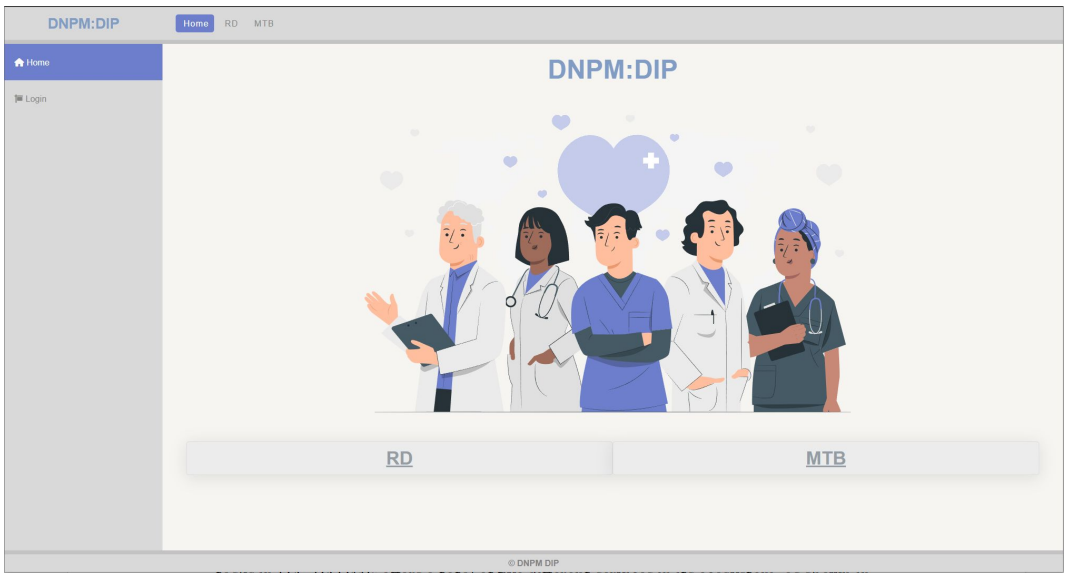

Figure 4.1: Portal - Home Page

4.1.3 Login

On the login page, at the top of the main content, there is a form for logging in with a name/email and password. In addition to logging in with a local user account managed by the Authup service, other authentication strategies based on OAuth2 or LDAP can also be defined in the admin area. More on this later in section 4.2.

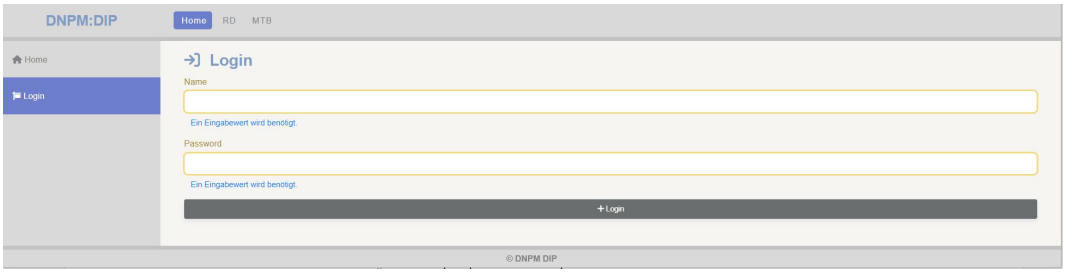

Figure 4.2: Portal - Login Page

After a successful login, two elements appear at the top of the page on the right-hand side. One is the username display; the other is a logout button.
A real-time countdown is displayed in the lower area of the sidebar, showing how long the session is still valid. Based on information collected during the login process, the session will automatically be renewed shortly before it expires.

4.2 Admin

The admin area contains functions that regular users do not typically interact with. These are described below for the sake of completeness and to explain all the portal functions.

4.2.1 Identity Provider(s)

The first menu item in the admin area is identity provider management. Initially, the overview page is displayed, listing all previously defined identity providers. These can be modified or deleted, provided the corresponding authorizations are available. In addition to the overview page, a new identity provider can also be defined. An identity provider can either be created based on a predefined preset, in which some attributes for the configuration are already predefined, or a protocol such as LDAP, OAuth2, or OpenID Connect can be selected. However, the respective forms will not be discussed further in this paper since doing so would exceed the scope of the work.

4.2.2 User(s)

Local users can be managed under the "users" menu item. Both user accounts created manually and user accounts created using an identity provider's login flow can be edited here. After creating or editing a user, general information can be edited. In addition, individual authorizations or roles can be assigned in the user view tabs. The authorizations that a user has are either explicitly assigned authorizations or are assigned via a role assignment.

4.2.3 Role(s)

Roles are the last top-level resource that can be managed in the admin area. Similar to managing identity providers and users, roles can also be managed here on an overview page. In addition to user-defined roles, there is an admin role that was created by default and to which the default user and robot are assigned. This role cannot be deleted. Like the user view, individual permissions or users can be assigned in the roles tab view.

4.3 RD

The RDs module was the first module to be planned and implemented as part of the project. The search, which can be found in the representative area of the portal, serves as the entry point to the module.

4.3.1 The Search

The search screen, which is shown in Figure 4.9, offers numerous filter options to limit the number of results. These filters were elected to meet the requirement of finding patients with similar characteristics.

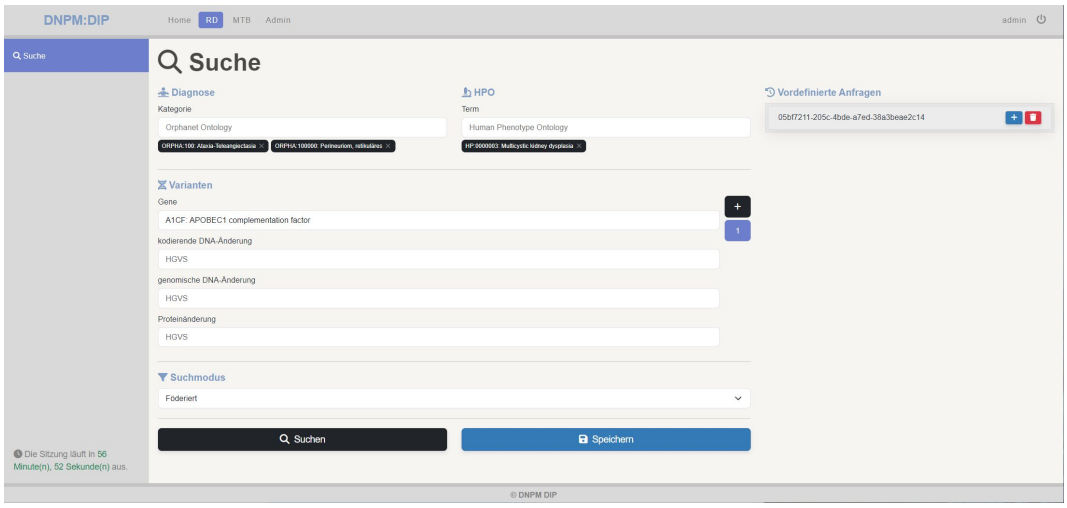

Figure 4.3: Portal - RDs Search Page

1. **Disease category**: This filter makes it possible to select several diagnosis categories based on the Orphanet catalog using a text-based search (see section 3.1.1).

2. **HPO**: The HPO filter makes it possible to refine the search based on phenotype characteristics. HPO is a standardized terminology that describes phenotypes (observable characteristics) associated with genetic diseases.

3. **Variant**: This filter consists of a group of sub-filters. Between 0 and 6 groups can be defined. In the first sub-filter, a "gene" can be selected based on the HGVS nomenclature. This refers to a DNA change that can be selected based on a text search in the nomenclature. In addition, further sub-filters such as "coding DNA change", "genomic DNA change" and "protein change" can be used to specify changes based on the HGVS nomenclature.

The result set of the search can be further reduced after submitting the search using so-called patient filters, but more on this in section 4.3.3 and section 4.3.4.
In addition to submitting the search, the selection of filters can also be saved as a prepared query. The search form can always be restored from these saved search queries. The saved search queries are displayed next to the search filters as shown in Figure 4.9.
An essential aspect of the search is that it can be sent to all nodes or only to the local node. The drop-down field above the search button allows users to specify this.

A so-called "query session" is created when the search is executed. Data from the local or all nodes is aggregated for the selected filters for a specific duration. At the end of the session, the temporarily aggregated data is discarded.

4.3.2 Query

After the search has been carried out, a query session is created as described above. The query session view is divided into three different views: 1) "Overview", 2) "Patients" and 3) "Customize", which are described beneath. For the entire query session, the results can still be reduced using so-called patient filters, located on the right-hand side of the view and available in all three views.

4.3.3 Query Summary

The query summary is the initial view (see figure 4.4) loaded when the search is started.

Figure 4.4: Portal - RDs Query Summary

It is divided into two parts, both are intended to give the user a quick overview of the distribution of various properties and help define a promising treatment recommendation.

1. **Demographics**: In this sub-view, the distribution of patients per location regarding gender and age is displayed using pie charts and bar charts (see figure 4.4).

2. **Diagnostics**: This sub-view displays the overall distribution of diagnostic categories and HPO terms. The distribution of diagnostic categories

and HPO terms can also be broken down by a specific variant in the gene (figure 4.5).

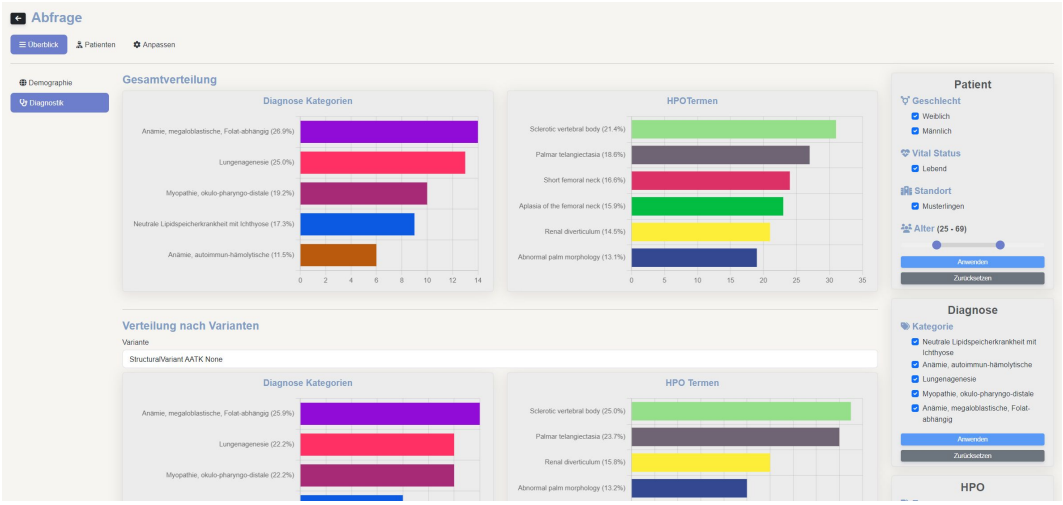

Figure 4.5: Portal - RDs Query Summary Diagnostics

4.3.4 Query Patients

In addition to the query summary above, the second tab, "patients," offers a view (see figure 4.6) of all patients who match the search criteria. The respective patient card contains general information about the patient, such as location, gender, age, vital status, and the matched search criteria.

Figure 4.6: Portal - RDs Query Patients

With the help of this list, the user should get an overview of a possible cohort for a study. It is also possible to switch from this view to a full view of a patient. The dedicated view is also described in the following section 4.3.5.

4.3.5 Query Patient

Like the query session, the patient record view is divided into two views: 1) "Overview" (see figure 4.7) and 2) "Diagnostics" (see figure 4.8). The patient view is a visual representation of the patient file and is intended to provide the user with more detailed information about a patient and thus facilitate case preparation.

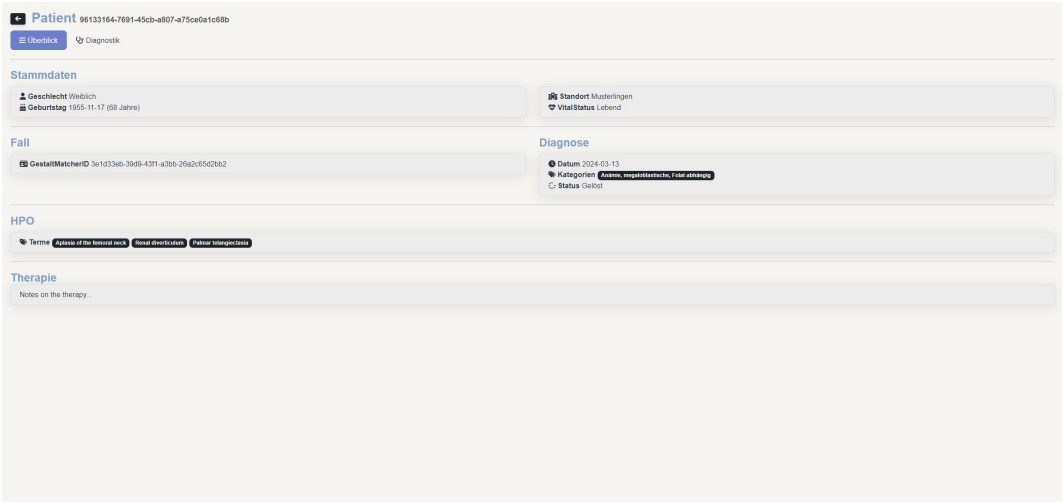

Figure 4.7: Portal - RDs Query Patient

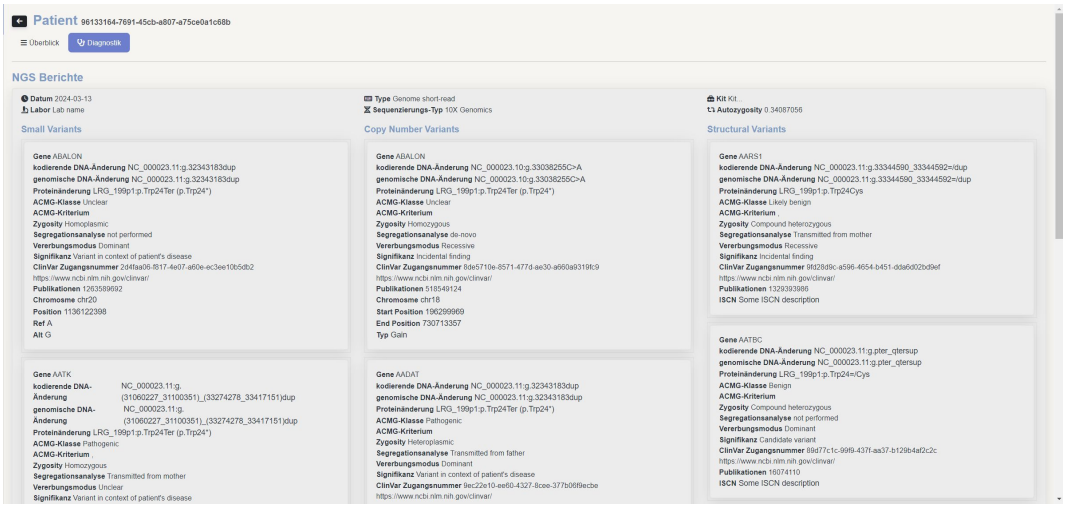

Figure 4.8: Portal - RDs Query Patient Diagnostics

4.4 MTB

The MTB module is the second module planned and implemented as part of the project. Like the RDs module, the search view is the entry point for interacting with it.

4.4.1 The Search

The search screen shown in Figure 4.9 is similar to the RDs module's but offers significantly more filter options to limit the result set. The reason for this is that more such criteria are relevant. The filters described below were chosen to find patients with similar characteristics.

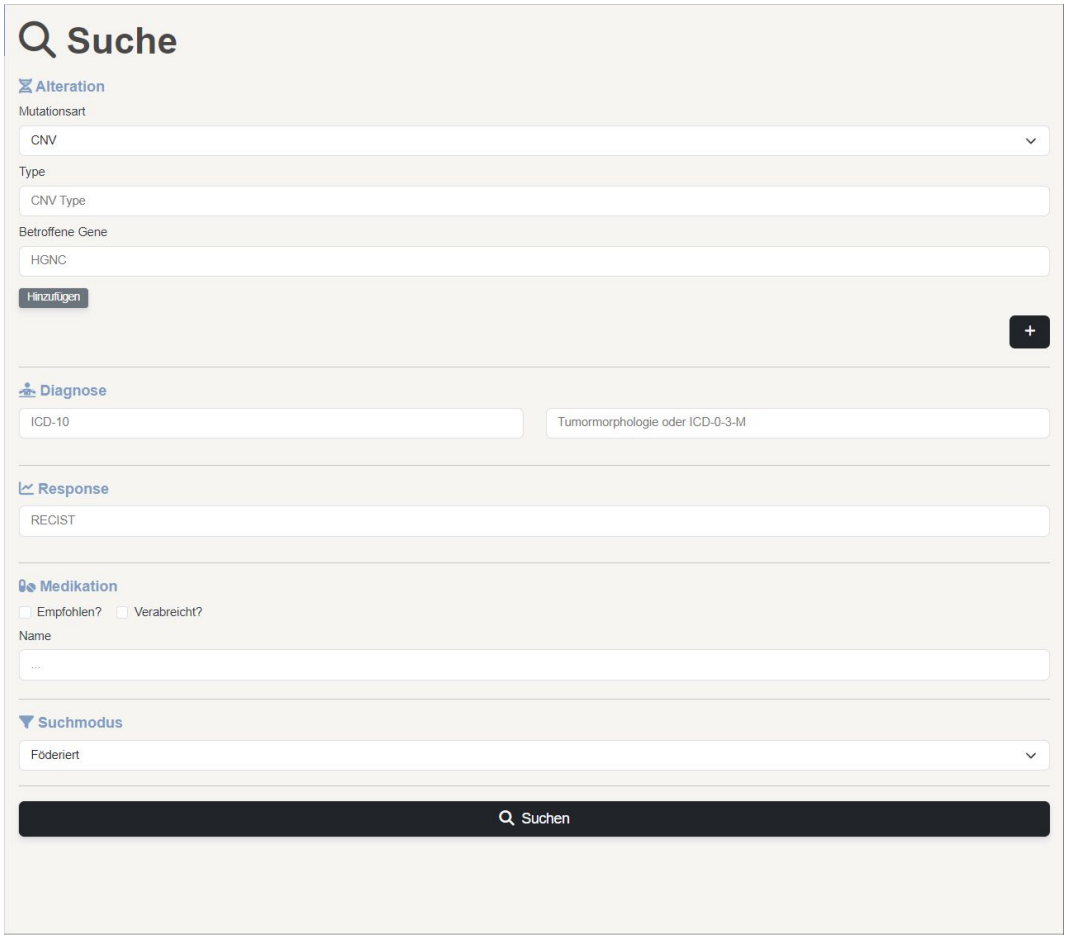

Figure 4.9: Portal - MTB Search Page

1. **Mutation Type**: This filter consists of a group of sub-filters. 0 and 6 groups can be defined. The sub-filters vary depending on the mutation type. It is possible to choose the following types:

   (a) The **SNV** mutation type refers to a single nucleotide change. In the first sub-filter, a variation in the gene can be selected based on the values of the HGNC catalog using a text search. In addition, the amino acid sequence of the encoded protein and the specific DNA sequence change can be selected based on the HGVS nomenclature.

   (b) The **CNV** mutation type describes the changes in the number of copies of a specific DNA section. The intensity of the genetic change can be selected in the first sub-filter. In the second sub-filter, any affected genes can be selected based on a text search in the HGNC catalog.

   (c) For the mutation types **DNA fusion** and **RNA fusion**, mutations in a gene can be specified in two sub-filters based on the HGNC catalog with 3' and 5' ends.

2. With the **diagnosis** filter, a diagnosed disease can be specified based on the ICD-10-GM classification for diseases and related health problems or based on the ICD-0-3M classification specifically for cancers with a focus on morphology, which is based on the results of the histological examination.

3. Using the **response** filter, the patient cohort can be filtered based on the tumor's response to treatment. Several categories can be selected here based on the RECIST catalog.

4. In the last filter, administered or recommended **drugs** can be filtered using a text search based on selections from the ATC catalog for pharmacological agents.

Similar to the RDs module, the search can be executed federated or locally. A query session is also created after the search is initiated.

4.4.2 Query

After the search has been submitted, a query session is created the same way as before. The query session view is divided into three different views: 1) "Overview", 2) "Patients" and 3) "Customize", which are described below. For the entire query session, the results can still be modified using so-called patient filters, located on the right-hand side of the view and available in all three views.

4.4.3 Query Summary

The query summary is the view that is initially displayed. It is divided into four parts to give the user a quick overview of the distribution of the various properties. The view is divided as follows:

1. **Demographics**: In this sub-view, the distribution of patients per location regarding gender and age is displayed using pie charts and bar charts (figure 4.10).
2. **Diagnostics**: This sub-view shows the overall distribution of diagnostic categories based on the ICD-10-GM and ICD-O-3M classifications. The distribution of diagnostic terms can also be broken down by a specific variant in the gene (figure 4.8).
3. **Medication**: This sub-view shows the overall distribution of recommended and administered medications (figure 4.12). Based on the supporting molecular alteration, explicitly recommended medications can be viewed. (figure 4.13). This also applies to the tumor's response to the treatment. In addition to these displays, there is a visualization of the average duration for which a drug was administered (figure 4.13).
4. **Survival report**: In this sub-view, the overall survival is shown without grouping and instead organized by tumor entity (figure 4.14). In the last view, the progression-free survival is shown as grouped by therapy (figure 4.15). All charts are based on Kaplan-Meier.

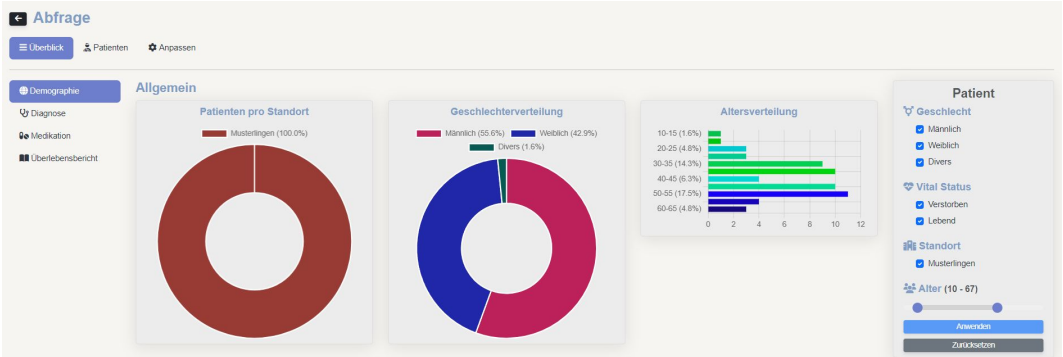

Figure 4.10: Portal - MTB Query Summary

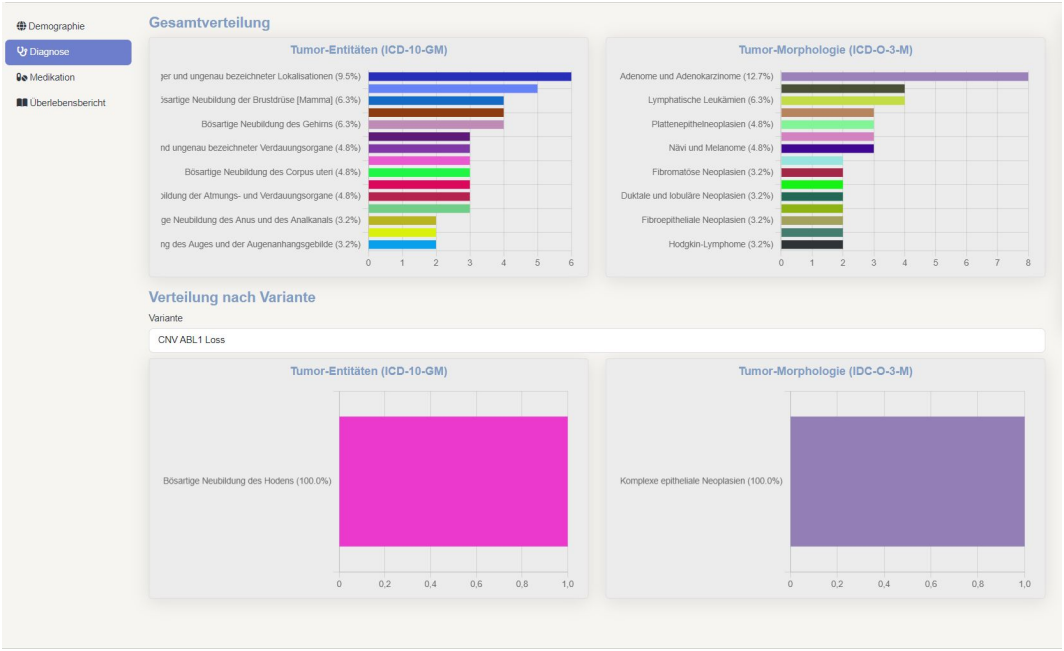

Figure 4.11: Portal - MTB Query Summary Diagnostics

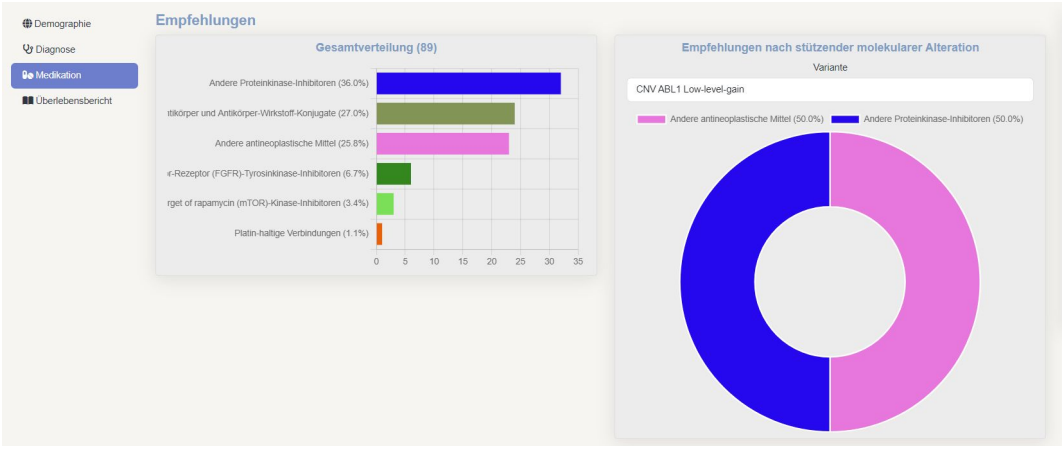

Figure 4.12: Portal - MTB Query Summary Medication #1

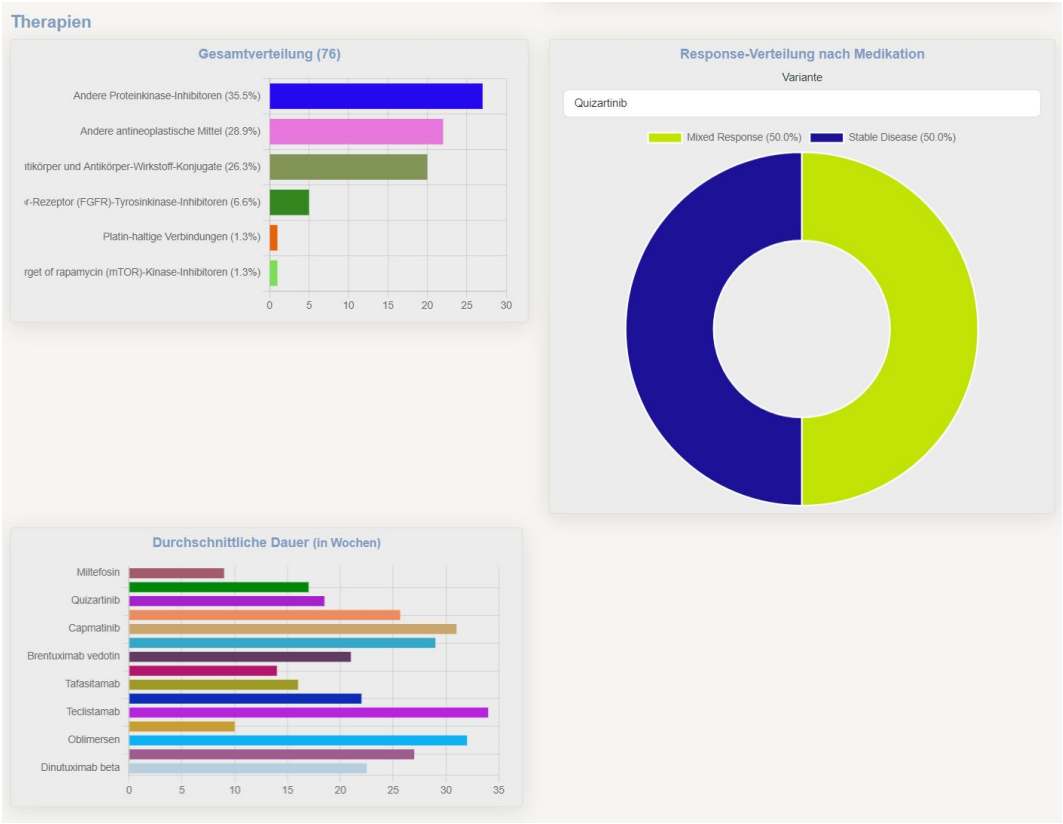

Figure 4.13: Portal - MTB Query Summary Medication #2

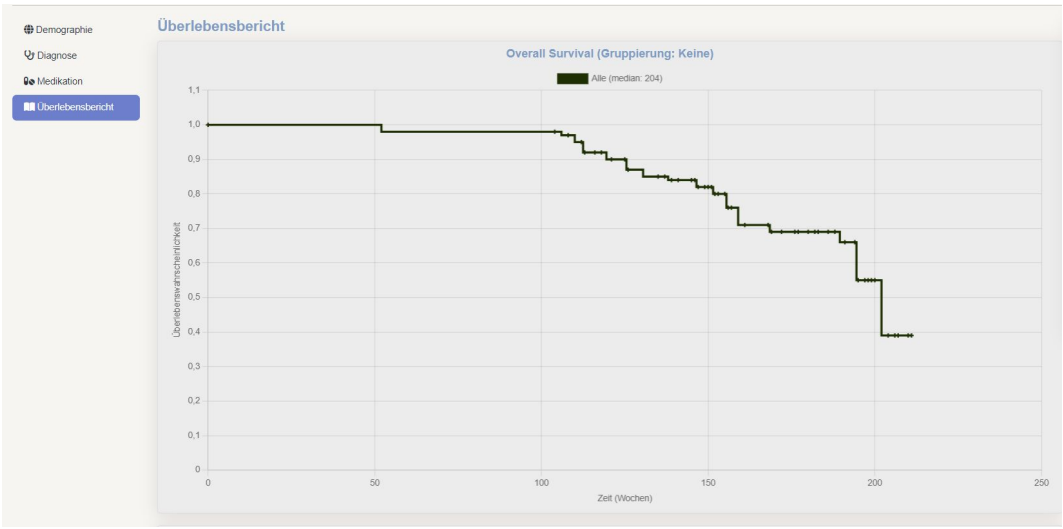

Figure 4.14: Portal - MTB Query Summary Overall Survival

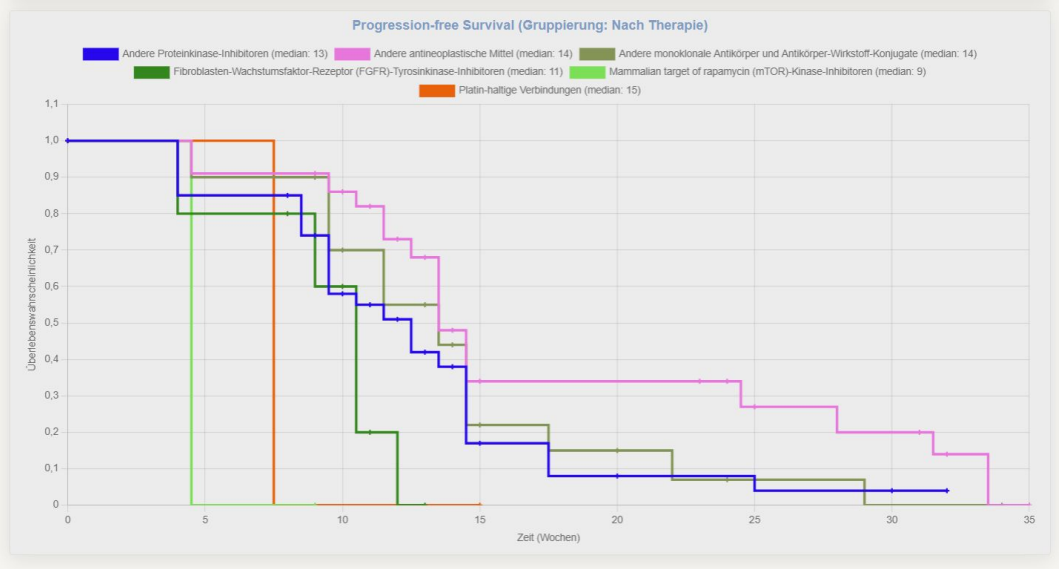

Figure 4.15: Portal - MTB Query Summary Progression-free Survival

4.4.4 Query Patients

In addition to the previous overview display of the query, the second "patients" tab presents a tabular display (see Figure 4.6) of all patients who match the defined search criteria. Each patient overview card provides basic information about the patient, including location, gender, age, vital status, and relevant search criteria.
This list provides the user with an overview of potential cohorts for scientific studies. It is also possible to navigate from this view to a specific patient view, which is explained in more detail in section 4.4.5.

4.4.5 Query Patient

Similar to the query session, the patient view is also divided into four separate sections: "Overview" (figure 4.16), "Diagnostics", "Treatment decisions" (figure 4.17) and "Follow-up". The patient view is a graphical representation of the patient record. It aims to provide the user with comprehensive insights into a patient's information to facilitate the preparation of treatment cases.

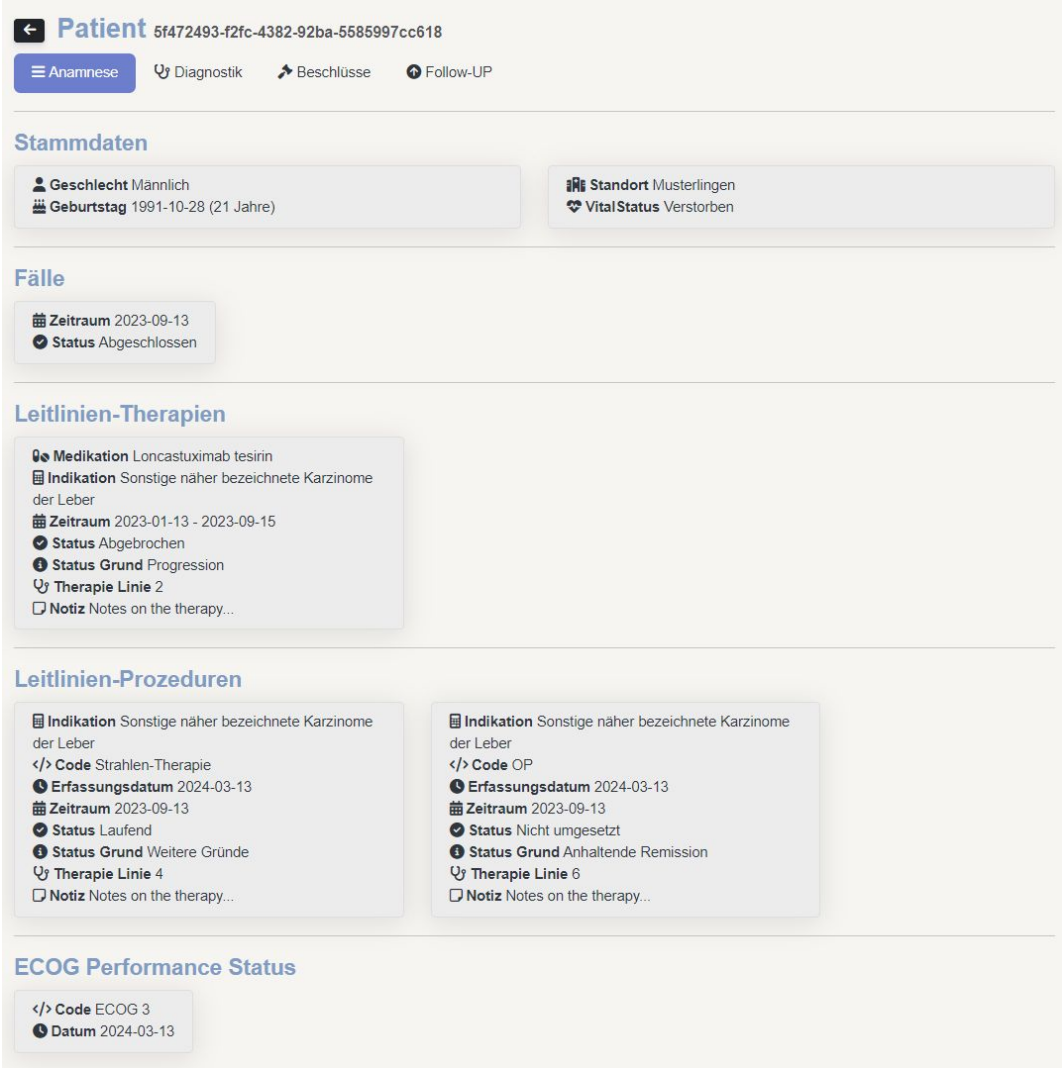

Figure 4.16: Portal - MTB Query Patient

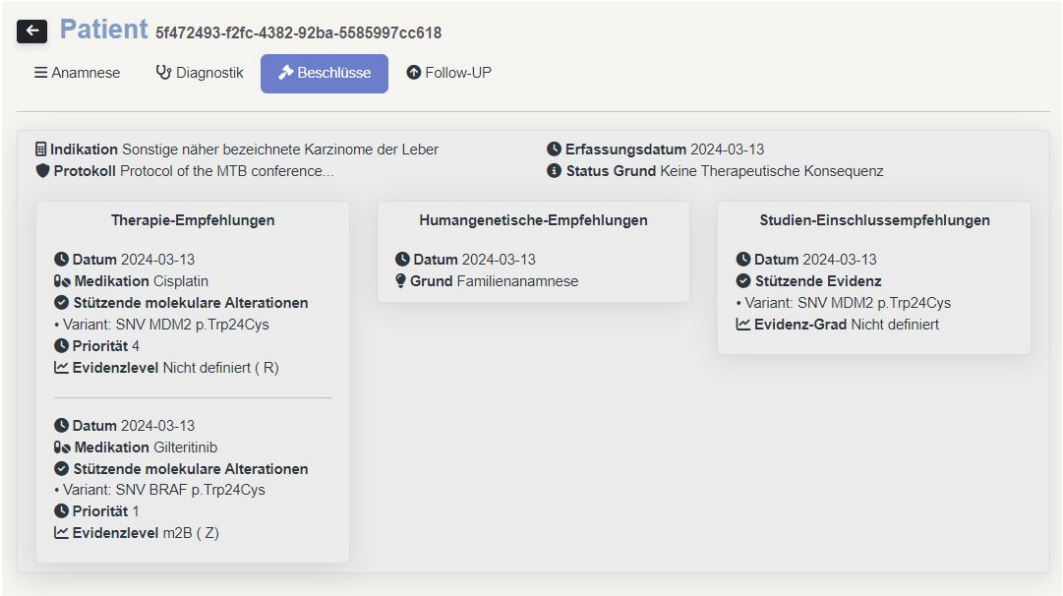

Figure 4.17: Portal - MTB Query Patient Treatment

Chapter 5

Results - Evaluation

In the following section, a user survey has been conducted to evaluate to what extent the implementation of the previously identified requirements successfully met the users' needs and expectations. The focus of the user evaluation is on the graphical visualization of the aggregated patient data. First, the structure and framework of the survey conducted are described. The questions asked as part of the user evaluation are then listed. In chapter 6, Discussion, the survey results and implementation are discussed in detail.

5.1 Setup

A test environment for the platform was set up at the hospital for user evaluation. This test environment is based on synthetic patient records, which means it can be used for the project without any concerns. Only doctors or people who will later actively use the platform participate in the user evaluation. This includes doctors who carry out the case preparation for an MTB conference. Participants could participate in the user survey on the portal using the online survey tool LimeSurvey.

5.2 Result

The questions asked as part of the user evaluation can be divided into three main categories: 1. search view (section 5.2.1), 2. results view (section 5.2.2), 3. general aspects (section 5.2.3) and 4. System Usability Scale (SUS) (section 5.2.4). Each of these categories is subdivided into up to two subcategories. The first subcategory contains questions that can be rated on a scale (scale) from 1 (strongly disagree) to 5 (strongly agree). In contrast, the second subcategory contains questions that can be answered with a free text (free text).

One focus was on the graphic presentation of patient information. It was investigated whether these representations offer added value for the user and support them in their work.
Due to the domain-specific nature of the system, only certain people could take part in the survey at the time. The survey could not be carried out earlier as work on the portal was still pending. Regrettably, only four people took part in the survey. For this reason, questions that could be answered on a scale were given the standard deviation (SD) and the average (Avg) to present the most meaningful results possible.
The user evaluation questions and their corresponding results are listed below. However, only the results for the quantitatively assessable questions were considered. The answers to the text questions are covered in Chapter 6.

5.2.1 Search view

Scale

Question	Avg	SD
How easy is it to define a search for patients with certain characteristics?	4	0
Are all relevant/necessary search parameters covered by the search form?	3.75	0.43

Table 5.1: Results - Evaluation: Search View

Text

1. Could you imagine that the process or usability for submitting a search could be improved, and if so, how?

2. What additional search parameters or filters would you like to see in order to further refine the search for patients?

3. What suggestions do you have for improving the user-friendliness of the search form?

5.2.2 Result view

The following category refers to the views of the query summary, the queried patients, and the individual queried patients that can be displayed after a search.

Scale

Question	Avg	SD
1. How would you rate the overall quality of the graphical representation of the patient cohort?	4	0
2. How helpful is the presentation of age distribution, gender distribution and distribution per location?	4	1
3. How meaningful is the representation of the distribution based on tumor entities and tumor morphologies?	4.25	0.82
4. How well can medication recommendations be recognized according to a specific supporting molecular property?	3.5	0.5
5. How clear is it which medication has been prescribed?	2.5	0.5
6. Can you easily see what the response was to a particular medication?	4.75	0.57
7. Does the visualization of "Overall Survival" of patients help?	3.25	0.43
8. Does the visualization "Overall Survival" grouped by tumor entity help?	3	0.7
9. Does the visualization "Progression-free Survival" grouped by medication help?	3	1
10. How helpful is the patient list to navigate to a specific patient in the detailed view?	2.25	1.08

Table 5.2: Results - Evaluation: Result view

Text

1. What other visualizations of the patient record do you think could be useful?
2. Is there any specific data or information that you feel is missing from the current visualizations?
3. Are there any areas of the user interface that you find confusing or unnecessarily complicated?

5.2.3 General

The following category contains general questions independent of a specific portal area or a metric.

Scale

Question	Avg	SD
1. How easy is it to navigate the portal?	4.5	0.5
2. How intuitive are the functions of the system (e.g. search)?	3.5	0.5
3. How quickly can required information be found?	3.25	0.43
4. How would you rate the user-friendliness of the system?	3.5	0.5
5. How well does the portal support you in case preparation?	2.5	0.5

Table 5.3: Results - Evaluation: General

5.2.4 System Usability Scale (SUS)

The following category contains questions for determining the System Usability Scale (SUS), a method for evaluating the user-friendliness of systems. The SUS, developed by John Brooke in 1986, is described as a reliable and efficient method for evaluating the usability of systems, as it is fast, inexpensive, and statistically valid. Compared to other tests, the SUS is cheaper and faster but still provides valid results [Tho].
Based on the answers to the survey, an SUS score of 70.62 was calculated. This value is just above the average SUS score of 68 [Tho].

Scale

Question	Avg	SD
1. I think I would like to use the portal frequently.	3.25	0.43
2. I found the portal unnecessarily complicated.	2	0
3. I found the portal easy to use.	3.25	0.43
4. I think I would need the support of a technical person to use this portal.	1.75	0.82
5. I found the various functions on the portal well integrated.	4.25	0.43
6. I thought the portal was too inconsistent.	2.25	0.43
7. I would imagine that most people would learn to use the portal very quickly.	3.25	0.86
8. I found the portal very cumbersome to use.	2.25	0.06
9. I felt very confident using the portal.	3.5	0.5
10. I had a lot to learn before I could get going with this system.	1	0

Table 5.4: Results - Evaluation: SUS

Chapter 6

Discussion and Outlook

This chapter discusses the implementation and user evaluation results in detail. It also discusses the developed solution's limitations and potential weaknesses. Finally, an outlook outlines possible improvements. The chapter also highlights the opportunities and possibilities that the portal can offer for medical care and PM in general.

6.1 Discussion

The developed portal presents a modular ecosystem that enables the integration of various use cases. Modules can be used to integrate almost any functionality into the portal. Nevertheless, careful examination of third-party modules is essential to prevent potential security risks, such as the introduction of malicious code into the application context.
The presentation of resources obtained via the API is done exclusively through interfaces that do not provide validation or security at runtime to ensure that the resources received fulfill the expected schema. Implementing schema validation when receiving resources could be beneficial, even if this could slow down the application speed due to the resources' often complex and nested structures.
Another important aspect is that the portal is currently only available in German. Many researchers and clinicians still need to be fully proficient in German so that a multilingual portal design could be beneficial. The survey confirms this, as participants responded to the text questions with English sentences.
In addition, a central focus was on the ecosystem and architecture to create a solid foundation for future implementations. The aim was also to provide a clear, simple, and uncluttered presentation.
In summary, the portal offers a robust ecosystem with numerous generic

components and tools, which facilitate and promote the integration of new modules and further development of existing modules.

The user survey results show that certain functionalities, such as the possibility of searching for combination therapies, could be beneficial to the participants. It was noted that it is currently possible to search for multiple therapies, but only via a logical "or" link. In addition, some participants expressed difficulties in answering the survey questions, as the dataset underlying the test system contains synthetic data. Despite these difficulties, participants were optimistic that the system's functions are well integrated and that no external support is required for its operation, which is a satisfactory result for a complex system. Nevertheless, most participants expressed a desire for an improved presentation of the patient record and hoped that the system would provide even better support for their case preparation.

6.2 Outlook

CDSS will play an increasingly important role in PM today and in the foreseeable future. Despite the successful implementation of the portal and fulfillment of the associated requirements formulated during the requirements engineering process and confirmed by the evaluation through the online survey, it is crucial to repeat this cycle regularly. In this way, doctors with extensive expertise can communicate their needs to computer scientists without in-depth background knowledge to create a system that best supports their work.
In Europe, especially in Germany, the improvement of patient care through PM has so far been severely limited due to numerous regulations and technical obstacles. People suffering from rare diseases, in particular, could receive better care thanks to the attention gained and hopefully also thanks to the portal developed. The development of this portal will contribute to improving patient care in Germany and positively impact society. My goal is that by using this portal, doctors will be able to work more effectively by having medical information and resources easily accessible to optimize the treatment and care of their patients. In this way, the portal will help to close the gaps in care for patients with rare diseases or cancer and improve healthcare overall.

Bibliography

[atc] Atc. https://www.ema.europa.eu/en/glossary/atc-code.

[bwh] bwhealthcloud. https://gesundheit-wird-digital.de/projekt/bwhealthcloud-12/.

[con] Conventional commits 1.0.0. https://www.conventionalcommits.org/en/v1.0.0/.

[CV15] Francis S Collins and Harold Varmus. A new initiative on precision medicine. *New England journal of medicine*, 372(9):793–795, 2015.

[dnpa] Datenintegrationsplattform des deutschen netzwerks für personalisierte medizin. https://forschen-fuer-gesundheit.de/projekt_dnpm_dip.php.

[dnpb] Dnpm — deutsches netzwerk fÜr personalisierte medizin. https://dnpm.de/.

[Gra23] Julia Carolin Graf. Anwendung von kaplan-meier-statistiken im bwhc-system. 2023.

[GS18] Laura H Goetz and Nicholas J Schork. Personalized medicine: motivation, challenges, and progress. *Fertility and sterility*, 109(6):952–963, 2018.

[HC10] Margaret A Hamburg and Francis S Collins. The path to personalized medicine. *New England Journal of Medicine*, 363(4):301–304, 2010.

[hgn] Hgnc. https://www.nlm.nih.gov/research/umls/sourcereleasedocs/current/HGNC/index.html.

[hgv] Hgvs. https://hgvs-nomenclature.org/stable/.

[icda] Icd-10-gm. https://www.bfarm.de/EN/Code-systems/Classifications/ICD/ICD-10-GM/_node.html.

[icdb] Icd-o-3. https://www.bfarm.de/EN/Code-systems/Classifications/ICD/ICD-O-3/_node.html.

[jav] An introduction to javascript. https://javascript.info/intro.

[KGL+21] Bart Koopman, Harry JM Groen, Marjolijn JL Ligtenberg, Katrien Grünberg, Kim Monkhorst, Adrianus J Langen, Mirjam C Boelens, Marthe S Paats, Jan H Thüsen, Winand NM Dinjens, et al. Multicenter comparison of molecular tumor boards in the netherlands: Definition, composition, methods, and targeted therapy recommendations. *The Oncologist*, 26(8):e1347–e1358, 2021.

[KKRB+20] Katarzyna Kolasa, W Ken Redekop, Alexander Berler, Vladimir Zah, and Carl V Asche. Future of data analytics in the era of the general data protection regulation in europe. *PharmacoEconomics*, 38(10):1021–1029, 2020.

[Kok20] Lau Tiam Kok. *Hands-on Nuxt. js Web Development: Build universal and static-generated Vue. js applications using Nuxt. js*. Packt Publishing Ltd, 2020.

[LC23] O. Kohlbacher et al. L. Clin, A.-C. Polatkan. A federated data infrastructure for personalized oncology. 2023.

[LLMS20] Claudio Luchini, Rita T Lawlor, Michele Milella, and Aldo Scarpa. Molecular tumor boards in clinical practice. *Trends in Cancer*, 6(9):738–744, 2020.

[LvdKD+18] Garm Lucassen, Maxim van de Keuken, Fabiano Dalpiaz, Sjaak Brinkkemper, Gijs Willem Sloof, and Johan Schlingmann. Jobs-to-be-done oriented requirements engineering: a method for defining job stories. In *Requirements Engineering: Foundation for Software Quality: 24th International Working Conference, REFSQ 2018, Utrecht, The Netherlands, March 19-22, 2018, Proceedings 24*, pages 227–243. Springer, 2018.

[moz] Mozilla - what is javascript? https://developer.mozilla.org/en-US/docs/Learn/JavaScript/First_steps/What_is_JavaScript.

[MSMW18] David Merry, Christoph Schickhardt, Katja Mehlis, and Eva C Winkler. Trust and responsibility in molecular tumour boards. *Bioethics*, 32(7):464–472, 2018.

[orp] Orphanet. `https://www.orpha.net/`.

[rec] Recist. `https://en.wikipedia.org/wiki/Response_evaluation_criteria_in_solid_tumors`.

[SGK19] Ute I Schwarz, Markus Gulilat, and Richard B Kim. The role of next-generation sequencing in pharmacogenetics and pharmacogenomics. *Cold Spring Harbor Perspectives in Medicine*, 9(2):a033027, 2019.

[SHDA08] Arrigo Schieppati, Jan-Inge Henter, Erica Daina, and Anita Aperia. Why rare diseases are an important medical and social issue. *The Lancet*, 371(9629):2039–2041, 2008.

[SPB+20] Reed T Sutton, David Pincock, Daniel C Baumgart, Daniel C Sadowski, Richard N Fedorak, and Karen I Kroeker. An overview of clinical decision support systems: benefits, risks, and strategies for success. *NPJ digital medicine*, 3(1):17, 2020.

[SRTP+21] James Scheibner, Jean Louis Raisaro, Juan Ramón Troncoso-Pastoriza, Marcello Ienca, Jacques Fellay, Effy Vayena, and Jean-Pierre Hubaux. Revolutionizing medical data sharing using advanced privacy-enhancing technologies: technical, legal, and ethical synthesis. *Journal of medical Internet research*, 23(2):e25120, 2021.

[Tho] Nathan Thomas. How to use the system usability scale (sus) to evaluate the usability of your website. `https://usabilitygeek.com/how-to-use-the-system-\usability-scale-sus-to-evaluate-the-usability\-of-your-website`.

[TLM03] Juan Trujillo and Sergio Luján-Mora. A uml based approach for modeling etl processes in data warehouses. In *International Conference on Conceptual Modeling*, pages 307–320. Springer, 2003.

[tsV] Typescript vs javascript: When to use. `https://yellow.systems/blog/typescript-vs-javascript`.

[uRKR13] Tousif ur Rehman, Muhammad Naeem Ahmed Khan, and Naveed Riaz. Analysis of requirement engineering processes, tools/techniques and methodologies. *International Journal of Information Technology and Computer Science (IJITCS)*, 5(3):40, 2013.

[VdVVHVL+17] DL Van der Velden, CML Van Herpen, HWM Van Laarhoven, EF Smit, HJM Groen, Stefan Martin Willems, PM Nederlof, MHG Langenberg, Edwin Cuppen, Stefan Sleijfer, et al. Molecular tumor boards: current practice and future needs. *Annals of Oncology*, 28(12):3070–3075, 2017.

[vue] Vue.js. `https://vuejs.org/`.

[zpm] Zpm-verbund bw. `https://zpm-verbund.de/`.

Selbständigkeitserklärung

Hiermit versichere ich, dass ich die vorliegende Masterarbeit selbständig und nur mit den angegebenen Hilfsmitteln angefertigt habe und dass alle Stellen, die dem Wortlaut oder dem Sinne nach anderen Werken entnommen sind, durch Angaben von Quellen als Entlehnung kenntlich gemacht worden sind. Diese Masterarbeit wurde in gleicher oder ähnlicher Form in keinem anderen Studiengang als Prüfungsleistung vorgelegt.

Ort, Datum Unterschrift