

**HealthCare and Nutrition Tracking System “HNTS”**

**Senior Project Presented to the Faculty of Business Administration, Department of Management Information System and Technology**

**Al Maaref University**

**As a partial Fulfillment of the Requirements for the**

**Degree of Bachelor of Sciences in Information Technology and Management Systems**

**Haydar Mohamad**

**Summer 2024**

**Al Maaref University**

**Faculty of Business Administration**

**Approval of the Senior Project**

**Applicant:** Haydar Mohamad

**Degree:** Bachelor of Sciences in Information Technology and Management Systems

**Title:** HealthCare and Nutrition Tracking System

The Following Professor nominated to serve as the advisor of the above candidates has approved this work.

Advisor: Dr. Kassem Danach

# **Acknowledgements**

To my esteemed university and dedicated instructors, As I approach graduation, I am deeply grateful for the invaluable guidance, support, and inspiration I have received throughout this wonderful journey. To Al Maaref, thank you for your nurturing environment that encourage intellectual growth, and personal development. The commitment to excellence that you provide, has changed me to the person I am today, and I forever will be owing what I accomplish in life to this institution that has become my second home. To my instructors, your role was the most significant within my journey, since your passion, guidance, and knowledge have left a mark in my academic growth. Your dedication to teach us about what is right, and what is wrong, and the mentorship you provided, has expanded my knowledge, and taught us exactly how to step on the road of success. I am thankful for being a student under such exceptional instructors.

For Dr. Kassem Danach the supervisor of my senior project, you have my sincere prayers, and gratitude to you for the great effort you spent on me, and on all of my fellow students during this long lasting journey.

# **Abstract**

The “Healthcare and Nutrition Tracking System” (HNTS) is a comprehensive software solution that is designed for healthcare facilities, to facilitate the monitoring, tracking, and management of individual health, and nutrition goals. It was developed using the C# programming language, and MySQL database, HNTS provides users with such a user-friendly interface to track various aspects of individual healthcare, and nutritional habits. The system enables users to manage patient’s health, set nutrition and exercise plans, record daily food intake, monitor health metrics such as, weight, blood pressure, and calorie intake.

Features of HNTS include a robust database architecture for efficient data storage, and retrieval, secure user authentication, and authorization mechanism to safeguard sensitive information, and intuitive data visualization tools to present meaningful insights to users.

The development of HNTS represents a significant contribution to the field of healthcare, and nutrition management, offering individuals an accessible tool to take manage patient’s well-being.

By leveraging the power of technology, HNTS empowers users to make informed decisions about individual health, and nutrition leading to improved overall wellness and quality of life.

**Table of Contents**

[Acknowledgements 2](#_Toc170463481)

[Abstract 3](#_Toc170463482)

[Symbols and Abbreviations 7](#_Toc170463483)

[Chapter 1: Introduction 8](#_Toc170463484)

[1.1 Overview 9](#_Toc170463485)

[1.2 Research Objectives 9](#_Toc170463486)

[1.3 Research Questions 9](#_Toc170463487)

[1.4 Research Hypothesis 10](#_Toc170463488)

[1.5 Scope 10](#_Toc170463489)

[1.6 Ethics 10](#_Toc170463490)

[1.7 Research Gap 10](#_Toc170463491)

[1.8 Chapter Distribution 11](#_Toc170463492)

[Chapter Two: Literature Review 12](#_Toc170463493)

[2.1. Introduction of the literature Review 13](#_Toc170463494)

[2.1.1. Definition and Evolution of Healthcare and Nutrition Tracking Information Systems 13](#_Toc170463495)

[2.1.2. Key Features and Functionalities of Healthcare and Nutrition Tracking Systems 14](#_Toc170463496)

[2.1.2.1 Tracking and Managing Healthcare and Nutrition Data 14](#_Toc170463497)

[2.1.2.2 Patient and Client Management 15](#_Toc170463498)

[2.1.2.3 Resource Procurement and Content Development 15](#_Toc170463499)

[2.1.2.4 User Support and Personalization 15](#_Toc170463500)

[2.1.2.5 User-Centered Design and Interface 16](#_Toc170463501)

[2.1.3 Challenges and Emerging Trends 16](#_Toc170463502)

[2.1.3.1 Interoperability 16](#_Toc170463503)

[2.1.3.2 Open-Source Solutions 17](#_Toc170463504)

[2.1.3.3 Cloud-Based Systems 17](#_Toc170463505)

[2.1.3.4 Artificial Intelligence (AI) and Machine Learning (ML) 17](#_Toc170463506)

[2.1.3.5 Remote and Mobile Access 17](#_Toc170463507)

[2.1.4 Impact on Healthcare and Nutrition Tracking Services and User Satisfaction 18](#_Toc170463508)

[2.2. Real types of Health Software Systems 19](#_Toc170463509)

[2.3. Development Strategies 22](#_Toc170463510)

[2.3.1 SDLC Models 23](#_Toc170463511)

[2.3.2. SDLC Stages 27](#_Toc170463512)

[Chapter Three: Methodology 33](#_Toc170463513)

[3.1 Research Philosophy 33](#_Toc170463514)

[3.2 Data Collection Methods 34](#_Toc170463515)

[3.3 Data Analysis 34](#_Toc170463516)

[3.4 Ethical Considerations 34](#_Toc170463517)

[3.5 Limitations 35](#_Toc170463518)

[3.6 Summary 35](#_Toc170463519)

[Chapter Four: Analysis and Design 36](#_Toc170463520)

[4.1 SDLC Analysis for HNTS 36](#_Toc170463521)

[4.1.1. Planning phase 36](#_Toc170463522)

[4.1.2. Analysis Phase 37](#_Toc170463523)

[4.1.3. Design Phase 43](#_Toc170463524)

[4.1.4. Used Technologies 45](#_Toc170463525)

[4.1.5. Development Environment Setup 46](#_Toc170463526)

[4.1.6. Testing 46](#_Toc170463527)

[4.2. HNTS Pages/ Forms 46](#_Toc170463528)

[Chapter Five: Conclusions and Recommendations 48](#_Toc170463529)

[5.1 Conclusion 48](#_Toc170463530)

[5.2 Recommendation 48](#_Toc170463531)

[5.3 References 49](#_Toc170463532)

**Table of Figures**

[Figure 1: Healthie Interface 19](#_Toc170659047)

[Figure 2: Nutrium System Interface 20](#_Toc170659048)

[Figure 3:Practice Better Home Page 21](#_Toc170659049)

[Figure 4: Simple Practice Home Page 22](#_Toc170659050)

[Figure 5: Water-Fall Model 24](file:///C:\Users\USER\3D%20Objects\SeniorProj.%20Report.docx#_Toc170659051)

[Figure 6:V-Shaped Model 25](file:///C:\Users\USER\3D%20Objects\SeniorProj.%20Report.docx#_Toc170659052)

[Figure 7:Prototyping Model 26](file:///C:\Users\USER\3D%20Objects\SeniorProj.%20Report.docx#_Toc170659053)

[Figure 8: Iterative Model 26](file:///C:\Users\USER\3D%20Objects\SeniorProj.%20Report.docx#_Toc170659054)

[Figure 9: SDLC Stages 27](file:///C:\Users\USER\3D%20Objects\SeniorProj.%20Report.docx#_Toc170659055)

[Figure 10: SWOT Analysis 29](#_Toc170659056)

[Figure 11: DFD at level 0 of HNTS 42](file:///C:\Users\USER\3D%20Objects\SeniorProj.%20Report.docx#_Toc170659057)

[Figure 12: DFD at level 1 of HNTS 43](file:///C:\Users\USER\3D%20Objects\SeniorProj.%20Report.docx#_Toc170659058)

[Figure 13: DFD at level 2 of HNTS 43](file:///C:\Users\USER\3D%20Objects\SeniorProj.%20Report.docx#_Toc170659059)

[Figure 14: Activity Diagram 44](#_Toc170659060)

[Figure 15: Use-Case diagram of HNTS 45](file:///C:\Users\USER\3D%20Objects\SeniorProj.%20Report.docx#_Toc170659061)

[Figure 16: HNTS DB's ERD Model 46](file:///C:\Users\USER\3D%20Objects\SeniorProj.%20Report.docx#_Toc170659062)

[Figure 17: HNTS System Login Page 49](file:///C:\Users\USER\3D%20Objects\SeniorProj.%20Report.docx#_Toc170659063)

[Figure 18: HNTS System Home Page 49](file:///C:\Users\USER\3D%20Objects\SeniorProj.%20Report.docx#_Toc170659064)

**Table of Tables**

[Table 1: Functional Requirements 39](#_Toc170463658)

[Table 2: Non-Functional Requirements 40](#_Toc170463659)

# **Symbols and Abbreviations**

HNTS Healthcare and Nutrition Tracking System

Mu Al Maaref University

FBA Faculty of Business Administration

VS Visual Studio

C# C-Sharp Programming Language

MySQL Open Source Database Server

MIT Management Information System

DB Database

SDLC System Development Life Cycle

DFD Data Flow Diagram

URL Uniform Resource Locator

UML Unified Modeling Language

ERD Entity Relationship Diagram

RDBMS Relational Database Management System

EHR Electronic Health Record Systems

UI User Interface

IDE Integrated Development Environment

HWC Health & Wellness Clinic

AI Artificial Intelligence

ML Machine Learning

HL7 Health Level Seven

API Application Programming Interface

TAM Technology Acceptance Model

# **Chapter 1: Introduction**

## **Overview**

In a world where bytes meet bites, and data duels our daily lives, the fusion of technology, and nutrition heralds a new era of wellness management. Nowadays, the digital landscape is evolving rapidly, where the integration of technology into various sides of our lives has become normal, revolutionizing the way we work, communicate, and manage our daily activities. And from the early arrival of information systems to sophisticated technologies of today, the field of Management Information Systems and Technology (MIS/MIT) has played a substantial role in harnessing the power of technology to optimize business processes, enhance decision-making, and drive organizational success. And as we go through the complexities of modern life, we cannot overstate the importance of healthcare, and nutrition management, since with the rising concerns about health, and wellness, individuals are seeking effective solutions to monitor, and improve their well-being. And so, with response to this growing demand, I Introduce to you “Healthcare and Nutrition Tracking System” (HNTS) a system that emerges as a cutting edge software solution designed to empower individuals to take control of their health, and nutrition goals. And as any Information System, this application is based on main components: Hardware, Software, Data, People, and Procedure. And each component is very essential on how this application is structured, and built. This application will allow the users or clients to check on patients, view their medical records effectively, tracking nutrition, designing a daily routine that transforms their lives, attach a training program that suits them, monitoring patients progress, and last but not least providing educational resources for patients.

## **Research Objectives**

1. Analyzing how ‘HNTS’ contributes to enhancing operational efficiency, reducing manual tasks, and optimizing healthcare procedures.
2. Investigate the level of acceptance among healthcare professionals, as well as the effectiveness of training programs for users.
3. Evaluate the HNTS’s long-term sustainability, ability to adapt to emerging healthcare technologies, and ongoing support.
4. Ensure that the HNTS effectively meets user needs by assessing its response time and overall performance, particularly during periods of increased demand.

## **Research Questions**

1. Why is selecting a comprehensive “Health Information System” important for modern healthcare facilities?
2. Which software development tools and programming languages are best for constructing “Electronic Health Record Systems (EHR), and how can these applications be engineered to deliver the best user experience for healthcare providers?
3. Which “Software Development Life Cycle (SDLC)” model aligns best with the complex design and development of “HNTS” systems?
4. What are the essential functional and performance requirements for creating robust “HNTS” software solutions tailored based on the needs of healthcare facilities, and providers?

## **Research Hypothesis**

Hypothesis: Implementing a “HNTS” will enhance the efficiency and efficacy in optimized resource allocation, streamlined clinical workflows, and elevated patient care experiences.

## **Scope**

The “Healthcare and Nutrition Tracking Information System (HNTS)” attempts to automate healthcare processes, enhance operational efficiency, and present to the users a transparent and intuitive experience. Its objective is to centralize and categorize medical records, diagnostic reports, and patient information, enabling healthcare professionals to access, manage, and utilize data in an easier way.

## **Ethics**

As we grow older, we pick up certain principles that guide our actions. These are like our personal rulebook, saying what’s right and wrong. When it comes to research, it’s essential to stick to these ethics, since they are the foundation of what this research is built on from the beginning.

## **Research Gap**

In today’s fast-paced world of healthcare and nutrition sector, the traditional methods that were used of manually organizing and managing data are starting to feel outdated and inefficient. With the increasing amount of information, diverse forms of data, and evolving patient needs, healthcare facilities and nutrition centers require a modern, automated system to keep up with the fast pace and, with the demands of the industry.

# **Chapter Distribution**

1. **Chapter Two: Literature Review**

The literature review chapter on “HNTS” provides an overview of the evolution, current technologies, challenges, benefits, and future trends in the field. This chapter is structured into the following sub-sections:

* **Historical Background:** Discusses the historical background and evolution of healthcare and nutrition tracking systems.
* **Features / Challenges**: Discusses the key features of “HNTS” and the challenges it encounters.
* **Impact on Health Sector:** Talking about the impact the healthcare and nutrition system has brought into the healthcare facilities.
* **Types of Systems:** Showing best implementation, and practices of other types of healthcare systems.

1. **Chapter Three: Methodology**

The methodology chapter outlines the approach, procedures, analysis used and the limitations encountered in developing the healthcare and nutrition tracking system.

1. **Chapter Four: Analysis and Design**

The analysis and design phase of the healthcare and nutrition tracking system (HNTS) is a critical stage in the SDLC, laying the foundation for system implementation. This chapter includes:

* **Planning & Analysis:** Explaining the planning, and analysis stages of the “SDLC”.
* **Gathering Techniques Used:** Discussing different techniques used to establish procedures across different scenarios.
* **Design Phase:** Talking about the designing phase of the ‘HNTS”.
* **Technologies Used:** Different types of technologies used in the creation of the “HNTS”.

1. **Chapter Five: Conclusion and Recommendation**

A pivotal component, summarizing the findings, insight, and proposing actionable steps for future endeavors. It includes:

* **Summary of Findings:** Clear summary of the key findings from the study.
* **Recommendations:** Actionable recommendations based on the findings.
* **References:** Resources used as reference to enrich the study.

# **Chapter Two: Literature Review**

Any research endeavor, whether it’s developing a healthcare information system or exploring nutrition trends, should begin with a literature review. This review presents a deep understanding of existing knowledge, old reviews, relevant theories, and previous studies in the field of healthcare, laying the foundation for the project. The findings of the literature review play a crucial role in the design, features, and overall success of the project.

In this chapter, we’ll delve into a mini research about various healthcare and nutrition tracking systems utilized by different organizations, analyzing their features and differences.

## **2.1. Introduction of the literature Review**

### **2.1.1. Definition and Evolution of Healthcare and Nutrition Tracking Information Systems**

Healthcare and nutrition tracking systems have always played a significant role in promoting individual wellness and fostering a culture of proactive health management. These systems have improved with the advancements in technology (Haux R. , 2006). The advent modern healthcare and nutrition tracking systems marks the dawn of what we now recognize as the “information age” in the health sector, with its roots coming back to the earliest human attempts to document and understand health related information(Smith, 2010)**.** In contemporary times, healthcare and nutrition tracking systems are categorized into four main types: public, academic, school, and special**.** These categories encompass a wide range of systems built to diverse user needs and contexts. Traditionally, the focus of healthcare and nutrition tracking systems has been on facilitating access to information rather than merely cataloging content. Healthcare professionals have viewed health data primarily as objects to be managed and utilized for informed decision-making. As a result, there has been a concerted effort within the field to leverage cutting-edge technologies to enhance the delivery of healthcare services and improve patient outcomes (Chaudhry B. , 2006)**.**

The realm of healthcare and nutrition tracking systems has seen significant evolution over the past few decades. The history is traced back 130 years, marked by milestones such as the standardization of patient records and the development of machine-readable formats for medical data (Smith, 2010). In the early 1980s, universities and healthcare technology vendors pioneered the first integrated healthcare systems (IHS) (Haux R. , 2006)**.** With the advent of the digital era, characterized by the widespread digitization of medical records and the proliferation of health-related content online, new services emerged to serve to both normal users and medical professionals (Haux R. , 2006). These services include open URL linking for an easy access to medical literature and meta-searching functionalities for comprehensive research. Additionally, healthcare facilities began adopting electronic resource management systems to direct operations and improve patient care (Berger M. , 2006). Healthcare institutions are now facing pressures from users where the demands for quick and effective discovery and delivery. Patients and medical professionals alike expect intuitive interfaces and seamless access to healthcare information, akin to their experiences with popular internet platforms. In response, a new wave of Healthcare 2.0 applications is emerging, designed to enhance the user experience and improve the efficiency of healthcare delivery. These innovative solutions allow healthcare facilities to maintain their existing back-office systems while incorporating new technologies as needed. By leveraging Web 2.0 services, institutions can easily integrate systems from different vendors, facilitating a progressive transition toward a more advanced automation platform for healthcare and nutrition tracking.

### **2.1.2. Key Features and Functionalities of Healthcare and Nutrition Tracking Systems**

Throughout history, the concept of a healthcare and nutrition tracking system has been complete to understanding and managing personal well-being. These systems are made to gather and manage health related data, ranging from medical records to dietary information. As society has fostered the internet and the World Wide Web, the landscape of healthcare information access and management has experienced a profound transformation (Griffin, 2019). In today's interconnected world, healthcare and nutrition tracking systems are essential components of the modern health sector. They provide individuals with access to a wealth of information on various aspects of health and wellness, facilitating informed decision-making and proactive health management (Jensen T. , 2013). Within the academic sphere, these systems are indispensable tools for promoting excellence in health education and research, empowering students and professionals alike with the resources they need to succeed in the evolving field of healthcare. Moreover, in the business world, where competition is fierce and innovation is key, healthcare and nutrition tracking systems play a vital role in collecting and analyzing data to drive enterprise growth and development. By harnessing the power of these systems, businesses can gain valuable insights into consumer trends, identify areas for improvement, and enhance their competitive edge in the marketplace. Thus, healthcare and nutrition tracking systems serve as indispensable assets in navigating the complexities of modern life, facilitating informed decision-making and fostering personal and professional growth.

### **2.1.2.1 Tracking and Managing Healthcare and Nutrition Data**

**Traditional Tracking Methods**: Historically, managing healthcare and nutrition data involved heavy manual processes, such as handwritten records and filing systems (Smith, 2010). **Modern Tracking Methods**: Today, healthcare and nutrition tracking systems utilize sophisticated software and standards like HL7 (Health Level Seven) and SNOMED CT (Systematized Nomenclature of Medicine -- Clinical Terms) to direct data management and retrieval (Haux R. , 2006). Additionally, advancements in technology have led to the exploration of linked data and Semantic Web technologies to improve interoperability and facilitate seamless exchange of health information across different platforms.

### **2.1.2.2 Patient and Client Management**

**Traditional Patient Management:** In the past, patient management relied heavily on manual processes, including paper record keeping and physical appointment cards (Smith, 2010). **Modern Patient Management:** Today, healthcare and nutrition tracking systems provide advanced features such as self-service kiosks for appointment check-ins, RFID technology for tracking patient movements within healthcare facilities, and online portals for appointment scheduling and renewal of prescriptions (Chaudhry B. , 2006). Patient records are often integrated with digital identification systems, user accounts, and mobile applications, ensuring seamless access to healthcare services and information.

### **2.1.2.3 Resource Procurement and Content Development**

**Traditional Procurement:** Historically, acquiring resources for healthcare and nutrition education involved manual processes such as ordering physical textbooks and materials (Smith, 2010).

**Modern Procurement**: Nowadays, healthcare and nutrition tracking systems employ electronic resource management tools, online procurement platforms, and automated systems to streamline the acquisition of digital resources, e-books, and online journals (Berger M. , 2006). Collection development strategies are informed by usage analytics and user preferences, ensuring that resources meet the evolving needs of students and professionals in the field (Haux R. , 2006).

### **2.1.2.4 User Support and Personalization**

**Traditional User Support:** In the past, support services in healthcare and nutrition tracking systems were often generic and not tailored to individual needs (Smith, 2010).

**Modern User Support:** Today, personalized features enhance the user experience in healthcare and nutrition tracking systems. These may include customizable alerts for medication reminders, saved searches for quick access to relevant health information, and personalized recommendations for diet and exercise plans based on individual goals and preferences (AKTAS M. , 2023). Additionally, healthcare and nutrition tracking systems may offer mobile applications for convenient access to health data, online chat support for fast assistance, and virtual consultations with healthcare professionals for guidance and advice.

### **2.1.2.5 User-Centered Design and Interface**

Research efforts in the realm of healthcare and nutrition tracking systems have traditionally focused on technical development, often led by experts in the field. Despite substantial investments in creating user-friendly interfaces, previous studies have suggested that potential users may still hesitate to adopt these systems (Griffin, 2019). This study aims to enhance our understanding of user acceptance in healthcare and nutrition tracking systems by applying the technology acceptance model (TAM) (AKTAS M. , 2023). It identifies three key interface characteristics, three organizational context variables, and three individual differences as crucial factors influencing adoption intentions through perceived usefulness and perceived ease of use of the system. Data was gathered from 397 users of an acclaimed healthcare and nutrition tracking system (AKTAS M. , 2023). The findings highlight the significance of both perceived usefulness and perceived ease of use in determining user acceptance. Furthermore, the study reveals that interface characteristics and individual differences impact perceived ease of use, while organizational context influences both perceived ease of use and perceived usefulness of healthcare and nutrition tracking systems (AKTAS M. , 2023).

### **2.1.3 Challenges and Emerging Trends**

Healthcare and nutrition tracking systems encounter various challenges in today's landscape, prompting ongoing evolution and innovation in system design and management.

### **2.1.3.1 Interoperability**

**Problem:** Healthcare and nutrition tracking systems often utilize diverse platforms (such as electronic health records, fitness tracking apps, and dietary analysis tools) that struggle to communicate seamlessly, leading to inefficiencies and fragmented data. **Trend:** Interoperability is gaining significance in healthcare and nutrition tracking systems. To facilitate a smooth data exchange between platforms, they are adopting standards like HL7 (Health Level Seven) and APIs (Application Programming Interfaces) (Raghupathi, 2014). Open-source solutions are accepted more for their adaptability and compatibility with various systems.

### **2.1.3.2 Open-Source Solutions**

**Challenge:** Healthcare and nutrition tracking systems often face vendor lock-in and financial constraints when it comes to investing in proprietary software.

**Trend:** A growing trend among healthcare and nutrition tracking systems is the adoption of free and open-source solutions, such as OpenMRS, GNU Health, and MyFitnessPal (Fontelo, 2015). These platforms provide greater control over customization and development while offering cost-effective alternatives to proprietary software solutions.

### **2.1.3.3 Cloud-Based Systems**

**Problem:** Healthcare and nutrition tracking systems often face challenges in maintaining on-site hardware and software, which can be costly and resource-intensive. **Trend:** To capitalize on benefits such as enhanced accessibility, scalability, and cost-effectiveness, healthcare and nutrition tracking systems are transitioning to cloud-based solutions (Sultan, 2010). These platforms offer automatic updates which is far from being costly and resource-intensive, recovery capabilities, and reduce the burden on IT personnel by outsourcing infrastructure management.

### **2.1.3.4 Artificial Intelligence (AI) and Machine Learning (ML)**

**Problem:** Managing and analyzing the vast amount of data generated in healthcare and nutrition tracking systems can be exhausting.

**Trend:** To streamline processes such as data analysis, personalized recommendations, and predictive analytics, healthcare and nutrition tracking systems are integrating AI and machine learning technologies (Rajkomar, 2019). Virtual assistants and chat bots are employed to improve user interactions and provide support services.

### **2.1.3.5 Remote and Mobile Access**

**Problem:** Ensuring that healthcare and nutrition tracking system resources and services are accessible to users from remote places and on mobile devices can be challenging. **Trend:** Healthcare and nutrition tracking systems are increasingly designed to be responsive and mobile-friendly. The integration of mobile apps and remote access options is becoming a part of these systems, offering users convenient access to health data and nutrition tracking tools anytime, anywhere (Kamel, 2013).

## **2.1.4 Impact on Healthcare and Nutrition Tracking Services and User Satisfaction**

The introduction of Healthcare Information Systems (HNTS) has brought significant changes to the services offered by healthcare and nutrition tracking systems, impacting resource discovery, information access, and user satisfaction in several ways:

1. Expanded Access to Digital Resources:

E-books and Online Journals: HIS provides users with convenient access to digital resources such as online journals and e-books (Haux R. , 2006). This increased accessibility expands the range of materials available in healthcare and nutrition tracking systems, particularly benefiting users who are off-campus or connected remotely.

Remote Access: To serve to the needs of learners and remote users, there is a growing emphasis on remote authentication and access to digital resources (Kamel, 2013).

1. User-Centric Service:

Personalization: Healthcare and nutrition tracking systems often offer personalized features like tracking histories, saved searches, and customized recommendations. This personalization fosters user engagement and enhances the overall user experience (AKTAS M. , 2023).

Virtual Support Services: By integrating virtual support services such as email or chat-based assistance, users receive prompt help, leading to higher satisfaction levels (Griffin, 2019).

1. Analytics for Informed Decision-Making:

Data-Driven Resource Management: By analyzing usage data and user behavior, healthcare and nutrition tracking systems can make informed decisions about resource allocation and collection development (Haux R. , 2006).

User Behavior Insights: Insights derived from HIS enable healthcare and nutrition tracking systems to tailor services to match the evolving needs and preferences of their users (Chaudhry B. , 2006).

1. Integration of AI and Machine Learning:

Recommendation Engines: AI-powered recommendation engines in healthcare and nutrition tracking systems suggest relevant resources based on user behavior and preferences, facilitating resource discovery (Rajkomar, 2019).

Chat bots and Virtual Assistants: AI-driven chat bots offer immediate assistance, improving user satisfaction and reducing response times for user inquiries (AKTAS M. , 2023).

## **2.2. Real types of Health Software Systems**

For practical insights, this review will examine successful implementations of Healthcare Information Systems and best practices employed by healthcare facilities to optimize their systems. I searched for several Healthcare systems, here are some of them:

1. **Healthie**

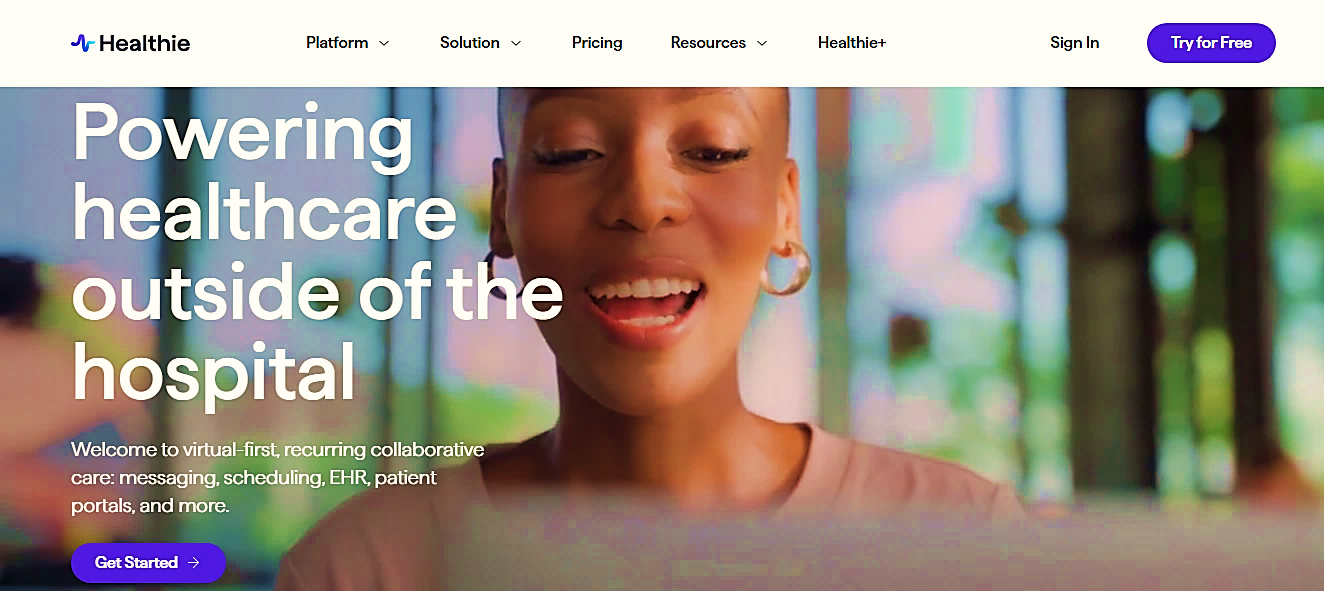
****

Figure 1: Healthie Interface

Healthie offers a wide range of services and features tailored to the needs of healthcare professionals and their clients **(see Figure 1):**

* **Client Management:** Healthie allows practitioners to efficiently manage patient information, track their progress, and communicate securely with the patients through messaging and video calls.
* **Appointment Scheduling:** Practitioners can schedule appointments with their patients directly within the Healthie platform, set availability, and send appointment reminders to minimize missed appointment.
* **Nutrition Tracking:** Practitioners can track their patient’s nutritional intake and habits using Healthie's intuitive tracking tools, enabling them to monitor progress, identify trends, and make informed decisions about their patient’s health and wellness journey.
* **Meal Planning:** Healthie includes robust tools for creating personalized meal plans tailored to patients' dietary preferences, restrictions, and health goals, empowering practitioners to support the patients in achieving their nutritional objectives.
* **Billing and Invoicing:** With Healthie, practitioners can streamline their billing processes by generating invoices, processing payments, and tracking financial transactions within the system.

1. **Nutrium**

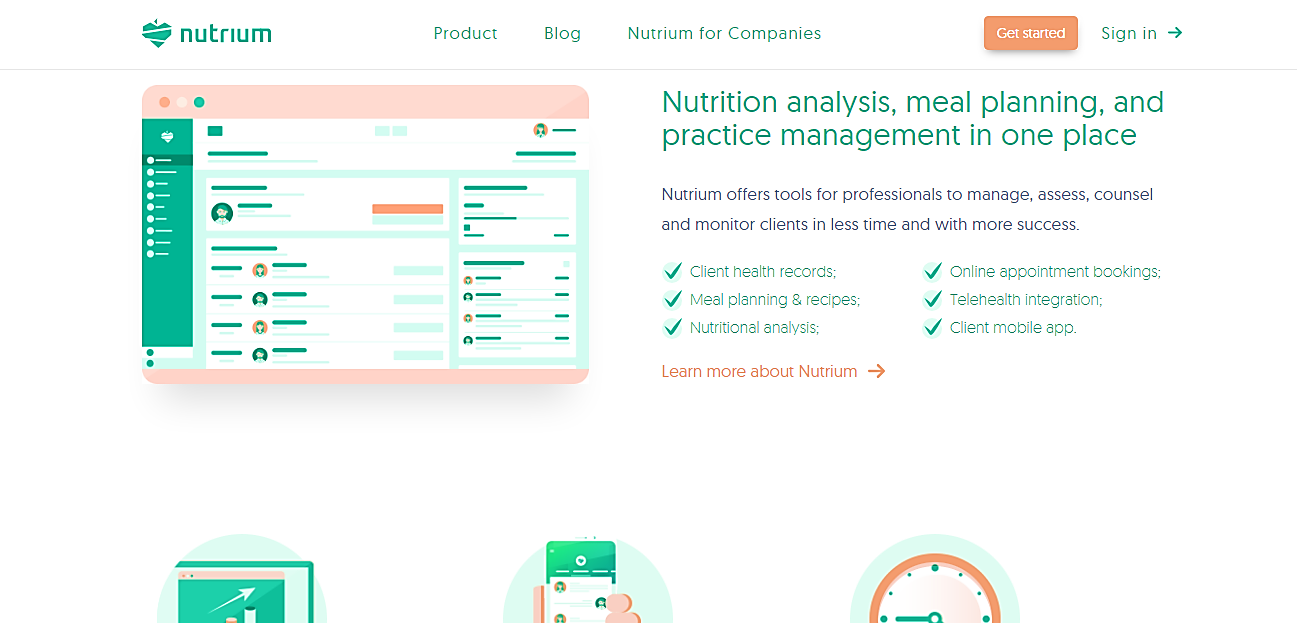
****

Figure 2: Nutrium System Interface

Nutrium is a practice management software designed specifically for dietitians, nutritionists, and healthcare professionals **(see Figure 2).** It offers a range of features to support practitioners in managing their practices and providing personalized care to patients:

* **Client Management:** Nutrium allows practitioners to organize and manage patient information, including health records, dietary preferences, and treatment plans, all in one centralized platform.
* **Nutritional Analysis:** Nutrium includes a comprehensive nutritional analysis feature that enables practitioners to assess patients' dietary intake, track progress, and make data-driven recommendations for optimal health outcomes.
* **Goal Setting and Tracking:** Nutrium enables practitioners to set personalized health goals with patients and track progress over time. Whether it's weight loss, improved dietary habits, or managing specific health conditions, practitioners can monitor patients' achievements and adjust plans accordingly.
* **Outcome Evaluation:** Nutrium offers tools for evaluating patient outcomes and measuring the effectiveness of interventions. Practitioners can track changes in patients' health parameters, assess adherence to recommendations, and identify areas for improvement to optimize care delivery.

1. **Practice Better**

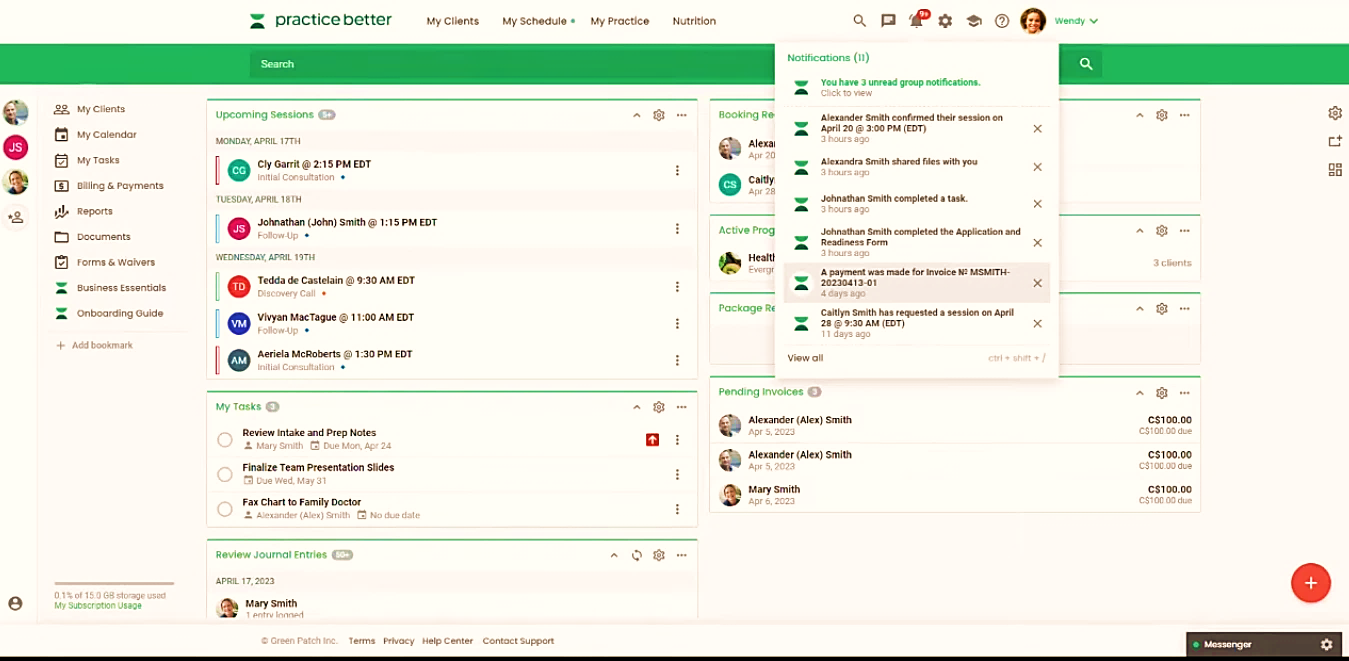
****

Figure 3:Practice Better Home Page

Practice Better is a comprehensive practice management platform designed specifically for health and wellness professionals, including dietitians, nutritionists, therapists, and coaches **(see Figure 3).** Here are some of the key features offered by Practice Better:

* **Customizable Forms:** Practice Better provides customizable intake forms, assessment questionnaires, and consent forms that practitioners can tailor to their specific practice needs and patient populations.
* **Client Portal:** Practice Better includes a patient portal where patients can access appointment details, complete forms, track progress, and communicate securely with their practitioner.
* **Outcome Measurement:** Practice Better includes outcome measurement tools that enable practitioners to evaluate the effectiveness of interventions, track changes in patients' health parameters, and measure progress towards goals.

1. **Simple Practice**

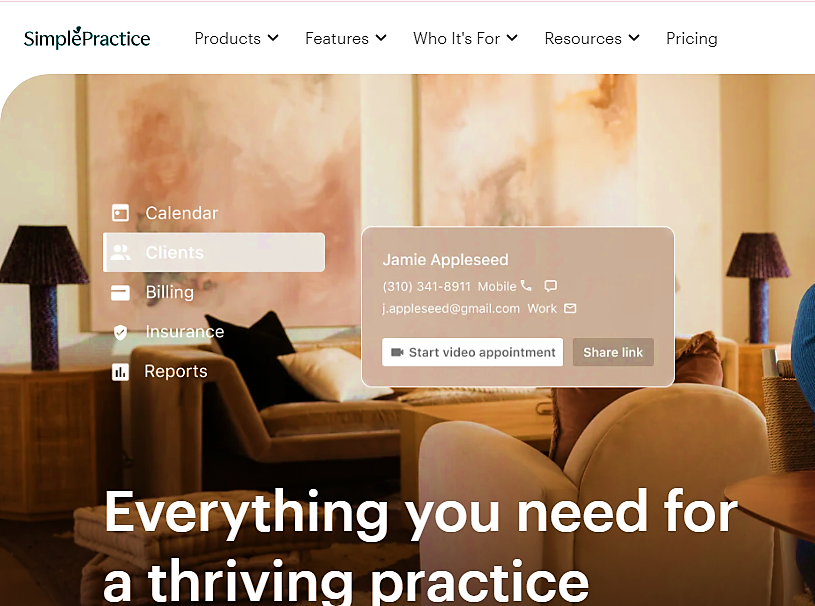
****

Figure 4: Simple Practice Home Page

Simple Practice is a versatile practice management platform designed for healthcare providers, therapists, counselors, and wellness professionals **(see Figure 4).** Here are some key features offered by Simple Practice:

* **Client Management:** Simple Practice provides tools for managing client information, including demographics, session notes, treatment plans, and billing details, all in one secure and intuitive platform.
* **Appointments Reminders:** Simple Practice includes automated appointment reminders sent via email or SMS to clients, reducing the likelihood of missed appointments and improving overall scheduling efficiency.
* **Practice Insights:** Simple Practice provides analytics and reporting tools that allow practitioners to track key practice metrics, such as appointment volume, revenue, and client engagement. This helps practitioners make informed decisions to optimize their practice operations and improve client outcomes.

## **2.3. Development Strategies**

### **2.3.1 SDLC Models**

Software Development Life Cycle, or SDLC, is a systematic approach to software development that guarantees the production of top notch software that satisfies user needs. Software development life cycle (SDLC) models are frameworks that direct the process of developing software, offering an orderly and structured method for creating software systems. Throughout the software development process, these models assist developers and project managers in managing resources, time, and risks. SDLC Models consist of waterfall, V-shaped, Prototype, Iterative, and RAD models (Pressman, 2014).

#### **2.3.1.1 Water-Fall Model**

The Waterfall Model is a widely used and traditional software development life cycle (SDLC) model. It is a sequential and linear approach to software development, where progress flows downwards like a waterfall. In this model, each phase of the development process is completed before moving to the next phase (Pressman, 2014). The Waterfall Model follows a systematic and structured approach, which makes it easy to understand and manage (Pressman, 2014). Subsequent phases of the Waterfall Model include testing, deployment, implementation, and maintenance **(see Figure 5).** There is no repetition or overlap between the steps; they are all carried out one after the other (Pressman, 2014). It is simpler to plan and manage projects using this model since it gives software development a precise and well-defined framework. Nevertheless, it has several drawbacks, like its rigidity and incapacity to adapt to changes once a phase is finished. The Waterfall Model is still widely used when the needs are clear and steady, despite these disadvantages. Each phase is completed in a specified period after that it moves to the next phase. As it is a linear model, it’s easy to implement (Pressman, 2014).

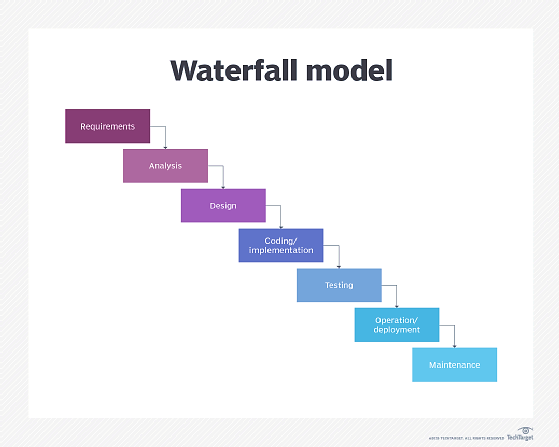


Figure 5: Water-Fall Model

#### **2.3.1.2. V-Shaped Model**

The verification and validation model, often referred to as the V-model, is a project management methodology that follows a sequential path similar to the waterfall model (Pressman, 2014). Each phase must be completed before proceeding to the next. However, unlike the waterfall model, testing occurs in parallel with each development stage, allowing for early detection and correction of issues (Pressman, 2014) **(see Figure 6).** This approach is particularly useful for small projects where requirements are well-defined and understood. The V-model's structure supports early problem identification, reducing the cost and effort needed to fix issues. Additionally, it provides a clear framework and roadmap for project planning and management (Pressman, 2014). A significant advantage of the V-model is its ability to facilitate systematic progress tracking and quality assurance. However, the V-model also has limitations. Its rigid structure requires precise and well-defined requirements from the beginning, making it difficult to accommodate changes or additions once development is underway. This inflexibility can pose challenges when project requirements evolve during the development process (Pressman, 2014).

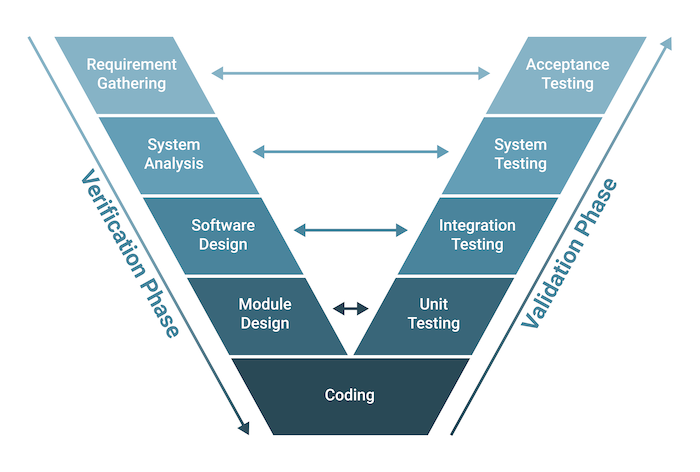


Figure 6:V-Shaped Model

#### **2.3.1.3. Prototyping Model**

Prototyping involves creating a working model that showcases the features of a proposed product, service, or system (Pressman, 2014). This method employs an iterative approach to system development, where each iteration involves analyzing requirements and exploring alternatives, followed by making necessary adjustments (Pressman, 2014). The process repeats until the reviewing stakeholder is satisfied **(see Figure 7).** The prototyping process begins with an initial model, which is continually refined through subsequent versions until the final system is developed. Although the prototype is not a complete system and lacks many details, its primary purpose is to demonstrate overall functionality and gather feedback for improvements (Pressman, 2014). This iterative refinement helps ensure that the final system meets user expectations and requirements.



Figure 7:Prototyping Model

#### **2.3.1.4. Iterative Model**

Based on the ideas of repetition and gradual development, the Iterative paradigm is a well- liked and frequently applied Software Development Life Cycle (SDLC) model (Pressman, 2014). Because it incorporates ongoing iterations and improvements to the software product, it is frequently referred to as the "Iterative and Incremental Development" paradigm. Large-scale and complicated projects can benefit from the Iterative Model's flexibility and adaptability, which set it apart from other linear SDLC models (Pressman, 2014). In the Iterative Model, the development process is divided into several iterations or stages, each of which consists of planning, designing, building, testing, and evaluation **(see Figure 8).** Thus, this model is particularly suitable for complex projects where requirements may evolve, making it a valuable approach in the field of software development (Pressman, 2014).

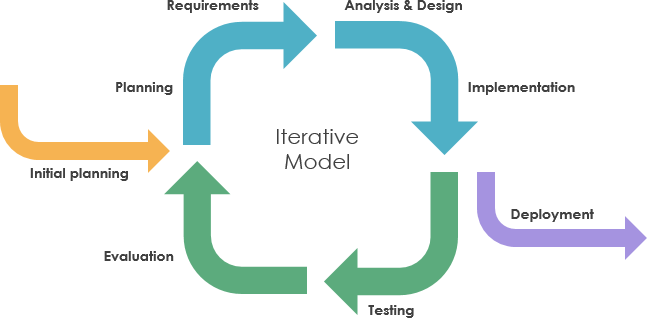


Figure 8: Iterative Model

#### **2.3.1.5. RAD Model**

The Rapid Application Development (RAD) model emphasizes prototyping and iterative development without detailed initial planning (Pressman, 2014). In RAD, the planning phase is integrated into the software development process itself. This approach prioritizes gathering customer requirements through workshops or focus groups, early prototype testing by customers, reusing existing components, continuous integration, and quick delivery. RAD is especially advantageous in scenarios where requirements are unclear or likely to change (Pressman, 2014).

#### **2.3.1.6. Spiral Model**

An iterative systems development in which the stages of analysis, design, code, and review repeat as new features for the system are identified. The four main activities of this model are as follows:

* **Planning:** setting project objectives, and defining alternatives.
* **Risk Analysis:** analysis of alternatives and the identification and solution of risks.
* **Engineering**: equivalent to the build phase of the SDLC with coding and testing.
* **Customer Evaluation:** testing of the product by customers. Mainly it is used for adding new features to the system, there is no need to re-run all the SDLC steps, only implement four of them.

### **2.3.2. SDLC Stages**

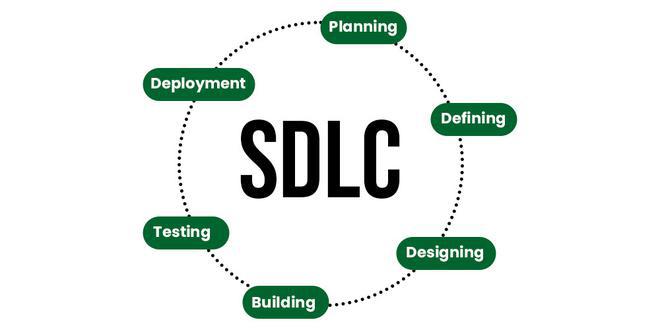


Figure 9: SDLC Stages

#### **2.3.2.1. Planning**

During the planning stage as shown in **(Figure 9),** potential problems and opportunities are investigated in relation to the business's goals and strategies. Systems planning aims to define the problem, determine the necessity of solving it, and outline the scope of the solution (Sommerville, 2016). When multiple issues or cases are present, the planning step involves selecting the most critical one that will be most beneficial to the organization. Additionally, the planning phase addresses project management aspects such as resource allocation, timeline estimation, and risk assessment. Project managers evaluate available resources, including personnel, budget, and technology, to ensure the project's success (Sommerville, 2016). This stage involves setting clear objectives, defining deliverables, and establishing a project timeline with milestones to track progress. Risk assessment is essential to identify potential obstacles and develop contingency plans to mitigate any negative impact on the project's timeline and budget. By thoroughly addressing these elements during the planning stage, organizations can establish a strong foundation for the subsequent stages of the SDLC, ensuring efficient development and implementation of the proposed solution (Sommerville, 2016).

#### **2.3.2.2. Analysis**

During the system analysis step, the primary focus is on gathering, documenting, and comprehensively understanding the business requirements for the proposed system (Sommerville, 2016). This involves engaging with stakeholders, end-users, and subject matter experts to elicit their needs and expectations from the new system. The culmination of this step is the creation of a set of requirements and priorities derived from various end-user requests, which serve as the blueprint for the system's development (Sommerville, 2016). Typically, this process involves conducting a SWOT analysis of the existing systems to identify their strengths, weaknesses, opportunities, and threats. The findings of this analysis are then organized into a SWOT matrix, delineating strengths and weaknesses in one row and opportunities and threats in another (Sommerville, 2016) **(see Figure 10).** This matrix provides a visual representation of the current system's status across four quadrants: offensive, defensive, adjustment, and severe trouble. Understanding this information is crucial as it helps evaluate the current system's position and define its goals before delving deeper into the software development process (Sommerville, 2016). By establishing a clear understanding of the system's requirements and objectives at this stage, organizations can ensure that the subsequent stages of the SDLC are aligned with the desired outcomes, ultimately leading to the successful development and implementation of the software solution.

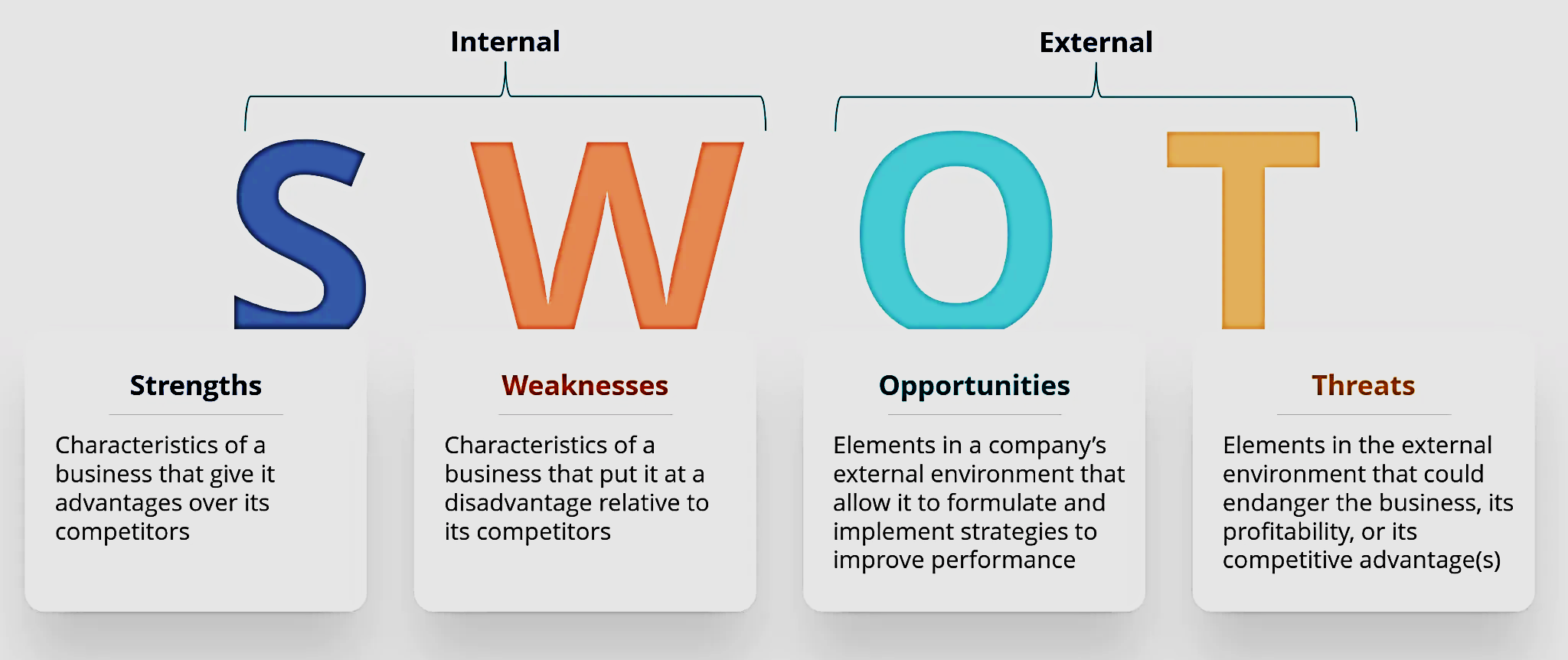


Figure 10: SWOT Analysis

#### **2.3.2.3. Design**

During the system design phase, the primary objective is to create a technical blueprint detailing the development or modification of the proposed system (Sommerville, 2016). This blueprint encompasses various elements, including the technical architecture, which specifies the necessary hardware, software, and telecommunications equipment to support the system. Additionally, the system design defines system components such as outputs, inputs, user interface elements, procedures, and personnel role (Sommerville, 2016). Furthermore, the design phase involves assessing the interrelationships among these components to ensure seamless integration and functionality. By meticulously planning the technical aspects of system design, organizations establish a foundation for subsequent development stages, enabling efficient implementation and minimizing the risk of technical issues during deployment (Sommerville, 2016). Ultimately, the system design phase plays a crucial role in translating identified requirements into a tangible plan, guiding the development team in constructing the desired software solution.

#### **2.3.2.4. Development**

In the system development phase, the focus shifts towards actualizing the components outlined in the design blueprints (Sommerville, 2016). This step involves either creating or procuring the necessary system elements as per the specifications laid out in the design phase. Here, the technical architecture comes to life, with any required equipment being purchased and deployed. Simultaneously, software development takes place, and the developed software is integrated into the system's hardware environment (Sommerville, 2016). It's worth noting that as development progresses, the involvement of end users typically diminishes, as their role transitions from active participation to providing feedback and testing. By the conclusion of this stage, the new or modified system is prepared for testing, marking a significant milestone in the SDLC process (Sommerville, 2016).

#### **2.3.2.5. Testing**

System testing is a critical stage in the SDLC, encompassing the verification process to ensure that the information system operates effectively and meets the specified requirements outlined by users (Pressman, 2014). During this phase, detailed test procedures, along with their expected outcomes, are documented and executed rigorously. Any necessary modifications identified during testing are implemented, and in some cases, it may be imperative to revert to previous stages of system development to address issues effectively (Pressman, 2014).

Typically, testing follows one or more of the following approaches:

* **Unit Testing**: Each individual unit of the system is tested independently. For instance, specific pieces of code such as printing, data input, transaction processing, and invoice generation are tested to ensure their functionality (Pressman, 2014).
* **System Testing:** This involves verifying that the different units of code, developed for a particular system, operate correctly when integrated into a cohesive system. For instance, in an accounting system, interactions among various components like transaction processing, inventory management, and invoicing are tested to ensure seamless operation (Pressman, 2014).
* **Integration Testing:** Similar to system testing but at a broader level, integration testing verifies that different constituent systems function properly when operating together. For example, integration testing might validate if the accounting system accurately updates the human resources system when processing sales transactions, ensuring that incentives and commissions are correctly calculated and recorded (Pressman, 2014).

By conducting thorough testing at each of these levels, organizations can identify and rectify any issues or discrepancies, ensuring that the final system meets the desired standards of performance and functionality.

#### **2.3.2.6. Implementation**

The Implementation step marks the activation of the system's operation, but before transitioning to full-scale usage, two essential tasks must be completed:

1. Firstly, comprehensive documentation must be prepared, covering every aspect of the system in meticulous detail. This documentation encompasses system description, operation procedures, troubleshooting guides, and maintenance protocols (Pressman, 2014).
2. Equally vital is the provision of training for end users prior to system launch. Training sessions may be conducted through workshops, classroom sessions, or accessible online resources, or often, a combination of these methods is employed to ensure optimal user comprehension and proficiency (Pressman, 2014).

Once documentation is finalized and training is completed, the system is ready for installation and operation (Pressman, 2014). There are four primary approaches to implementation, often referred to as the four p's:

* **Plunge Implementation:** In this approach, the old system is discarded entirely, and the new system is immediately put into use. While this method is expedient, it carries a significant risk if the new system encounters issues or fails to operate correctly (Pressman, 2014).
* **Parallel Implementation:** Both the old and new systems are utilized concurrently for a specified period, after which the old system is phased out. While this approach mitigates risk, it may lead to concerns among employees about managing dual systems (Pressman, 2014).
* **Phased Implementation: T**his method involves introducing the new system gradually, typically by implementing it in phases, such as room accounting followed by restaurant accounting, and so on. Each phase's success precedes the transition to the next, making it a conservative approach that may require an extended implementation period (Pressman, 2014).
* **Pilot Implementation:** A small group of employees initially operates the new system until its functionality and reliability are verified. Once proven successful, the system is gradually rolled out to the wider user base. For instance, in a retail chain, the new accounting system may be piloted in select stores before being implemented across the entire chain (Pressman, 2014).

#### **2.3.2.7. Maintenance and Review**

During this step, the primary objective is to sustain long-term system functionality and address any modifications or updates required to align with evolving business needs. Central to achieving this goal is the establishment of a helpdesk to provide ongoing user support and implement necessary modifications (Pressman, 2014). By establishing a dedicated helpdesk, organizations ensure continuous support and service for the system, thereby enhancing its longevity and effectiveness (Pressman, 2014). Furthermore, fostering a culture of receptiveness to suggestions for improvement and fault correction among system users is essential. Encouraging users to provide feedback and suggestions promotes continuous refinement and optimization of the system, ultimately enhancing its performance and usability over time. This proactive approach ensures that the system remains adaptive and responsive to the dynamic requirements of the business environment, thereby maximizing its value and utility to the organization (Pressman, 2014).

# **Chapter Three: Methodology**

Developing a Healthcare and Nutrition Tracking System (HNTS) in Lebanon presents unique challenges, particularly in sourcing existing systems and gathering relevant data. Despite these hurdles, diligent research efforts, including extensive online searches and inquiries via social media platforms, were undertaken to identify healthcare facilities and nutrition tracking systems within the region (AKTAS M. , 2023). Regrettably, the search yielded limited results, with few healthcare facilities in Lebanon having dedicated software solutions for nutrition tracking. However, leveraging online resources proved invaluable, as we discovered existing systems and applications from other regions that could serve as models or inspiration for the HNTS. One notable find was the Health & Wellness Clinic (HWC), a prominent healthcare facility in Lebanon, which lacked a specialized nutrition tracking system. Recognizing this gap, we embarked on designing and implementing a tailored software application to address the specific needs of HWC and its patient. To inform our development process, we adopted a structured research approach, starting with a broad overview and gradually narrowing down to specific insights. Despite the challenges of accessing local data, we were fortunate to secure interviews with healthcare professionals and subject-matter experts, including the clinic manager of HWC (AKTAS M. , 2023). These interviews provided invaluable insights and guidance, ensuring that the HNTS would be meticulously tailored to meet the clinic's unique requirements and operational goals.

## **3.1 Research Philosophy**

The "HNTS" research philosophy includes the basic concepts and assumptions that guide the system's development, implementation, and use. It shapes the researcher's understanding and approach to issues in healthcare facility management. Within the "HNTS" context, researchers commonly employ positivism, interpretivism, and pragmatism as their guiding research philosophies.

**Positivism:** The empirical and scientific methods of study play a significant role in positivism, particularly within the context of healthcare under "HNTS" (AKTAS M. , 2023). A positivist perspective in healthcare may emphasize the objective evaluation and validation of healthcare practices. Such an approach would probably lean towards utilizing qualitative data analysis methods.

**Interpretivism:** Interpretivism focuses on understanding how individuals interpret their experiences, particularly in the realm of healthcare within "HNTS" (AKTAS M. , 2023). This philosophy entails exploring the beliefs, attitudes, and behaviors of users and patients. Qualitative methods such as interviews and observations help reveal insights into the social dynamics and user experiences within healthcare facilities.

**Pragmatism:** Pragmatism in the development of "HNTS" is characterized by a focus on practical problem-solving (AKTAS M. , 2023). Researchers adopting a pragmatic approach prioritize blending quantitative and qualitative methodologies to address specific challenges and generate practical outcomes. The aim is to identify effective solutions that enhance user satisfaction and streamline healthcare administration processes.

## **3.2 System Analysis Approach**

In the development of the healthcare and nutrition tracking system (HNTS), the System Development Life Cycle (SDLC) approach was utilized to ensure a structured and efficient process. Specifically, the Waterfall model, a linear and sequential SDLC methodology, was employed to maintain a clear progression through each phase. The SDLC stages (planning, analysis, design, implementation, and maintenance) were well adhered to. During the planning phase, the project's scope, objectives, and feasibility were defined. The analysis phase involved gathering detailed requirements through testing different applications, reading users feedback, watching videos, and reading articles to understand the needs and expectations of users and patients. The design phase focused on creating a robust architecture for the system, emphasizing a secure login with hashed passwords, a comprehensive dashboard, and an intuitive side menu for managing functionalities such as patient records, nutritional plans, and scheduling. Implementation was carried out using C# for the desktop application and MySQL for the database, ensuring seamless integration and performance. Finally, in the maintenance phase, continuous improvements and updates were planned to adapt to evolving user requirements and technological advancements. The Waterfall model's structured and sequential nature ensured that each phase was completed thoroughly before moving on to the next, leading to the development of a reliable, efficient, and user-friendly healthcare and nutrition tracking system.

## **3.3 Data Collection Methods**

In this methodology chapter of “HNTS” report, data collection methods include a variety of techniques tailored to capture comprehensive insights into user behaviors and system requirements. Observations serve as a foundational method, allowing researchers to directly observe user interactions with the system in real-world contexts, providing valuable firsthand observations. Additionally, extensive reading of relevant documents and books supplements this approach, offering theoretical frameworks and best practices in healthcare and nutrition tracking systems. Experimenting with various applications enables researchers to compare functionalities, identify strengths and weaknesses, and gather feedback from the users. Furthermore, watching instructional videos and user tutorials provides a good understandings of user experiences and challenges encountered during different user experiments. By employing this approach to data collection, the research aims to ensure a holistic understanding of user needs and preferences, which will lead to informing the development and enhancement of the healthcare and nutrition tracking system.

## **3.4 Data Analysis**

In the data analysis section of this chapter, I’ll carefully look at the information I gathered about the healthcare and nutrition tracking system. I'll check numbers and answers from documents or reports to see if there are any important trends or patterns. This helps me understand things like how healthy people are eating and how they're using such systems. I’ll also go through what people said in different interviews, videos, or wrote down, paying attention to common themes or ideas. This helps me learn about what users like or find hard about such system. By doing this, I hope to understand better what users need from the healthcare and nutrition tracking system, so I can make it work better for them in healthcare settings.

## **3.5 Ethical Considerations**

This section about ethical considerations, discusses how I am making sure everything I do is fair and respectful. This means I’ll talk about things like getting people's permission before using their information and keeping it private. It's important for me to be honest and clear about what I’m doing and why. This helps build trust with the people who are helping me with my research.

## **3.6 Limitations**

In discussing limitations, it's important to be open about the challenges I faced during the research process. One limitation was the difficulty in accessing systems that are like “HNTS”, as there weren't any nearby facilities available for direct observation. This required me to rely on online resources, which may not fully represent real-world scenarios. Additionally, technical issues with hardware devices, such as my laptop, posed challenges during data collection and analysis. These limitations affected the scope and depth of my research, and I acknowledge that they may have impacted the accuracy and comprehensiveness of my findings. Despite these challenges, I made every effort to mitigate their effects and provide transparent reporting of my research process.

## **3.7 Summary**

In summary, the methodology chapter outlines the approach to researching the healthcare and nutrition tracking system. I described how I collected data using various methods such as observations, online research, apps experiments and interviews. My data analysis procedures included examining both quantitative data, like reading documents, and qualitative data, such as observing users’ actions toward such systems, to gain a comprehensive understanding of user needs and preferences. I emphasized the importance of ethical considerations, ensuring privacy and informed consent throughout the research process. Despite limitations such as difficulty accessing physical facilities and technical issues with hardware devices, I remain committed to conducting rigorous and transparent research to inform the development of the healthcare and nutrition tracking system.

# **Chapter Four: Analysis and Design**

In this section, I will analyze the Software Development Life Cycle (SDLC) stages that are specific to my software ‘HNTS’. In addition to the database design and relationship diagrams, it will have comprehensive diagrams that show every step involved in the creation of ‘HNTS’. Nevertheless, to illustrate ‘HNTS’ software's capabilities and interface, we shall offer the data flow within the system and screenshots of its functional forms.

## **4.1 SDLC Analysis for HNTS**

The traditional Systems Development Life Cycle (SDLC) is a structured step-by-step methodology to develop an information system. The steps necessary to carry out the SDLC are the seven steps listed below:

### **Planning phase**

Throughout the initial stage of the software development life cycle, planning seems essential for the accomplishment of the project. The planning phase for a healthcare and nutrition tracking information system involves numerous benefits including feasibility assessment, clear requirement analysis, efficient project achievement, and cost and time estimation that help in budgeting resource management, and decision-making.

1. **Feasibility Study**: The activity that occurs at the start of ‘HNTS’ to ensure that it is a viable business proposition (Vogelsang, 2022). The feasibility report analyzes the need for and impact of the system and considers different alternatives for acquiring software. It helps in defining business objectives and outlining systems requirements, evaluating acquisition alternatives (estimating the costs and benefits), defining scope, assessing risks, identifying constraints, and developing a project plan.
2. **Organizational Feasibility:** Responsible for reviewing how ‘HNTS’ meets the needs of the healthcare facilities (must improve the performance of Healthcare facilities) (Vogelsang, 2022). It focuses on evaluating whether the organization's structure, culture, and resources align with the goals and requirements of implementing the HNTS. As a description of the organizational context, consider the healthcare culture and whether it aligns with the goals of implementing the HNTS. If the organization values innovation, collaboration, and continuous improvement, it may be more conducive to successful system adoption. Also, 29 evaluate the readiness of the healthcare providers to embrace the new system. Consider factors such as their technical skills, willingness to adapt to change, and the availability of training programs to bridge any skill gaps.
3. **Operational Feasibility**: An assessment of how HNTS will affect the daily working practices within healthcare facilities. First of all, it is user-friendly and easy to use. HNTS will facilitate the working of users who will have more time to improve other things in the time required for managing, and tracking patient’s health (Vogelsang, 2022). This project will give reports containing organized data about every single piece of information in it. HNTS is workable on a day-to-day basis, and it will be accepted by end-users (Healthcare providers) in their day-to-day work.
4. **Technical Feasibility:** According to the technical side of this project, it evaluates to what degree the proposed solutions will work as required and whether the right people and tools are available to implement the solution (Vogelsang, 2022). Thus, HNTS must work efficiently according to performance, availability, and stability. HNTS requires the implementation of a user-friendly interface, a DB system, and a secure infrastructure for data storage and retrieval. It is not necessary to obtain new hardware equipment, where the software can be downloaded on the existing computers. Healthcare facilities have the technical expertise for any update. For future capacity requirements, new features may be added so it may be needed for a new larger hard disk to store a lot of data added. Moreover, HNTS will be developed using a combination of an object-oriented language ‘C#’, and MySQL database.
5. **Economic Feasibility:** An assessment of the cost and benefits of different solutions to select that which gives the best value. Economic feasibility is divided into two parts cost and benefit.

### **Analysis Phase**

The investigation of the business and user requirements of an information system. Fact finding techniques are used to ascertain the user`s needs, and these are summarized using a requirements specification and a range of diagramming methods. System architecture (DB structure, UI, and integration requirements) was also defined as well as the potential challenges or constraints that need to be addressed.

#### **Requirements Gathering Techniques Used**

Requirement-gathering techniques establish procedures for task execution across different scenarios, which may lack associated procedures or have one or more. Each technique should be associated with at least one task. In my project, I exclusively employ interviews with Dr. Kassem Danach (MIT advisor). This technique facilitates in-depth information gathering through two-way dialogue. Multiple meetings were conducted to discuss requirements and outline objectives for the application. Utilizing interviews for gathering requirements necessitates posing several types of questions, including open-ended, closed, and probing questions. These questions help ensure comprehensive exploration of the project's needs and objectives.

#### **SRS Document**

The Software Requirements Specification (SRS) document is a crucial component in the software development process. It serves as a formal agreement between the client and the software development team, outlining the requirements and functionalities that the software should possess (Pressman, 2014). The SRS document contains two important parts: functional requirements, and non-functional requirements. Functional requirements consist of requirements that perform the activities that run the business. Which describes the functionalities and features that the software system should possess. It includes specific requirements such as user interfaces, data manipulation, system behavior, and performance criteria. Nonfunctional requirements define the performance levels of the business functions to be supported. This includes requirements related to performance, availability, security, reliability, usability, and interoperability.

#### **Functional Requirements**

The functional requirements for “HNTS” application are outlined in **(see Table 1).** These requirements detail the specific tasks and operations the system must be able to perform effectively to meet the user needs.

**Table 1: Functional Requirements**

|  |  |
| --- | --- |
| Requirement Number | Requirement Statement |
| 1 | The Information system should have a “log in” page which requires a username and password.. |
| 2 | In the “Edit Patients” form the system shall provide functionality for health providers to add, edit, and delete patient’s records. |
| 3 | In the “Vitals” form the system shall display details about patient’s health vitals and metrics |
| 4 | In the “Health Records” form the system shall view the health records of patients |
| 5 | In the “Appointments” form , it aims to schedule new appointments, view appointments date, sent the date for patients |
| 6 | The “Exercise” forms aims to allow health providers to set exercise plans, and assign them to patients. |
| 7 | The “Nutrition” forms aims to allow health providers to set nutrition plans, and assign them to patients |
| 8 | The “Report” form will generate report based on patient’s needs |
| 9 | The form has on close button which will exit the program. |
| 10 | If any “Backup” button is clicked will save a backup for the data, and restored by “Restore” Button. |

#### **Non-Functional Requirements**

The non-functional requirements for “HNTS” application are showed in **(see Table 2).** These requirements address the system's performance, security, usability, and other operational criteria essential for overall effectiveness.

**Table 2: Non-Functional Requirements**

|  |  |  |
| --- | --- | --- |
| Requirement Number | Requirement Name | Statement |
| 1 | Performance | The system shall respond to user requests within 4 seconds under normal load conditions |
| 2 | Availability | The HNTS shall be accessible to users 24/7, with scheduled maintenance communicated in advance. |
| 3 | Security | The system shall implement robust security measures, including encryption of sensitive data, secure login mechanisms |
| 4 | . Reliability | The system must be reliable with error-free |
| 5 | Usability | Easy to use |
| 6 | interoperability | The system shall implement robust security measures, including encryption of sensitive data, secure login mechanisms |

#### **Data Flow Diagram**

A data flow diagram represents a simple diagram showing how information is routed between different parts of an organization. It has an information focus rather than a process focus (Pressman, 2014). Nodes in DFD represent a set of interconnected components that include processes, data stores, and data flows. DFD is a graphical representation that illustrates how data moves through a system and how processes operate on that data. It is a modeling technique used in system analysis and design to show the flow of information within a system and the transformations that occur at various points (Pressman, 2014). Moreover, DFD is organized into multiple levels in which each level represents a different level of detail. Levels start at level ‘zero’ and progress to higher levels reaching level one or two’. Level zero provides an overview of the entire system, showing the interactions between the system and external entities. Level one decomposes the context diagram into major processes, data stores, and data flows, providing more detail about the system's internal workings. Level two further decomposes processes from the Level 1 DFD into sub-processes, providing increasing levels of detail (Pressman, 2014).

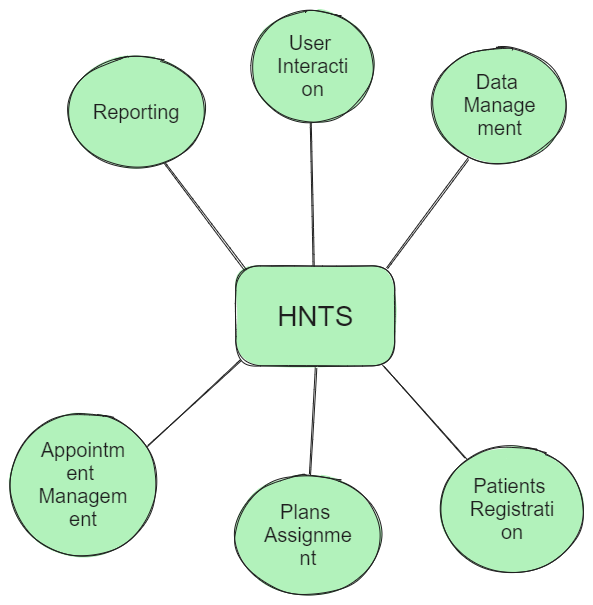


Figure 11: DFD at level 0 of HNTS

* **Level ‘0’ DFD for “HNTS”:** In this part, it provides an overarching view of the entire system, illustrating interactions between the system and external entities **(see Figure 11).** Each one of them provides a specific feature:

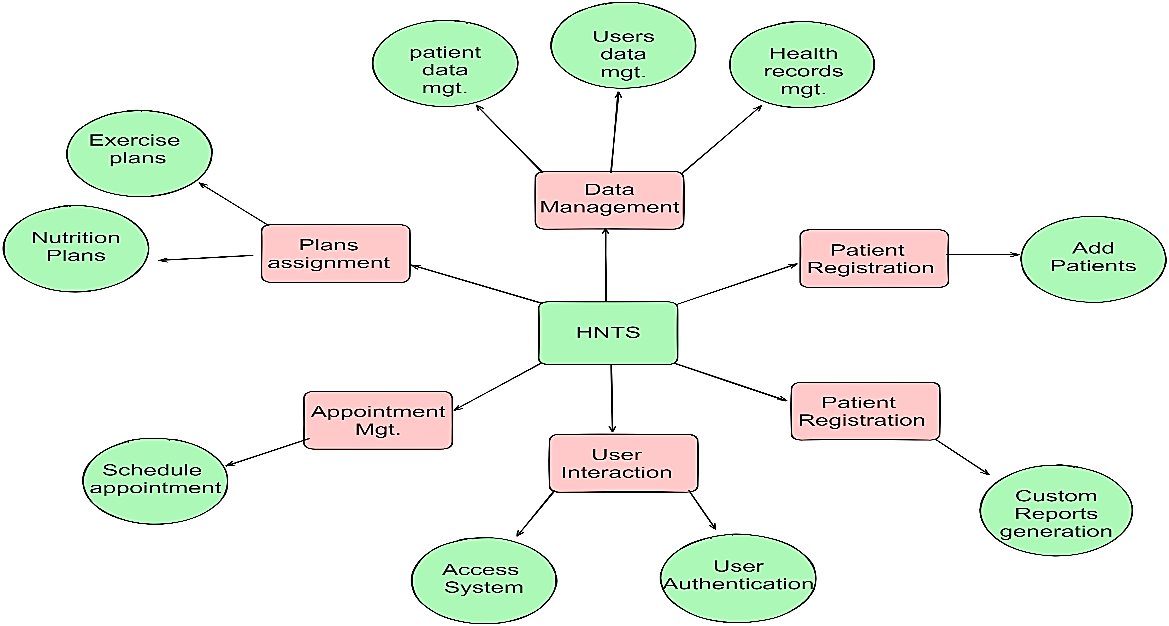
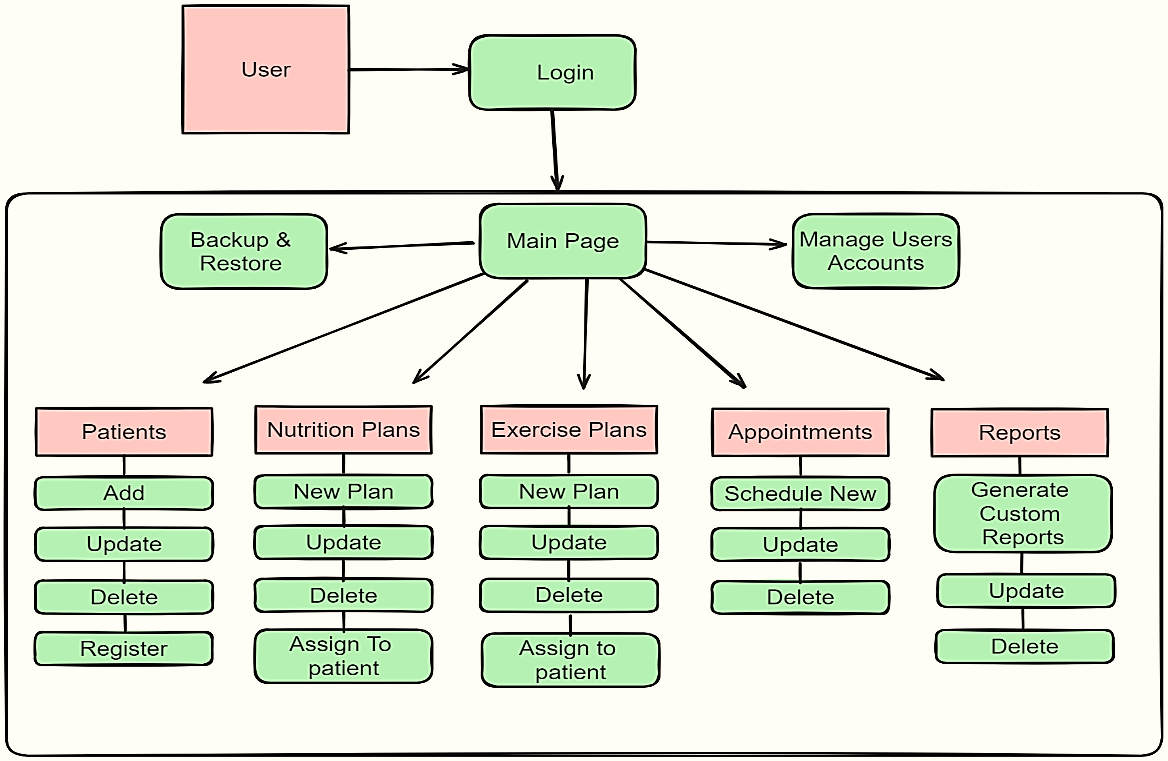


Figure 12: DFD at level 1 of HNTS

* **Level ‘1’ DFD for “HNTS”:** In this part, processes are divided into sub-processes **(see Figure 12).** Each one of them provides a specific feature:

Figure 13: DFD at level 2 of HNTS



* **Level ‘2’ DFD for HNTS:** A user logs in into the main page where he manages patient’s data, health records, vitals, create new nutrition and exercise plans and assign them to the user, schedule appointments, generate reports, manage user data, and backup/restore for the whole system database**. (see figure 13)**

#### **4.1.2.4 Activity Diagram**

An activity diagram is a crucial component of the Unified Modeling Language (UML) used to represent the dynamic aspects of systems (Booch, 1999). Specifically, for the healthcare and nutrition tracking system (HNTS), an activity diagram visually depicts the workflow and the sequence of activities involved in various processes, such as user authentication, patient management, nutritional plan management, and appointment scheduling. This type of diagram helps in understanding the flow of control from one activity to another, providing a clear visualization of the system's operational process. Each activity represents a single step in the workflow, while the transitions between activities show the flow of control. Using UML activity diagrams facilitates the identification of potential bottlenecks and inefficiencies in the system's processes, thereby aiding in optimizing and streamlining operations. Activity diagrams are particularly useful for modeling the dynamic behavior of a system and for detailing complex workflows (Booch, 1999).

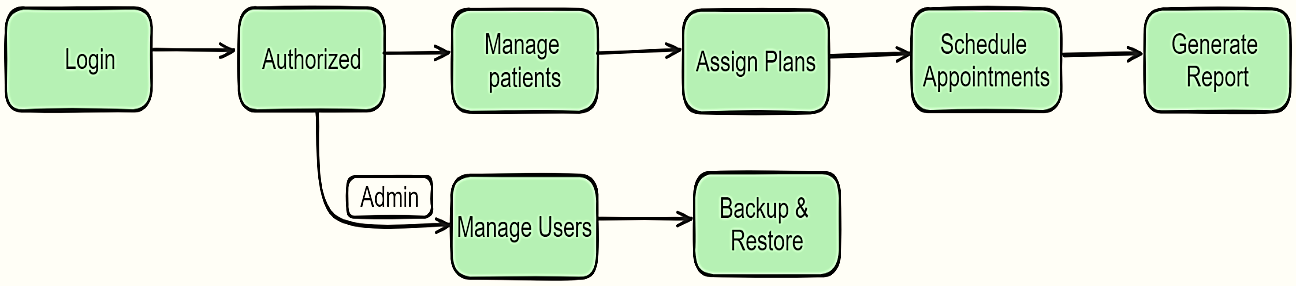


Figure : Activity Diagram

### **Design Phase**

The design phase of the lifecycle defines how the finished information system will operate.

#### **UML diagrams for LMS**

Unified Modeling Language (UML) diagrams are a standardized set of graphical notations used for modeling software systems, visualizing their design, and documenting their architecture. There are several types of UML diagrams, each serving a specific purpose such as use case, activity diagram, and class diagram.

#### **Use Case Diagrams**

Illustrates the interactions between a system and its external entities (actors) by representing various use cases and their relationships. **(see Figure 14)**

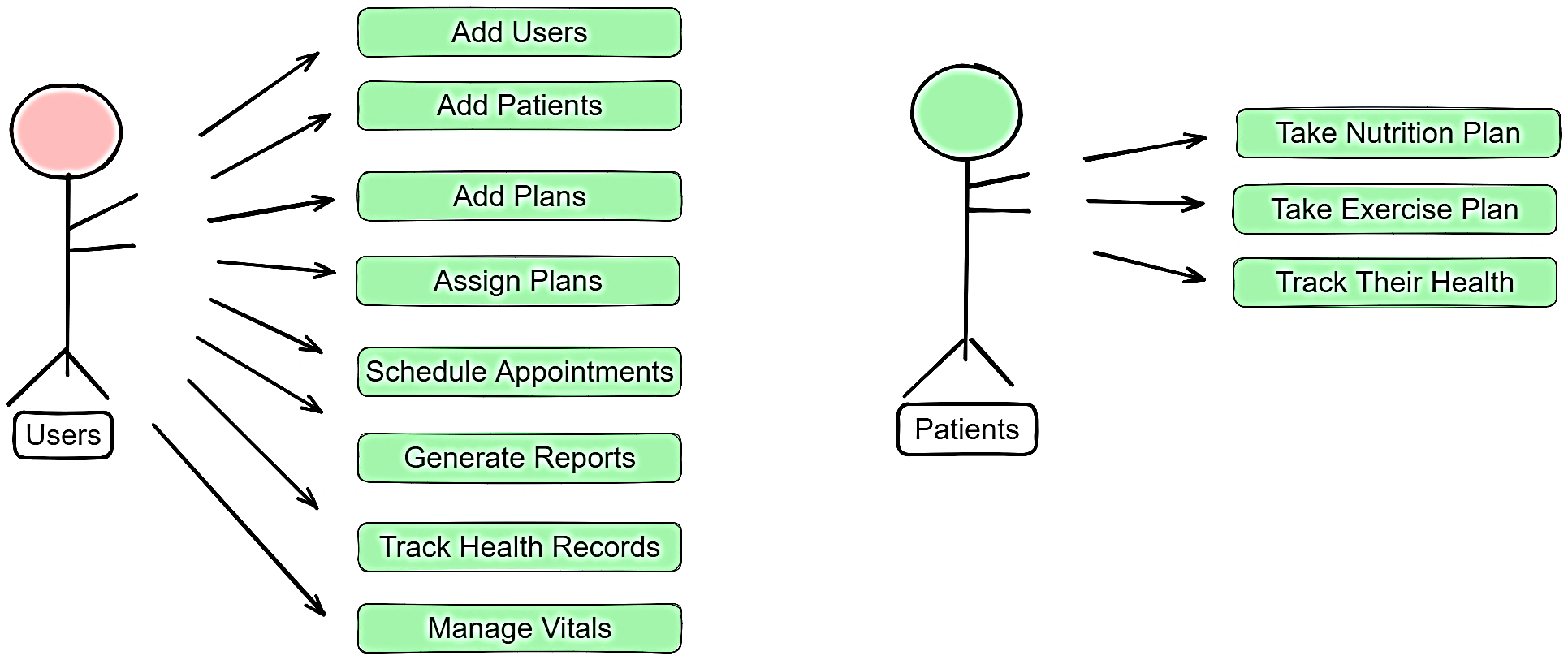


Figure 15: Use-Case diagram of HNTS

#### **DB Design for HNTS**

This section contains visual representations of a database structure for HNTS and relationships between entities.

#### **ERD**

Figure 16: HNTS DB's ERD Model

#### **4.1.3.2.2. Relational Schema**

A relational schema is a set of organized structures, including tables, defining how data is stored and accessed in a relational database. It provides a blueprint for the logical structure of the database, specifying the tables, their attributes (columns), data types, and the relationships between the tables. **(see Figure 15)**

### **Used Technologies**

#### **Visual Studio**

Visual Studio is an integrated development environment (IDE) created by Microsoft for software development. It provides a comprehensive set of tools, features, and services to support the entire development lifecycle, from designing and coding to testing, debugging, and deployment. Visual Studio is widely used by developers for building various types of applications, including desktop, web, mobile, cloud, and enterprise-level applications. VS is available in both a free “community” edition and a paid commercial version. It contains C#, a programming language that allowed the build of HNTS. In addition, it allows designers and developers to test their work without an internet connection.

#### **MySQL Database**

MySQ**L** is an [open-source](https://en.wikipedia.org/wiki/Open-source_software) [relational database management system](https://en.wikipedia.org/wiki/Relational_database_management_system) (RDBMS) . A [relational database](https://en.wikipedia.org/wiki/Relational_database) organizes data into one or more data tables in which data may be related to each other; these relations help structure the data. SQL is a language that programmers use to create, modify and extract data from the relational database, as well as control user access to the database. In addition to relational databases and SQL, an RDBMS like MySQL works with an [operating system](https://en.wikipedia.org/wiki/Operating_system) to implement a relational database in a computer's storage system, manages users, allows for network access and facilitates testing database integrity and creation of backups.

#### **C# Programming Language**

C# (pronounced "C sharp") is a modern, object-oriented programming language developed by Microsoft. It is part of the Microsoft .NET platform and is designed for building a wide range of applications, including desktop, web, mobile, cloud, and enterprise applications. C# is known for its simplicity, type-safety, and versatility, making it a popular choice among developers for various software development projects. It is a free software tool and can be related to several databases software (MySQL). First, Visual Studio tools were used for the UI design. Second, coding was used to integrate UI tools. Thirdly, utilizing a C# code that links the database to Visual Studio and uses SQL commands to manage, store, and retrieve data as needed. Testing was done to make sure everything was running smoothly and that every code was operating precisely as needed, following each coding step, procedure, and function.

#### **Microsoft Word**

Microsoft Word is a word-processing software application developed by Microsoft. Played a significant role in the development of the Question Master System (HNTS). It is part of the Microsoft Office suite, which includes a range of productivity applications such as Excel, PowerPoint, and Outlook. Microsoft Word is one of the most widely used word processing programs globally and is designed to create, edit, format, and print documents.

### **Development Environment Setup**

Initially, Visual Studio (VS) was installed to be utilized for UI design and coding in the object-oriented C# language. After that, Visual Studio for HNTS was connected and an installation of the MySQL database was made.

### **Testing**

Testing for a Healthcare and Nutrition Tracking System (HNTS) involves the process of systematically evaluating the system to ensure that it meets specified requirements, functions as intended, and is free of defects or errors. Testing is a critical phase in the software development life cycle, helping identify and address issues before the HNTS is deployed for actual use.

## **HNTS Pages/ Forms**

An overview of the Healthcare and Nutrition Tracking (HNTS) graphical user interface (GUI) forms is given in this section. Every screenshot demonstrates a particular form inside the program and focuses on its main features and user interface. The screenshots are meant to serve as a visual representation of the “HNTS” user interface design and a guide for comprehending its different features and parts. By providing patients with a visual representation of the “HNTS” application's user interface and overall user experience, the screenshots improve the readability and comprehension of the text.

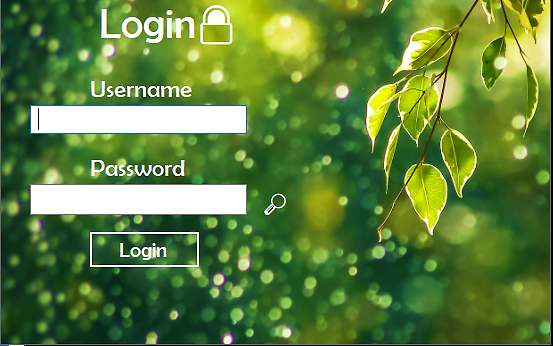


Figure 17: HNTS System Login Page

* This is the login page of “HNTS” that prompts users to enter their credentials, such as username and password, to gain access to the home page.

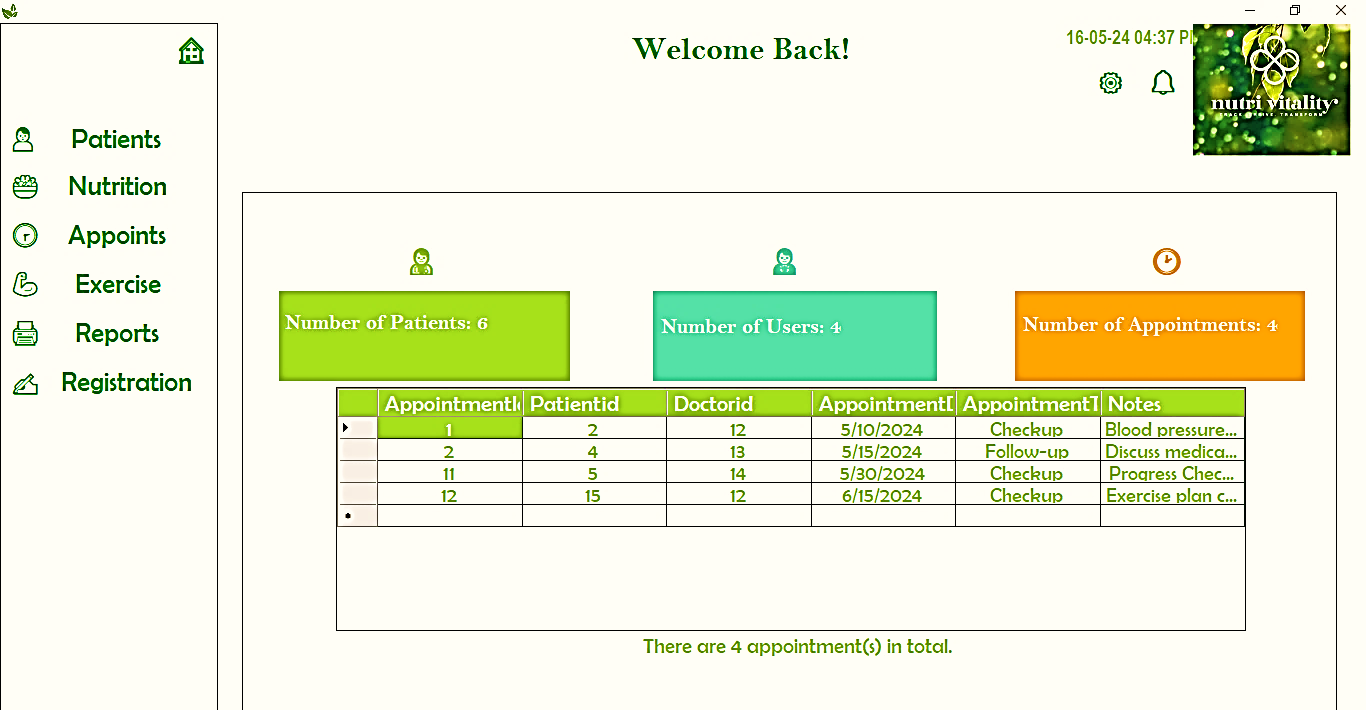


Figure 18: HNTS System Home Page

* This is the home page of “HNTS” where all actions, functions are done, such as: adding patients, users, nutrition plans, assigning plans to patients, appointment scheduling.

# **Chapter Five: Conclusions and Recommendations**

## **Conclusion**

The healthcare and nutrition tracking system is a very important system for organizing and optimizing various healthcare processes efficiently including managing patient’s health, tracking vitals, and user management. Healthcare and Nutrition Tracking System offers multifaceted benefits, it improves accessibility with a user-friendly interface, enabling easy navigation, and access. Moreover, HNTS contribute to effective resource utilization by automating routine tasks such as adding new patients, and plans management. This automation not only reduces manual workload for healthcare providers but also minimizes the likelihood of errors, ensuring the integrity of health records. In life, a healthcare and nutrition tracking system is an important tool that modernizes and enhances the work of healthcare facilities. By combining technological advancements with effective management practices, an HNTS not only benefits healthcare providers by simplifying their tasks, but also enriches the experience of health system users, ultimately contributing to the continued relevance and success of health facilities in the digital age.

## **Recommendation**

To enhance the efficiency of healthcare and nutrition tracking systems, it is important to grasp IT solutions for managing and utilizing data effectively. Staying informed about system updates and upgrades is essential. These updates ensure access to new features, enhanced measure of security, and bug fixes, thereby improving the reliability and performance of the health tracking system. Additionally, customizing the healthcare and nutrition tracking system to meet the required needs is crucial. Configure settings, fields, and workflows to be aligned with the unique requirements of your organization and user base. Given the rapid advancements in technology, it is important to provide regular training and updates to staff. This ensures they remain skilled in using the latest features and functionalities, leading to better data management and improved healthcare and nutrition outcomes.

## **5.3 References**

AKTAS. (2023).

AKTAS, M. (2023). User-engagement in digital health applications: a scoping review .

al., C. e. (2006).

Berger. (2006).

Berger, M. (2006). impact of a computerized information system on quality support in the ICU . *Nutrition*, 221-229.

Booch, G. (1999). The Unified Modeling Language User Guide. *Addison-Wesley*, 210.

Chaudhry. (2006).

Chaudhry, B. (2006). Systematic review: impact of health information technology on quality, efficiency,and costs of medical care. *Annals of internal Medicine*, 742-752.

Chung, G. &. (2019).

Fontelo, P. (2015). Open source software for libraries: Exploring the past, examining the present, envisioning the future. *Medical Reference Services Quarterly*, 94-104.

Griffin, A. C. (2019). Health tracking and information shring in the patient-centered are: a health information national trends survey (HINTS) study. *AMIA Annual Symposium Proceedings*, 1041.

Haux. (2006).

Haux, R. (2006). Health information systems-past, present, future. *International journal of medical informatics*, 268-281.

Jensen. (2013).

Jensen, T. (2013). Design principles for achieving integrated healthcare information systems. *Health informatics Journal*, 29-45.

Kamel, M. (2013). Remote patient tracking and monitoring system. *International Journal of Computer Science and Mobile Computing*, 88-94.

Ngafeeson. (2015).

Pressman, R. S. (2014). Software engineering: a practitioner's approach. *McGraw-Hill Education*.

Raghupathi, W. (2014). An empirical study of chronic diseases in the United States: A visual analytics approach to public health. *International Journal of Environmental Research and Public Health*, 1776-1792.

Rajkomar, A. (2019). Machine learning in medicine. *New England Journal of Medicine*, 1347-1358.

Smith, J. (2010). The evolution of healthcare information systems has been integral to modern medicine. *Journal of Medical Informatics*, 123-130.

Sommerville, I. (2016). Software Engineering. *Pearson Education Limited*.

Sultan, N. (2010). Cloud computing for education: A new dawn? *International Journal of Information Management*, 109-116.

Vogelsang, L. (2022). The Role of organizational Cutlure in Healthcare Innovation Adoption: A Case study analysis. *Journal of healthcare management*, 27-42.