Software Requirements Specification

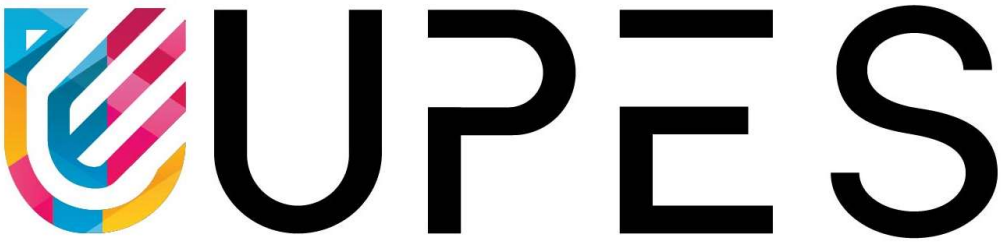
For

D.A.R.T.S: Doctor Allocation and Registration Tracking System

14th February, 2024

Prepared by

Specialization	SAP ID	Name
CCVT	500093618	Aditya Sharma
CCVT	500091345	Lakshit Joshi
CCVT	500093004	Vibhor Minocha



Department of Systemics
School Of Computer Science
UNIVERSITY OF PETROLEUM & ENERGY STUDIES,
DEHRADUN- 248007. Uttarakhand

Table of Contents

Revision History

Date	Change	Reason for Changes	Mentor Signature

1. INTRODUCTION

In the evolving landscape of healthcare, the Digital Automated Registration and Tracking System (D.A.R.T.S) emerges as a pivotal solution aimed at revolutionizing patient intake procedures and optimizing resource allocation within medical facilities. Amidst the complexities of modern healthcare administration, D.A.R.T.S stands as a beacon of innovation, offering a comprehensive platform to streamline processes, enhance patient care, and improve operational efficiency.

At the heart of D.A.R.T.S lies its ability to automate the cumbersome process of form filling, which has long been a bottleneck in patient intake procedures. By integrating seamlessly with hospital databases and electronic health records (EHRs), D.A.R.T.S significantly reduces the burden of manual data entry, expediting patient registration and minimizing errors. This not only enhances the patient experience but also enables healthcare providers to focus more on delivering quality care rather than administrative tasks.

Moreover, D.A.R.T.S revolutionizes resource allocation within healthcare facilities by dynamically assigning ICU beds, ward beds, and doctors to patients based on real-time data analysis and patient acuity levels. ICU beds, being critical resources in healthcare settings, are often in high demand and require meticulous management. D.A.R.T.S facilitates real-time monitoring of bed availability and patient acuity, enabling healthcare providers to allocate ICU beds judiciously, ensuring that the most critical patients receive prompt and appropriate care.

Similarly, D.A.R.T.S streamlines the allocation of ward beds by optimizing bed utilization and minimizing turnover times. By analyzing bed availability, patient acuity, and anticipated length of stay, D.A.R.T.S ensures efficient utilization of resources, reducing wait times for admission and enhancing patient flow within the hospital. This not only improves the overall efficiency of healthcare delivery but also enhances patient satisfaction by ensuring timely access to care.

Furthermore, D.A.R.T.S fosters a patient-centric approach to healthcare delivery by intelligently matching patients with appropriate healthcare providers based on factors such as specialty, availability, and patient preferences. Through its intelligent algorithms and patient profiling capabilities, D.A.R.T.S ensures that patients receive personalized and timely care from the most suitable medical professionals, optimizing clinical outcomes and patient satisfaction.

In emergency situations, where rapid intervention is paramount, D.A.R.T.S excels in its ability to expedite critical care delivery. By prioritizing patients based on acuity levels and resource availability, D.A.R.T.S ensures that the most urgent cases receive immediate attention, including prompt assignment of ICU beds and mobilization of medical teams. This not only improves patient outcomes but also enhances the overall resilience of healthcare systems in responding to emergencies.

Overall, D.A.R.T.S represents a paradigm shift in healthcare administration, enabling hospitals to operate more efficiently, effectively, and compassionately. By streamlining administrative processes, optimizing resource allocation, and fostering a patient-centric approach to care delivery, D.A.R.T.S is poised to transform the way healthcare is delivered and experienced, ultimately improving outcomes and saving lives.

1.1 Purpose of the Project

The purpose of the project is to develop and implement the Digital Automated Registration and Tracking System (D.A.R.T.S), a cutting-edge solution designed to revolutionize patient intake procedures, optimize resource allocation, and enhance healthcare delivery within medical facilities. Through automation of form filling processes, dynamic assignment of ICU beds, ward beds, and doctors to patients, and fostering a patient-centric approach to care delivery, D.A.R.T.S aims to streamline administrative tasks, improve operational efficiency, and ultimately, enhance the quality of patient care while reducing wait times and errors.

1.2 Target Beneficiary

The target beneficiaries of the Digital Automated Registration and Tracking System (D.A.R.T.S) project encompass patients, healthcare providers, and healthcare facilities alike. For patients, D.A.R.T.S streamlines administrative processes, reduces wait times, and enhances the overall healthcare experience. Healthcare providers benefit from improved efficiency, streamlined communication, and enhanced decision-making capabilities, leading to better patient outcomes. Healthcare facilities experience operational improvements, cost savings, and strengthened competitive positioning, ultimately resulting in enhanced quality of care and patient satisfaction across the healthcare ecosystem.

1.3 Project Scope

The scope of the project encompasses the development and implementation of the Digital Automated Registration and Tracking System (D.A.R.T.S) within healthcare facilities. This includes the automation of administrative tasks such as form filling, the dynamic assignment of ICU beds, ward beds, and doctors to patients based on real-time data analysis, and the integration of D.A.R.T.S with existing hospital systems and electronic health records. Additionally, the project involves training healthcare staff on the use of D.A.R.T.S, ensuring seamless adoption and effective utilization of the system to enhance patient care delivery, optimize resource allocation, and improve overall operational efficiency within healthcare facilities.

1.4 References

[1] Research Paper:
Title: Online Medical Form Filling Auto Typer Software
URL: https://rvsdataconversion.com

[2] Software:
Title: Online Medical Form Filling Auto Typer Software
URL: https://rvsdataconversion.com

[3] Website:
Title: Spring Boot Documentation
URL: https://spring.io/projects/spring-boot

[4] Website:
Title: MySQL Documentation
URL: https://www.mysql.com/products/enterprise/document_store.html

[5] Website:
Title: Gradle Documentation
URL: https://docs.gradle.org/current/userguide/getting_started_eng.html

2. PROJECT DESCRIPTION

2.1 Data/ Data structure

To implement the Digital Automated Registration and Tracking System (D.A.R.T.S), several data structures can be utilized to efficiently manage and process the required information. Here are some key data structures that can be employed:

1. **Linked Lists**: Linked lists can be used to represent patient records, where each node contains information such as patient ID, name, age, contact details, and medical history. Additionally, linked lists can be utilized to maintain queues for patient intake, ICU bed allocation, and ward assignment, facilitating efficient data management and processing.
2. **Hash Tables**: Hash tables can be employed to store and retrieve patient information based on unique identifiers such as patient ID or QR codes. This allows for fast access to patient records and enables efficient search operations, enhancing the responsiveness of the system.
3. **Priority Queues**: Priority queues are essential for managing the allocation of ICU beds, ward beds, and doctors to patients based on priority criteria such as medical urgency and severity of condition. By utilizing priority queues, the system can ensure that critical patients receive timely attention and appropriate care.
4. **Graphs**: Graphs can be utilized to model the relationships between patients, doctors, and healthcare facilities within the system. For example, a graph representation can be used to depict the network of hospitals and the routes for patient referrals. This allows for effective navigation and coordination of patient care across multiple healthcare providers.
5. **Trees**: Trees can be employed to represent hierarchical structures within the system, such as the organizational hierarchy of healthcare facilities or the specialization hierarchy of medical staff. This enables efficient organization and management of resources, facilitating streamlined communication and decision-making processes.

2.2 SWOT Analysis

Strengths:

1. Efficiency: D.A.R.T.S automates administrative tasks, reducing paperwork and streamlining patient intake processes, thereby saving time and improving efficiency.
2. Resource Optimization: By dynamically allocating ICU beds, ward beds, and doctors based on real-time data analysis, D.A.R.T.S optimizes resource utilization within healthcare facilities, ensuring timely access to care and improving patient outcomes.
3. Improved Patient Experience: D.A.R.T.S enhances the patient experience by minimizing wait times, reducing errors, and enabling personalized care delivery, leading to higher levels of patient satisfaction.

Weaknesses:

1. Initial Investment: Implementing D.A.R.T.S requires a significant initial investment in terms of technology infrastructure, software development, and staff training, which may pose financial challenges for some healthcare facilities.
2. Technical Dependencies: D.A.R.T.S relies on the integration of various technologies and systems, making it susceptible to technical failures, interoperability issues, and cybersecurity risks, which could potentially disrupt operations and compromise patient data security.

Opportunities:

1. Market Expansion: D.A.R.T.S presents opportunities for market expansion by targeting healthcare facilities looking to enhance operational efficiency, improve patient care delivery, and stay competitive in the rapidly evolving healthcare landscape.
2. Partnerships and Collaborations: Collaborating with technology vendors, healthcare providers, and regulatory bodies can facilitate the adoption and implementation of D.A.R.T.S on a larger scale, driving widespread adoption and market penetration.

Threats:

1. Regulatory Compliance: D.A.R.T.S must comply with stringent regulatory requirements, privacy laws, and data protection regulations, failure to do so could result in legal liabilities, penalties, and damage to reputation.
2. Resistance to Change: Resistance from stakeholders, including healthcare providers and staff, towards adopting new technology and workflows could hinder the successful implementation and adoption of D.A.R.T.S, necessitating effective change management strategies.

2.3 Project Features

The Digital Automated Registration and Tracking System (D.A.R.T.S) project boasts several key features designed to enhance healthcare delivery and streamline administrative processes:

1. **Automated Patient Intake**: D.A.R.T.S automates the patient intake process by digitizing and streamlining administrative tasks such as form filling and data entry. This reduces paperwork, minimizes errors, and expedites patient registration, leading to improved efficiency and patient satisfaction.

2. **Real-time Resource Allocation**: D.A.R.T.S dynamically assigns ICU beds, ward beds, and doctors to patients based on real-time data analysis and patient acuity levels. This ensures timely access to care, optimizes resource utilization, and enhances patient outcomes.

3. **Integration with Hospital Systems**: D.A.R.T.S seamlessly integrates with existing hospital systems and electronic health records (EHRs), facilitating efficient data exchange and interoperability. This allows for seamless communication, streamlined workflows, and enhanced decision-making capabilities for healthcare providers.

4. **Personalized Care Delivery**: D.A.R.T.S enables personalized care delivery by matching patients with appropriate healthcare providers based on factors such as specialty, availability, and patient preferences. This ensures that patients receive tailored and timely care, leading to improved clinical outcomes and patient satisfaction.

5. **Secure Data Storage**: Patient data collected by D.A.R.T.S is securely stored in the cloud, ensuring privacy and compliance with regulatory requirements. This allows for easy access to patient information for healthcare providers while maintaining the confidentiality and integrity of patient data.

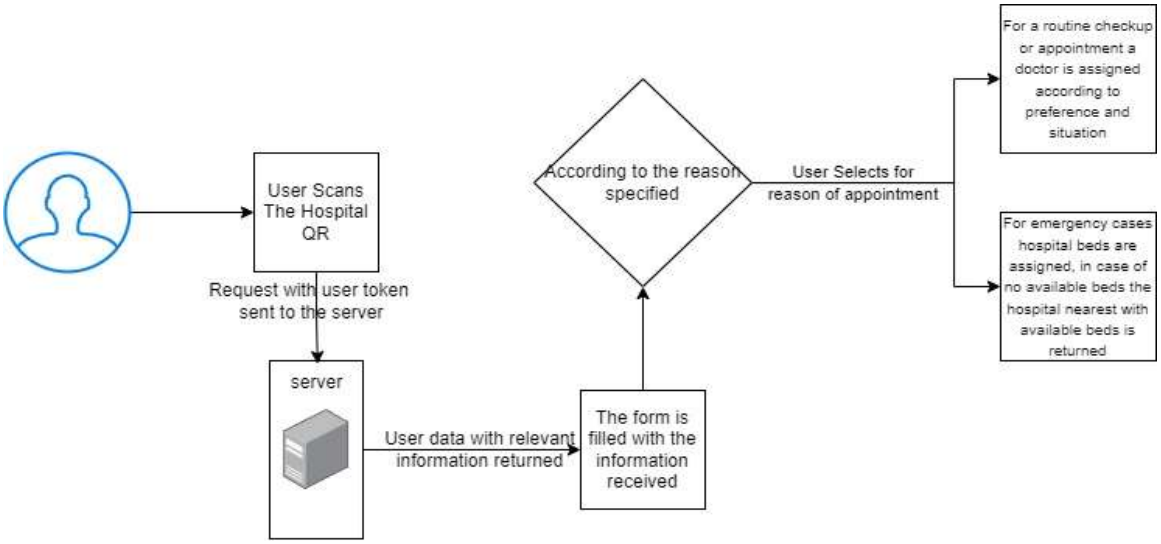
6. **Scalability and Flexibility**: D.A.R.T.S is designed to be scalable and adaptable to the needs of different healthcare settings, from small clinics to large hospitals. Its modular architecture allows for easy customization and expansion, ensuring that it can grow alongside evolving healthcare requirements.

Overall, these features make D.A.R.T.S a comprehensive solution for optimizing patient intake processes, enhancing resource allocation, and improving the overall quality and efficiency of healthcare delivery.

2.4 Design and Implementation Constraints

1. Technical Compatibility
2. Data Security and Privacy
3. Interoperability
4. Resource Limitations
5. Regulatory Compliance
6. User Training and Adoption
7. Reliability and Availability

2.5 Methodology Diagram:



3. SYSTEM REQUIREMENTS

3.1 User Interface

Our project's user interface, built using HTML and CSS, offers seamless access to features like checking crime statistics, registering complaints, and searching missing person registries. Optimized for both desktop and mobile use, it provides a user-friendly experience. Developed in Visual Studio Code, it efficiently captures user inputs, processes them, and communicates with the backend, ensuring smooth data flow and effective interaction.

3.2 Protocols

Wired Communication Protocol using HTTP:
HTTP (Hypertext Transfer Protocol) is a software-based communication protocol that enables the exchange of data between a client and a server over a wired network. It operates over a synchronous connection, where the client sends requests and receives responses in a well-defined manner, unlike UART's asynchronous communication.

IEEE Wireless Standard using HTTP:
IEEE 802.11 [Wi-Fi] Wireless LAN Media Access Control and Physical Layer specification can be thought of as the underlying infrastructure for wireless networks that support HTTP based communication. Devices compliant with 802.11 standards, such as 802.11a, b, g, etc., provide the foundation for wireless connectivity. In this context, HTTP is used for transmitting data over Wi-Fi networks, and products that meet 802.11 standards are certified for seamless Wi-Fi communication.

4. NON-FUNCTIONAL REQUIREMENTS

4.1 Performance requirements

The following are the four basic requirements needed:

- Provisioning and Authentication
- Compliance
- Data Processing Speed
- Software Updates and Maintenance

4.2 Security requirements

- Authentication and Authorization: Users must authenticate themselves securely, and the system should implement role-based access control to ensure that users can only access authorized information and functionality.

- Data Integrity: The system should guarantee data integrity by preventing unauthorized tampering or modification of records, especially for critical information like reported incidents and missing person records.

- Backup and Disaster Recovery: Regularly back up data to ensure it can be restored in case of data loss or system failure.

4.3 Software Quality Attributes

Availability:

Availability of our project depends upon the availability of individual services used in our project. As we are using MySQL and its availability as per Google Service level agreement (SLA) is 99.95 %.

Usability:

Usability is defined as the ease with which users can achieve their goals using the web interface. Our project prioritizes a simple and user-friendly web interface, keeping ease of-use in mind to ensure effective and efficient interaction for all users.

Testability:

The project is designed with modularity in mind, enabling easy breakdown into sub-components. Each service, can be individually tested and verified for functionality and reliability. This approach enhances the project's overall testability and maintainability.

Portability:

The server-side application is platform independent and can be run on any architecture.