INDIANA UNIVERSITY
BLOOMINGTON

ENGR-E516 Engineering Cloud Computing

Cloud based Blood Bank System

Anurag Ganguly
Ayantika Nandi
Atharva Atul Joshi

Indiana University Bloomington

March 22, 2024

Project Title

Our project, **Cloud-Based Blood Bank System** (CBBBS), is designed to revolutionize the way blood banks operate by leveraging cloud computing technologies. It aims to streamline the process of blood donation and acquisition, making it faster and more efficient for blood banks to match donors with recipients, manage storage spaces, and locate blood deposit centers

Project Goals

The Cloud-Based Blood Bank System (CBBBS) is an innovative solution to the numerous issues that traditional blood banks encounter. These issues include inefficient inventory management, the difficulty of matching donors with beneficiaries, and physical infrastructure constraints that affect accessibility and scalability. The CBBBS's move to a cloud-based framework intends to overcome these difficulties by providing a stable and extensible platform. This platform can handle large amounts of data, allows for real-time processing, and supports complex algorithms for accurate matching of blood types and donor compatibility.

The primary goal of CBBBS is to create an efficient, scalable, and reliable system that enhances the management of blood banks by automating the storage, retrieval, and matching of blood types and donor information. This system will be hosted on the cloud, providing global accessibility, high availability, and secure storage solutions.

The CBBBS's novel features include secure storage of donor information and blood types, data privacy through encryption, and adherence to regulatory norms. The technology improves the efficiency of storage space management by giving real-time information on blood stock levels and expiration dates, allowing for improved resource planning and allocation. It also incorporates GPS and mapping technology to assist users in discovering nearby blood donation centers, as well as enabling blood banks to locate places with high donation potential for mobile donation camps. Our system should handle a vast amount of transactions and data, serving both small and large blood banks. The CBBBS's use of AWS enables it to access a variety of cloud services, including database management for donor information, platform services for bespoke application development, and infrastructure services, to provide a scalable and reliable hosting solution.

Furthermore, CBBBS has an improved donor-recipient matching algorithm that greatly decreases the time required to locate appropriate blood matches, potentially saving lives in emergency scenarios. The system features an easy and user-friendly interface that is available via both online and mobile platforms, allowing blood bank operators and donors to interact with the system at any time and from any location. Cloud-Based Blood Bank System, which leverages the scalability, flexibility, and cost-effectiveness of cloud computing via AWS, not only aims to improve operational efficiencies and donor-recipient matching, but also sets a new standard in blood bank management, ultimately contributing to life saving. This initiative is a big step forward in using technology to address critical healthcare concerns.

Some of the goals of our system include:

Blood Type and Donor Information Management: To securely store and manage detailed records of donors, including their blood types, contact information, and donation history. The system will use encryption and comply with data protection regulations to ensure the privacy and security of this sensitive information.

Storage Space Management: Dynamically manage the storage space for different blood types, enabling blood banks to efficiently use their resources and reduce wastage. The system will provide real-time updates on blood stock levels, expiry dates, and space utilization, aiding in the planning and allocation of resources.

Cloud-Based Infrastructure: The entire system will be hosted on a secure cloud platform, ensuring scalability, reliability, and accessibility. The cloud infrastructure will enable the system to handle high volumes of transactions and data, making it suitable for both small and large blood banks.

Donor-Recipient Matching Algorithm: Implement advanced algorithms to match donors with recipients quickly and accurately, based on blood type compatibility and other medical criteria. This feature will significantly reduce the time needed to find suitable blood matches, potentially saving lives in emergency situations.

User Interface and Accessibility: Design an intuitive and user-friendly interface for both blood bank operators and donors. The interface will be accessible via web ensuring that users can interact with the system anytime, anywhere.

The choice of cloud computing as the backbone of the CBBBS is strategic, aimed at leveraging the cloud's scalability, flexibility, and cost-effectiveness. The project will explore the use of various cloud services, including database as a service (DBaaS) for managing donor and blood type information, platform as a service (PaaS) for developing custom applications and interfaces, and infrastructure as a service (IaaS) for ensuring robust, scalable hosting solutions. Additionally, the project will implement cloud security best practices to protect data integrity and confidentiality.

This midterm project proposal outlines the vision for a transformative Cloud-Based Blood Bank System.

Related Work and Gap Analysis

P. Priya, V. Saranya, S. Shabana, and Kavitha Subraman developed a system in 2014 that uses Technopedia Geographic Information System, Blood Bank, and Push technology to improve blood donor information and management via an Android mobile app. This programme sought to alleviate the limitations of offline blood bank systems by implementing Cloud Computing. Their suggested system provides users with information on nearby donors, hospitals, and blood banks,

thereby increasing accessibility and efficiency.

However, one significant flaw in present cloud-based blood bank systems is poor interaction with the broader healthcare ecosystem, which includes healthcare institutions and Electronic Health Record (EHR) systems. Current systems frequently function in isolation, preventing the smooth interchange of critical patient information such as blood type, medical history, and transfusion requirements with healthcare professionals. Addressing this integration gap could transform patient care by providing healthcare professionals with instant access to critical data such as compatible blood units and transfusion records, streamlining the blood ordering process, lowering administrative burdens, and improving patient safety and outcomes. Furthermore, improved coordination between blood banks and healthcare facilities, particularly during emergencies or mass casualty events, could provide prompt access to critical blood products. Bridging this integration gap represents a significant opportunity for innovation in blood transfusion service management and delivery.

Proposed Tasks

For a comprehensive project like the Cloud-Based Blood Bank System (CBBBS) utilizing Amazon Web Services (AWS), the development lifecycle encompasses a range of tasks from initial planning and system design to deployment, testing, and eventual use. Below is an expanded task list that covers these stages in detail:

Task 1: Project Planning and Requirements Analysis

Gather and analyze system requirements through group meetings.

Define project scope and select appropriate AWS services and technologies.

Task 2: System Design and Architecture

Design system architecture focusing on scalability, reliability, and security.

Plan data backup and disaster recovery strategies with AWS S3 (Simple Storage Service is a scalable cloud storage solution for storing and retrieving any amount of data, anytime, from anywhere on the web) and AWS Backup (backup service that makes it easy to centralize and automate the backup of data across AWS services in the cloud and on-premises).

Task 3: Development Environment Setup

Configure the development environment using AWS and establish version control with AWS CodeCommit or Git.

Task 4: Application Development

Develop backend and frontend components using CSS, HTML and JavaScript.
Implement authentication with AWS KMS.

Task 5: Testing and Quality Assurance

Objective: To ensure the CBBBS is efficient, scalable, and reliable by implementing thorough

testing methodologies that measure system performance and data processing capabilities.

System Performance Metrics

In the context of the CBBBS, our performance evaluation metrics will include:

- Transaction Throughput: Measure the number of successful blood type match queries and donor information transactions processed per second.
- Response Time: Assess how quickly the system responds to queries for blood availability, donor registration, and emergency requests under different load conditions.
- Resource Utilization: Monitor the usage levels of CPU, memory, and storage, especially during peak transaction periods to identify potential bottlenecks.
- Data Integrity and Consistency: Ensuring all donor records, blood type data, and transaction logs maintain integrity and are free from errors during operations.

Evaluation Strategy and Tools

We plan to employ the following strategies and tools for a comprehensive evaluation:

- Implement AWS CloudWatch to monitor real-time metrics for CPU, memory, and disk usage within the AWS environment.
- Develop and deploy custom scripts to validate the consistency and integrity of donor and blood inventory data across multiple database transactions.
- Scalability in our Cloud-Based Blood Bank System (CBBBS) means the system can handle increased load by adding more cloud instances. Ideally, adding instances improves performance linearly by allowing more concurrent processing. However, real-world limitations like network latency or database speed may prevent perfect linear scalability. We plan to perform scalability testing to understand how performance changes with additional instances and identify any non-linear relationships due to bottlenecks.
- Elasticity allows the CBBBS to automatically adjust resources in response to changing demands, ensuring optimal performance without manual intervention. Using AWS's Auto Scaling, the system can efficiently scale up or down based on actual needs, ensuring responsiveness during varying workload conditions, such as during emergencies or blood drives.

Task 6: Deployment and Provisioning

Use AWS Elastic Beanstalk/ECS/EKS for deployment and AWS CloudFormation/CDK for resource provisioning.
Implement automated deployments with AWS.

Task 7: Monitoring, Scaling, and Maintenance

Monitor health and performance with AWS CloudWatch and adjust resources with AWS Auto Scaling.
Plan for regular updates and optimization.

Task 8: Launch and Operational Use

Officially launch the application, monitoring system performance and user feedback closely. Implement a feedback loop for continuous system improvement using AWS services.

Please Note: We aim to look into certain **PHI issues** which refer to concerns related to the privacy and security of individuals' health information. This includes any information that can be used to identify an individual and that relates to their past, present, or future physical or mental health condition, the provision of health care to them, or the payment for that health care.

Some of the common PHI issues to be looked at are:

- **Unauthorized Access**: When individuals or entities gain access to PHI without proper authorization, it can lead to privacy breaches and compromise the confidentiality of personal health information.
- **Data Breaches**: Data breaches occur when PHI is accessed, disclosed, or used by unauthorized individuals or entities. This can happen due to cybersecurity vulnerabilities, insider threats, or other means.
- **Inadequate Security Measures**: Weak security measures, such as insufficient encryption or lax access controls, can leave PHI vulnerable to unauthorized access or disclosure.

Timeline

Week 1-2: Conduct group meetings to gather and analyze system requirements and data to work on.
Week 3: To plan data backup and disaster recovery strategies with AWS S3 and AWS Backup
Week 4-6: Develop backend and frontend components using HTML, CSS, JavaScript / React>> midterm project report
Week 7: Perform system testing
Week 8: Deployment and constant monitoring
Week 9: Summarize the contributions and prepare a final project report.

Given the project's scope involves frontend, backend, database management, and cloud infrastructure, here's the workload division:

Name	**Role**	**Responsibilities**	**Average Time Spent(in hours)**
Anurag Ganguly	Backend Development & Cloud Infrastructure	● Lead backend development and API integration. ● Set up and manage AWS deployment and resource	~15

		provisioning. ● Implement donor-recipient matching algorithm. ● Contribute to disaster recovery and backup strategies.	
Ayantika Nandi	Database Management & Cloud Infrastructure	● Design and manage database schema and ensure data security. ● Manage real-time updates on blood stock levels. ● Share responsibility for cloud infrastructure ● Participate in monitoring and maintenance.	~15
Atharva Atul Joshi	Frontend Development & User Interface Design	● Design and implement user interfaces for web and mobile. ● Integrate GPS and mapping for locating blood centers. ● Ensure frontend-backend integration. ● Assist in frontend testing and QA.	~15

Progress on Proposed Tasks

1. **Project Overview**

- *Objective*: The project aims to transform blood bank operations through cloud technology, enhancing their efficiency and accessibility.
- *Current Stage*: The team is in the midst of developing and testing both front and back-end components, with a focus on utilizing AWS cloud infrastructure.

2. **Achievements to Date**

- *Requirements Gathering*: The team completed the initial phase of project planning and gathering requirements.
- *System Architecture*: A design for the system architecture has been laid out, prioritizing scalability, reliability, and security.
- *Development Milestones*:
    - The back-end development is ongoing, featuring the integration of initial APIs.
    - A database schema has been developed with an emphasis on security and the capability for real-time updates.
    - Front-end development is also in progress, particularly focusing on the design and integration of the user interface.

3. **Challenges Encountered**

- *Technical Issues*: Tackling scalability challenges and the integration of sophisticated algorithms for matching donors with recipients.
- *Security Concerns*: Guaranteeing the integrity and confidentiality of PHI (Protected Health Information).
- *Resource Allocation*: Distributing workload effectively among team members and managing time efficiently.

4. **Next Steps**

- *Testing and Quality Checks*: Set to begin in the 7th week, this phase will focus on assessing system performance metrics and enhancing user experience.
- *Deployment and Oversight*: Scheduled for the 8th week, focusing on leveraging AWS services for scalable and thorough operations.
- *Comprehensive Report*: A final report will compile the project's contributions, milestones, and key learnings.

5. **Team Contributions**

- ***Anurag Ganguly***: Heads the back-end development and cloud infrastructure work, concentrating on AWS deployment and the implementation of algorithms.
- ***Ayantika Nandi***: Manages database architecture and cloud infrastructure, ensuring data security and prompt updates.
- ***Atharva Atul Joshi***: Focuses on front-end development and designing the user interface, facilitating smooth integration with the back-end systems.

6. **Technical and Ethical Considerations**

- *PHI Security*: Tackling issues related to unauthorized access, data breaches, and the adoption of robust security measures to safeguard sensitive health data.
- *Data Ethics*: Complying with data protection laws and maintaining ethical standards in the management of donor and patient data.

Preliminary Observations Report: Cloud-Based Blood Bank System (CBBBS)

While choosing this project topic as well as working on it, we have observed that traditional blood banks face issues such as inefficient inventory management, which leads to wastage of valuable blood resources or shortages that can imperil lives. Additionally, the process of matching donors with recipients is often cumbersome and slow, hindered by manual systems that cannot efficiently handle the complexity and urgency of the task. Furthermore, the physical infrastructure of traditional blood banks imposes limitations on scalability and accessibility, making it difficult to expand services or adapt to increasing demand without significant investment.

Our project seeks to address these challenges by leveraging the power of cloud computing to revolutionize the blood banking sector. By moving to a cloud-based framework, our system aims to provide a more scalable, reliable, and secure platform than is possible with conventional IT infrastructure.

Some of the observations that we made while working on this project are as follows:

1. **Data Latency and Real-time Processing**: Achieving real-time updates in a cloud-based system is essential for blood banks to manage donations and demands efficiently. However, ensuring minimal data latency, crucial for the timely processing of information across widespread geographical areas, poses a significant challenge. We have been trying to optimize data flow and reduce latency to ensure immediate action can be taken.
2. **Cloud Security and Compliance**: Protecting sensitive health information in the cloud requires stringent security measures that comply with healthcare regulations. This has also been one of our key observations.
3. **Scalability vs. Cost Management**: While cloud services offer the advantage of scaling resources according to demand, managing the associated costs effectively remains a challenge. It's crucial to balance the need for scalability with budgetary constraints, requiring continuous monitoring and optimization of cloud resource usage to prevent unnecessary expenditures. We were looking for free services to support and deploy our project.
4. **Disaster Recovery and Data Backup**: Ensuring the cloud-based system's resilience in the face of system failures or cyberattacks requires a comprehensive disaster recovery plan. This plan must guarantee data integrity and availability, necessitating the strategic deployment of cloud resources, regular data backups, and swift recovery mechanisms to maintain continuous operation and trust in the system.

Additional Pointers

1. Qualitative Changes (Setup of the System/Framework)

Initially, the project's goal was to use cloud technology to transform how blood banks operate, aiming for improved scalability, reliability, and easy access. The main focus areas were on securely storing data, managing information about blood types and donors efficiently, and developing sophisticated algorithms for matching donors with recipients.

Updates and Progress:

- *Enhanced Security Measures*: A significant effort is being put into developing strong security measures to safeguard sensitive health information (PHI), a key step for maintaining trust and meeting health regulations.
- *Improved Real-time Data Handling*: The team has worked on making data processing faster and reducing delays, ensuring that blood management and donations are handled as swiftly as possible.
- *Scalability Versus Costs*: The challenge of keeping the system scalable while controlling

costs is being addressed by fine-tuning the use of cloud resources, aiming for an efficient balance to keep expenses in check.

2. Quantitative Changes (Initial Profiling/Benchmarking Result)

Regarding quantitative aspects, the initial proposal outlined objectives for system performance metrics such as transaction throughput, response time, resource utilization, and data integrity. At present, specific benchmarking results are not available, as the focus has been on laying the groundwork for these assessments.

Preparations for Benchmarking:

- Preparing to use AWS CloudWatch for real-time metrics monitoring.
- Planning scalability tests to understand performance changes with additional cloud instances and identify potential bottlenecks.

3. What has been changed since the first proposal?

Since the first proposal, our project has progressed largely in line with our initial plans, without significant deviations from the proposed objectives and methodologies. We've been focusing on developing the system's back-end components, emphasizing utilizing AWS cloud infrastructure for enhanced scalability and security. Our dedication to implementing strong security measures and managing data efficiently continues to be a priority, showcasing our consistent alignment with the core objectives we established at the start of the project.

4. Comparison with the Initial Timeline

The timeline outlined in our proposal included a structured approach to development, from requirements gathering to deployment. As per the report, our team is currently in the midst of development and testing, which aligns with the planned activities for the mid-phase of the project (Weeks 4-6 for development, Week 7 for testing). It suggests that the project is on track according to the initial timeline.

5. Takeaway So Far

From the work done so far and the challenges encountered, a significant takeaway is the importance of early and continuous attention to scalability, security, and integration challenges.

Potential Pitfalls:

- *Underestimating the Complexity of Integration*: We realized that ensuring seamless integration of front-end, back-end, and cloud services can be more complex than initially anticipated, particularly when aiming for real-time data processing.
- *Security and Compliance*: The critical nature of PHI security demands ongoing vigilance. Any oversight here can lead to significant setbacks. We are also exploring more on this.
- *Resource Management*: Effectively managing the balance between scalability and cost can emerge as a more intricate task as the system scales.