SOFTWARE ENGINEERING

SECURITY ENGINEERING

## TABLE OF CONTENTS

[EXECUTIVE SUMMARY 2](#_TOC_250008)

[INTRODUCTION 3](#_TOC_250007)

[DEPENDABILITY 3](#_TOC_250006)

[POSSIBLE VULNERABILITIES OF OUR SYSTEM 4](#_TOC_250005)

[HOW’S THE APPLICATION DEPENDABLE? 5](#_TOC_250004)

[HOW THE APPLICATION DESIGN AND OPERATIONS WILL COMPEL THE SYSTEM TO BE RESISTANT TO EXTERNAL ATTACKS? 7](#_TOC_250003)

[HOW THE SOFTWARE DESIGN AND OPERATION WILL COMPEL THE SYSTEM TO RECOVER FROM AN ISSUE? 8](#_TOC_250002)

[CONCLUSION 10](#_TOC_250001)

[BIBLIOGRAPHY 11](#_TOC_250000)

# EXECUTIVE SUMMARY

OctoFlex Technologies is a start-up IT firm that is focused and specialized in developing all kinds of software solutions for businesses from small-sized to mid-sized. With experienced professionals and a team approach to most of the projects, OctoFlex Technologies will be able to offer a more balanced quality services by providing software solutions which are customizable and can be tailored to meet the specific needs of any business. Our goal is to help businesses to improve their productivity, efficiency, and profit by using the services that we provide in this technological world.

### The Company

OctoFlex Technologies is a team of eight IT professional. Each professional specializes in some particular discipline and has broad knowledge of other disciplines as well. All of our IT professionals which include developers, designers and support staff work together and closely with each client to make sure that they get most of the benefit from our software solutions.

### The Market

OctoFlex Technologies promises to provide the best service and customer experience. With our team of experienced IT professionals, we believe that OctoFlex Technologies has the potential to give tough competition to its competitors in the market.

# INTRODUCTION

The Blood Support+ Application developed by our company OctoFlex Technologies focuses on facilitating the blood donation process flexible and easily accessible. It focuses on connecting the blood donors and blood recipients. The application will have many confidential information about the users such as medical records, blood types, and some other personal information which necessarily needs to be protected. The system security and resilience are the important aspects on which our application is dependent. The security and resilience are vital to make sure the application is reliable. This report will showcase our approach to development of a dependable, secure and resilient application.

# DEPENDABILITY

Dependability of the system is the most important system property for many computer systems. The dependability of a system refers to the ability of a system to provide all the required services which reflects the user’s degree of trust in that application. It makes the user confident enough to believe that the system will operate the way he/she has expected and that it will not fail when it is in normal use.

To achieve a secure and dependable a system, our company will follow a secure development process which will verify the secure design and developmental practices. Our company will take all the necessary steps to make the system secure which will require conducting a security risk assessment by which we will be able to recognize the threats and vulnerabilities. Our development team will follow a secure design and development practices which will include

From a client’s perspective a secure and dependable system will have some of the system attributes which are:

Security: To secure sensitive data, such as user passwords, medical records, and personal information, the application will be created with strong security features.

Together with appropriate systems for access control and authentication, this will also involve the encryption of data both in transit and at rest.

Reliability: The application is made to function with little downtime or disturbance across a range of situations.

This will have safeguards like redundancy and fault tolerance as well as routine testing and maintenance to guarantee the program keeps performing as intended.

User experience: The application will be having a simple and intuitive user interface that is intuitive and easy to use.

Users will have a little trouble navigating the application and carrying out the desired tasks, including looking for and donating/receiving blood.

Overall, a dependable application will be designed with the user's needs and safety in mind, with a focus on security, reliability, availability, user experience, compatibility, and ongoing support and maintenance.

# POSSIBLE VULNERABILITIES OF OUR SYSTEM

The Blood Support+ application can have the possible weaknesses that might be exploited by attackers, just like some other software. Below are a few instances:

Attacks known as "injections" take place when an attacker inserts malicious code into an application's input fields, like login information or search terms. This may result in data in the application being accessed or altered without authorization.

Cross-site scripting (XSS): This happens when an attacker inserts malicious code into an application's web pages so that the victim's browser can use it to carry out the attack. This could be used to steal user information or carry out other harmful deeds.

Broken session management and authentication: The application's authentication method might not be handled correctly, because of which attackers may be able to access user accounts and sensitive information. Many techniques, including brute-force attacks and session hijacking, may be used to achieve this.

Inadequate logging and monitoring: The application's logging and monitoring systems might not be set up correctly, which may let attackers have the ability to access sensitive information or run harmful code undetected.

Insecure data storage: The application's data storage features might not be properly protected, which may let the hackers gain access to private information such as user login information or medical records.

That’s why It's crucial for us to use secure coding techniques and put in place the necessary security safeguards, including input validation, access control, encryption, logging, and monitoring. In order to make sure that the security measures in place are still effective and current with the most recent security best practices, it's also crucial to regularly examine and update them.

# HOW’S THE APPLICATION DEPENDABLE?

The application's dependability is taken into account. Availability, reliability, safety, security, resilience, repairability, maintainability, and error tolerance are all aspects of dependability, which is the capacity of the program to provide a service that can be trusted.

To ensure the security of the application, we will follow some security processes and procedures.

* Secure coding practices: We will make sure the code is written securely with no known weaknesses or vulnerabilities. We will scan the code for security problems and fix them using tools and procedures.
* Authentication and access control: We will provide a reliable authentication system to guarantee that only permitted users can access the application. Passwords, two-factor authentication, and biometric authentication are some examples of this.
* Encryption: We will protect the sensitive data both in transit and at rest by using encryption. This might encrypt user login information, information kept in databases, and information sent back and forth between the client and the server.
* Regular updates: We will follow the program's most recent security updates.
* Security testing: We will conduct regular security testing of the application to identify and fix any potential security vulnerabilities. This could include penetration testing, vulnerability scanning, and code review.

In terms of the costs of dependability, it's important for us to consider the upfront costs of designing and implementing these security measures, as well as the ongoing costs of maintaining and updating the software to ensure its continued dependability.

Dependable systems are desirable since they are “trustworthy,” Dependable systems are typically characterized by the following attributes:

* + Reliability: The system behaves as expected, with very few errors.
  + Availability: The system and services are mostly available, with very little or no down time.
  + Safety: The system do not pose unacceptable risks to the environment or the health of users.
  + Confidentiality: Data and other information will not be divulged without intent and authorization.
  + Survivability: The system services will be robust enough to withstand accidents and attacks.
  + Integrity: System data will not be modified without intent and authorization.
  + Maintainability: Maintenance of system hardware and services will not be difficult or excessively expensive.

# HOW THE APPLICATION DESIGN AND OPERATIONS WILL COMPEL THE SYSTEM TO BE RESISTANT TO EXTERNAL ATTACKS?

1. SHIFT SECURITY TO THE LEFT IN THE SDLC

By integrating security as early in the software development lifecycle as practicable, shift left security replaces legacy security procedures and equipment created for the waterfall release approach (SDLC). This change will be facilitated by the following techniques:

* + Using threat modelling
  + Incorporating security considerations into code design and architecture
  + Testing code while it is being written instead of waiting until applications are live in the production environment

1. INJECTION & INPUT VALIDATION

A set of attack techniques called injection involves inserting malicious code into browsers or other entry forms. SQL injection and cross-site scripting are two instances of injection that use malicious SQL code and malicious scripts in website frontends, respectively. Input validation techniques should be used to ensure that only properly formatted data may be inputted, barring any malicious code from entering a system. This will guard against injection attacks.

1. USER AUTHENTICATION MANAGEMENT

Authenticating users is a crucial component of web-based security. Security administrators will have a variety of choices to make sure that only authorized users are accessing their apps thanks to user authentication management, which strengthens usernames and passwords. Multi-factor authentication is one such approach, which demands users to demonstrate their identity using at least two different methods of authentication.

1. DATA ENCRYPTION

Failures related to cryptography (or lack of it) may lead to breaches of sensitive information. Encrypting data, both at rest and in transit, will be the key protection in the event of a breach.

1. FIND AND FIX MISCONFIGURATIONS

Misconfigurations — like failing to implement the principle of least privilege access — make it easier for third parties to access sensitive data. Most misconfigurations are introduced by manual error, so using infrastructure as code (IaC) and automation will help us prevent them. Additionally, scanning tools like Snyk IaC will be used to detect and remediate misconfigurations before they reach production environments.

1. WEB APPLICATION FIREWALLS

A web application firewall (WAF) sits between clients and web servers and serves as a proxy for traffic between them. By setting up rules in a WAF, we will protect our application against common attacks like injection.

1. AUTHORIZATION

Executing authorization processes prevents privilege escalation, an attack where a user gains access to an application then changes their privileges or role to expand access. Privilege escalation will be uncovered through penetration testing, mitigated by running applications with least privilege access, and will be prevented by properly configuring authentication keys.

# HOW THE SOFTWARE DESIGN AND OPERATION WILL COMPEL THE SYSTEM TO RECOVER FROM AN ISSUE?

While designing the Blood Support+ Application, it’s important for us to consider the approach to software design and operation to make sure the system will recover from any kind of issue that may emerge.

Our company OctoFlex Technologies which is currently developing a Blood Support+ Application will have to keep in mind the resilience while designing the application. The Blood Support+ Application allows users to register as blood donors as well as blood recipients. The users would also be able to search the blood banks and schedule appointments. There are many approaches that we can consider to recover from the issues that may emerge which are:

1. Redundancy: Implementation of the redundancy will help the application to keep on continuing to function when any part of the system fails or if there is a successful cyberattack. Copies of data and software will be maintained on separate computer systems on separate server. The application will be developed on multiple servers. The application will keep on functioning well even when one of the servers goes down because there will be the other server that will take over with its redundant database.
2. Graceful Degradation: This can help to make sure the user experience is smooth even when there are some kind of issues. For instance, if the application is not able to get access to user’s location then the user will have the option to search for the blood banks near him/her by typing their zip code or pin code. This will make the processes more flexible and adaptable
3. The system can be designed for fast recovery and automated monitoring and alerting to decrease downtime whenever there is a failure in the system. If there is any kind of issues related to database, then then system will immediately notify the development team to take some action and will automatically switch to backup database.

An important part of designing and operating the system for resilience is identifying the critical services. Our focus is on recognition and resistance from problems, recovery of critical services and assets and reinstatement of the system. Our system will be designed with many defensive layers of different types such as authentication controls, data protection, and access measures.

Data protection will be required to safeguard the information resources. The processes that will be used to data protection will be data hashing and encrypted backups.

Access Measures will be taken to make the software to identify and regulate who is allowed to enter into company’s network and to which extent.

# CONCLUSION

In conclusion, a dependable, secure, and resilient application is important to make sure the safety of both the blood donors and the recipients. The application will be designed in a way that it handles all the confidential information of the all the users. The application will have multi-factor authentication, encryption, secure data storages and will also include redundant backups and recovery plan whenever the system experiences a failure. Additionally, the applications will be accessible, reliable, and user-friendly to make the user experience smooth reflecting the user’s degree of trust in the application. The application will be built by keeping in mind that all the users can easily access all the information and communicate with each other. Regular updates will be done on the application to make sure the application secure and resilient from the emerging threats and cyberattacks. Overall, security and resilience are the important aspects on which the software is mainly dependent, and which are to be considered to protect the privacy and safety of the blood donors and recipients.

# BIBLIOGRAPHY

*Just a moment..***.** (n.d.). Just a moment... https://[www.sciencedirect.com/topics/computer-](http://www.sciencedirect.com/topics/computer-)

science/dependable-system

*Web application security risks & 9 best practice tips*. (2022, November 1).

Snyk. https://snyk.io/learn/application-security/web-application-security/