Sri Lanka Institute of Information Technology

Group Assignment: Critically evaluate the Information Security Policy and Management practices of a Software solution provider

IE3072-Infromation Security Policy and Management
B.Sc. (Hons) in Information Technology Specializing in Cyber Security

Submitted by:

Student registration number	Student name
IT21188572	Fernando S.D.K.
IT21162596	Gunasekara M.V.G.R. S
IT21195402	Gunathilaka D.J.V.
IT19212364	Mohammed Saabith M. J
IT21278976	Perera U.L.S.A.

Date of Submission: **07/10/2023**

Group Number: **IE3072_23_28**

Table of Contents

Abstract

This paper delves into the real-world implementation of an Enterprise Information Security Policy (EISP) within a discreet company. Over time, the EISP has been in operation, and the organization has observed a reduction in security incidents.

The central argument presented in this paper emphasizes the indispensable nature of adopting an EISP for organizations, regardless of their size or industry. An EISP provides a structured framework for crafting and implementing security measures, concurrently ensuring that all employees understand their roles in maintaining security standards.

In the concluding section, this paper explores the impact of the EISP on the undisclosed company's practices in secure software engineering. The EISP has played a pivotal role in enhancing the organization's secure software engineering practices by specifying security requirements, integrating security measures throughout the software development lifecycle, and providing training to employees.

This research paper contributes significantly to the field of information security by presenting a case study on the effective implementation of an EISP. Additionally, it imparts valuable insights for companies contemplating the adoption of an EISP.

Introduction

Enterprise Information Security Policies (EISPs) are a crucial necessity for organizations, regardless of their size or industry. These policies provide a structured framework for designing and implementing security controls, ensuring that all employees are fully aware of their roles and responsibilities concerning security.

One of the significant merits of implementing an EISP is its potential to enhance an organization's secure software engineering practices. By incorporating security prerequisites into the EISP, organizations can ensure that security remains a top priority at every stage of the software development lifecycle.

Due to concerns surrounding confidentiality and security, the organization under scrutiny in this research paper has requested that all evidence related to its EISP and secure software engineering practices be anonymized or omitted from this paper.

This paper embarks on an exploration of how a prominent Information Technology solutions provider in Sri Lanka, hereafter referred to as "Asia Pacific Technology Systems (Pvt) Ltd" adopted an EISP and the subsequent impact on the organization's secure software engineering practices.

We'll commence by highlighting the overarching significance of EISPs. Then, we'll delve into the specifics of the EISP implemented at "Asia Pacific Technology Systems (Pvt) Ltd", encompassing its notable features and the strategies used for its implementation. Subsequently, we'll examine the influence of this EISP on the company's secure software engineering practices.

In conclusion, we'll consider the broader implications of this research for other organizations contemplating the integration of an EISP.

1) The overall Information security philosophy (Enterprise infosec program) of the organization.

In short, an Enterprise Information Security Policy (EISP) describes an organization's security executive philosophy and helps set the direction, scope, and tone of all an organization's security efforts. This type of management document is typically written by the chief officer (CEO), chief information officer (CIO), or another person in that role. Once completed, the EISP will serve as a roadmap for developing future safety programs and set the tone for how the company addresses specific safety issues.

The mission of the EISP is to clarify an organization's views on the structure of its security program in relation to the different types of roles and responsibilities that exist in the organization's security landscape and protect critical information from intruders. The document should also identify the basic principles of an effective security policy and define the appropriate level of security through security standards and policies. The EISP must also ensure that appropriate responsibilities are assigned to relevant organizational elements to achieve maximum security effectiveness.

Legal Compliance

Ultimately, based on the goals of the organization, each company will have a different preferred usage of an EISP. A hospital that manages a large volume of Protected Health Information (PHI) electronically may state in their EISP that their main priorities are to protect PHI from unauthorized access and unintentional disclosure. The company's reputation with regard to its moral and legal obligations is protected by including these goals in the EISP's tenets.

An official organization that deals with sensitive and/or classified material will approach legal compliance through its EISP differently than a corporation that just deals with the general public. Based on the legal compliance criteria that its organization is required to follow, the EISP must address the appropriate use of penalties and disciplinary proceedings. The establishment of procedures and guidelines that can answer the question of what should be done in a particular scenario and who would then be responsible for it are guided by these legal compliance rules.

The core values, concepts, and beliefs of an organization towards the security of digital assets and sensitive data are reflected in the organization's information security philosophy. This ideology acts as the foundation for the company's information security strategy. The specifics of an information security philosophy may change based on the organization; however the following fundamental ideas frequently serve as its cornerstone:

1. **Integrity:** This goal requires organizations to focus on protecting information from unauthorized access and inappropriate use. It is critical to implement security measures and processes that increase your chances of catching hackers through ongoing monitoring, testing, and training.
2. **Confidentiality** – This goal requires protecting policies, processes, or systems from unauthorized intentional or accidental changes. This goal is compromised by both tool vulnerabilities and human error, which is why it is so important to develop measures to protect against loss of integrity.
3. **Availability:** This goal requires fast and reliable access and use of information, regardless of what is currently impacting the world around the company. These include threats such as natural disasters, hardware failures, programming errors, human error, distributed denial of service (DDoS) attacks, and malicious code. An organization should implement

availability guarantees to ensure effective emergency preparedness and disaster recovery planning when the time comes.

Information Security Program (EISP)n security philosophy, often referred to as the Enterprise Information Security Program (EISP), takes a comprehensive, strategic approach to protecting digital assets and sensitive information. This philosophy forms the basis for designing, implementing and managing security measures across the organization. Here are the key elements and principles that typically make up the EISP of the organization.

1. **Risk Management**: Information security is essentially about risk management. The goal of EISP is to identify, assess, and prioritize threats to an organization's data and systems. This includes understanding potential threats, vulnerabilities, and their potential impact on the business.
2. **Security Policies and Standards**: The EISP establishes clear and comprehensive security policies and standards that define the organization's expectations and requirements for protecting information. These policies cover areas such as data classification, access controls, encryption, incident response, and more.
3. **Compliance and Legal Requirements**: EISP ensures that your organization complies with applicable laws, regulations and industry standards related to information security and data protection. Depending on the industry, these include GDPR, HIPAA, PCI DSS and more.
4. **Security awareness and training**: The weakest link in information security is frequently employees. To inform personnel about security threats and best practices, EISP emphasizes the value of continual security awareness and training programmers.
5. **Access Control**: Strong access controls are implemented to guarantee that only authorized persons have access to sensitive data. User authentication, authorization processes, and the least privilege principle are all included in this.

2) Implementation of security software engineering at the organization

Introduction

Secure software is designed to protect against data loss attacks, with the likelihood of attacks based on the software's weaknesses. The Internet of Things (IoT) has increased the likelihood of attacks, making it crucial to critically assess, analyze, and update security elements in software development processes. Many developers focus on functional needs, neglecting non-functional needs like security, leading to regret and loss. Security measures should be applied during software development to avoid delays and maximize productivity. The secure software development lifecycle (SDLC) is a practical framework that helps developers create, design, and maintain software effectively while ensuring all functional and non-functional requirements are met. This approach enables earlier vulnerability detection and issue amendment. Security is achieved when software meets Confidentiality, Integrity, and Availability (CIA) requirements, which are crucial for maintaining data privacy and preventing attacks like denial-of-service attacks and ransomware and malware attacks. More laws and regulations have been suggested to help achieve security, including authentication and authorization.

The importance of software security

Customers depend on their applications' security. Cyberattacks and other serious internet threats are protected by software security. It employs a proactive method to handle security early on rather than waiting for security issues to arise.

Cyberattacks increase yearly, and this pattern doesn't seem to be changing. Hackers have greater opportunities to exploit us as more of our regular activities, both personal and professional, move online.

Given this, developers need to understand how important software security is and give it top priority when designing new systems. Early issue and vulnerability identification enables faster and cheaper remediation.

How to maintain and improve software security

The security of software is a continuous process. By making security an integral part of your software design process, investing in training, and doing careful vulnerability testing, you can constantly endeavor to increase your security.

To enhance and maintain the security of your program, use these 6 tactics.

1. Design security considerations for your decisions.
   It is essential to include security in the early phases of development to reduce security concerns. Making software security a top priority when making design choices helps stop attacks and save time. This strategy is less expensive than the break/fix strategy. Software can be protected, security holes and flaws can be avoided, and this can be done by making

sure security is considered throughout the design process, especially when making important product decisions. As a security breach or application outage might have a negative impact on stakeholders and customers, keeping security at the forefront of decision-making is comparable to prioritizing customer demands.

2. Spend investment on team education and training.
   Employee training is crucial for software development to ensure high security. Regular training in software security best practices, SDLC, and evolving threats keeps teams informed. Developers should receive secure coding instruction and safety tutorials, while new hires receive computer security training. Compliance and participation requirements are essential, along with regular phishing test runs.

3. Establish in place policies and procedures.
   All team members need to be aware of and able to access your security policies. Ensure you have robust procedures to ensure that nothing eludes you.

   What procedures do you now use to guarantee that software security is considered at every stage of development? Who is responsible for maintaining and upgrading these security protocols? Are team members informed of the protocols and what is expected? Does everyone on your team know about them?

4. Integrate software security across your SDLC.
   Include security concerns for software in the SDLC while developing software. This may be made sure of by intentionally including secure software in your SDLC and integrating it into your standard operating procedures.

   Ensuring that security is properly represented in the software development life cycle (SDLC) will take time, but the work is well worth it. Take the time up front to do tasks like risk analysis, software composition analysis, security vulnerability detection, and code review. The sooner vulnerabilities and bug patches are fixed, the better.

5. Thorough risk assessment and rigorous testing
   A vulnerability may be fixed as soon as you become aware of it. The more you test, the more likely you are to discover problems, weak spots, or programming errors that hackers may take advantage of.

   Complete a thorough risk analysis and conduct several tests often and early. Use several analysis techniques, such as penetration testing (sometimes called pen testing), to assess the security of your applications. This may help you find all the possible ways that your system could be compromised.

6. Least privilege access should be used.
   The concept of lowest privilege (PoLP), a notion in information security, states that individuals, programs, or other systems should only be given the minimal access or rights

required to perform their jobs. It describes the ability of a person or program to get through security restrictions and safeguards privileged access to valuable information and resources. To prevent security difficulties, access should only be granted to those who require it. Access levels for interns or contract workers vary from those for managers or business owners. Privilege creep, which happens when administrators fail to remove access control and other rights once they are no longer required, may be harmful to security. Businesses should set up security teams, evaluate user privileges, and implement processes for tracking access to avoid this.

Current Application Security Landscape: Common Secure Software Engineering Issues

Many software applications are developed without security safeguards, which can be dangerous if not prioritized. Organizations may still be caught off stride regardless of whether security is prioritized, and secure software development practices are used. Current issues with application security include:

- Third-party libraries and frameworks may have security holes: third-party libraries and frameworks, which, if not updated often, might bring vulnerabilities into the program.

- Injection attacks: An attacker can gain unauthorized access to an application or database by introducing malicious code or commands into its input fields, such as login forms or search boxes.

- Cross-site scripting (XSS): In XSS attacks, malicious code is inserted into a website or software program by the attacker, which the user's browser can then run to potentially steal sensitive data or engage in unauthorized activity on the user's behalf.

- Insecure authentication and authorization: If authentication and authorization processes are poorly planned or implemented, attackers may be able to sneak past security protections and access crucial data or functionality.

- Insufficient logging and monitoring: Finding security events, responding to them, or figuring out the root of security issues may be difficult without proper recording and monitoring.

- Mobile application security: It is now more important than ever to defend mobile apps due to the proliferation of mobile devices. Mobile applications can be vulnerable to a range of crimes, especially those that attack the gadget itself or the program's backend servers.

- Cloud security: Protecting the security of applications developed in the cloud is crucial given the rise in popularity of cloud computing. Cloud-based apps may be vulnerable to a

range of attacks, including those that target the infrastructure used by the cloud, the application itself, or the data stored there.

3) Roles and responsibilities of the ensuring secure software engineering.

What are the roles and responsibilities of the different members of a software development team?

- **Business analyst**

A business analyst (BA) is a professional who bridges the gap between the business and IT worlds. BAs are responsible for gathering, analyzing, and documenting business requirements, and then working with IT teams to develop and implement solutions that meet those requirements.

Here are some of the key roles and responsibilities of a business analyst:

- **Identify and understand business needs**. To understand their requirements and concerns, BAs collaborate with stakeholders from throughout the organization. Additionally, they examine corporate data and procedures to look for areas that may be improved.
- **Gather and document business requirements**. BAs gather and record precise requirements for solutions after they have a firm grasp of the business demands. Determining the system's features, functionality, and performance needs is a part of this process.
- **Develop and maintain use cases and user stories**. Users' interactions with a system are outlined in papers called use cases and user stories. BAs create and update these papers to make sure the system satisfies the requirements of its users.

Business analysts (BAs) play a vital role in ensuring secure software engineering practices:

- **Gathering and analyzing security requirements**. BAs collaborate with stakeholders to identify and comprehend a software system's security needs. Understanding the dangers a system confronts, the assets that must be safeguarded, and the intended security posture are all part of this.
- **Translating security requirements into technical requirements**. BAs transform security requirements into technical specifications that the development team may use once they have a

clear understanding of them. Identifying the security measures that must be put into place and how they should be included into the system architecture are part of this process.

- **Working with the development team to implement security controls**. To guarantee the proper and efficient implementation of security controls, BAs collaborate closely with the development team. Among other things, this may entail giving developers comments, examining code, and testing the system for flaws.

Product owner

The Development Team's work is put out as a product, and the Product Owner (PO) is the only one in charge of increasing the product's value. The team, the company, and the product being produced will all have an impact on how this is done.

The PO is in charge of making sure that the product satisfies the demands of its customers and serves as their voice. The PO works closely with the Development Team to prioritise the work and make sure that everyone is on track to deliver the product on time and within budget.

Here are some of the key roles and responsibilities of a Product Owner:

- **Create and assign a priority to the product backlog**. All the tasks that need to be completed to create and release the product are included in the product backlog and are given a priority. The PO is in charge of outlining the backlog items and ranking them according to how valuable each one is to the client and the company.
- **Communicate the product vision to the team.** The PO must inform the Development Team and other stakeholders about the product's vision. A declaration of the product's goals and the reason for its significance is known as the product vision.
- **Accept or reject work completed by the Development Team.** The work produced by the Development Team must be approved or rejected by the PO. The PO accomplishes this by assessing the work and making sure it complies with the specifications of the product backlog item.

  Ensuring safe software engineering practises is crucially dependent on the product owner (PO). The PO is in charge of ensuring the security of the product’s overall success. As a result, the PO is required to be aware of the security risks the product confronts and to take action to reduce those risks.

Here are some specific ways that the PO can ensure secure software engineering practices:

- **Include security in the product vision and roadmap**. In the product strategy and vision, the PO should make security a top priority. The development lifecycle will be made more secure as a result of this.
- **Prioritize security requirements.** In the product backlog, the PO should give security needs priority. This means that other needs, such functional requirements, should be given the same weight as security requirements.
- **Communicate security risks to stakeholders**. The PO should consult with stakeholders to identify security concerns and then collaborate with them to create mitigation strategies. This is crucial since it's possible that stakeholders are unaware of the product's security concerns or the potential effects those risks may have on them.

Project manager

There are some basic tasks that all project managers are accountable for, while the roles and responsibilities of a project manager might vary based on the industry and the project.

Here are some of the key roles and responsibilities of a project manager:

- **Planning and developing the project idea**: The project manager collaborates with stakeholders to establish the project's objectives, needs, and scope. They also create a thorough project plan that lists all of the tasks, deadlines, and materials required to finish the project.
- **Monitoring project progress and setting deadlines**: The project manager compares the project's progress to the plan and makes any modifications. Additionally, they coordinate the project's progress and set deadlines for tasks and milestones.
- **Managing the money**: The budget of the project must be managed and used intelligently, and this is the job of the project manager. This entails keeping track of expenditures, predicting cost trends, and implementing necessary corrections.

Project managers play a critical role in ensuring secure software engineering practices. They are responsible for overseeing the entire software development lifecycle (SDLC) and ensuring that security is considered at every stage.

Here are some of the keyways that project managers can help to ensure secure software engineering:

- **Define and incorporate security requirements into the project plan:** This involves identifying the security risks that the program confronts and creating specifications to reduce those risks.
- **Monitor and report on security-related activities:** This entails monitoring the development of security tasks, recognizing any problems or obstacles, and reporting on the project's overall security posture.
- **Provide guidance and support to other stakeholders:** This involves informing stakeholders of the value of security and assisting them in realizing how important it is to do their part in making sure the software is safe.

UX/ UI designer

The task of designing user-friendly and aesthetically pleasing digital goods falls to user experience (UX) and user interface (UI) designers. In order to comprehend consumer demands and then create solutions that address those needs, they collaborate closely with product managers and engineers.

The overall usability, information architecture, and product flow are the main areas on which UX designers concentrate. Users' interactions with goods are better understood through user research, which is then used to create more effective and pleasurable products.

The layout, typography, colour scheme, and overall look and feel of the product are the main areas of concentration for UI designers. They strive to make goods that are both aesthetically pleasing and user-friendly.

Roles and responsibilities of UX/UI designers:

- **Conduct user research**: Understanding the requirements of the consumers who will use their products is important for UX/UI designers. They carry out user interviews, surveys, and usability testing to achieve this.
- **Create user personas and user stories**: User personas, which are fictitious representations of ideal users, are created by UX/UI designers using data from user research. Additionally, they provide user stories, which are succinct summaries of how users will interact with the product.

- **Test the user experience:** Usability testing and user feedback are two ways that UX/UI designers evaluate the user experience of their products.

UX/UI designers can play a vital role in ensuring secure software engineering practices by following these tips:

- **Authentication and authorization:** To make it difficult for unauthorized users to access the product, secure authentication and authorization systems may be designed with the aid of UX/UI designers.
- **Data protection:** Products that safeguard sensitive user data against unauthorized access, usage, or disclosure might benefit from the aid of UX/UI designers**.**
- **Vulnerability prevention:** UX/UI designers may aid in the development of products that are less susceptible to well-known attack methods like phishing and cross-site scripting (XSS) assaults.

Software architect

For the creation of software systems, software architects are accountable. In order to comprehend the system's needs and build a solution that satisfies them, they collaborate closely with stakeholders including product managers, business analysts, and developers. Aside from overseeing the development phase, software architects also make sure that the system is put into use in accordance with the plan.

Here are some of the specific roles and responsibilities of a software architect:

- **Gather and analyze requirements:** The functional and non-functional needs of the system are understood by software architects through collaboration with stakeholders. Performance, security, and scalability are a few examples of the needs that may be present.
- **Design the system architecture:** The software architect creates the system architecture after fully comprehending the requirements. This entails characterizing the system's constituent parts and the dynamics of their interactions.
- **Select the technology stack:** Additionally, the technology stack that will be utilized to build the system is chosen by the software architect. Selecting the appropriate programming languages, frameworks, and databases is part of this.

Here are some specific ways that software architects can help to ensure secure software engineering practices:

- **Incorporate security into the software development lifecycle (SDLC):** Don't put off thinking about security. From gathering requirements to design, implementation, testing, and deployment, software architects should collaborate with the development team to include security at each stage of the SDLC.
- **Use a threat modeling approach:** An evaluation of possible security vulnerabilities to a software system is done through the technique of threat modelling. Threat modelling should be used by software architects early in the design process to find and fix possible security flaws.
- **Apply security design principles:** There are several documented security design concepts, including defence in depth, separation of roles, and least privilege. These guidelines should be used while designing software systems.

4) The effectiveness of the current implementation of secure software engineering

Effectiveness of the current implementation.

1. **Secure coding best practices**

- A coding standard is a set of policies and procedures that an organization uses to minimize development issues and security vulnerabilities. These guidelines are followed by the software developers to prevent and detect cyber security attacks. Coding standards make the company's software products more secure and improve readability and maintainability. It reduces costs and development time as well as makes it easy to develop software because of the simplicity of code segments.
- Some code segments on the internet can be vulnerable to the products. They make a backdoor code called trap doors to gain unauthorized access by the real programmer to the software product. Software developers in this project team don’t copy and paste code

segments from the internet into their projects. Backdoors bypass normal security access paths and hide the impact from the user of the product. Therefore, it is hard to detect.

- Using functions in coding is reusability, makes more easier to implement codes, reduces cost, effort, and time, and increases effectiveness. The software developers use functions in coding. If threat agents apply vulnerable codes, then SQL injection attacks occur.
- Check that the code conforms with the criteria using review checklists. As soon as human-readable code is checked into the code repository, use automated techniques to continuously discover and correct documented and confirmed dangerous software practices. Identify and note the underlying source of each problem that is found. Publish the knowledge base's lessons learned after code review and analysis for developers to access and browse.
- Software engineers store codes in a secure environment like github. It is providing work and project with the exposure it requires, such as fluid collaborations, templates, integrations, and repositories. Git can make it easier to collaborate remotely. Multiple developers in the organization can collaborate on the same project simultaneously without hindering one another's progress thanks to a variety of alternatives.

2. <u>Standard frameworks</u>

- In time-sensitive projects, the interaction of management and developer agents leads to the adoption of security practices. Assess the adoption of security practices for developers with various preferences and methods under individual and group punishments using the framework. The model compares the use of security among engineers with various preferences and gives managers advice on how to choose the best sanctions for promoting the use of security technologies in software development.
- Aligning their procedures with a recognized framework, such as the Secure Software Development Framework (SSDF) developed by NIST, is advantageous for many enterprises. A variety of foundational secure software development resources have been developed by groups like OWASP and SAFEcode that go into detail about software security. These resources provide the required instructions to lower the incidence of vulnerabilities in software releases, diminish their impact, and avoid them in the future.

3. **Physical environment**

- Physical security is the safeguarding of employees, equipment, networks, and data against physical acts and occurrences that could seriously harm the organization. They use a disaster recovery plan as well as a business continuity plan. Even if the majority of these are insured, physical security prioritizes damage prevention to reduce the loss of time, money, and resources as a result of these occurrences.
- They implement common security controls to ensure environment security. The company put into practice specific adjustments, such as guest access rules, password change policies, employee training, door locking, and limiting access to places with critical information or equipment, in order to create a significant impact and increase physical security in the workplace.
- They limit access to areas that contain crucial machinery or information. Because it is, good firms treat their data with the utmost respect. Their documents, projects, hardware, and software are secured in the company. They don't allow visitors to roam freely in their vaults. The company seals off specific rooms and locations from anyone who doesn't need access to them. Working areas are segmented to ensure the CIA tried. These areas should not be targeted for theft, and only reliable people should be present there. They pay particular attention to regions that contain critical data, hardware, and software as assaults on physical and cybersecurity systems frequently originate in these locations.

Recommendations for improvements

- **Train development team with threat model experts** – Develop and attack threat models for secure software with them. Examining the program architecture is one way to spot potential security threats and weaknesses. This makes it easier to set the necessary security safeguards and create software with security in mind. Teach students how to assess risks, adopt a risk-based approach to handle them, and put mitigations in place.
- **Develop coding secure software practices - ** Developers must use secure coding techniques, such as input validation, safe data storage, and secure communication

protocols. By adhering to secure code standards, common security issues like SQL injection, cross-site scripting, and buffer overflow attacks can be prevented.

- **Teams follow the organization's policies or guidelines for code review -** Examining developer-written code to look for any potential security holes is known as code review. This helps find security problems early on in the development process and repair them. Follow the organization's policies or standards regarding when and how code analysis should be undertaken.
- **Secure Configuration Management -** The deployment of software systems with secure configurations is guaranteed by configuration management. This involves setting up network settings, access restrictions, and other security-related settings to reduce the likelihood of unauthorized access.
- **Recurrent Patches and Updates -** Patching and updating software on a regular basis helps to fix security flaws and lower the likelihood of security breaches. It's critical to keep all software components utilized in the system updated with security patches and other fixes.
- **Security Training** - To ensure that they comprehend the value of security and the best practices for secure software development, developers and other staff members participating in the software development process should regularly attend security training.
- **Reaction to incidents -** To respond to security issues, organizations should have a clearly defined incident response plan in place. This entails spotting possible security breaches, minimizing their effects, and recovering from security breaches**.**
- **Constant Watching -** Real-time detection and response to security events are made possible through continuous monitoring. This entails keeping an eye on user activity, network traffic, and system records for any indications of security lapses.

The organization may create safe and dependable software applications that can withstand potential security risks and weaknesses by adhering to these best practices. To avoid unwanted access and safeguard sensitive data, it is essential to give security top priority throughout the whole software development process.

Appendices and supportive evidence

Given the nature of this topic and the need to protect the privacy of those involved this document does not include another materials or supporting evidence. It is essential to mention though that all the information presented here is obtained from sources and we have taken care to ensure the accuracy of the data.

ISO 9001:2015 QUALITY POLICY

Asia Pacific Technology Systems (Pvt) Ltd

It is the policy of **Asia Pacific Technology Systems (Pvt) Ltd,** committing ourselves to enrich and deliver innovative ICT solutions to our communities with best quality, and meet their uttermost **satisfaction, needs and expectations** always and exceed where possible.

We promise to function as a globally proficient professional team in the Arena of ICT that provide solutions beyond duty that create value, improve efficiency and integrate multiple technology platforms to deliver human needs

Hence we ensure solutions, the change, and reality of innovative ideas with the business practices through honesty and continual grow through valued services.

In achieving above goals we

- Establish ***quality management system*** with Plan Do Check & Act framework complying with ***ISO 9001:2015 standards, statutes and regulations with other requirements.***
- Implement systematic ***process approach*** that streamlines the effective flow of business activities to deliver quality output.
- Train our team members to acquire knowledge and deliver productivity
- Implement ***Risk Based Thinking practices*** to foresee ***risks*** and to generate related ***opportunities*** through potential challenges in the ICT industry
- Regularly assess our **performance** against ***objectives and system compliance*** to improve ourselves continually.

.....

Approved By

01.01.2022

.....

Issue Date

References

[1] "https://apts.lk"

[2] Donna Dodson, Murugiah Souppaya, Karen Scarfone, "Secure Software Development Framework (SSDF) Version 1.1: Recommendations for Mitigating the Risk of Software Vulnerabilities," pp. 1-28, 2022.

[3] D. M. Mehta, "Effective Software Security Management choosing the right drivers for applying application security," pp. 1-16.

[4] Shams Al-Amin, Nirav Ajmeri, Hongying Du, Emily Z. ,Berglund, Munindar P. Singh, "Toward Effective Adoption of Secure Software," pp. 1-32, 2018.