Confidence Building

Language Barrier, Impromptu Speech – IT Polices Briefing

Introduction

- My Self Rudra Vamshy
- I have completed B.Tech in Mechanical Engineering from Sri Indu College of Engineering & Technology, JNTU, Hyderabad.
- I have been working as a Senior E-commerce Executive since 4 years.
- And pursuing a Software Testing in Tops Technologies.

Academic Projects

- **MAJOR PROJECT:**

  Title: Design and Analysis of CAMSHAFT

  Description: Objective of our project is to design and analyze a camshaft in a multi design cylinder engine and modelled using pro/Engineering. WE have done structural and modal analysis on the camshaft to verify the strength of the camshaft. using the two material Aluminum Alloy A360 and Forged steel. Present used material for camshaft is forged steel. We are replacing with Aluminum Alloy A360. By using Aluminum Alloy A360, weight of the camshaft reduces almost 4 times that by used Forged Steel, since its density is less than forged steel. By observing the Analysis result, Aluminum Alloy A360 for camshaft is safe since von mises stress obtained by analysis is less than its yield stress. By comparing the results for both materials, the stresses developed in the camshaft using Aluminum Alloy A360 is less than that by using forged steel. So we can conclude that using Aluminum Alloy A360 is better.

Academic Projects

- **MINOR PROJECT:**

- **Title:** Study and Redesign of VIM in MIDHANI

- **Description:** Objective of the project is to redesign the VIM (Vacuum Induction Melting) 2.2 ton to 4.5 ton and maintenance of Vacuum Boosters in MIDHANI.

Methods for Building Confidence

- **Set Achievable Goals**: Start with small, manageable goals and gradually increase their difficulty. Achieving these goals will help build your confidence.
- **Gain Knowledge and Skills**: Take courses on software testing methodologies, tools, and best practices. Websites like Coursera, Udemy, and LinkedIn Learning offer many relevant courses. And obtain certifications such as ISTQB (International Software Testing Qualifications Board) to validate your skills and knowledge.
- **Practical Experience**: Work on real-world projects, either through your by contributing to open-source projects. Hands-on experience is invaluable. Learn and practice using automation testing tools like Selenium, JUnit, and TestNG. Automation can enhance your testing efficiency and effectiveness.
- **Join Communities**: Participate in software testing forums, attend webinars, and join professional groups on platforms like LinkedIn. Networking with other testers can provide insights and support.

- **Continuous Learning**:

  Stay Updated: The tech industry is always evolving. Keep up with the latest trends, tools, and best practices by reading blogs, attending conferences, and following industry leaders.

  Learn from Mistakes: Analyze your mistakes and learn from them. Every error is an opportunity to improve.

- **Develop Soft Skills**:

  Communication: Work on your communication skills. Being able to clearly and effectively communicate your findings is crucial.

  Critical Thinking: Practice analyzing problems from different angles and coming up with creative solutions.

- **Build a Portfolio**: Keep a record of your testing projects, methodologies used, challenges faced, and solutions implemented. A portfolio showcases your experience and skills to potential employers or clients.

Techniques for Successful Impromptu Speeches

From APJ Abdul Kalam's impromptu speeches, one can glean several important lessons and qualities that made him an iconic speaker and leader.

- **Clarity of Vision and Purpose**:
  - **Focused Messaging**: Kalam’s speeches were known for their clear focus on important themes such as education, youth empowerment, and national development.
  - **Defined Objectives**: He articulated his vision for a better future with clarity and conviction.
- **Inspiration and Motivation**:
  - **Empowering Language**: Kalam’s speeches were filled with words of encouragement and empowerment, inspiring individuals to dream big and work hard to achieve their goals.
  - **Personal Anecdotes**: He often shared personal stories and experiences to illustrate his points and connect with his audience.

- **Humility and Approachability**:
    - **Accessible Language**: Kalam communicated in a simple and accessible manner, making complex ideas understandable to a wide audience.
    - **Humility**: Despite his achievements, he remained humble and approachable, which endeared him to people from all walks of life.
- **Communication Skills**:
    - **Effective Communication**: Kalam's ability to convey complex ideas with simplicity and clarity underscored his effective communication skills.
    - **Connection with Audience**: He established a strong connection with his audience through sincerity, empathy, and relevance.

APJ Abdul Kalam's impromptu speeches were not only informative and inspiring but also reflected his core values of integrity, humility, and visionary leadership. Learning from his speeches can impart valuable lessons in effective communication, leadership qualities, and the importance of personal integrity and societal contribution.

Briefing on Software Testing IT Policies

- **Quality Assurance (QA) Policy:**
    - **Purpose**: Defines the objectives and responsibilities for ensuring software quality through testing.
    - **Key Points**:
        1. Establishes guidelines for testing methodologies, tools, and best practices.
        2. Outlines the roles and responsibilities of QA teams and stakeholders.
        3. Sets standards for test planning, execution, reporting, and defect management.
- **Security Testing Policy**:
    - **Purpose**: Addresses security vulnerabilities and risks associated with software applications.
    - **Key Points:**
        1. Defines security testing requirements, including penetration testing, vulnerability assessments, and security code reviews.
        2. Specifies protocols for handling sensitive information and protecting data privacy.

- **Testing Environment Policy**:
    - **Purpose:** Ensures the availability and stability of testing environments**.**
    - **Key Points**:
        1. Defines procedures for setting up, configuring, and maintaining testing environments**.**
        2. Specifies requirements for hardware, software, network configurations, and data sets used in testing.
        3. Establishes protocols for environment access, version control, and synchronization with production environments.
- **Compliance and Regulatory Policies**:
    - **Purpose:** Ensures software testing aligns with legal and regulatory requirements.
    - **Key Points**:
        1. Specifies compliance with industry standards (e.g., IEEE standards for software testing).
        2. Addresses regulatory requirements specific to sectors like healthcare (HIPAA), finance (PCI DSS), and data protection (GDPR).
        3. Ensures documentation and audit trails to demonstrate compliance with policies and standards.

- **Documentation and Reporting Policies**:
    - **Purpose**: Ensures transparency and accountability in software testing processes.
    - **Key Points**:
        1. Defines requirements for test plans, test cases, test scripts, and test results documentation.
        2. Specifies reporting formats, metrics, and key performance indicators (KPIs) for measuring testing effectiveness.
        3. Ensures timely and accurate communication of test findings, defects, and recommendations to stakeholders.
- **Training and Skill Development Policies**:
    - **Purpose**: Enhances the competency of testing teams and stakeholders.
    - **Key Points**:
        1. Defines requirements for ongoing training and certification in testing methodologies, tools, and technologies.
        2. Encourages knowledge sharing and collaboration within testing teams.
        3. Supports career development and succession planning for testing professionals.

- **Change Management Policy**:
    - **Purpose**: Manages changes to software applications and their impact on testing.
    - **Key Points**:
        1. Establishes procedures for documenting and assessing changes before testing.
        2. Defines the roles and responsibilities of change control boards and stakeholders.
        3. Ensures coordination between development, testing, and deployment teams to minimize risks and disruptions.

- **Implementation and Adherence**:
    - **Implementation**: Policies should be communicated effectively to all stakeholders and integrated into the software development lifecycle (SDLC).
    - **Adherence**: Regular audits and reviews ensure compliance with policies, identify areas for improvement, and promote continuous enhancement of testing practices.

Thank You