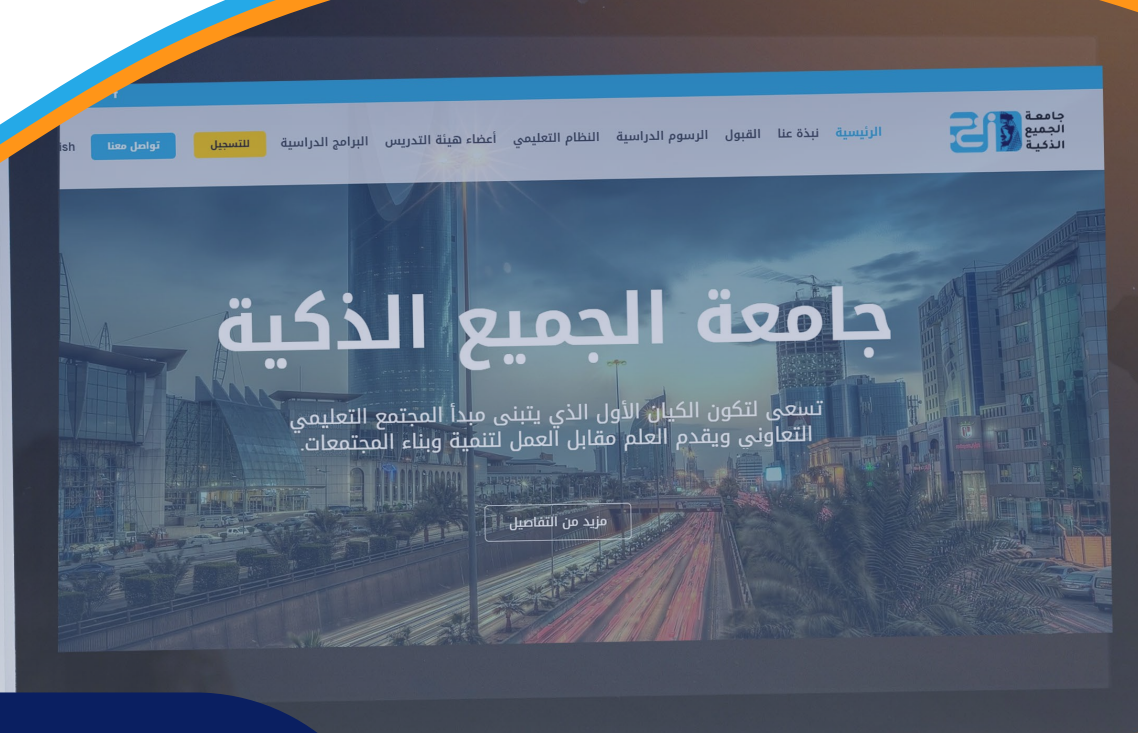




جامعة الجميع الذكية
EVERYONE'S SMART UNIVERSITY



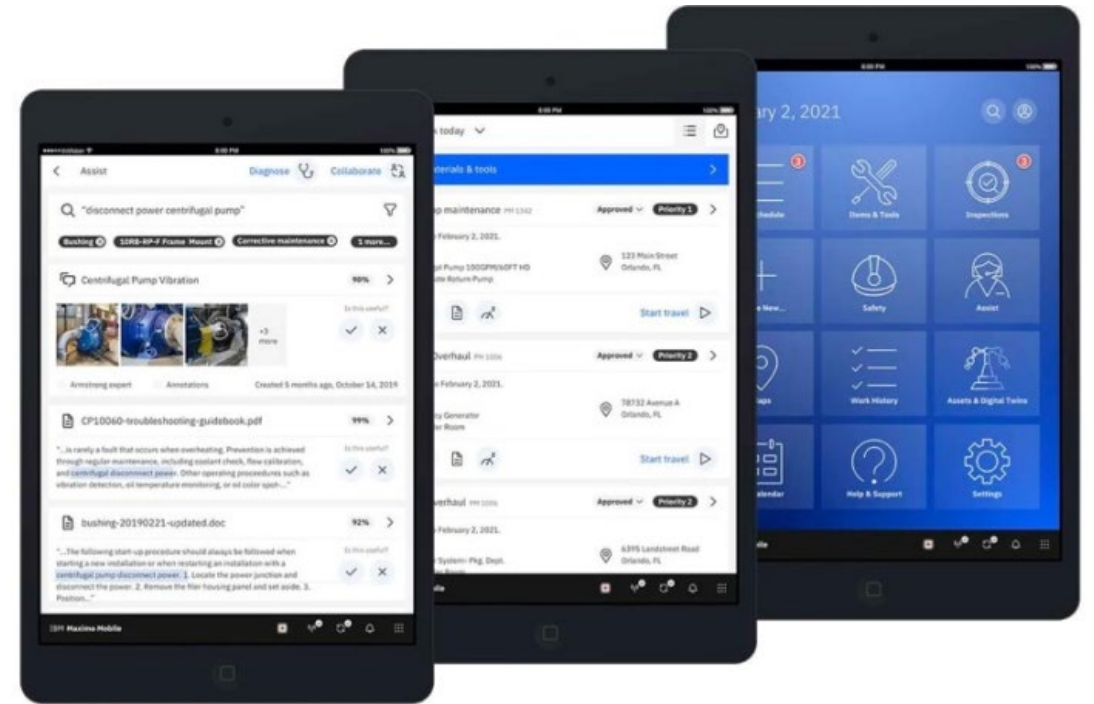
IoT Security
Prepared by Dr. Abdelrhman Ahmed
Lecture 10

Manage
Software & Firmware Risks

Chapter 10

What is IoT Software?

IoT software addresses its key areas of networking and action through **platforms, embedded systems, partner systems, and middleware**. These individual and master applications are responsible for data collection, device integration, real-time analytics, and application and process extension within the IoT network. They exploit integration with critical business systems (e.g., ordering systems, robotics, scheduling, and more) in the execution of related tasks.



Data Collection:- This software manages sensing, measurements, light data filtering, light data security, and aggregation of data. It uses certain protocols to aid sensors in connecting with real-time, machine-to-machine networks. Then it collects data from multiple devices and distributes it in accordance with settings. It also works in reverse by distributing data over devices. The system eventually transmits all collected data to a central server.

Device Integration:- Software-supporting integration binds (dependent relationships) all system devices to create the body of the IoT system. It ensures the necessary cooperation and stable networking between devices. These applications are the defining software technology of the IoT network because without them, it is not an IoT system. They manage the various applications, protocols, and limitations of each device to allow communication.

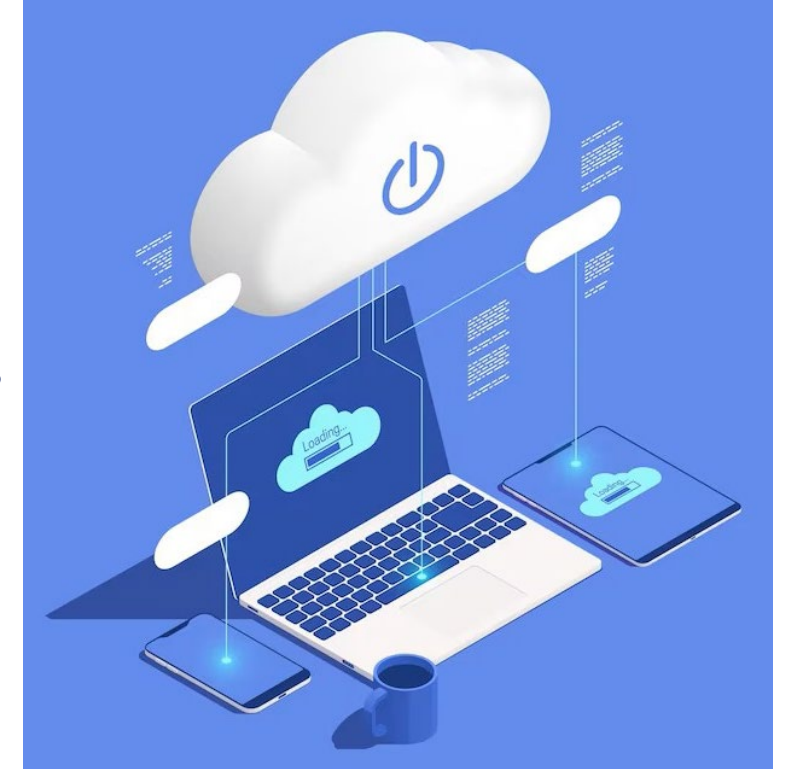
Real-Time Analytics:- These applications take data or input from various devices and convert it into viable actions or clear patterns for human analysis. They analyze information based on various settings and designs in order to perform automation-related tasks or provide the data required by industry.

Application and Process Extension:- These applications extend the reach of existing systems and software to allow a wider, more effective system. They integrate predefined devices for specific purposes such as allowing certain mobile devices or engineering instruments access. It supports improved productivity and more accurate data collection.

Chapter 10

What is Risk in Software Development?

Very simply, a risk is a potential problem. It's an activity or event that may compromise the success of a software development project. Risk is the possibility of suffering loss, and total risk exposure to a specific project will account for both the probability and the size of the potential loss. Guesswork and crisis management are never effective. Identifying and aggregating risks is the only predictive method for capturing the **probability that a software development project will experience unplanned or inadmissible events**. These include terminations, discontinuities, schedule delays, cost underestimation, hardware failures and overrun of project resources



Chapter 10

What is IoT Risk Management?

IoT risk management is a set of processes and practices used to identify and help eliminate potential dangers and negative consequences of IoT vulnerabilities. IoT risk management applies risk management practices and manages the business risk involved based on all facets of how that business uses technology. In particular, the significant growth of the internet of things (IoT) has dramatically increased the number of risk management and security challenges businesses face. Cybercriminals are launching more sophisticated and potentially damaging attacks, and the number of devices that need securing continues to grow as the IoT expands.



Chapter 10

Risks & Benefits of IoT Software

Most Internet of Things (IoT) devices have limited computing ability. IoT software facilitates the movement and processing of data collected by those devices so they can remotely perform their action or function. The software also performs the critical duty of creating an interface for users to automate and control the smart devices, enables the access of data for just-in-time information, and improves sensor data collection.



SAP Internet of Things is a business technology platform that allows users to collect, manage, and embed data from an IoT device into business processes and uses cases such as in product design, manufacturing, supply chain, and operation. It handles a large volume of sensor data that users can easily manage and consume. SAP's IoT platform can sense identified events within a business context to automatically guide businesses in managing their supply chain processes. The IoT software can run business processes and real-time analytics autonomously on edge devices and allow users to orchestrate a process or analysis report from the cloud.

Pros

- Simple and fast IoT development
- Easy device integration
- Responsive customer support

Cons

- Dashboard needs improvement
- Higher license cost compared with other IoT solutions

Microsoft Azure IoT Central provides numerous cloud computing services including an IoT platform. Users can create and develop industry-specific cloud solutions using edge-to-cloud technology with built-in IoT security, privacy, and compliance. Azure IoT offers platform services for companies with cloud solutions and device expertise. Microsoft also offers a managed app platform through Azure IoT Central so companies can quickly build IoT apps with a fully managed solution. It is highly secure, scales as the business grows, and integrates with existing business apps. It provides a ready-made web experience and API surface that simplify IoT development services. Tools include a job scheduler, data storage, custom dashboards, rules, data export, and a public REST API.

Pros

- All tools in one place
- Easy device management
- Responsive customer support

Cons

- High learning curve
- Users wish for more how-to videos

IBM Watson IoT Platform is a SaaS for device management, monitoring, and data storage. It provides users with an IoT system where they can build and extend apps to take advantage of IoT analytics and data from connected devices and sensors. In combination with IBM Cloud technologies, businesses can extract data that provides insights to improve operations and enable innovative business models. Watson IoT includes the Maximo Application Suite for asset management, remote monitoring, predictive maintenance, and mobile enterprise asset management (EAM). Users can manage connected devices and applications including a mobile app can access live and historical data. Using secure REST and real-time APIs, users can connect apps with data from devices and create analytic apps in the cloud.

Cons

- Pricing is on the higher side
- Jenkins integration needs improvement

Pros

- Monitors, tracks, and manages all IoT devices
- Intuitive interface
- Simple and reliable deployment

AWS IoT offers IoT solutions and services that let users connect and manage devices on an enterprise scale. Companies can collect, store, and analyze data from any device used in industrial, consumer, commercial, and industrial settings. Users can quickly build solutions with built-in secure device connectivity, management, storage, and analytics. AWS IoT addresses every security layer of the IoT architecture. IoT security includes preventative mechanisms like encryption and access control with tools to audit and monitor device configurations. The platform also provides access to AI and machine learning together with IoT to make devices more intelligent. It easily integrates with other AWS services for scalability.

Pros

- Fast and reliable service
- Smooth device connections and integration
- Easy deployment to edge devices

Cons

- Pricing is on the high side
- Missing multi-factor authentication (MFA) support for some devices

Google Cloud IoT Core is a managed service for connecting, managing, and ingesting data from dispersed devices in an easy and secure way. In combination with other cloud services, users get a complete solution to collect, process, analyze, and visualize IoT data in real-time so operations can be more efficient. Google Cloud IoT Core can aggregate dispersed device data and integrate it with data analytics services. IoT data stream can be further analyzed and visualized using machine learning to improve operations, anticipate problems, and build new business models. The platform supports standard MQTT and HTTP protocols, 2-way communication with devices, and central management that works with leading hardware manufacturers.

Pros

- Ease of setup
- Support for MQTT and HTTP protocols
- Responsive customer support

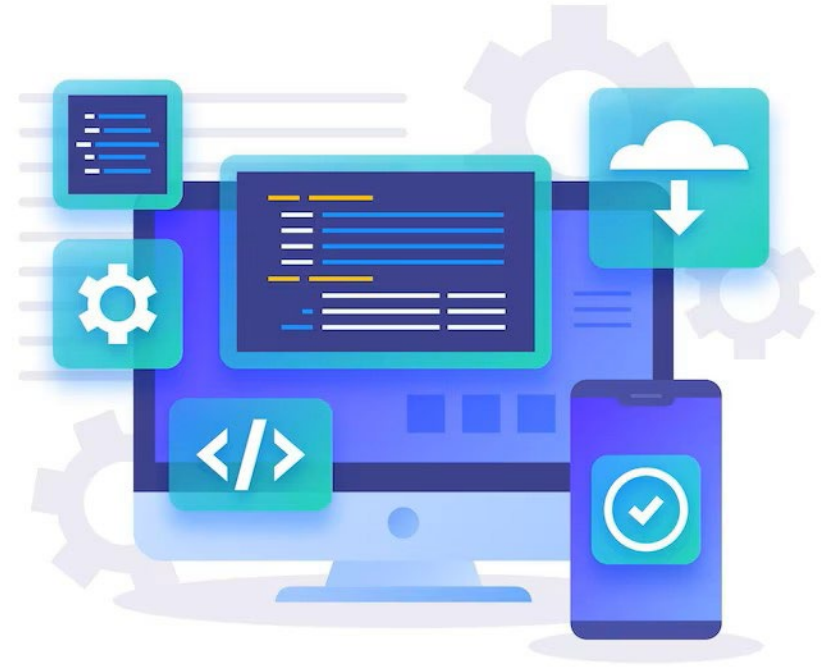
Cons

- High learning curve
- Planned retirement of the software on August 2023

Chapter 10

Manage General Software Risks

Risk management in the software development of any IoT System or Device is the process of identifying, accessing, evaluating risks, then limiting and mitigating them to ensure project success. Any threat, explicit or implicit, small or big, internal or external that threatens the successful rollout of your product should be carefully analyzed and mitigated. Everybody involved in the project is responsible for identifying these threats and communicating them to the project manager for further processing.



Risk management in software development of IoT systems involves the following tasks:

- Identify
- Classify and prioritize
- Develop an action plan template to mitigate
- Continuously monitor throughout the project life cycle
- Implement the action plan if any threat materializes
- Constant communication within the team to identify new threats



Unrealistic Expectations:-

What it means: You expect something from the software, but it doesn't deliver.

Before implementing software, you will have some expectations from it, such as:

- It should streamline operations through automation.
- It should lower costs for one or more business units.
- It should be easy to deploy and use.
- It should seamlessly integrate with the existing technology stack.
- It should help the organization to meet its customer experience or customer success goals.

While it's perfectly OK to have any of these expectations, if expectations don't align with what the software can actually deliver, you will struggle to meet these goals. Such misalignments can turn out to be costly mistakes.

Strategies to Reduce the Risk of Unrealistic Expectations:-

To avoid these problems, vendor communication is critical.

Before the rollout, ask the software vendor pointed questions such as:

- What are the software's unique or new features?
- What can it do? Equally important, what can't it do?
- Can it help us meet "X" goal? How?

Define the implementation timeline and critical milestones to keep the project on track. Finally, identify the must-have deliverables that the vendor must provide to assure that stakeholder expectations will be met.



Adopting Off-the-Shelf Software:-

What it means: A ready-to-use software without modifications may seem attractive – until it's not. Many organizations use commercial off-the-shelf (COTS) software for a wide range of applications, business challenges, and use cases. **COTS can be beneficial if:**

- You want to buy a ready-made, pre-tested, and trusted software.
- You don't need to customize it.
- You don't have a development team to build a bespoke solution.
- You want to save money and avoid expensive greenfield software projects.

End-users can use a COTS in an almost “plug and play” manner.

Adopting Off-the-Shelf Software:-

All that said, COTS can also create risks that harm the organization later. For example, the vendor may stop supporting the software with little or no notice. This can be especially problematic if the COTS has security vulnerabilities that can jeopardize your software supply chain security and increase the risk of cyberattacks and data breaches. The vendor may also provide only generic support that's inadequate for your specific business needs. Many COTS don't offer a simple upgrade path from one version to another. Finally, the software may lack a feature request process and product enhancements such as plugins or add-ons to extend its functionality for your specific use cases.

Strategies to Reduce the Risk of Commercial Off-the-Shelf Software:-

Before investing in COTS, confirm that the vendor will support the product for the entire period you plan to use it.

Inquire about premium support or support per your specific **service level agreements (SLAs)**.

If you reuse any components, your team should have the right skills and access to the source code for a successful implementation. Team members should have control over the application's performance and functionality and be able to add plugins or add-ons per business need.

It's also crucial to find and fix security issues and errors before they open the door to cyber-attacks. One 2021 report found that the average time to fix critical cybersecurity vulnerabilities had increased to 205 days.

That's 205 days a hacker has access to your systems and data because of a software code vulnerability. Ask the vendor about its security patching and version upgrade schedule to protect your organization.



Inadequate Resources for Implementation and Testing:-

What it means: Implementing new software without a dedicated team to manage implementation and testing can affect its ability to meet your needs in a real-world environment.

One reason for software implementation failure is that organizations depend on the vendor for end-to-end implementation and testing. While the vendor's job is to implement the software and customize it to your specifications, only you know your business best, and you must have the right resources in place to manage its implementation, testing, and go-live.

Strategies to Reduce the Risk of Inadequate Resources:-

Identify dedicated resources to work with the vendor on implementation and pre-deployment testing within the project timeframe. Involve the relevant stakeholders and process owners throughout the implementation lifecycle to assure you get the solution you expect and need.

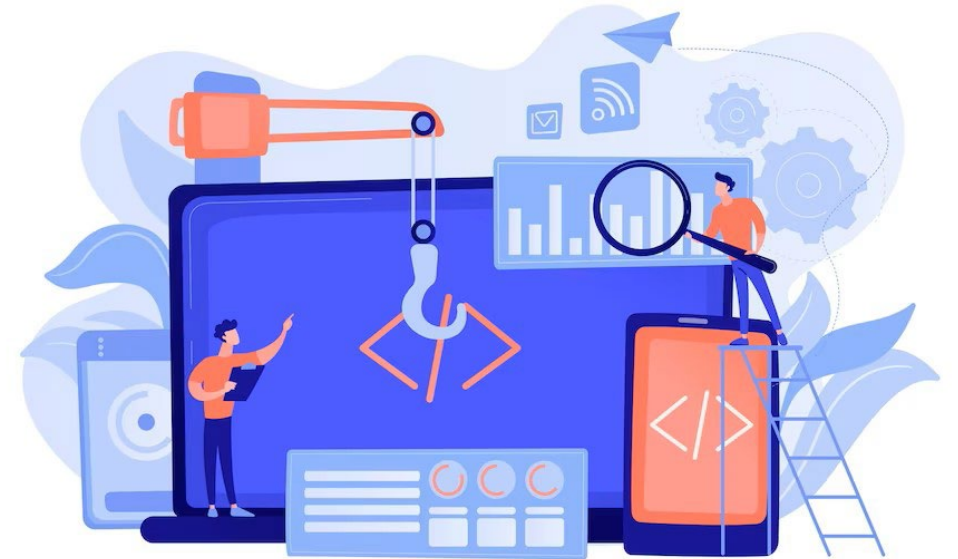
Chapter 10

Risks related to software Installation

Poor Change Management

What it means: A failure to prepare users for the change to new software can limit adoption and result in failure to achieve expected goals.

Humans are creatures of habit, so any new change may cause resistance. When implementing new software, people will push back, delaying implementation or creating adoption challenges after implementation. Poor change management may also lead to a decline in productivity and a demotivated or burned-out workforce.



Strategies to Reduce the Risk of Poor Change Management

For successful software implementation that delivers on its goals, you must manage the people-side of the change. Prepare staff with proper training, onboarding, and guidance to help them adapt to the new system and modify their behavior accordingly.

If possible, identify a few testers and “evangelists” from the end-user group to test the product and drive its adoption after deployment. This will also help the team identify administrative or business process changes that will be necessary to facilitate the success of the new software.

Senior leadership should set and drive the communication strategy to show staff the advantages of the new software. Make sure to clarify how it can benefit their day-to-day work and improve the organization’s efficiency and quality.

Poor Data Migration and Tech Integration

What it means: If the new software cannot integrate with other systems or maintain data integrity, that will affect your data architecture and business outcomes.

Does the new ERP software or CRM platform require data to be migrated from an old system? If yes, the new system should maintain data integrity to assure you don't lose any valuable intelligence that informs your organization's decision-making.

The data migration should happen seamlessly, without affecting existing processes or business continuity. Also, the two applications should operate in tandem until you have completed the migration and implementation.

The new software should also flawlessly integrate with your existing tech stack and offer interoperability with other software. If it doesn't, that might cause usability problems after deployment.

Strategies to Reduce the Risk of Poor Data Migration and Tech Integration

During implementation, your project team must continually verify data integrity as it moves between the old and new applications. They should also assure its security so that nothing falls through the cracks concerning privacy, security, and compliance. It's also important to check that the software aligns with the organization's IT architecture and is fully interoperable with other systems and applications.



Configuration management (CM) is a systems engineering process for establishing and maintaining consistency of a product's performance, functional, and physical attributes with its requirements, design, and operational information throughout its life.

Configuration management is a form of **IT service management (ITSM)** as defined by ITIL that ensures the configuration of system resources, computer systems, servers and other assets are known, good and trusted. It's sometimes referred to as IT automation.

Most configuration management involves a **high degree of automation** to achieve these goals. This is why teams use different tools like Puppet, Ansible, Terraform, and other configuration management tools.

By using automation, it's easier to build in checks and redundancies, improving the potential for omissions due to human error and the accuracy for keeping assets in the desired state.

Chapter 10

Software Configuration Management

Configuration management applies to a variety of systems, but most often, you'll be concerned with these:

- Servers
- Databases and other storage systems
- Operating systems
- Networking
- Applications
- Software



- Manual effort (months) to determine which system components should change when requirements change.
- Failed implementations because your project's requirements changed, and you didn't communicate the changes to all parties.
- Lost productivity by replacing system components with flawed new versions, without the ability to quickly revert to a working state.
- Unexpected outages from incorrectly modifying system components, because you couldn't accurately determine which components were impacted by a change.

Chapter 10

Benefits of Configuration Management

- Reduced risk of outages and security breaches through visibility and tracking of the changes to your systems.
- Cost reduction by having detailed knowledge of all the elements of your configuration, avoiding wasteful duplication of your technology assets.
- Improved experience for your customers and internal staff by rapidly detecting and correcting improper configurations that could negatively impact performance.
- Strict control of your processes by defining and enforcing formal policies and procedures that govern asset identification, status monitoring, and auditing.



Chapter 10

Benefits of Configuration Management

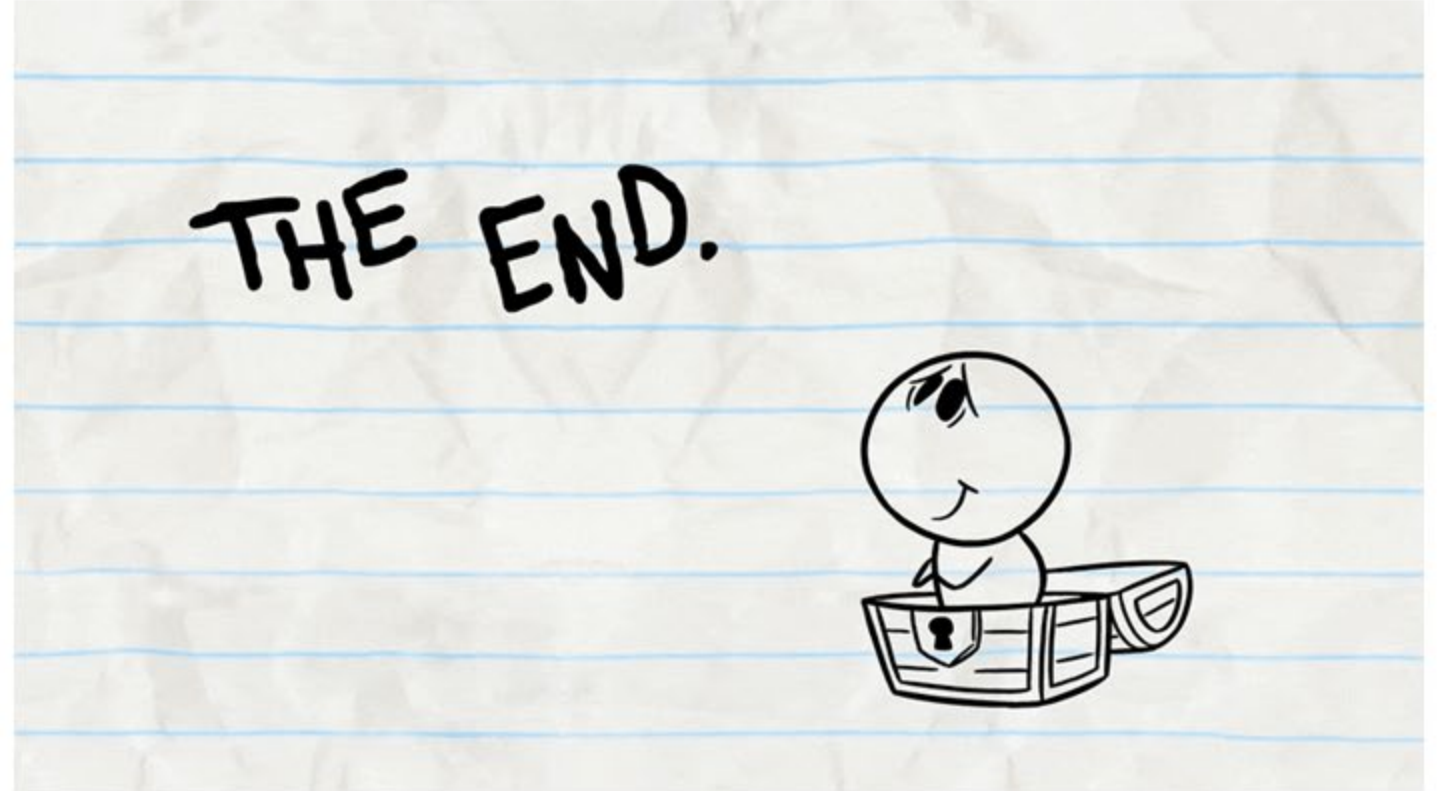
- Greater agility and faster problem resolution, enabling you to provide a higher quality of service and reduce software engineering costs.
- Efficient change management by knowing your baseline configuration, and having the visibility to design changes that avoid problems.
- Quicker restoration of service. In an outage, you'll be able to recover quickly as your configuration is documented and automated.
- Better release management and clear status accounting.



The End

Please feel free to contact me anytime if you have any questions

Abdelrhman Ahmed Refaat:
abdelrhmanahmedrefaat@gmail.com



جامعة القادسية
EVERYONE'S SMART UNIVERSITY