

Contents

[1. Introduction to Software Engineering 3](#_Toc161827608)

[A. What is Software Engineering 3](#_Toc161827609)

[B. Professional software development 3](#_Toc161827610)

[C. Software engineering ethics 4](#_Toc161827611)

[1. Software Evolution 4](#_Toc161827612)

[2. DevOps 4](#_Toc161827613)

[A. Process of DevOps 5](#_Toc161827614)

[3. Background Project 5](#_Toc161827615)

[A. Problem statement 5](#_Toc161827616)

[B. Objective 6](#_Toc161827617)

[C. Scope of the project 7](#_Toc161827618)

[D. Data collection 7](#_Toc161827619)

[3. Conclusion 10](#_Toc161827620)

[4. Reference 10](#_Toc161827621)

# Introduction to Software Engineering

Software Engineering គឺជាផ្នែកមួយនៃ Computer Science​​ និងត្រូវបានកើតឡើងនៅក្នុងអំឡុងឆ្នាំ​​ 1960s, ដែល​ deals ជាមួយនិង​ Designing, Development, Testing, and Maintenance etc នៃ​ Software Application. Software Engineers ធ្វើការអនុវត្តន៍​នូវ​គោលការណ៍​ហើយនិងចំណេះដឹងនៃ​ Programming Languages ដើម្បីបង្កើតនៅ​ Software Solutions សម្រាប់អ្នកប្រើប្រាស់ (End users)​​​​​ ។

## What is Software Engineering

Software Engineering គឺជាដំណើរការនៃការ​ Developing, Testing, and Deploying Computer Application ដើម្បីដោះរាល់ស្រាយបញ្ហាដែលកើតមានឡើងជាក់ស្ដែង​ (Real-World) ដោយប្រកាន់ខ្ជាប់នូវ​គោលការណ៍​​ហើយនិងការ​អនុវត្តន៍​អោយមានភាពល្អបំផុត។​ ហើយផ្នែក Software Engineering នេះគឺបាន​អនុវត្តន៍​ទៅតាមវិន័យហើយនិងវិធី​សាស្រ្ត​ដែលបានរៀបចំរួចរាល់ដើម្បី​ Improve quality, time and budget efficiency, ទៅដល់ Software Development រួមជាមួយនិងការធានាផង។

## Professional software development

Professional Software Development គឺជាដំណើរការនៃការ Designing, and Building Computer Program ដោយ Software Developers​ ។ ហើយ Software Developers គឺពួកគេធ្វើការបង្កើតនូវ Software ឡើងដើម្បីអោយត្រូវនិងតម្រូវការរបស់អ្នកប្រើប្រាស់ (End users)​ ឬ​ក៏​បង្កើតទៅតាម​ Employing Diagrams and Models, and writing code with functionality​ ។ នេះគឺជាដំណើរការរបស់​ Professional Software Development:

* Requirements Analysis: Understanding and gathering requirements from stakeholders to determine the functionalities and features the software must have. Translating business needs into technical specifications.
* Design: Creating a high-level architecture and detailed design for the software. Considering factors like scalability, maintainability, and performance during the design phase.
* Implementation/Coding: Writing code based on the design specifications. Following coding standards and best practices to ensure code quality.
* Testing: Conducting thorough testing, including unit testing, integration testing, and system testing, to identify and fix bugs. Implementing automated testing to improve efficiency and reliability.
* Version Control: Using version control systems (e.g., Git) to manage code changes, collaborate with team members, and track the history of the project.
* Collaboration: Working collaboratively with cross-functional teams, including designers, product managers, and quality assurance professionals. Effective communication is crucial for successful collaboration.
* Documentation: Creating and maintaining documentation for code, design decisions, and project processes. Ensuring that documentation is accessible and understandable by team members and stakeholders.
* Deployment: Deploying software to production environments. Implementing continuous integration and continuous deployment (CI/CD) practices to streamline the release process.
* Maintenance and Support: Addressing and resolving issues that arise after the software is deployed.

Implementing updates, patches, and improvements based on feedback and changing requirements.

* Security: Incorporating security best practices throughout the development lifecycle to protect against potential vulnerabilities and threats.
* Adaptability and Continuous Learning: Staying updated with new technologies, methodologies, and best practices in the rapidly evolving field of software development. Being adaptable to changing project requirements and business needs.
* Quality Assurance: Ensuring that the software meets quality standards and complies with relevant regulations. Conducting code reviews and other quality assurance processes.

## Software engineering ethics

Software engineering ethics can be approached from three directions. First, it can describe the activity of software engineers making practical choices that affect other people in significant ways. Second, it can be used to describe a collection of principles, guidelines, or ethical imperatives that guide or legislative action, and third, it can be used to name a discipline that studies the relationship between the other two senses of ethics. Software engineering ethics is clearly both an activity and a body of principles. The discipline of software engineering ethics that studies this activity and formalizes these principles, however, is in its infancy.

# Software Evolution

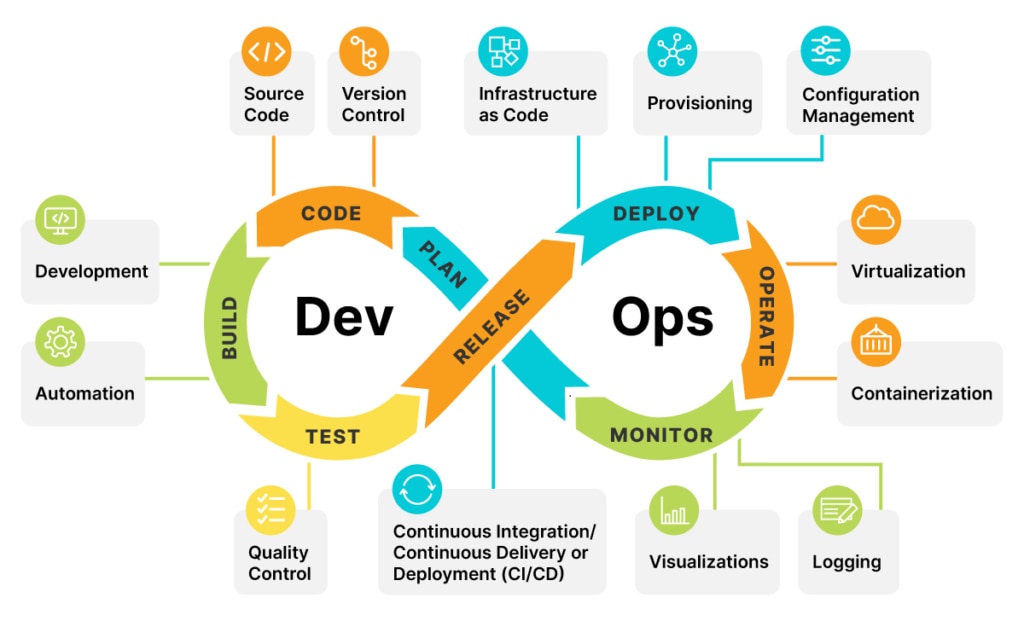
Software evolution is the process of developing software initially and then updating it for various reasons, such as adding new features, removing obsolete functionalities, or meeting changing requirements. It is an important process because organizations invest large amounts of money in their software and are completely dependent on this software. Software evolution helps software adapt to changing business requirements, fix defects, and integrate with other changing systems in a software system environment. The process of software evolution is ongoing and requires continuous investment of time and resources.

# DevOps

DevOps is a set of practices, tools, and a cultural philosophy that automate and integrate the processes between software development and IT teams. It emphasizes team empowerment, cross-team communication and collaboration, and technology automation.

The DevOps movement began around 2007 when the software development and IT operations communities raised concerns about the traditional software development model, where developers who wrote code worked apart from operations who deployed and supported the code. The term DevOps, a combination of the words development and operations, reflects the process of integrating these disciplines into one, continuous process.

## Process of DevOps



# Background Project

UGamer is a computer store that was Open on 22/05/2019 with high percentage of people visiting, it’s situated in The Olympia mall (1st floor Manager luch visal) that is open at 7/12 form (Monday until Sunday) .

## Problem statement

នៅពេលដែលComputer store មិនទាន់មានប្រព័ន្ធគ្រប់គ្រង ពេលនោះគេបានជួបបញ្ហាមួយចំនួនដូចជា៖

1. ពិបាកគ្រប់គ្រងលើStock
2. ពិបាកគ្រប់គ្រងទិន្នន័យរបស់អតិថិជន, បុគ្គលិកនិងSupplierប្រចាំថ្ងៃ​
3. អាចគណនាទឹកប្រាក់ខុសមានកម្រិតខ្ពស់
4. មានផលវិបាកក្នុងការរក្សាទុកឯកសារ
5. ពិបាកគ្រប់គ្រងប្រាក់ចំណូលនិងចំណាយប្រចាំកំឡុងពេល ពិបាកធ្វើរបាយការណ៍ប្រចាំកំឡុងពេល។

## Objective

ដោយសារមានបញ្ហាទាំងនោះកើតឡើងទើប Ugamer Computer storeបង្កើតនូវប្រព័ន្ធគ្រប់គ្រង ដើម្បីអោយមានភាពងាយស្រួលដូចជា៖

1. ការគ្រប់គ្រងStock
2. ការគ្រប់គ្រងទិន្នន័យរបស់អតិថិជន, បុគ្គលិកនិងSupplierប្រចាំថ្ងៃ​
3. គណនាទឹកប្រាក់ និងឆាប់រហ័ស
4. រក្សាទុកឯកសារ
5. ការគ្រប់គ្រងប្រាក់ចំណូលនិងចំណាយប្រចាំកំឡុងពេល ធ្វើរបាយការណ៍ប្រចាំកំឡុងពេល។

## Scope of the project

ប្រព័ន្ធគ្រប់គ្រង Computer Store នេះគឺមាននូវលទ្ធភាពនិងមុខងារដូចជា៖

1. ប្រព័ន្ធសុវត្តិភាពនិងការផ្ដល់សិទ្ធទៅអោយអ្នកប្រើប្រាស់
2. ការបញ្ចូលនិងការកែប្រែព័ត៌មានរបស់អតិថិជន, បុគ្គលិកនិងSupplier
3. ការគ្រប់គ្រងទៅលើស្តុក
4. ការធ្វើវិក្ក័យប័ត្រលក់និងវិក្ក័យប័ត្រទិញ
5. ការត្រួតពិនិត្យទៅលើចំណូលនិងចំណាយផ្សេងៗ
6. មានDatabaseសម្រាប់រក្សាទុករាល់ទិន្នន័យទាំងអស់  
   (SQL Server)
7. អាចបង្ហាញពីរបាយការណ៏ប្រចាំថ្ងៃឬកំឡុងពេល។

## Data collection

* Employee ID
* Employee Name
* Employee Contact
* Employee Gen
* Employee Position
* Employee Salary
* Employee DOB
* Employee ADD



. Employees



* Customer ID
* Customer Name
* Customer Gen
* Customer Contact
* Customer Address



. Products

* Product ID
* Brand
* Product Category
* Product Name
* Product Cost
* Sale Price
* Quantity

. Supplier

* Supplier ID
* Supplier Name
* Supplier Gen
* Supplier Contact
* Supplier ADD

. Customers



. Purchase order

. Invoice

* Invoice ID
* Invoice Date
* Employee Name
* Customer Name
* Product Name
* Quantity
* Unit Price
* Total
* Grand Total
* Invoice ID
* Invoice Date
* Employee Name
* Customer Name
* Product Name
* Quantity
* Unit Price
* Total
* Grand Total
* PO ID
* Date
* Ordered By
* Delivery By
* Items
* Items Price
* Total Price
* PO ID
* Date
* Ordered By
* Delivery By
* Items
* Items Price
* Total Price

# Conclusion

In conclusion, DevOps has revolutionized the software development and deployment process. It has helped organizations to bring developers and operations teams together to work seamlessly towards a common goal, leading to a better understanding of the application's requirements and how it will be deployed.

# Reference

for Software engineering ethics [(PDF) Software Engineering Ethics (researchgate.net)](https://www.researchgate.net/publication/227991826_Software_Engineering_Ethics)

https://www.devopsschool.com/slides/devops/module-5-conclude-devops/index.html#/10