

## TTSA02 : Patterns of Enterprise Application Architecture

### Description :

หลักสูตรนี้จะให้ความรู้ความเข้าใจเชิงลึกในการใช้ patterns ทางด้านสถาปัตยกรรม (Architectural Patterns) ในการแก้ไขปัญหาในการวิเคราะห์และออกแบบสถาปัตยกรรมของแอปพลิเคชัน ระดับองค์กร (Enterprise Application Architecture) เพื่อให้การพัฒนาแอปพลิเคชันสำหรับองค์กรมีประสิทธิภาพ แข็งแรง ปรับปรุงแก้ไขได้ง่าย และรองรับการเกิดขึ้นของเทคโนโลยีที่หลากหลายและรวดเร็ว และรวมถึงแพลตฟอร์มที่เป็น object-oriented แบบ multi-tier เช่น Java และ .NET ซึ่งใช้กันอย่างกว้างขวาง และรวมถึงสถาปัตยกรรมแบบกระจาย (Distributed Computing) และสถาปัตยกรรมเชิงบริการ (Service-Oriented Architecture: SOA)

### Instructor :



Training Date : 27 พ.ย. 2567 - 29 พ.ย. 2567

fee : 12,000 ฿ (ราคายังไม่รวม Vat 7%)

Days & Duration : 3 Day(s) | 18 Hour(s)

Time : 10:00:00 - 17:00:00

Language : Thai

Venue : Software Park Training Room 3rd floor, Software Park Building Chaengwattana Road, Pakkred Nonthaburi

Type : Classroom

Category : Software Architecture and Design

### นายณรงค์ จันทร์สร้อย

### Objectives :

Patterns เปรียบเสมือนสูตรลับหรือสมการทางคณิตศาสตร์ หากผู้ปฏิบัติรู้จักมากเท่าใดและสามารถนำไปประยุกต์ให้เหมาะสมได้มากเท่าใด จะช่วยทำให้การคำนวณ การพัฒนา หรือการสร้างงานสำเร็จลุล่วงได้รวดเร็วยิ่งขึ้น มีความถูกต้อง และมีมาตรฐานยิ่งขึ้น ความผิดพลาดที่พบได้บ่อยสำหรับนักออกแบบหรือ Software Architect หรือผู้ที่รับผิดชอบในการออกแบบสถาปัตยกรรมซอฟต์แวร์ในระดับองค์กร (Enterprise Architecture) เช่น

- การแบ่งและจัดระเบียบลอจิก (logic partitioning) ออกเป็น layer โดย เรียกว่า Architectural Layer
- วิธีการ ต่าง ๆ ในการจัดการกับตรรกะทางธุรกิจ (business logic)
- การทำ mapping ระหว่างการออกแบบ Data Model ในรูปแบบ object (Object Data Model) และในรูปแบบฐานข้อมูลแบบ relational (ER Diagram)
- การใช้ Model-View-Controller ในการจัดการ Web presentation
- การจัดการ concurrency ของหลาย transaction รวมถึงเทคนิคการทำ locking แบบต่างๆ
- การออกแบบ interface ของ distributed object
- การจำกัด ช่องทาง (channel) ในการติดต่อขอใช้บริการ (service) โดยโคลเ็นต์
- การจำกัด ช่องทาง (channel) ในการติดต่อขอใช้บริการ (service) ของคอมพิวเตอร์หรือแอปพลิเคชันที่อยู่เครื่องอื่น
- การสร้าง เพรมเวิร์กขึ้นใช้เอง
- การออกแบบ transaction ให้มีประสิทธิภาพ รวมถึงการจัดการด้าน session
- การติดต่อ กับฐานข้อมูลในรูปแบบต่าง ๆ และการไหลของข้อมูลในรูปแบบต่าง ๆ

หลักสูตรนี้จะให้ความรู้ความเข้าใจเชิงลึกในการใช้ patterns ทางด้านสถาปัตยกรรม (Architectural Patterns)

ในการแก้ไขปัญหาในการวิเคราะห์และออกแบบสถาปัตยกรรมของแอปพลิเคชัน ระดับองค์กร (Enterprise Application Architecture)

เพื่อให้การพัฒนาแอปพลิเคชันสำหรับองค์กรมีประสิทธิภาพ แข็งแรง ปรับปรุงแก้ไขได้ง่าย และรองรับการเกิดขึ้นของเทคโนโลยีที่หลากหลายและรวดเร็ว และรวมถึงแพลตฟอร์มที่เป็น object-oriented แบบ multi-tier เช่น Java และ .NET ซึ่งใช้กันอย่างกว้างขวาง และรวมถึงสถาปัตยกรรมแบบกระจาย (Distributed Computing) และสถาปัตยกรรมเชิงบริการ (Service-Oriented Architecture: SOA)

แนวคิดและเทคโนโลยีใหม่เหล่านี้สามารถที่จะสร้างแอปพลิเคชันที่มีศักยภาพสูง นอกจากนี้จะเน้นหนักที่การฝึกสร้างระบบความคิดให้มีประสิทธิภาพ สอดแทรกแนวคิดปรัชญาการออกแบบ และจะแสดงให้เห็นว่าการศึกษาและฝึกฝนด้าน Pattern ที่ดีต้องไม่เน้นที่การจำ โดยเฉพาะการให้ความสำคัญกับซอร์สโค้ดและไดอะแกรม UML มากเกินไป จนพยายามจะทำความเข้าใจและจดจำซอร์สโค้ดและไดอะแกรม UML ในตัวอย่างให้ได้ ซึ่งแท้จริงไม่จำเป็นหรือไม่ใช่สาระสำคัญหลัก และเป็นการกระทำที่เสียเวลา เพราะโลกนี้มี Pattern นับร้อย ดังนั้นหลักสำคัญคือต้องเน้นที่การศึกษาและฝึกที่ระบบความคิด โดยเฉพาะการวิเคราะห์ปัญหา การทำความเข้าใจแก่นหรือบริบทของปัญหา (context / problem domain) และการนำ Pattern มาใช้และเรียงร้อย (Weave / orchestrate) เพื่อสร้างโซลูชันที่เหมาะสมกับงาน

การออกแบบที่ทำการสอนในหลักสูตรนี้มีการใช้ UML บ้าง แต่ไม่ซับซ้อนมากนักเกินไป จึงไม่มีปัญหาสำหรับผู้เรียนที่เพิ่งเริ่มต้นแต่อย่างใด และหลักสูตรนี้ไม่เน้นการอธิบายหรือยกตัวอย่างด้วยซอร์สโค้ด ซึ่งไม่มีที่ไหนที่สอน Pattern ในลักษณะแบบนี้ เพราะหลักสูตรนี้ไม่ต้องการให้ผู้เข้าอบรมเน้นซอร์สโค้ด ดังนั้นผู้เข้าอบรมจึงสามารถนำองค์ความรู้ความเข้าใจได้ภายหลังอบรมไปประยุกต์กับการทำงานที่ใช้ภาษาโปรแกรมอะไรก็ได้

### Target Group :

- การอบรมนี้ออกแบบมา สำหรับโปรแกรมเมอร์ นักออกแบบซอฟต์แวร์ และสถาปนิกซอฟต์แวร์ (Software Architect) ซึ่งพัฒนาแอปพลิเคชันสำหรับองค์กร (Enterprise Application) และผู้ที่ต้องการพัฒนาความเข้าใจ หรือการสื่อความหมายในเรื่องเกี่ยวกับสถาปัตยกรรมซอฟต์แวร์

### พื้นฐานของผู้เข้ารับการอบรม (Prerequisites)

- ประสบการณ์ทางด้าน การออกแบบและพัฒนาระบบซอฟต์แวร์

- มีความเข้าใจในหลักการพื้นฐานของสถาปัตยกรรมซอฟต์แวร์
- มีความเข้าใจสัญลักษณ์และไดอะแกรม ของ UML มาบ้าง

#### Benefits :

- หลักสูตรนี้ผู้เข้ารับการอบรมจะเข้าใจถึงการออกแบบสถาปัตยกรรมซอฟต์แวร์ระดับองค์กรให้มีศักยภาพสูงสุดโดยใช้ Architectural Patterns ซึ่งผู้เข้ารับการอบรมจะได้เรียนรู้ patterns ถึงประมาณ 50 ตัวตลอด 3 วันที่อบรม
- ผู้เข้ารับการอบรมจะเข้าใจมุมมองต่างๆ ของ patterns ทางด้านสถาปัตยกรรม และจะสามารถนำไปใช้ในการออกแบบและพัฒนาสถาปัตยกรรมของแอปพลิเคชันสำหรับองค์กรที่ทำอยู่ได้
- ผู้เข้ารับการอบรมสามารถนำความรู้ไปใช้ได้กับทั้งแอปพลิเคชันที่พัฒนาด้วย Java และ .NET
- นอกจากนั้นการอบรมต้องการให้ผู้เข้าอบรมเข้าใจถึงแก่น และการนำ Pattern ไปใช้เป็นเครื่องมือประกอบการแก้ไขปัญหาในการออกแบบ ซอฟต์แวร์ และเข้าใจถึงปรัชญาการออกแบบที่ดี และพื้นฐานการเป็นนักกลยุทธ์ด้านการออกแบบซอฟต์แวร์ที่ดี
- ได้ศึกษา และฝึกกระบวนการคิดเพื่อเป็นพื้นฐานที่ดีต่อการเป็นนักออกแบบซอฟต์แวร์โดยใช้ Pattern เป็นเครื่องมืออย่างมีประสิทธิภาพ โดยผู้เข้าอบรมจะได้เข้าใจถึงแก่นของ Pattern สามารถศึกษา Pattern อื่นๆ อีกมากมายต่อไปได้ด้วยตนเอง สามารถต่อยอดและสร้าง Pattern ใหม่ๆ ได้ด้วยตนเอง และสามารถนำองค์ความรู้ด้าน Pattern ไปประยุกต์หรือปรับปรุงกับกระบวนการพัฒนาซอฟต์แวร์ในองค์กรและเพื่อสร้างนักออกแบบซอฟต์แวร์ที่ใช้ Pattern ได้อย่างชำนาญให้เกิดขึ้นมากๆ ภายในองค์กรของตน
- นอกจากนั้นผู้เข้าอบรมจะเข้าใจถึงหลักการวิเคราะห์และทำความเข้าใจกับปัญหา เพราะหากไม่เข้าใจปัญหาแล้ว แม้รู้จัก Pattern นับร้อยก็ไม่เกิดประโยชน์ใดๆ

#### Course Outline :

- Introduction to Patterns
- Introduction to Enterprise Application
- Architectural Layering
- Domain Logic Patterns
- Data Source Architectural Patterns
- Object-Relational Behavioral Patterns
- Object-Relational Structural Patterns
- Object-Relational Metadata Mapping Patterns
- Web Presentation Patterns
- Distribution Patterns
- Offline Concurrency Patterns
- Session State Patterns
- Base Patterns

#### Payment Condition :

##### Payment can be made by:

1. Cash or Credit Card or Bank Cheque payable to "สำนักงานพัฒนาวิทยาศาสตร์และเทคโนโลยีแห่งชาติ" (a post-dated cheque is not accepted) on the first day of the service or within the last day of the service.
2. **Account transfer** and send the proof of the payment (the deposit slip) to email [ttd@swpark.or.th](mailto:ttd@swpark.or.th)

- ธนาคารกรุงเทพ สาขาอุทยานวิทยาศาสตร์
  - Saving Account Number: 080-0-00001-0
  - Account Name: สำนักงานพัฒนาวิทยาศาสตร์และเทคโนโลยีแห่งชาติ
- ธนาคารกรุงไทย สาขาสลาดไทย
  - Saving Account Number: 152-1-32668-1
  - Account Name: สำนักงานพัฒนาวิทยาศาสตร์และเทคโนโลยีแห่งชาติ

#### Notes:

- Withholding tax (3%) is exempt.
- Should you need to withdraw, you must send the notice of the withdrawal in writing no later than 7 working days before the commencement date. The cancellation less than 7 days will be subject to a fine of 40% of the fee.
- Software Park Thailand reserves the rights to cancel courses due to unforeseen circumstances.

#### Contact Person :

For more information, contact our course coordinator on:

Namfhon Pongyat

Tel: +66-2583-9992 Ext. 81427

Email: [namfhon@swpark.or.th](mailto:namfhon@swpark.or.th)



You are encouraged to use the course schedule as a guide to plan your training.  
The schedule is accessible at [www.swpark.or.th](http://www.swpark.or.th) for more information.