لغة النمذجة الموحدة (UML)

مقدمة

- تقدم لغة النمذجة الموحدة مجموعة من أفضل خبرات الممارسات الهندسية التى ثبت نجاحها فى نمذجة النظم الكبيرة والمعقدة، وهى جزء مهم من التطوير الكائنى المنحى للبرمجيات وعمليات تطوير البرمجيات.
- ليست لغة النمذجة الموحدة منهجية لبناء أو تصميم البرمجيات وتطويرها.
- لا ترتبط لغة النمذجة الموحدة بمنهجية أو طرق إنتاج البرمجيات، ويمكن توظيف هذه اللغة على مختلف العمليات البرمجية بغض النظر عن المنهجية المتبعة.
- تتألف لغة النمذجة الموحة من أربع طبقات أساسية تقسم كل طبقة بدورها إلى طبقات فرعية

- **لغة (UML) :**

هي اختصار إلى (Unified Modeling Language) .
باللغة العربية تسمى (لغة النمذجة الموحده).

- **استخدام لغة (UML) :**

في عمل التخطيط لمشاريع البرمجة قبل البدا في عمل المشروع البرمجي عملياً.

- **مكونات لغة (UML) :**

١. أشكال هندسية .
٢. رموز مثل (مربع ، مستطيل ، خط مستقيم ، أسهم ، سداسي) .

ملاحظة :- كل رمز من الرموز يرمز إلى شيء في المشروع البرمجي.

- **ابرز سمات لغة (UML) :**

١. لا ترتبط لغة النمذجة الموحدة بمنهجية أو طرق إنتاج البرمجيات .

٢. تقدم لغة النمذجة الموحدة مجموعة من أفضل خبرات الممارسات الهندسية .

٣. ليست لغة النمذجة الموحدة منهجية لبناء أو تصميم البرمجيات وتطويرها.

لماذا تستخدم لغة النمذجة الموحدة UML

و بناءا على استراتيجية زيادة اهمية البرامج للعديد من الشركات ، فمجال الصناعة سعى بالبحث عن تقنيات لجعل إنتاج البرمجيات أوتوماتيكي، مع تحسين النوعية والضغط على التكلفة والوقت لزيدة القدرة التنافسية في السوق. وتشمل هذه التقنيات تقنية المركبات " component" ، و القوالب "patterns" ، والبرمجة المرئية ""visual programming" technology" الشركات تسعى أيضا إلى تقنيات لإدارة تعقيدات الأنظمة ."frameworks" ومنصات العمل " لأنها في زيادة من حيث الحجم و المدى. وعلى وجه الخصوص ، فهي تعترف بالحاجة إلى حل المشاكل الهندسية المتكررة ، مثل التوزيع المادي " physical distribution" ، التزامن "load balancing" ، الأمن ، الموازنة ""replication" ، التكرار ""concurrency" و بالإضافة إلى ذلك ، قد أدى تطور شبكة الويب العالمية ، ."fault tolerance" الاحتمال الخطأ " مما جعل بعض الأمور أكثر بساطة ، في زيادت المشاكل الهندسية. وقد تم تصميم لغة النمذجة للاستجابة لهذه الاحتياجات. "UML الموحدة "

من أربع طبقات أساسية UML تتألف لغة النمذجة

١. طبقة كائنات المستخدم User Objects Layer

هي الطبقة السطحية العامة التي يستخدمها الذين يتعاملون مع لغة النمذجة الموحدة،
وتتألف من تسعة مخططات رئيسية بالإضافة إلى كائنات وأدوات مساعدة
وهى الطبقة الأكثر وضوحاً وتوصيفاً، المستخدم هنا المقصود هو مستخدم اللغة وليس المستخدم النهائي للبرمجية أو المنتج البرمجي .

تشمل الطبقة الأولى المخططات التسعة التالية :

* مخطط حالات لاستخدام.
* مخطط الفئة.
* مخطط الكائن.
* مخطط النشاط.
* مخطط التعاون.
* مخطط الحالة.
* مخطط التتابع .
* مخطط المكونات .
* مخطط التوزيع.

٢. الطبقة الثانية :تسمى طبقة النموذج Model Layer

وتكون فى المرحل الأولى من التحليل حيث تحتوى على مفاهيم موضوع التحليل كفهم النظام ويستخدم هذه الطبقة محلل النظام أثناء عمله قبل نضج فكرة النظام أو وضوحه بتوصيف أقل من طبقة المستخدم الأولى.

٣. الطبقة الثالثة: طبقة ما وراء النموذج Meta Model Layer

وتعنى هذه الطبقة بالمفاهيم المتعلقة بلغة النمذجة الموحدة كمفهوم الصنف والظاهرة ونوع البيانات والتجريد والأنماط وغيرها من مفاهيم اللغة، وهى طبقة تصف ما يجرى فى النموذج وتتألف من ثلاث حزم رئيسية هى:

* حزمة الأساس .
* حزمة العناصر السلوكية.
* حزمة إدارة النموذج

٤. الطبقة الرابعة : طبقة ما وراء ما وراء النموذج
(Meta meta model layer)

طبقة لا تهم معظم محللي النظم تهتم بلغة كتابة النمذجة الموحدة، وتهم مطوري أدوات لغة النمذجة الموحدة مثل البرامج التي تؤتمت المخططات وترسمها .

١. مخطط حالة الاستخدام (Use Case Diagram) :
يعرض العلاقة بين الجهات الفاعلة (actors) وحالات الاستخدام(use cases).

٢. مخطط الفئة (Class Diagram):
نماذج هيكل الفئة (class structure)ومحتوياتها تستخدم عناصر التصميم مثل
الفئات(classes)، والحزم والكائنات (objects).

ويعرض أيضا علاقات مثل الاحتواء(containment) ، التوريث(inheritance) ، و التجميع (associations) وغيرها.

٣. مخطط التفاعل (Interaction Diagrams):

- **مخطط التسلسل (Sequence Diagram)**

يعرض التسلسل الزمني للكائنات (objects) المشاركة في التفاعل (interaction).
و هذا يتألف من البعد العمودي (الوقت) والبعد الأفقي (الكائنات المختلفة — different objects) .

- **مخطط التعاون (CollaborationDiagram)**

يعرض التفاعل المنظم حول الكائنات "objects" وعلاقاتها مع بعضها البعض. تستخدم الأرقام لإظهار تسلسل الرسائل.

٤ . مخطط الحالة (State Diagram) :

يعرض تسلسل الحالات التي يمر من خلالها كائن التفاعل (object of an interaction) في حياته في عملية الاستجابة لمحفزات وردت سابقا ، جنبا إلى جنب مع ردودها والإجراءات.

٥. مخطط النشاطات (Activity Diagram):

يعرض المخططات المميزة لمخطط الحالة (state diagram) ، بحيث ان معظم الحالات هي حالات العمل (action states) ومعظم الانتقالات يتم تشغيلها بواسطة إنهاء الإجراءات في مصدر الحالات. هذا المخطط يركز على التدفقات المدفوعة من المعالجات الداخلية (internal processing).

٦. المخططات المادية (Physical Diagrams) :

- **مخطط المركب (Component Diagram)**

يعرض المستوى المرتفع لهيكل حزمة الشفرة نفسها (packagedstructure ofthe code). بالاعتماد على المركبات (components) التي ظهرت ، بما في ذلك مركبات شفرة المصدر (sourcecode components) والمكونات البرمجية الثنائية (binary code components) ، والمكونات القابلة للتنفيذ (executable components).

- **تخطيط النشر (Diagram Deployment)**

يعرض التكوين (configuration) لعناصر التجهيز وقت التشغيل (run-time processing) ومكونات البرامج (software components) ، والعمليات (processes)، والكائنات (objects) التي تعمل فيها. حالات مكونات البرامج (Software component instances) تقدم مظاهر وقت التشغيل لوحدات التعليمات البرمجية (codeunits).