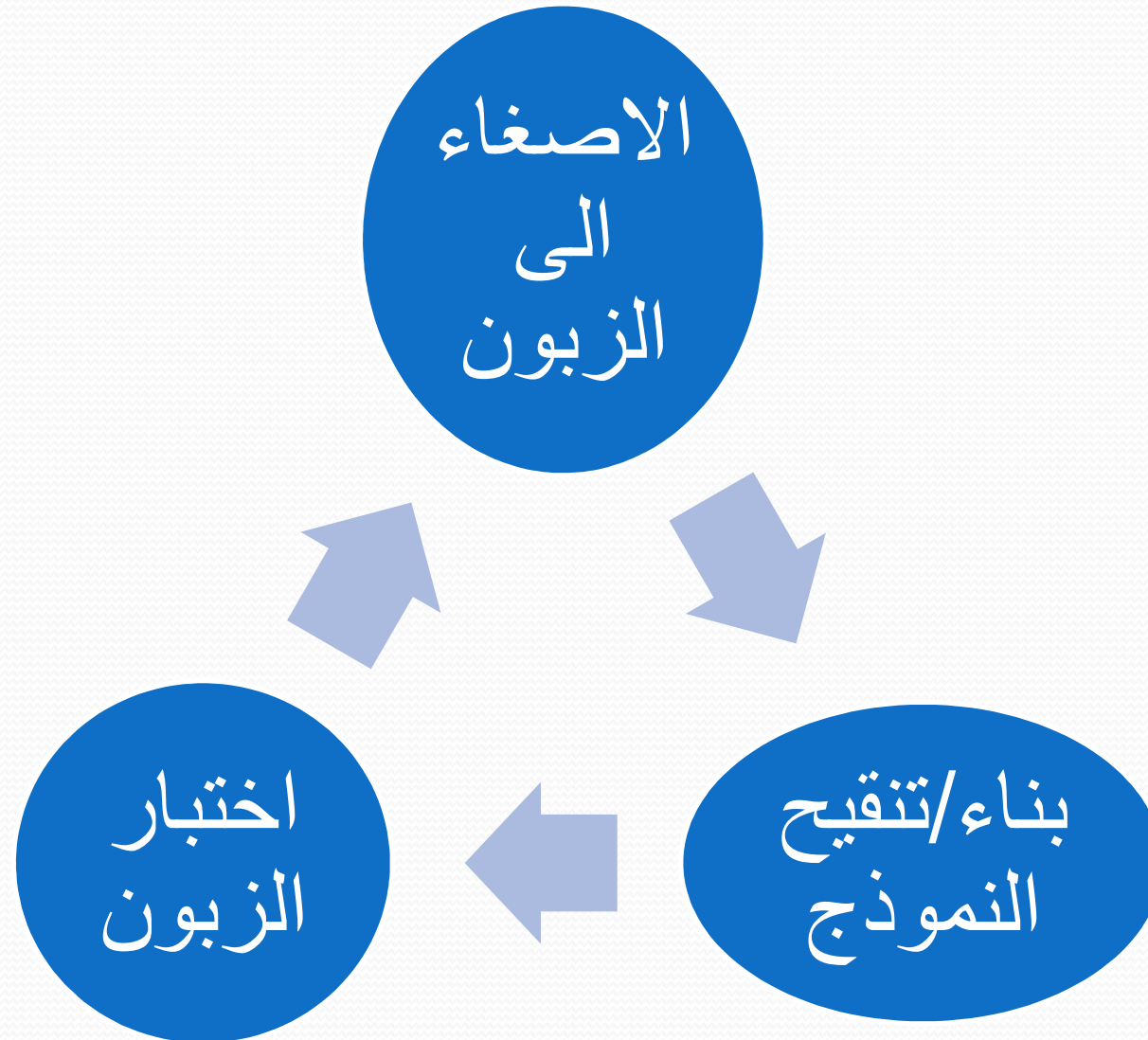


هندسة البرمجيات
المحاضرة الرابعة
نماذج عمليات البرمجيات
مدرس المادة: م.م نور حسن حسون

نموذج "النمذجة الأولية" Prototype Model

- يبدأ هذا النموذج من خلال تعريف الزبون مجموعة من الأهداف العامة للبرنامج، ولا يحدد بالتفصيل كل متطلبات الادخال أو المعالجة أو الاخراج،
- في بعض الحالات قد يكون مطور غير متأكد من فعالية الخوارزمية او النموذج، أو الشكل الذي يجب أن يأخذه تفاعل الإنسان مع الآلة، لذلك فان نموذج النمذجة الأولية (Prototyping Paradigm) جاء ليقدم الطريقة الفضلى في حل المشاكل في هذه الحالات وحالات كثيرة غيرها.
- الشكل التالي يبين نموذج النمذجة الاولية:

نموذج "النمذجة الأولية" Prototype Model



نموذج "النمذجة الأولية" Prototype Model

- خطوات هذه النموذج تكون كالآتي:

١. الاصغاء الى الزبون Listen to customer

- يبدأ هذا النموذج بجمع المتطلبات النظام من خلال اجتماع يحصل بين مطور النظام والزبون لتعريف وتحديد الأهداف الإجمالية للبرنامج.

٢. بناء النموذج build/revise mock-up

- تطوير نموذج أولي (مبدئي ، تجريبي): استنادا إلى حاجات الزبون او المستخدم، حيث يعمل مصممو النظم على تأمين نموذج سريع بواسطة البرمجيات والأدوات المساعدة.

نموذج "النمذجة الأولية" Prototype Model

٣. اختبار الزبون customer test drives mock-up

- تحديد إذا كان النموذج الأولي مقبولا أم لا (تجربة النموذج).

- في هذه المرحلة يعيد الزبون أو المستخدم تقييم النموذج الأولي، لضمان ان هذا النموذج قد لبي متطلبات تصميم النظام، وكذلك يساع المطور من ايجاد فهم افضل للمتطلبات الواجب تلبيتها للزبون.

- يحدث التكرار في هذه المرحلة لضبط النموذج الاولي ولضمان تحقيق متطلبات الزبون

نموذج "النمذجة الأولية" Prototype Model

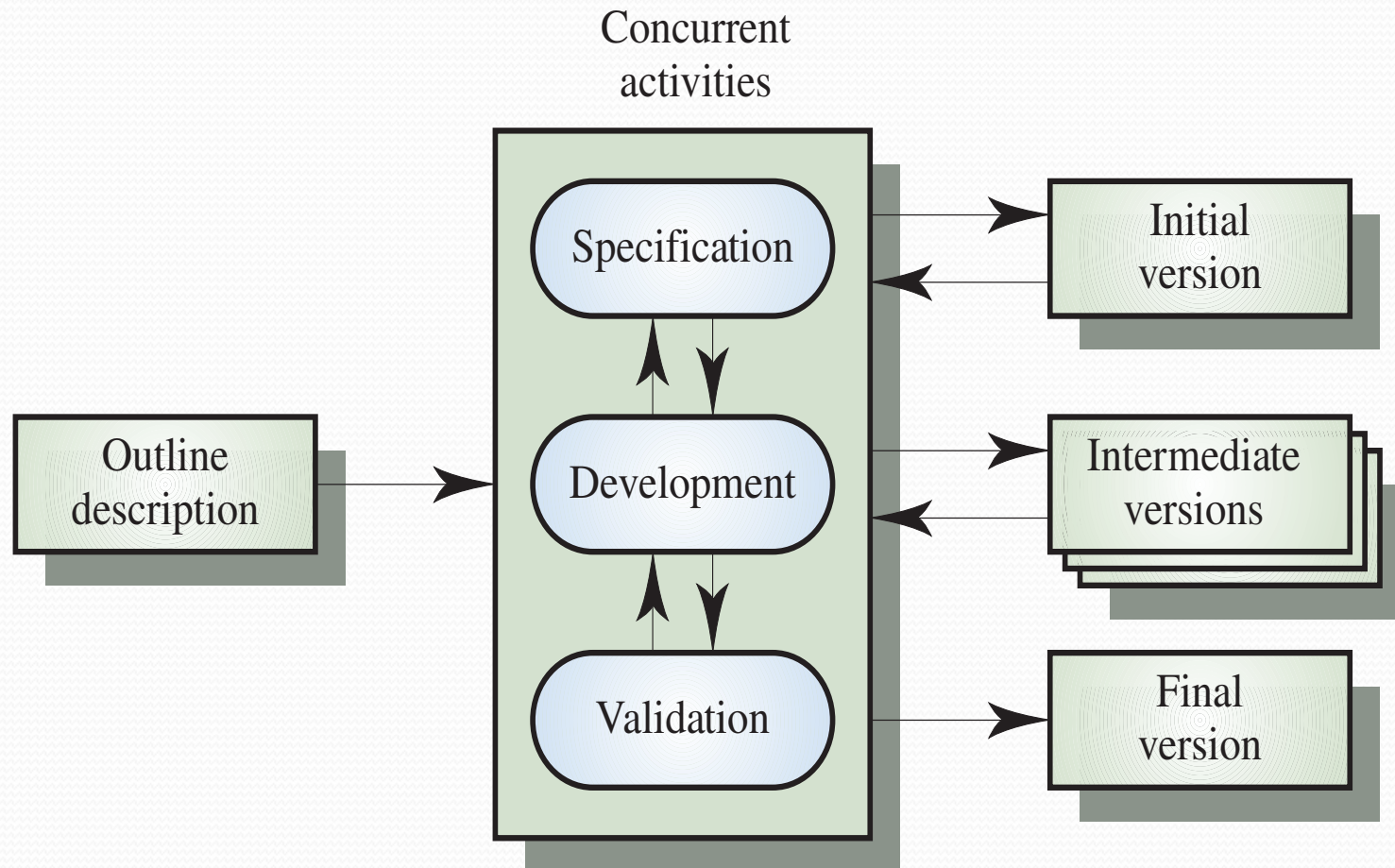
- مميزات النمذجة الأولية:

١. يمكن اعتبارها اقتصادية حيث تقل الكلفة بسبب تلافي الأخطاء والالتباس وسوء فهم الاحتياجات في مرحلة مبكرة.
٢. سهولة الفهم حيث تتجه الى بناء النظام خطوة بخطوة، ولا يتم الانتقال الى خطوة تالية إلا بعد التأكد من الخطوة الأولى.
٣. يستوعب المستخدم النظام جيداً قبل تركيبه وتنفيذه بشكل نهائي.

- عيوب النمذجة الأولية:

١. معظم المشاريع بالكاد يكون اول نظام يبني قابلاً للاستخدام فقد يكون بطئاً جداً او كبيراً جداً او صعب الاستخدام او الثلاثة معاً.
٢. اذا لم يتم التحكم بالمراحل يمكن أن تطول عملية النمذجة.

نموذج "النمذجة الأولية" Prototype Model



النماذج التطورية Evolutionary Models

- هناك اعترافٌ متنام بأن البرمجيات تتطور على مر الزمن. إذ تتغير متطلبات الزبون والمنتج مع تقدم عملية تطوير هذا المنتج، وهذا ما يجعل المسار المستقيم (النموذج الانحداري Waterfall Model) باتجاه إنتاجه غير واقعي. لقد صُمم النموذج التتابعي الخطي لحالات التطوير المباشر (خط مستقيم). وبمعنى آخر، تفترض هذه الطريقة التتابعية أن النظام كله سيسلم بعد اكتمال هذا التتابع الخطي.
- ومن جهة أخرى، فقد صُمم نموذج النمذجة الأولية (Prototype Model) لمساعدة الزبون (أو المطور) على فهم المتطلبات، ولم يصمم عموماً لتسليم نظام نهائي. فلم تُلاحظ الطبيعة التطورية للبرمجيات في كلا هذين النموذجين التقليديين لهندسة البرمجيات.

النماذج التطورية Evolutionary Models

- النماذج التطورية evolutionary models هي نماذج تكرارية، وتوصّف بطريقة تمكّن مهندس البرمجيات من تطوير نسخ أكثر تعقيداً من البرمجيات، وسنذكر فيما يلي اثنين من هذه النماذج:

١. النموذج التزايدى (Incremental Model)

- النموذج المُتزايد: الذي يجمع مكونات من النموذج المتتالي الخطي مطبقة بشكل متكرر، ويقوم النموذج المتزايد بعمل تتابع مخطوط بصيغة متتالية في الترتيب لكن مع تقدم زمن الإنتاج، فينتج كل تتابع خطي برمجيات متزايدة جاهزة للتسليم.
- هذا النموذج كما يقترحه اسمه يعمل بالتزايد، أي أن التطبيق الناتج لا يُصدر مرة واحدة و إنما على دفعات تسمى بالمتزايدة.

النماذج التطورية Evolutionary Models

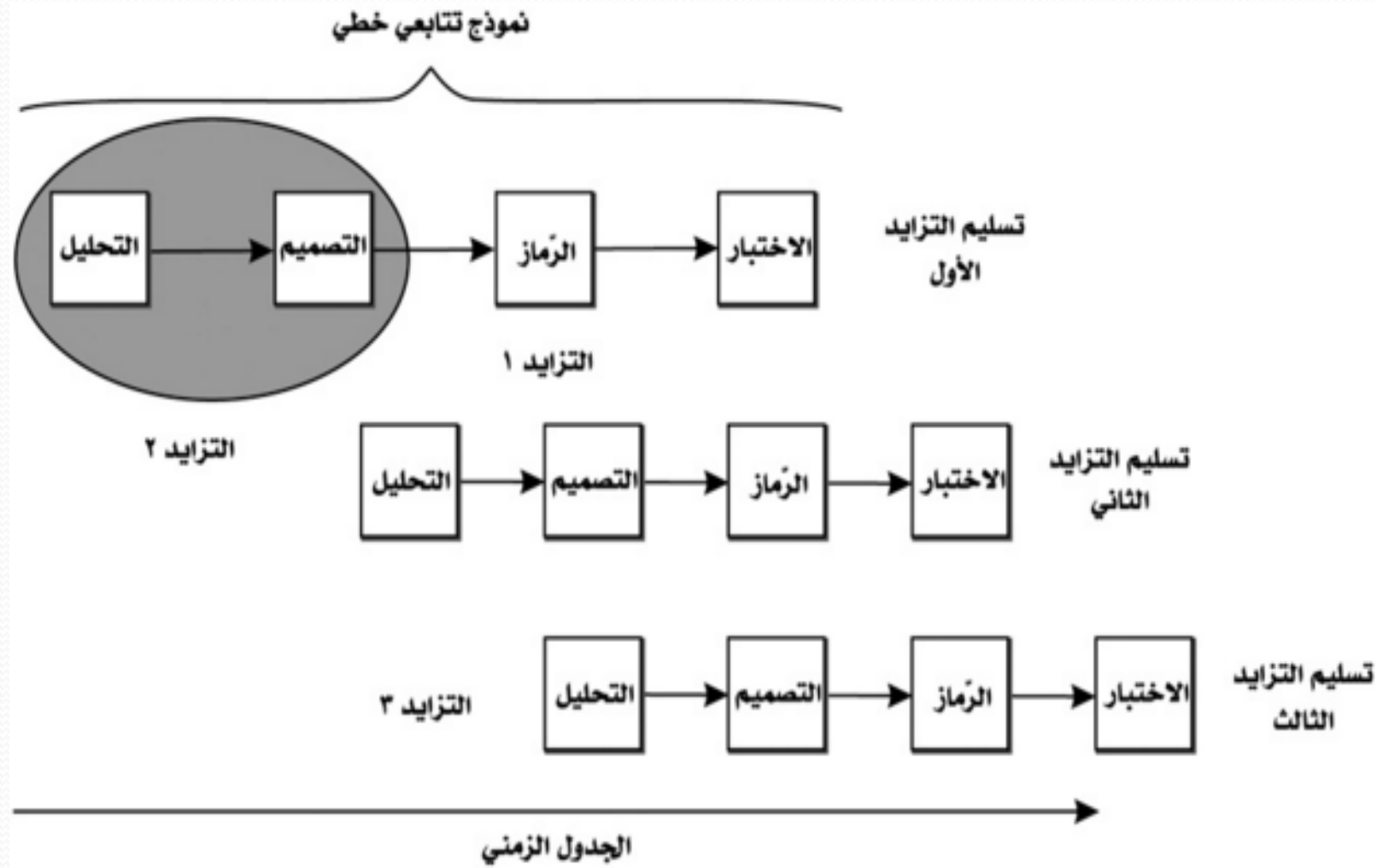
- كمثال، لنأخذ تطبيق على أجهزة المكتب، هذا التطبيق يسمح لك بإضافة عملاء و بياناتهم و بيانات شرائهم كما يسجل كل البضائع في المخزن و يتابع عددها بإنشاء تقارير اسبوعية.
- إذا كان التطبيق سيتم تطويره باستخدام هذا النموذج، فإن أول عملية يقوم بها المطورون هو تقسيم الوظائف إلى مجموعات غير مترابطة، بحيث يتم تطوير كل مجموعة على حدة بشكل متزايدة، يتم البدء بأهم و أعقد الوظائف و كون تحديد وظيفة ما بأنها مهمة راجع لأسباب منها رغبة العميل نفسه ،

النماذج التطورية Evolutionary Models

- ولنفترض أن إضافة العملاء هي أهم خاصية برأي العميل، ذلك يتم تطبيقها و اختبارها و اطلاقها في النظام و السماح للعميل لا لتجربتها فقط بل و البدء بالاستخدام الفعلي ، حالما تطلق الوظيفة يتم البدء حالاً بتطوير المتزايدة اللاحقة و بنفس المنظومة يتم اختبارها و من ثم دمجها مع الوظيفة السابقة واطلاقها للعميل و هكذا، اذاً نرى أن العميل له دور تفاعلي اساسي في هذا النموذج ، كما أن المطورون يحصلون على رأيه بشكل متتالي و سريع مما يسمح لهم بالتعديل بفاعلية اكبر.

- الشكل يبين النموذج التزايدي.

النماذج التطورية Evolutionary Models



النماذج التطورية Evolutionary Models

- من فوائد هذه النموذج انه

١. حقق تطورا فظيعا في تطوير البرمجيات
٢. انشئ مفاهيم جديدة، كما انه يتميز بملائمته للعصر الحالي
٣. الحصول على التطبيقات في وقت سريع حيث ان مدة المشاريع تصل الى شهور حينما تمتد مشاريع نموذج الشلال الى سنين .

- ولكن ان من عيوب هذا النموذج انه يطلق نسخ كثيرة من نفس التطبيق، هذا ربما يسبب الانزعاج لدى المستخدم و الذي لا يكون بالضرورة نفس شخص العميل ،

النماذج التطورية Evolutionary Models

- في مثالنا السابق مثلاً يكون العميل هو صاحب الشركة لكنه لا يستخدم النظام فعلياً و إنما موظفونه، ان الموظفين هم المستخدمون المقيقون لهذا التطبيق و كونهم لا يشترونه منك لا عني ان لا تضعهم في الحسبان ، لأن في النهاية الغرض من التطبيق دائماً هو لتحسين و تطوير فاعلية العمل و حين يكون الموظفون منزجون فإن فاعليتهم ستقل و سينزعج منك العميل في النهاية.
- كن هذا العيب يمكن اصلاحه بسهولة بتقسيم مناسب للوظائف بحيث في كل دفعة يتوجه الاهتمام لقسم معين ، او بالتدريب المناسب لموظفين حال اطلاق نسخة جديدة بحيث لا يشعرون بالضياع ، أو أسوأ: أن يشعروا بالغباء و هو السبب الحقيقي غالباً وراء كل انزعاج من المستخدم .

النماذج التطورية Evolutionary Models

٢. النموذج اللولبي او الحلزوني Spiral Model

- هو نموذج تطوري لعملية البرمجة يقرن الطبيعة التكرارية للنمذجة الأولية بالنواحي النظامية والمحكومة للنموذج التتابعى الخطي
- يقدم إمكانية تطوير سريع لنسخ تزايدية من البرنامج
- يكون الإصدار التزايدى خلال التكرار الأولي نموذجا ورقيا أو نموذجا أوليا
- ينقسم النموذج الحلزوني إلى عدد من نشاطات الهيكل تسمى منطقة المهمة task region هناك عادة ما بين ٣ إلى ٦ مناطق وهي:

النماذج التطورية Evolutionary Models

١. الاتصال بالزبون Contact the Customer (التواصل الفعال بين الزبون والمطور)،
٢. التخطيط Planning (تعريف الموارد والمسارات الزمنية للمشروع)،
٣. تحليل المخاطرة risk analysis (تقييم المخاطرة التقنية والإدارية)،
٤. الهندسة Engineering (بناء تمثيل أو أكثر للتطبيق)،
٥. البناء والإصدار construction & release (مهام البناء والاختبار)،
٦. تقويم الزبون customer evaluation (الحصول على تقييم الزبون للعمل).

النماذج التطورية Evolutionary Models

- حالما تبدأ عملية البرمجة التطورية يتحرك فريق هندسة البرمجيات حول الحلزون باتجاه عقارب الساعة بدءاً من النواة
- خلافاً للنماذج التقليدية التي تنتهي بتسليم البرنامج يمكن تكيف النموذج الحلزوني لتطبيقه على امتداد كامل حياة البرنامج
- يستخدم النموذج الحلزوني النمذجة الأولية كآلية لتقليل المخاطر التقنية ولكنها تمكن المطور من تطبيق النمذجة الأولية في أي مرحلة من مراحل تطور المنتج

العيوب:

النموذج الحلزوني “قلل من المشاكل بشكل كبير” ولكن يمكن أن تظهر نفس المشكلة في التي كانت تظهر في الشلال حيث في حال أكتشف المبرمج أن هناك خطأ أو نقص في التصميم لا يمكن العودة للخلف ويجب الانتظار حتى البدء في الدورة الثانية

النماذج التطورية Evolutionary Models

- الشكل الاتي يبين النموذج اللولبي او الحلزوني

