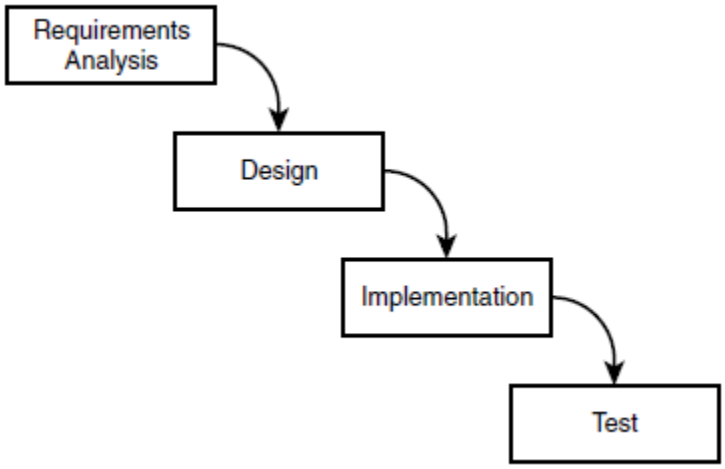
خطوات التحليل والتصميم الكائني

Object-oriented analysis and design **OOAD**

- **الطريقة 1 :**نموذج الشلال(**Waterfall Model**)

1. معرفة وتحليل المشكلة والمتطلبات وتسمى.(**Analysis**)
2. طرح حلول المشكلة وتسمى.(**Design**)
3. تنفيذ التصميم ويسمى.(**Implementation**)
4. إجراء إختبار ويسمى.(**Testing**)



وتعرف هذة الخطوات الأربع(**Analysis, Design, Implementation, Testing**)
باسم دورة حياة المشروع (**Software Development Life Cycle**)

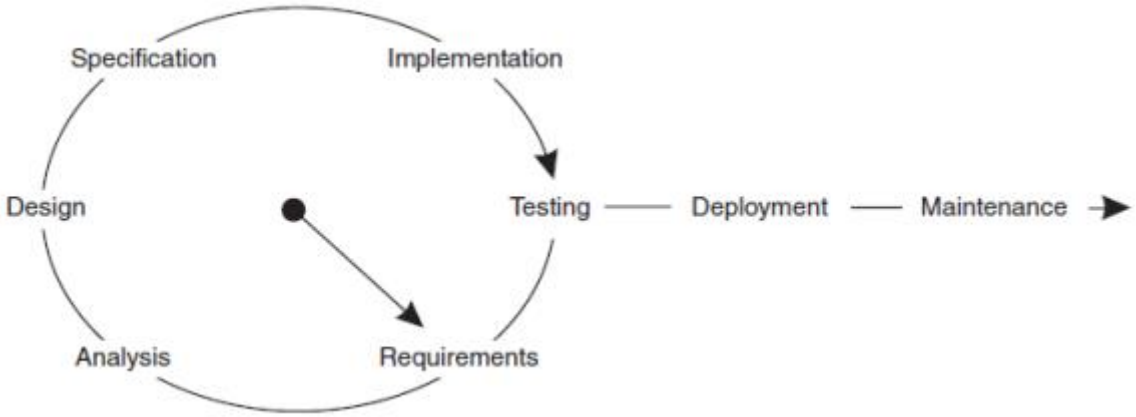
العيوب:
لأن هذة العملية تكون بالترتيب بشكل متنازل فإنها لاتصلح إلا في المشاريع
الصغيرة والتي لا يوجد فيها خطورة والسبب أننا لا نستطيع العوده للخلف بتاتاً
فمثلاً:
عندما يبدأ فريق العمل في مشروع معين يقوم المحلل أولاً بتحليل البرنامج بالكامل
وكتابة جميع الوثائق المطلوبة ومن ثم يمرر هذة الوثائق للمصمم الذي يقوم بتصميم
النظام ووضع حلوله أيضاً ، ومن ثم تمر بالمبرمج الذي يقوم بكتابة الكود وعندما ينتهي
تجرى الأختبارات على هذا البرنامج للتأكد من عدم وجود أخطاء وأخيراً يسلم الى الزبون
السؤال هنا:
ماذا لو تغيرت متطلبات هذا الزبون بعد مرور 3 أشهر من التحليل والتصميم وبالبدء بالكود ، كارثه!!
وحتى لو أن المتطلبات لم تتغير فإن وجود خطأ في مرحلة من المراحل يؤدي إلى دمار كل المشروع!!

الحل:
الحل أنهم قاموا بتطوير نموذج الشلال (**Waterfall Model**) إلى نموذج جديد يسمى:

- **الطريقة 2 :**النموذج الحلزوني(**Spiral Model**)

1. أخذ المتطلبات من الزبون(**Requirements**)
2. تحليل جزء من المشكلة وتسمى.(**Analysis**)
3. طرح حلول المشكلة وتسمى.(**Design**)
4. التأكد من المواصفات(**Specification**)

5. تنفيذ التصميم ويسمى.(**Implementation**)
6. إجراء إختبار ويسمى.(**Testing**)
7. نشر هذا الجزء من البرنامج ويسمى.(**Deployment**)
8. صيانة الأخطاء التي تظهر فيه وتسمى.(**Maintenance**)

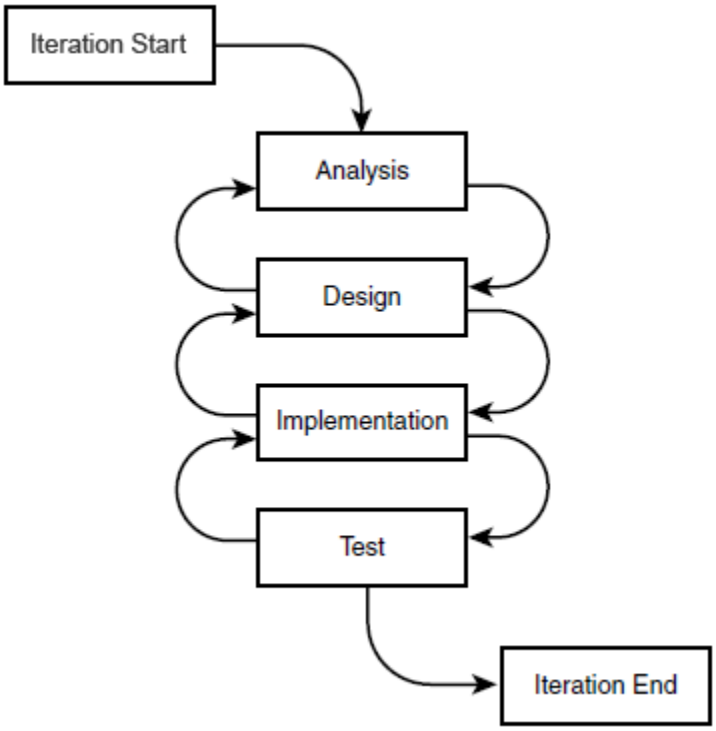


العيوب:
النموذج الحلزوني" **قلل من المشاكل بشكل كبير**"
ولكن يمكن أن تظهر نفس المشكله في التي كانت تظهر في الشلال
حيث في حال أكتشف المبرمج أن هناك خطأ أو نقص في التصميم
لا يمكن العودة للخلف ويجب الانتظار حتى البدء في الدورة الثانية

الحل:
الحل أنهم قاموا بتطوير نموذج الشلال (**Spiral Model**) إلى نموذج جديد يسمى:

- **الطريقة 3 :**النموذج التكراري:(**Iterative Model**)

1. بدء المشروع وتسمى.(**Iteration Start**)
2. تحليل المشكلة وتسمى.(**Analysis**)
3. تصميم حلول المشكلة وتسمى (**Design**) و إمكانية العودة إلى مرحلة(**Analysis**)
4. تنفيذ التصميم ويسمى (**Implementation**) و إمكانية العودة إلى مرحلة(**Design**)
5. إجراء إختبار ويسمى (**Testing**) و إمكانية العودة إلى مرحلة(**Implementation**)
6. نهاية المشروع وتسمى.(**Iteration End**)



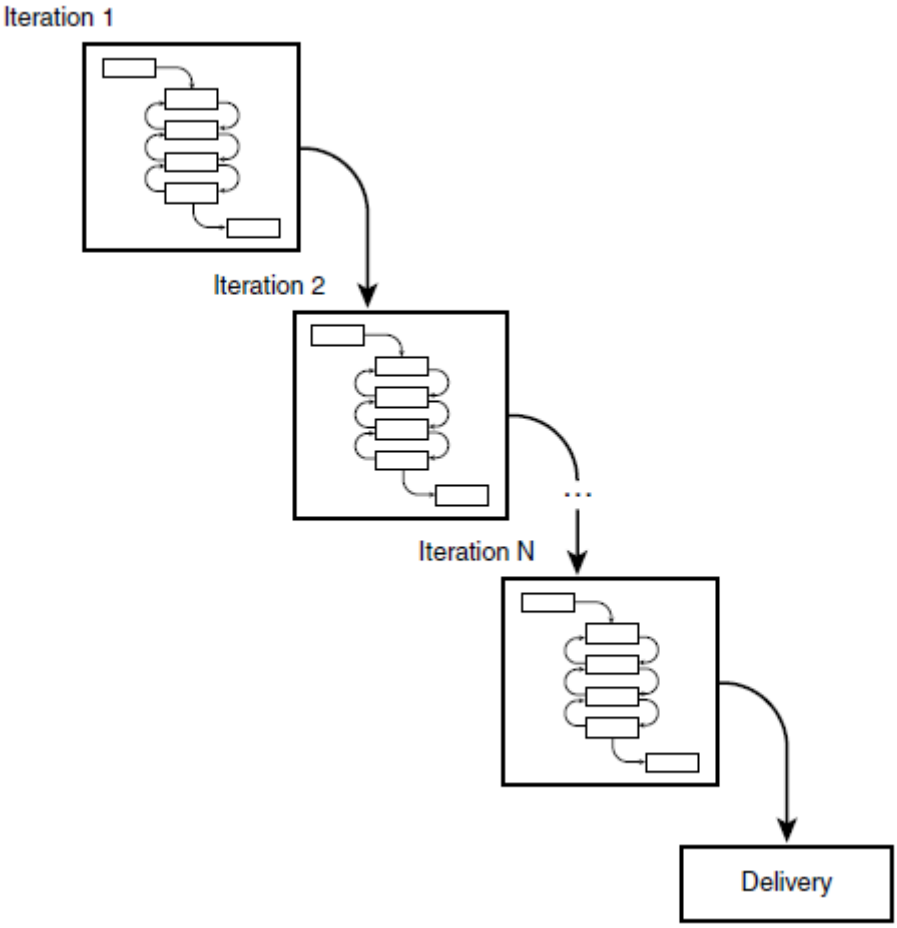
العيوب:

نعم لقد أصبحت عملية التطوير هي عباره عن عدة دورات
كل منها تستطيع التحرك للأمام أو للخلف حسب ما تريده
ولكن ما زالت المشاكل بالظهور حيث الدورات السابقه
كانت لإصلاح خلل وغالباً ما تكون 4 دورات فقط!!!

الحل:

نريد تقسيم المشروع لأكثر من دوره وفي كل مره نقوم بتطوير جزئيه معينه
ولهاذا جمعوا بين جميع هذه الطرق (الشلال ، الحلزوني ، المتكرر) كل منها
توجد فيه خاصيه مفيده في عمليه التطوير ولذا يمكن أن نجمع كل هذه الطرق
مع بعضها البعض وبالتالي نستفيد من جميع المزايا في كل واحدة من الطرق

الطريقة 4 : الجمع بين النماذج الشلال والحلزوني والمتكرر:(**Waterfall, Spiral, Iterative Models**)


Iteration 1
Iteration 2
...
Iteration N
Delivery