



*University of mustansiriyah /College of Education*

*Computer Science Department*

*Software Engineering 3<sup>rd</sup> Class*

*Lecturer maha ali hussain & muntaha abbod*

## **CHAPTER 3**

### ***Software Requirement Engineering***

### **هندسة المتطلبات البرمجية**

Topics:

3.1 Introduction

3.2 Requirement and its Problems

- Software Requirement Engineering Activities
  - Requirement Elicitation
  - Requirement Analysis
  - Requirement Specification
  - Requirement Validation
  - Requirement Management

3.3 Software Requirement Engineering and its Objectives

ويتم في هذا النشاط استخدام مجموعة طرق وادوات لهذا الغرض منها :

### 1 المقابلة الشخصية Interview

تعتبر المقابلة الشخصية احدى اهم طرق جمع المتطلبات الخاصة بالمنظومات المعلوماتية وفي بداية المشروع بقضي محلل النظم وقتا كافيا لعمل مقابلات شخصية مع الزبون والمستخدم من اجل فهم طبيعة العمل والبيانات والمعلومات المتداولة في بيئة الزبون وايضا القوانين واللوائح المستخدمة لتسيير دفة العمل .

### 2- الاستبيان Questionnaire

بالرغم من اهمية المقابلة الشخصية كطريقة من الطرق الرئيسية في الحصول على المتطلبات الا انها مكلفة ويتم اجراؤها على عدد صغير ومحدود بمعنى انه لن تكون الصورة واضحة على كل ما يجري . لهذا فان طريقة الاستبيان Questionnaire هي النقيض حيث تجري على عدد هائل من الناس في زمن قليل . وعادة يتم تجهيز الاستبيان على ورق الا انه يمكن اجراؤه على الهاتف او الانترنت . ويحتوي الاستبيان عادة على مجموعة من الاسئلة القصيرة والطويلة او الاسئلة ذات الاجابة المحددة من بين مجموعة اجابات Multiple Choice . وتجري الاستبيانات وتسمى ايضا المسح Survey للوصول الى هدف معين . وعلى سبيل المثال يمكن استخدامها لمعرفة رأي المستخدمين في عمل النظام ( او المنظومة ) الحالي .

### 3 الملاحظة Observation

تعتبر طريقة مباشرة لتفقد ومعرفة النظام بحسب الدراسة ويساعد على الوصول للمعلومة بشفافية وبموضوعية كما هي وليس كما يعتقدها الزبون والمستخدم / ولكن من عيوبها ان عملها ليس دائم ومستمر لهذا قد يلتجئ المستخدم للنصنع والعمل المثالي لحظة ملاحظة المحلل وبالتالي تكون المعلومات المجمعة لا تعكس الواقع المعيش . لهذا من الافضل اداء هذا العمل (في بعض الحالات) من دون علم الموظف القائم بالعمل وهذا لا يعني انه في حالات معينة يمكن ان يتفاعل محلل النظم مع الشخص المعني وان يوجه الاسئلة او يشاهد ويجمع بعض النماذج المستخدمة .

### 4 جمع وتحليل العينات والوثائق Document Gathering & Sampling

تعتبر طريقة جمع وتحليل النماذج Forms والوثائق Documents هامة ايضا ومجدية لاننا قد نجد طرفا اخرى تساعدنا على معرفة طريقة عمل النظام كما هي على ارض الواقع او كما تم شرحها اثناء المقابلة والملاحظة . لكن السؤال هو هل يعمل المستخدم بالطريقة المثالية والصحيحة المدونة بالفوانين واللوائح في شكل وثائق او لا ؟ وللاجابة عن السؤال اقول : عند جمع عينات من الوثائق الخاصة باللوائح والملفات والنماذج والتقارير والمخططات التنظيمية للمؤسسة وتحليلها يمكن معرفة ومقارنة طريقة عمل النظام بالطريقة المثالية وطريقة عمل النظام بالطريقة الفعلية . وبالطبع سستفيد محلل النظم من هذا عند تحديد المتطلبات للمنظومة الجديد .

### 5 تصميم التطبيق المشترك (JAD) Joint Application Design

بدأت هذه الطريقة في أواخر السبعينات من قبل شركة IBM لغرض جمع كل من الزبون (المدير ورؤساء الاقسام مثلا) والمستخدمين ومحلي النظم والمبرمجين وممرر الجلسات في عملية تحليل النظام . وانتشرت هذه الطريقة JAD بعد ذلك كأداة مهمة تسعمل في اعداد المنظومة وذلك بأشراك الزبون في معظم المراحل . وفي مرحلة التحليل يتم عمل ورش عمل (جلسات) لجمع المتطلبات من قبل الذين لهم علاقة بالنظام (Stakeholders) في مكان واحد وفي وقت واحد ولمدة قد تستغرق اسبوعا . وهي جلسات مكثفة لحل المشاكل او على الاقل معرفة السبب في صعوبة ايجاد الحلول . ان مشاركة المستخدم في اغلب مراحل المشروع اصبح من المفاهيم الحديثة والمرغوبة . وقد تم ايجاد علاقة بين مشاركة المستخدم ورضاه عن المنظومة حين تسلمها لان افضل الافكار الابداعية للمنتجات الجديدة وتحسنها تأتي عادة من الزبون وليس من معد المنظومة . ويجب اخذ الملاحظة بان مكان الاجتماع يجب ان يكون بعيدا من مكان عمل الزبون ويجب اعداد الحجرة التي ستقد فيها الجلسات بجيدا بمعنى وجود السبورة البيضاء وجهاز الحاسوب للمقرر وطابعه لطباعة الوثائق التي قد توزع على الحاضرين وجهاز عرض الشرائح باستخدام الحاسوب . اما بعد المكان فمردده الى الابتعاد عن الازعاج والارباك وزحمه العمل لتتسنى التركيز على العمل الذي كلفوا به . وتتلخص النقاط في هذه الجلسات في ما يلي :

1 لمحة على النظام الحالي والمشاكل التي تعترضه .

ان فكرة التحليل اساسا هي تقييم احتياجات المستخدم للوصول الى تعريف محدد للمتطلبات البرمجية المراد تجهيزها .

ويعني **تحليل المتطلبات (Requirements Analysis)**: عملية تفكيك وتجزئة المتطلبات العامة (العالية المستوى) وتحويلها الى متطلبات وظيفية تفصيلية (متدنية المستوى) حيث يتم استخدام ادوات مناسبة لتمثيلها نمذجتها Modeling .

وكملاحظة في هذا السياق يجب ان اقول انه يجب تصنيف وترتيب هذه المتطلبات ليتم تنفيذها حسب الاهمية . وهذا التصنيف للمتطلبات يمكن ان يأخذ شكل :

- متطلبات ضرورية .
- متطلبات مشروطة .
- متطلبات اختيارية .

- وعند تحليل المتطلبات اي نمذجتها نستخدم الادوات بناء على المنهجية التي يتم اختيارها لعملية التحليل للمنظومة .

وتوجد منهجيتان مشهورتان على نطاق واسع وهما :

اولاً: المنهجية الهيكلية Structured Methodology

ثانياً : المنهجية الشيئية Object-Oriented Methodology

ثالثاً- منهجية أجل Agile Methodology

رابعاً: منهجية اطار عمل الحلول Ms-Solution Framework Methodology (MSF)

اولاً: **المنهجية الهيكلية Structured Methodology**: هذه المنهجية تركز اكثر على وظائف المنظومة (المعالجة) حيث يتم استخدام الادوات المستخدمة في التحليل الهيكلي Structured Analysis لتمثيل ووصف نمذجة هذه الوظائف .

ومن هذه الادوات :

التعريف

1- **مخطط انسياب البيانات (Data Flow Diagram) (DFD)** / وهو مخطط هيكلي رسومي يبين صورة لحركة انسياب البيانات داخل النظام بين مخازن البيانات والمعالجة والكيانات الخارجية / وهناك نوعان من مخطط DFD :

الأول : يستخدم لتمثيل النظام كما هو على ارض الواقع مشتملا على المعلومات والمواد ويسمى المخطط الانسيابي المادي Physical DFD . وهو وسيلة تفاهم بين المستخدم ومحلل النظم حتى يفهم محلل النظم الدورة المستندية وتحرك المواد عبر النظام ليتسنى له فهم عمل النظام وبالتالي تزال عملية الغموض لديه .

الثاني : يستخدم ليكون ساسا لمرحلة التصميم ويسمى المخطط الانسيابي المعنوي Logical DFD ويشتمل على انسياب البيانات فقط محذوفا منه انتقال المواد .

وعملية رسم مخطط انسياب البيانات تبدأ برسم مخطط عام وعالي المستوى (وبدون تفاصيل) يسمى المخطط البيئي Context Diagram يبين تفاعل المنظومة مع الكينونات الخارجية (البيئة الخارجية للمنظومة) .

ثم يقوم محلل النظم بتفصيل المخطط البيئي ليشمل على مخططات تفصيلية تسمى DFD level 1 و DFD level 2 وهكذا تتم التجزئة تباعا . ومن مزايا هذه التجزئة الهيكلية ازدياد سهولة فهم النظام او المنظومة من خلال المناقشة حولها بين جميع الاطراف ذات العلاقة .

ويعتبر مخطط انسياب البيانات بسيط التكوين الا انه اداة قوية لتمثيل (نمذجة) وظائف النظام (او المنظومة) .

ويعتقد العديد من محلي النظم ان هذا المخطط هو كل ما يحتاجونه لمعرفة التحليل الهيكلي والحق انه بدون استخدام ادوات اخرى مساعدة لعملية التحليل تصبح هذه الاداة غير ذات جدوى في حد ذاتها .

لهذا يجب جمع هذه الاداة مع ادوات اخرى مثل قاموس البيانات Data Dictionary والانجليزية الهيكلية Structured English ومخطط الكائنات العلائقية Entity Relationship Diagram وشجرة القرار Decision Tree وغيرها .

## 2- قاموس البيانات Data Dictionary

يمكن تعريف اداة قاموس البيانات Data Dictionary بأنه قائمة او مستودع لكل عناصر البيانات (data objects) الخاصة بالمنظومة او وصف لمخازن البيانات وانسيابها والموجودة في مخطط DFD .

ونظرا لأهمية هذه الاداة لتحديد متطلبات البيانات Data Requirements الخاصة بالمستخدم فأنها يجب ان تكون دقيقة وواضحة لكي تساعد على الفهم المشترك بين كل من المستخدم ومحلل النظم والمصمم والمبرمج وابعاد اي لبس حول مدخلات ومخرجات المنظومة .  
وتعتبر هذه الاداة من الاساسات التي سيستفاد منها في تصميم قاعدة البيانات في مرحلة التصميم .

## 3- الانجليزية الهيكلية Structure English

تعتبر الانجليزية الهيكلية Structure English اداة تحليل نصية تستخدم جزء محدود من اللغة الانجليزية لتوضيح الخطوات المراد اداؤها لوظائف (عمليات) منظومة معلومات .

( مبيسة في قاموس البيانات )

وتستخدم افعال مثل : read , write , print , sort , add , subtract.....etc وعناصر بيانات مثل Customer-no , item-no , price وتركيبات البرمجة الهيكلية مثل While Do أو If-then else وعادة ما تستخدم الانجليزية الهيكلية بعد رسم مخطط انسياب البيانات لتوضيح كل معالجة Processes في DFD .  
وتعتبر الانجليزية الهيكلية اداة تواصل مع المستخدم لأزالة الغموض حول كل عملية . وهي اساس كتابة شبه الشفرة Pseudo code في مرحلة التصميم .

## 4- جدول القرار Decision Table

يعتبر جدول القرار Decision Table اداة تحليل في شكل جدول (مصفوفة) يبين الافعال Actions المحتملة بناء على شروط Conditions معينة .  
- ويساعد هذه الأداة (Decision Table) في توضيح القرارات التي تستخدم في الحالات المعقدة .

ويستخدم محلل النظم هذه الاداة لبيبين سياسة عمل النظام مثل :

- سياسة التخفيضات في نظم المبيعات .
- سياسة تنسيب طالب الثانوية للجامعات والمعاهد .
- سياسة تقييم المستوى الأكاديمي للطالب بناء على أدائه في الامتحانات والواجبات المدرسية .

## 5- شجرة القرار Decision Tree

شجرة القرار Decision Tree اداة تحليل على شكل شجرة تبين الحالات (الشروط) Conditions والافعال Actions ذات العلاقة بهذه الشروط وهي تبين سياسة عمل النظام . وهي تشبه الى حد بعيد عمل جدول القرار الا ان فروعها يجب ان تكون محدودة وهي بدئل لجدول القرار في النظم الغير معقدة .

## 6- مخطط الكائنات العلائقية ERD Entity Relationship Diagram

يعتبر مخطط الكائنات العلائقية ERD (Entity Relationship Diagram) بدايه صحيحة من قبل محلل النظم لفهم متطلبات البيانات Data Requirements الخاصة بالمنظومة تحت الاعداد . ويتكون مخطط الكائنات العلائقية من اشكال هندسية تشبه المخطط الانسيابي تبين الكائنات Entities والعلاقه Relationships بين هذه

الكائنات وايضا الخصائص Attributes (عناصر البيانات) لكل كائن أو علاقة . ويعتبر هذا المخطط أيضا أساسا قاعدة البيانات . وهو داه بواصل بين محلل النظم والمستخدم لفهم وتكوين عناصر البيانات المستخدمة وانتمائها للكائنات . وفي الوقت الحاضر وباستخدام العرض التجريبي Prototyping أصبح الحصول على عناصر البيانات عملا ميسرا .

### ثانياً : التحليل الشبيني Object-Oriented Methodology

يمثل التحليل الشبيني Object Oriented Analysis تغيرا دراميا مقارنة بالتحليل الهيكلي حيث يتم التعامل مع النظام على أساس أنه مجموعة من الكائنات (المادية والمعنوية) .

أما التحليل الهيكلي فيعتبر البيانات منفصلة عن العمليات التي تحصل على هذه البيانات . بمعنى أن البيانات ليس لها أهمية بالغة في التحليل الهيكلي Structured Analysis . حيث يتم تقسيم المنظومة الى وظائف رئيسية وتجزأ هذه الوظائف الى وظائف فرعية وهكذا . أما عرض التحليل الشبيني OOA فهو ربط البيانات والعمليات في مكان واحد وهو الكائن Object أو الفصيلة Class علما بأن الكائن حالة خاصة من الفصيلة . وفي هذا النوع من التحليل يمكن تعريف الفصائل والعلاقات بين هذه الفصائل والنظام . ويجب اتباع الخطوات التالية للحصول على هذا التحليل :

1 يبدأ محلل النظم في الحصول على بعض المتطلبات من الزبون والتي من أهمها المتطلبات الوظيفية ويستخدم مخطط استخدام الحالة Use Case Diagram (UCD) كأداة هامة لتحديد هذه المتطلبات . ويستخدم السيناريو Scenario النصي لوصف كل حالة استخدام .

2 التعرف على الفصائل Classes الخاصة بالنظام ( أيضا الخصائص Attributes والطرق Methods لكل فصيلة) .

- 3 رسم مخطط الفصائل Class Diagram ويتم ذلك اما يدويا او باستخدام احدى ادوات Case .
- 4 رسم مخطط السلسلة Sequence Diagram ليعبر عن وصف تفصيلي لكل حالة استخدام . وفي هذا البند نقوم بشرح ثلاث مخططات شبينة تستخدم في مرحلة التحليل :

#### 1 مخطط حالة الاستخدام Use Case Diagram UCD :

ويعتبر مخطط حالة الاستخدام UCD أداة تحليل شبينية مهمة لتوضيح المتطلبات الوظيفية للنظام . ويتكون من اشكال هندسية تعبر عن حالة الاستخدام Use Case وهي المعاملة أو الوظيفة التي يؤديها النظام والممثل أو الفاعل Actor وهو الذي يقوم بإداء هذه المعاملة (حاله الاستخدام) .

#### 2 مخطط الفصيلة Class Diagram

مخطط الفصيلة Class Diagram هو أداة تحليل شبيني رسومي يبين هيكلية الكائنات الساكنة للنظام . وهذه الهيكلية تبين فصائل الكائنات والعلاقة بين هذه الكائنات . ويبين مخطط الفصيلة الحركة الساكنة للمنظومة الشبينية .

#### 3 مخطط السلسلة Sequence Diagram

يعتبر مخطط السلسلة Sequence Diagram أداة تحليل شبينية تبين الكائنات والتواصل بين هذه الكائنات باستخدام الرسائل المتبادله بينهم عند تنفيذ حاله الاستخدام . ويبين مخطط السلسلة الحركة المتحركة (الديناميكية) للمنظومة الشبينية .

### ثالثاً- منهجية أجل Agile Methodology :

تعتمد منهجية Agile على اختيار افضل الادوات والطرق المناسبة لأداء المهام الخاصه بالمنظومة . ويعتمد على استعمال عدة منهجيات في المشروع الواحد مثل استخدام الاداه DFD مع الاداة UCD في نفس المشروع اي المنهجية الهيكلية والمنهجية الشبينية .

وبعبارة اخرى يمكن استخدام منهجيات وطرق مختلفة والهدف هو تسريع عملية اعداد المنظومة وهذه المنهجية تفاضل بين الانتاجية والجودة .

رابعاً: منهجية اطار عمل الحلول (MSF) Microsoft Solution Framework Methodology (MSF) في هذه المنهجية MSF التي تستخدمها شركة Microsoft في اعداد منظوماتها يستطيع محلل النظم ان يصمم عدة نماذج Models منها على سبيل المثال لا الحصر :

- نموذج المعالجة Process Model
- نموذج البيانات Data Model
- نموذج ادارة المخاطرة Risk Management Model

وكل نموذج يساهم في تحليل وتصميم المنظومة تحت الاعداد .  
وتستخدم منهجية MSF ادوات OOA مثل UCD وادوات CASE .  
وتعيد هنا القول أنه : للحصول على منظومة ذات جودة يجب ادماج والحاق المستخدم جنباً الى جنب مع معد المنظومة في هذه المرحلة المهمة الا وهي التحليل .

تعريف

### ثالثاً: المواصفات Specifications

في هذا النشاط يتم كتابة وتجهيز وثيقة هامة من وثائق المنظومة في نهاية مرحلة التحليل وتسمى وثيقة مواصفات المتطلبات Requirement Specification Document والتي تشتمل على كل متطلبات المنظومة المقترحة .  
وتلعب وثيقة المتطلبات دوراً مهماً في دورة حياة المنظومة لأنها تقودنا الى مراحل التصميم والتنفيذ وتعتبر اساساً للتعاقد بين الزبون ومعد المنظومة .  
ولقد ثبت ان حوالي 85% من اخطاء البرمجيات كان مرده الى المتطلبات ومشاكلها والتي هي :

- 49% افتراضات تتعلق بمتطلبات غير صحيحة .
- 29% متطلبات محذوفة (غير معلنة).
- 13% متطلبات متضاربة .
- 5% متطلبات غامضة وغير واضحة .

وقد ثبت من الاحصائيات بسبب مشكلة تحديد المتطلبات ايضا ان حوالي 30% من المشاريع يتم الغاؤها قبل ان تنتهي وان حوالي 50% من المشاريع تكلف الضعف من التقديرات الاولى .

### خصائص مواصفات المتطلبات البرمجية :

تناول العالم Boehm (1984) خصائص مواصفات المتطلبات البرمجية الجيدة فيما يلي :

- |             |                         |
|-------------|-------------------------|
| Complete    | ● كاملة                 |
| Measurable  | ● دقيقة وقابلة للقياس   |
| Correct     | ● صحيحة                 |
| Unambiguous | ● واضحة                 |
| Testable    | ● قابلة للاختبار        |
| Consistent  | ● متناغمة (غير متضاربة) |
| Concise     | ● موجزة ومحددة          |
| Verifiable  | ● قابلة للتحقق          |
| Changeable  | ● سهولة التعديل         |
| Design-free | ● بدون علاقة بالتصميم   |

وقبل ان نشرع في هذه الوثيقة نشرح معنى واهداف التوثيق :

### التوثيق Documentation

يعتبر التوثيق عنصرا مهما في اعداد البرمجيات واستمرار عملها بعد الاعداد . ويمكن تعريف التوثيق بأنه مجموعة اوصاف نصية ورسومية وشروح للمنتج البرمجي (المنظومة البرمجية) . وقد يشمل التوثيق ما يلي :

Narratives	● سرد أو نص
Charts	● مخططات
Tables	● جداول
Voice	● الصوت
Video clips	● قصاصات فيديو
Animations	● صور متحركة
Comments in program	● تعليقات في البرنامج

ويمكن ان تكون الوثيقة على شكل ورقة او تكون مخزنة في الحاسوب .

### اهداف ووظائف التوثيق :

يؤدي التوثيق الوظائف التالية :

- 1- مرجع تاريخي .
- 2- مرجع ارشادي وتوضيحي .
- 3- متابعة جودة المنتج البرمجي .
- 4- التواصل بين مراحل اعداد المنظومة .
- 5- التواصل بين المهام داخل المرحلة الواحدة .
- 6- اتفاق بين المستخدم او الزبون ومعد المنظومة .

### استخدام التوثيق :

يستخدم التوثيق المعد بصورة عامة من قبل :

- الادارة لغرض المراجعة .
- القائمين على صيانة البرنامج .
- فريق التفتيش .
- فريق المراجعة غير الرسمية من قبل زملاء العمل .
- موظفي التحقق والمصادقة .

نستعرض الان محتويات وثيقة مواصفات المتطلبات التي يجب ان يدها محلل النظم في نهاية مرحلة التحليل ولتتم مراجعتها من قبل الادارة لاتخاذ احد القرارات الاتية :

- الاستمرار في المشروع وتنفيذ المرحلة التالية وهي التصميم .
- وقف استمرار المشروع .
- اجراء بعض التعديلات ثم الاستمرار في المشروع .

وثيقة تحديد المتطلبات **Requirement Specification Document** :  
 تعريف وثيقة تحديد المتطلبات : هي وثيقة يتم اعدادها في نهاية مرحلة التحليل تتضمن وظائف  
 المنظومة المراد تنفيذها وخصائص الجودة المتعلقة بها : وهذه الوثيقة يجب ان تكون صحيحة ودقيقة  
 وكاملة ومتناسقة وقابلة للقياس والاختبار .  
بنود وثيقة مواصفات المتطلبات :

#### (المقدمة) Introduction (a)

##### 1- Overall description وصف عام :

تعريف المسألة	problem definition •
الاهداف	objectives of the system •
البيئات (العوامل ك)	interfaces of the system •
حدود المنظومة	scope of the system •
القيود	constraints of the system •

##### 2- الوصف الوظيفي Functional description :

قائمة الوظائف	list of system functions •
وصف كل وظيفة	Narrative for each function •

##### Data/ Information description (b) وصف البيانات :

مخطط ERD	Entity relationship diagram •
قاموس البيانات	Data dictionary •

##### Process/ logic description (c) وصف المعالجة والمنطق :

مخطط انسياب البيانات	(DFD) Data Flow Diagram •
مخطط استخدام الحالة	(UCD) Use Case Diagram •
الانجليزية الهيكلية	Structured English •
شجرة القرار	Decision tree •
جدول القرار	Decision table •

##### Performance Requirements (d) متطلبات الاداء:

زمن الاستجابة	Response Time •
الذاكرة	Memory •

##### Validation / Acceptance Criteria (e) معيار التحقق والقبول:

انواع الاختبارات	Types of test •
خصائص الجودة المطلوبة	Quality attributes required •
البنود القابلة للتسليم	Deliverables •



- on-line/off-line
- رسومات أم نص Graphic / text
- قاعدة بيانات أم ملفات database / files

ملاحظات عن وثيقة مواصفات المتطلبات :

- 1- تصف مشاكل وليس حلولاً .
- 2- هي منتج وليس عملية معالجة .
- 3- وثيقة بين الزبون والمحلل وتستخدم فيما بعد في التصميم .
- 4- تقوم بتحويل الاحتياجات الى متطلبات .
- 5- يجب مراجعتها من قبل المستخدم ومعد المنظومة .
- 6- تبين ما هو المتوقع من المنظومة وليس كيفية العمل .

**خامساً: اعتماد المتطلبات Requirement Validation**

**المصادقة Validation :** يعتبر هذا النشاط مهما للغاية بهدف في النهاية الى التأكد Confirmation من ان مواصفات المتطلبات التي تم تجهيزها في البند السابق تتوافق مع المعايير Standards في كتابة وثيقة المتطلبات وجاهزة لان تكون أساساً لعملية التصميم في المرحلة اللاحقة لمرحلة التحليل .

ويستخدم في هذا التحقق والاختبار عدة طرق للفحص والمراجعة والتأكد والتي منها :

- الفحص **Inspection Formal** من قبل متمرسين في الكشف عن الاخطاء لهم سابق خبرة في اخطاء سابقة لمنظومات مشابهة وقديمة .
  - المراجعة السريعة **Walkthrough** عن طريق زميل في فريق اعداد المنظومة .
  - التحقق **Verification** من قبل جهة مستقلة مثل مكتب استشاري أو محلل متمرس من ان اهداف واحتياجات الزبون قد تمت ترجمتها في شكل متطلبات .
  - المراجعة النهائية **Review** في نهاية مرحلة التحليل بحضور الزبون ورئيس واعضاء فريق المنظومة لاتخاذ القرار النهائي بخصوص وثيقة المتطلبات .
- ونلاحظ هنا اننا نقوم باختبار المتطلبات لايجاد الاخطاء ولكن ليس على جهاز الحاسوب بل على الوثائق علماً بأن التعرف على الاخطاء في بداية مراحل المشروع يقلل التكلفة .

**سادساً : ادارة المتطلبات Requirement Management**

ان ادارة المتطلبات : هي دراسة واستخدام الاجراءات والسياسات والعمليات التي تحكم كيفية التعامل مع التغيير في المتطلبات وبمعنى أدق :

التعريف

- 1- كيفية تقديم مستند طلب تغيير Change Request .
- 2- كيفية تحليل هذا الطلب ومعرفة تأثيره على التكاليف والجدول الزمني وحدود المشروع .
- 3- كيفية المصادقة والموافقة على اجراء التغيير
- 4- كيفية تنفيذ التغيير بعد اخذ الموافقة عليه .

ويهتم هذا النشاط في هندسة المتطلبات ايضا بالتخطيط Planning والمتابعة Controlling لنشاطات جميع المتطلبات والتحليل والمواصفات والتحقق .  
ومن المهام الادارية الخاصة بأدارة المتطلبات Requirement Management ما يلي :

- ادارة النسخ الخاصة بالمنظومة والتغيير Managing versions and change
- تخزين خصائص المتطلبات Storing requirement Attributes
- التواصل مع الذين لهم علاقة بالمنظومة Stakeholders

وتوجد برمجيات جاهزة لادارة المتطلبات Automated Requirement Management من قبل شركات متخصصة . ومن ابرز هذه البرمجيات :

- Doors
- Requisite Pro
- RTM Workshop
- Caliber-RM

ونظرا لاهمية المتطلبات والتعامل معها فقد أنشأت بعض الشركات ادارة تعهد اليها بمتابعة التغييرات التي تحدث في المتطلبات ومتابعة اصدار النسخ والاصدارات لهذه البرمجيات والتي تسمى ادارة مكونات البرمجيات .