

**ITSE311**

# **Software Requirements Analysis**

**تحليل متطلبات البرمجيات**

**مهارة فريق العمل 4: فهم احتياجات المستخدمين**

## **Lecture 14**

**التحقق من المتطلبات**

**أ.د. الحرميين محمد الحرميين**

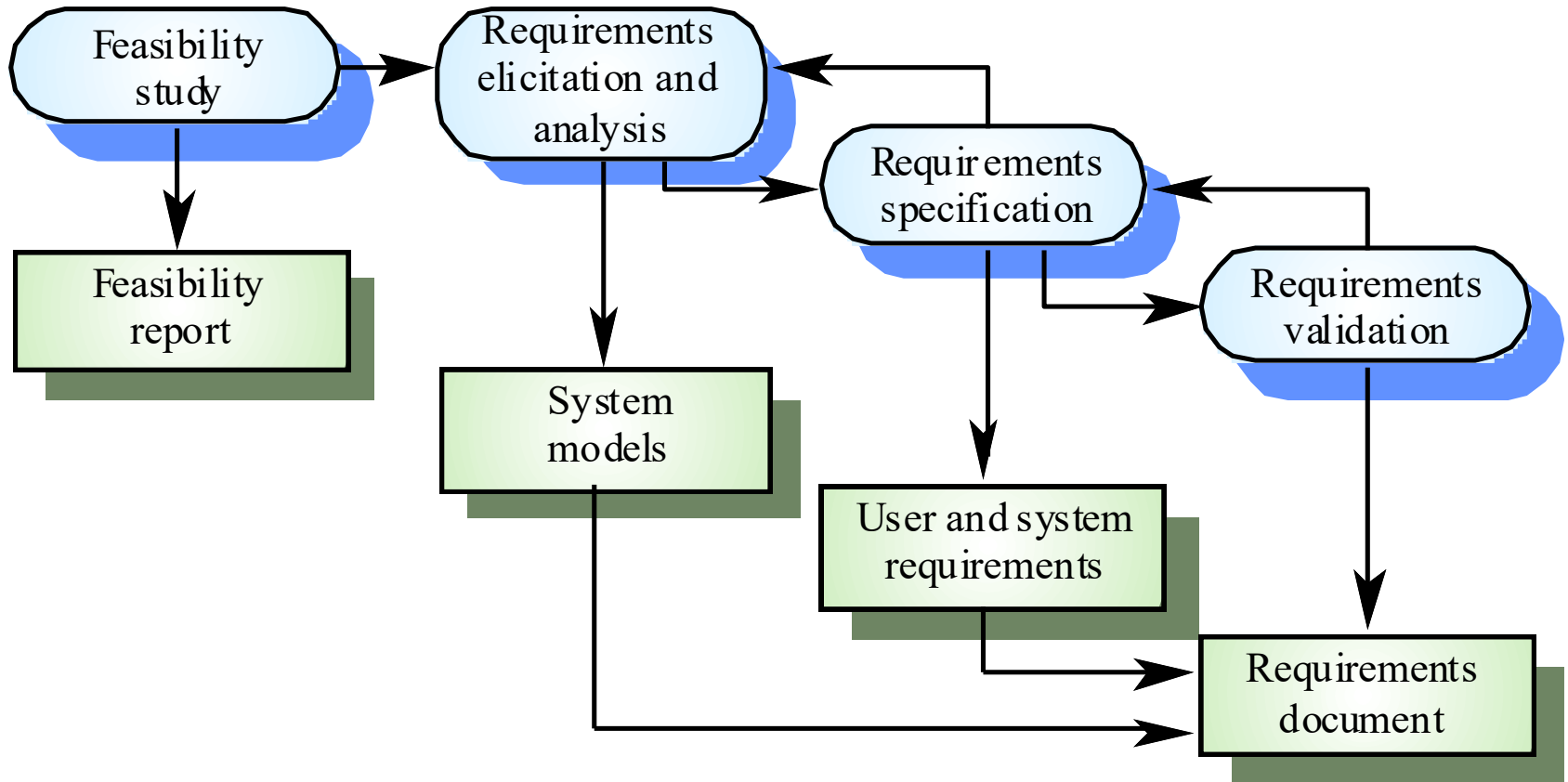
# The Road map

# خارطة الطريق

المواضيع	مهارات الفريق Team Skills	م
L03: the five steps in problem analysis L04: Business Modeling	Analysing the problem تحليل المشكلة	1
L05: The Requirements Elicitation L06: Traditional Elicitation Techniques L07: Modern Elicitation Techniques L08: Applying use-cases	Understanding the user needs فهم احتياجات المستخدمين	2
L09: The vision document	Defining the System تعريف النظام	3
L10: Software Requirements L11: Refining the Use-Cases L12: Modern SRS	Refining the system definition تحسين تعريف النظام الجديد	4
L13: From Requirements to implementation <b>L14: Requirements Validation</b>	Building the right system بناء النظام المطلوب	5

# The requirements engineering process

---



# المتطلبات

---

- المتطلبات تبين ما يجب أن يقوم به النظام وتعرف القيود على تشغيل هذا النظام وتنفيذه.
- المتطلبات الوظيفية تبين الخدمات التي يوفرها النظام.
- المتطلبات غير الوظيفية تبين القيود التي يتم خلالها تطوير النظام.

# دورة حياة تطوير البرمجيات

## Software Development Life Cycle

---

- هندسة البرمجيات عمل إبداعي يتم خطوة بخطوة بالتعاون فرق من المختصين لكل منهم مهمة محددة.
- دورة حياة أى منتج برمجى وتتضمن الأنشطة التالية:
  - تحديد وتعريف المتطلبات
  - تصميم النظام
  - كتابة البرامج
  - إختبار وحدات البرامج وإختبار النظام
  - تسليم النظام
  - صيانة النظام

# خطوات كتابة المتطلبات

1. الإجماعات مع ذوي العلاقة للتعرف على المتطلبات.
2. تسجيل المتطلبات في وثائق ورقة او الكترونية، وعرضها على ذوي العلاقة للمصادقة عليها باعتبار أنها ما يطلبونه بالفعل.
3. إعادة توثيق المتطلبات بطرق منظمة ليتمكن المصمم من تحويلها إلى تصميم جيد للنظام.
4. التحقق من المتطلبات.

# أنواع المتطلبات

- **متطلبات الاعمال:** وثيقة توضح الغرض من النظام الجديد والفوائد التي ستعود على صاحب العمل.
- **متطلبات المستخدم:** مجموعة جمل مكتوبة بلغة طبيعية (قصص المستخدم مثلا) مع أشكال توضيحية للخدمات التي يريدها المستخدم ويجب ان يوفرها النظام وقيود التشغيل.
- **متطلبات البرمجيات:** وصف تفصيلي للبرمجيات يعمل كأساس لتصميم وتنفيذ المنتج، مكتوب لمطوري النظام.

# المتطلبات الوظيفية وغير الوظيفية

## Functional and non functional Requirements

### • المتطلبات الوظيفية: وتوضح

- الخدمات التي يوفرها النظام.
- كيف يتعامل النظام مع المدخلات.
- كيفية تصرف النظام في حالة وجود اخطاء في المدخلات.

### • المتطلبات الغير وظيفية:

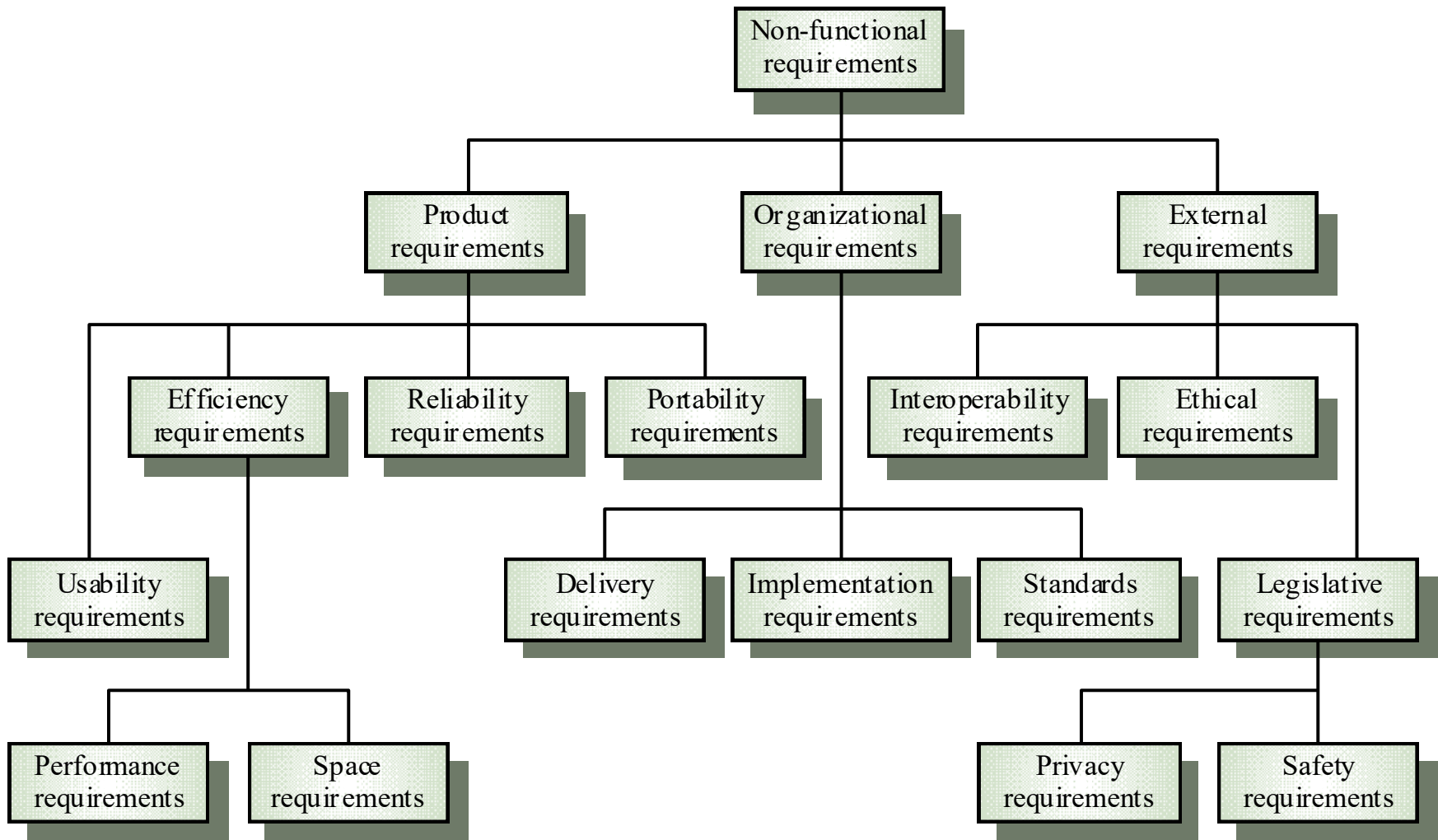
- تعريف خصائص النظام والقيود المفروضة عليه، مثل الإعتيادية وزمن الإستجابة ومتطلبات التخزين .....



# التصنيفات الغير وظيفية

- **متطلبات المنتج:** وهي متطلبات مواصفات عمل المنتج مثل سرعة التنفيذ والإعتمادية وغيرها.
- **المتطلبات التنظيمية:** وهي متطلبات تعبر عن نتائج سياسات المنظمة والإجراءات مثل المعايير المستخدم، ومتطلبات التنفيذ وغيرها.
- **المتطلبات الخارجية:** وهي متطلبات تنشأ من عوامل خارج النظام وعملية تطويره مثل متطلبات السلطة التشريعية وغيرها.

# Non-functional requirement types



مواصفات مبنية على شكل ثابت يعرف تفاصيل الوظيفة:

- تعريف الوظيفة، وصف المدخلات ومن أين تأتي، وصف المخرجات وإلى أين تذهب، الإشارة إلى أى مكونات أخرى مطلوبة، الشروط، والتأثيرات الجانبية إذا كانت موجودة.

# مثال اللغة المهيكله للمواصفات

الدالة: التحقق من المستخدم

الوصف: قبل استخدام النظام يجب التحقق من كلمة المرور واسم المستخدم.

المدخلات: اسم المستخدم وكلمة المرور

المصدر: المستخدم

المخرجات: مدخلات صحيحة او غير صحيحة

الشاشة التالية: الشاشة الرئيسية مع نتيجة التحقق

حالة النظام السابقة: عرض شاشة التحقق من المستخدم

حالة النظام اللاحقة: عرض الشاشة الرئيسية اذا تم التحقق او شاشة الخطا

اذا لم يتم التحقق.

- وثيقة المتطلبات هي مستند رسمي يوضح ما هو مطلوب من مطوري النظام، ويجب أن تتضمن تعريف المتطلبات ومواصفاتها.
- ليست مستند تصميم، ولكنها من اهم مدخلات مرحلة التصميم لأنها تحدد بدقة كل ما يجب أن يفعله النظام.

# هيكل وثيقة المتطلبات

---

- وثيقة المتطلبات تتضمن:

- مقدمة، مسرد المفردات، تعريف متطلبات المستخدم، بنية النظام، مواصفات متطلبات البرمجيات، نماذج النظام، ملاحق، وفهرس.

# التحقق من المتطلبات Requirements validation

---

- تهتم عملية التحقق من المتطلبات بإظهار وإثبات أن المتطلبات تقوم بتعريف النظام الذي يريده صاحب العمل (الزبون).
- اكتشاف أخطاء المتطلبات في مرحلة التحليل يقلل التكلفة ويساهم في انجاح المشروع.
- تصحيح أخطاء المتطلبات بعد تسليم النظام قد يكلف اضعاف تكلفة تصحيح أخطاء التنفيذ.

# التحقق من المتطلبات

---

- هذه المرحلة تمكننا من التحقق من أن المتطلبات:
  - متكاملة complete
  - متماسكة consistent
  - تتبع المواصفات المعيارية conforms to standards
  - واقعية Achievable
  - يمكن اختبارها Can be tested
  - خالية من الغموض free of ambiguity



# فحص وإختبار المتطلبات Requirements checking

**التحقق:** هل توفر المتطلبات وصف شامل لوظائف النظام بحيث تلبى احتياجات الزبون. يجب ان تكون المتطلبات:

1. **متكاملة:** هل تم تضمين كل الوظائف المطلوبة.

2. **متناسكة ومتينة:** هل تتعارض المتطلبات مع بعضها.

3. **تتبع المواصفات المعيارية :** مثلا تستخدم معيار IEEE 830

4. **واقعية:** هل يمكن تنفيذ كل المتطلبات فى حدود الميزانية المتاحة والتقنية المتوفرة.

5. **قابلة للاختبار:** هل يمكن فحص وإختبار المتطلبات.

6. **خالية من الغموض:** كل متطلب واضح وله تفسير واحد.

# مراجعة المتطلبات Requirements reviews

---

- مراجعات عامة تنفذ خلال مراحل إستنباط وصياغة تعريف المتطلبات.
- يقوم بهذه المراجعات كل من الزبون وفريق المحللين.
- يمكن أن تكون المراجعات مكتوبة في مستندات خاصة.
- التواصل الجيد مع كل ذوي العلاقة يساعد على حل المشاكل في مراحل مبكرة من العمل ويجنب الجميع التكلفة الزائدة والتأخير الكبير.

# دعم أدوات هندسة البرمجيات CASE tool support

---

- توثيق المتطلبات: إدارة المتطلبات وتخزينها بشكل منظم.
- إدارة التغيير: سهولة استرجاع المتطلبات وتغييرها مع الحفاظ على تماسك الوثيقة.
- إدارة عملية التتبع: الإستجلاب التلقائي للوصلات بين المتطلبات.