



*COMP208*

*2020-21*

*Lecture 4*

Requirement Analysis Review

# *REQUIREMENT REVIEW*

PLEASE READ CAREFULLY

PAGE 12 – 13 OF

notes.pdf

# *Organisational Details*

The documentation to be reviewed must be submitted on CANVAS as a single PDF file on or before 12 noon ***Friday 23 February*** **SIGNED BY ALL CONTRIBUTING TEAM MEMBERS**

**NO SIGNATURE, NO MARK!**

Feedback will be returned as soon as possible.  
Feedback document will rubber stamp the final set of requirements in each case.

# *Requirements Review*

- This event looks at the output of the requirement analysis process.
- Its main purpose is “to ensure that an **appropriate and feasible system** is planned for implementation by each team, and that each **team is working** following a reasonably **well-defined plan.**”
- This review accounts for **12%** of your mark.

# *Documentation Required*

**FIVE PAGES** of pdf CONTAINING

- Project description
  - Mission statement
  - Mission objectives
  - System Boundary Diagram
  - User Views (UCDs) and Their Requirements
  - Transaction Requirements
  - Systems Specification
- Conduct of the Project and Plan (including a project Gantt Chart if appropriate).
- Bibliography

# *Page Limit Hurdle?*

- Major point: synthesis; be concise
- Minor hacks:
  - Play with margins and font size **A BIT** (don't go under font size 10!)
  - Diagrams and pictures clarify matters, but use space, draw them carefully.
  - Sectioning. Headings fonts ..

# *FURTHER READINGS*

Descriptions and examples for some of the items are in Connolly and Begg Chapters 3 and 4 (1<sup>st</sup> edition) or Chapter 6 (2<sup>nd</sup> edition).

[Software Engineering notes](#) are always the most valuable resource.



# FEEDBACK

## COMP 208/214/215/216: Group Software Project REQUIREMENTS REVIEW FEEDBACK FORM

Team:

Project Title:

### Summary

Overall Project Quality:

Category	A++	A+	A	B	C	D	E+	E- to F-	G
Project appears to have enough content (here we assess the <u>scope</u> of the proposed set of requirements)									
Project appears to have enough originality (here we assess the <u>originality</u> of the proposal)									
Progress appears satisfactory (here we provide and overall assessment of the <u>quality of the requirements and their specification</u> : further analysis of the grade provided here is on the following pages)									

Grade:

### Detailed Assessment

Report: project description, deliverables, and plans:

Category	A++	A+	A	B	C	D	E+	E- to F-	G
Mission Statement									
Mission Objectives									
Boundary Diagrams (describing the architecture of the planned system)									
User Views and Their Requirements (using Use Case Diagrams, User Driven Scenarios, etc)									
Transaction requirements									
System Specification									
Project Plan and Gantt Chart (including Objective <u>Prioritisation</u> , Risk Assessment and Contingency Plans)									

### Formative Comments

# FEEDBACK

## Agreed Requirements

## Marking Descriptors:

Grade	Percentage	General Description	Written Material
A++	90% - 100%	Exceptional work	Shows critical understanding of the project aims, thorough analysis, clear, comprehensive and detailed plan
A+	80% - 89%	Outstanding work	Shows critical understanding of the project aims, thorough analysis, clear, comprehensive and detailed plan
A	70% - 79%	Excellent work	Shows comprehensive understanding of the project aims, clear analysis, clear, comprehensive and detailed plan
B	60% - 69%	Competent work	Shows good understanding of the project aims, clear analysis, clear, and detailed plan
C	50% - 59%	Satisfactory work	Shows satisfactory understanding of the project aims, some analysis, and a limited plan
D	40% - 49%	Adequate work	Shows some understanding of the project aims, limited analysis, and a sketchy plan
E+	35% - 39%	Marginal failure	Shows limited or fragmented understanding of the project aims, little analysis, and poor planning
E-	30% - 34%	Non-marginal failure	Shows limited or fragmented understanding of the project aims, very little analysis, and poor planning
F	20% - 29%	Work shows little effort	Shows very limited and fragmented understanding of the project with numerous errors, very little analysis, and poor planning
F-	10% - 19%	Work shows little adherence to the tasks	Shows very limited and fragmented understanding of the project with numerous fundamental errors, very little analysis relevant to the project, and poor planning
G	0% - 9%	Nominal or complete lack of work	Virtually no understanding of the project, no analysis, and virtually no planning

# *SUMMARY*

- Requirement analysis document deadline February 23<sup>rd</sup>
- PDF. 5 pages. Documenting your requirements
- Feedback will include agreed system requirements.

# *REQUIREMENT ANALYSIS*

# COMP201

## Software Engineering 1

### Lecture 4 – What Are Requirements?

***Lecturer: T. Carroll***

***Email: [Thomas.Carroll2@Liverpool.ac.uk](mailto:Thomas.Carroll2@Liverpool.ac.uk)***

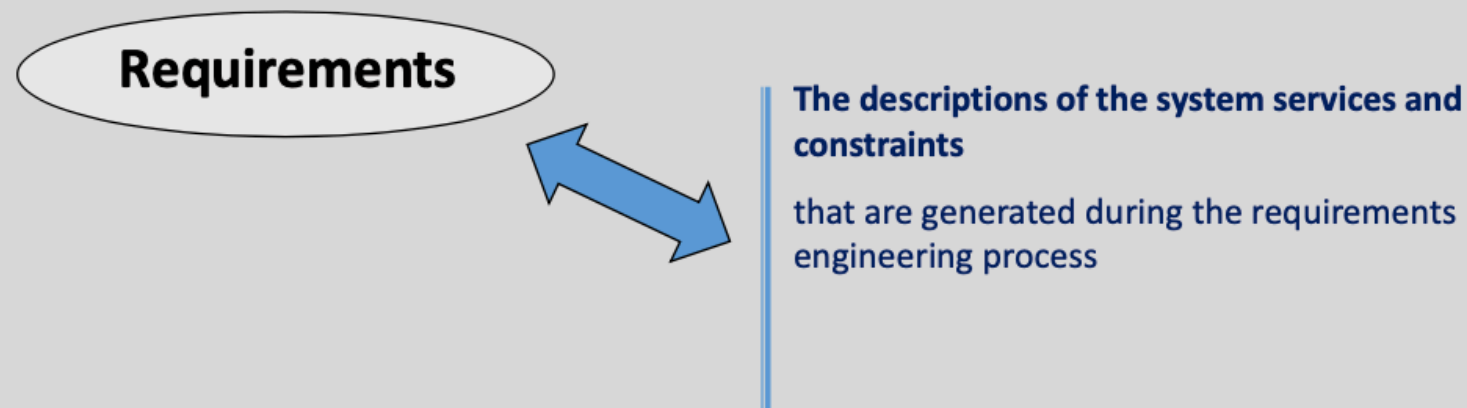
***Office: Ashton G.14***

***See Vital for all notes***

# What Are Requirements?

Requirements engineering is the process of establishing

- **the services** that the customer requires from a system
- **the constraints** under which it operates and is developed





# **Describing Requirements**

# Draw Me A Picture

- It is a sunny day
- Mallard is travelling through the countryside
- It is special, because it has wheels
- It is travelling very fast



# Did It Look Like This?



[https://i.guim.co.uk/img/media/7ae1efa7fc130741c5cbe43245b177b367127d50/150\\_89\\_3354\\_2013/3354.jpg](https://i.guim.co.uk/img/media/7ae1efa7fc130741c5cbe43245b177b367127d50/150_89_3354_2013/3354.jpg)

*DOCUMENTATION*

# *USEFUL EXERCISE*

- cross check these slides with the SE ones (e.g. System Boundary/UCDs)
- What is relevant / irrelevant?
- Try to come up with your own process
- Don't overdo it!
- If something is NOT needed, DON'T use it!

有益的锻炼 • 交叉检查这些幻灯片与se幻灯片（如系统边界/UCD） • 什么是相关/无关的？ • 尝试提出自己的流程 • 不要过度！ • 如果不需要，不要使用！

# ***DISCLAIMER***

1. Content of forthcoming slides **ADVICE ONLY**,
2. Most examples refer to a standard DB system
3. Adjustments may be needed because of specific aspects of each project

免责声明 1仅适用于即将发布的幻灯片建议的内容， 2大多数示例涉及标准数据库系统 三。由于每个项目的具体方面，可能需要进行调整 运用你的判断力！

**USE YOUR JUDGEMENT!**

# *Mission Statement*

- The mission statement is ***a single sentence*** which defines the overall purpose of the planned system: ***what*** it is ***for ...***
- It should be ***clear*** and ***unambiguous***
- It is ***not*** a list of functions that the system will perform: it is the ***reasons why*** those functions are wanted.

*StayHome example: p 64 of Connolly and Begg (p129 in 2<sup>nd</sup> edition).*

使命宣言 •任务说明是一句话，它定义了计划系统的总体目标：它的用途... •应明确无误 •这不是系统将执行的功能列表：而是需要这些功能的原因。例如：康诺利和贝格的第64页（第2版第129页）。

# *Mission Objectives*

- The Mission Objectives statement is a list of ***particular tasks*** that the system ***will support***
- Should be specified using domain specific language and terminology
- ***Tasks supported*** - not ***who*** will do them
- Should be as comprehensive as possible.

*Example: Figure 4.8 of Connolly and Begg (Figure 6.8 in 2<sup>nd</sup> edition).*

# *StayHome Objectives, revisited*

Remember that the example is for a DBMS system!

Very different list for a Computer Game scenario, or a Weather Forecast system

Could be shortened as follows:

- Maintain **BRANCHES, STAFF, VIDEOS, MEMBERS, RENTALS, SUPPLIERS, ORDERS**
- Search on **BRANCHES, STAFF, VIDEOS, MEMBERS, RENTALS, SUPPLIERS, ORDERS**
- Print status of **VIDEOS, RENTALS, ORDERS**
- Report on **BRANCHES, STAFF, VIDEOS, MEMBERS, RENTALS, SUPPLIERS, ORDERS**

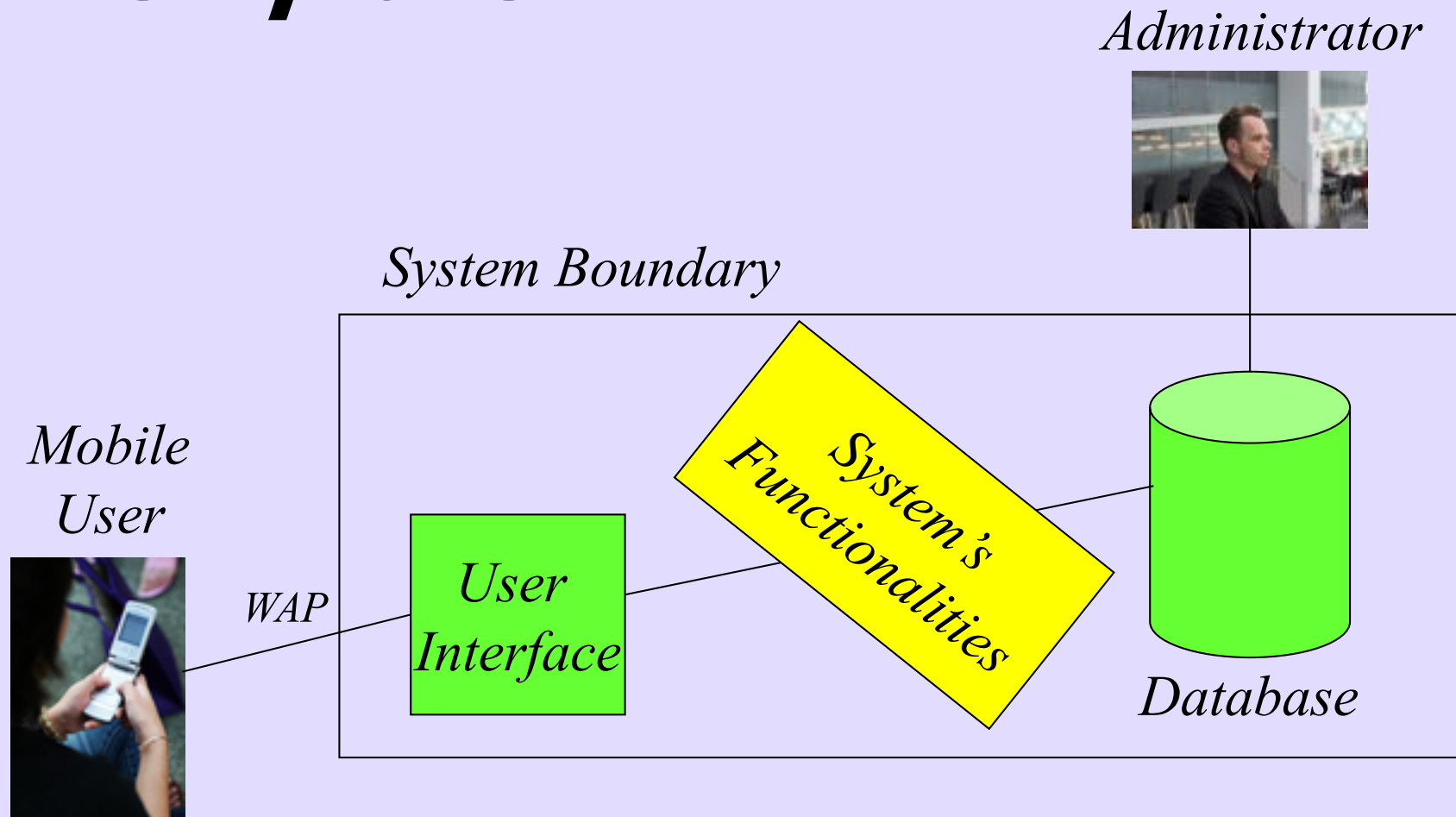
# *Systems Boundary Diagram*

- The intention of this diagram is to represent the main types of data relevant to the system, the main system components, and their relationships
- The boundary also shows what will be included in the system and *what will be not*. Data may be:
  - In the system
  - Available on other systems to which links will be provided
  - Not to be available at all.

*Figure 4.9 of Connolly and Begg provides an example (Figure 6.9 in 2<sup>nd</sup> edition).*



# *Example of System Boundary Template*



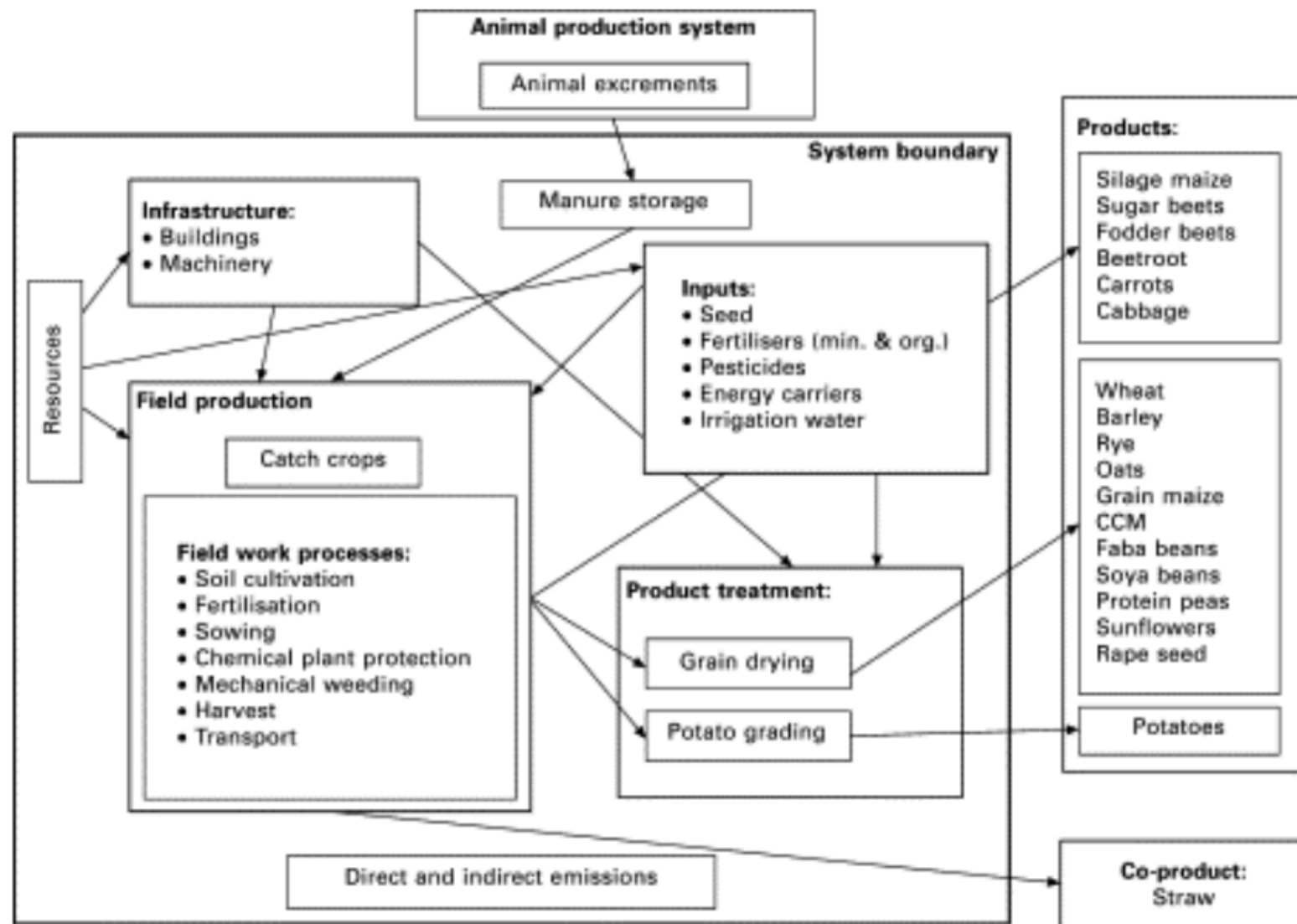


Fig 6.1. Typical system delimitation for a crop product.  
(after Nemecek *et al.*, 2005)

T. Nemecek, G. Gaillard,  
in *Environmental Assessment and Management in the  
Food Industry*, 2010

# *User Views and Requirements*

- Purpose: to identify the major ***classes*** of **user** and the ***functions*** they will ***require***.
  - e.g. administrator, teacher, pupil:
    - Administrator ***maintains*** the database: views, adds, modifies, deletes records
    - Teacher ***customises*** the database: views and adds records, but doesn't modify or delete records
    - Pupil ***uses*** the database: only views records.
- In other applications, different users may maintain and use different data items.

# *User Views and Requirements*

- Purpose: to identify the major **classes** of **user** and the **functions** they will **require**.

EXAMPLE!!

= e.g. administrator, teacher, pupil:

EXAMPLE!!

- Administrator ***maintains*** the database: views, adds, modifies, deletes records
- Teacher ***customises*** the database: views and adds records, but doesn't modify or delete records

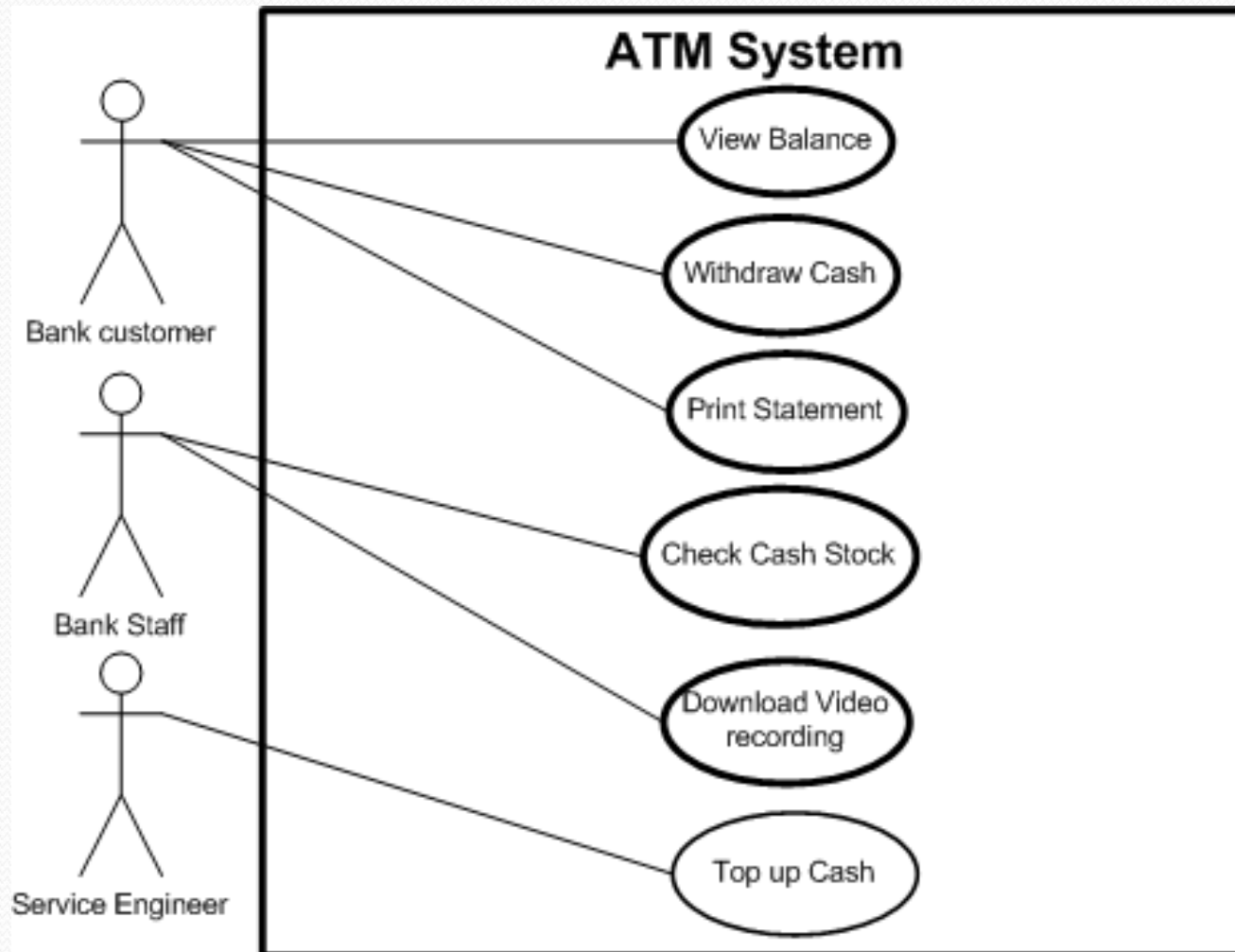
EXAMPLE!!

- Pupil ***uses*** the database: only view records.

EXAMPLE!!

- In other applications, different users may maintain and use different data items.

# Use Case Diagrams



## *More on User Views*

- We are producing a list of users and, for each user, the functions they need:
- Functions should relate to mission objectives:
  - Every user function should be included in mission objectives
  - Every mission objective should relate to some user function
  - Several users may use the same function.

*Example at figure 4.10 of Connolly and Begg  
(Figure 6.10 in 2<sup>nd</sup> edition).*

# *Transaction Requirements*

- Each **user view** will involve **certain transactions** (certain uses or transformations of data), stipulating how the data is to be used
- There are three broad categories of transactions
  - Data entry: every data item needs to be created somewhere
  - Data update and deletion
  - Data queries.
- **Transactions should be related to the user views** to ensure all functions are supported.
- Constraints on such manipulations should be recorded

# *System Requirements*

- Various aspects MIGHT BE covered here:
  - Initial Database Size
  - Rate of Growth
  - Expected type and frequency of searches
  - Network and Access requirements
  - Performance
  - Security
  - Back-up and Recovery
  - Legal Issues
- Detail required will vary according to application and environment
  - e.g. a stand alone single user application will need less detail on access and requirements than a commercial multi-user system.



# *Documentation Required*

**FIVE PAGES** of pdf CONTAINING

- Project description
  - Mission statement
  - Mission objectives
  - System Boundary Diagram
  - User Views (UCDs) and Their Requirements
  - Transaction Requirements
  - Systems Specification
- Conduct of the Project and Plan (including a project Gantt Chart if appropriate).
- Bibliography

# *Project **Plan** ~~(Gantt Chart)~~*

- This should include (but not be limited to) a chart showing the major milestones, tasks and deliverables of the project and when they are scheduled
- You need to report your past progress and future plans.
- You will need to update this chart for each Walkthrough.

# *Project Plan*

- This should include (but not be limited to) a chart showing the major milestones, tasks and deliverables of the project and when they are scheduled.
- You need to report your past progress and future plans.
- You will need to update this chart for each Walkthrough.

**DON'T FORGET TASK  
DEPENDENCIES  
AND TASK ASSIGNMENT TO  
TEAM MEMBERS**