

# CSI6207

## Systems Analysis and Database Design

# CSI6207 Team



Unit Coordinator



Tutor



Dr Viet Huynh

- [v.huynh@ecu.edu.au](mailto:v.huynh@ecu.edu.au)
- Office: JO18.314
- For all:
  - On/Off campus issues
  - Requests (all students)
  - Complaints (all students)
  - Outside allocated consulting time, **use email only!**

Mr Ali Hur

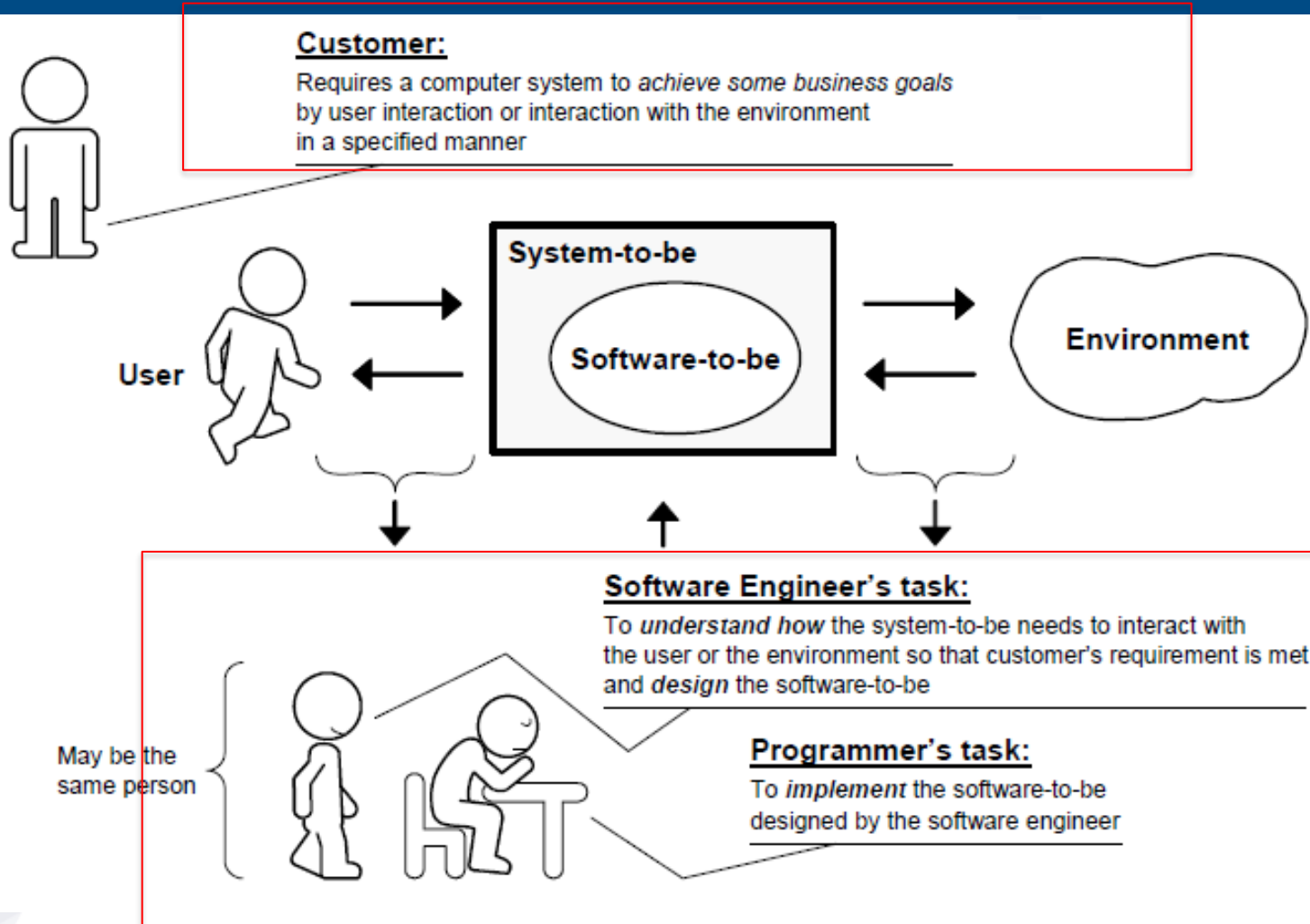
- [m.alihur@ecu.edu.au](mailto:m.alihur@ecu.edu.au)
- All workshop questions

# Introduction

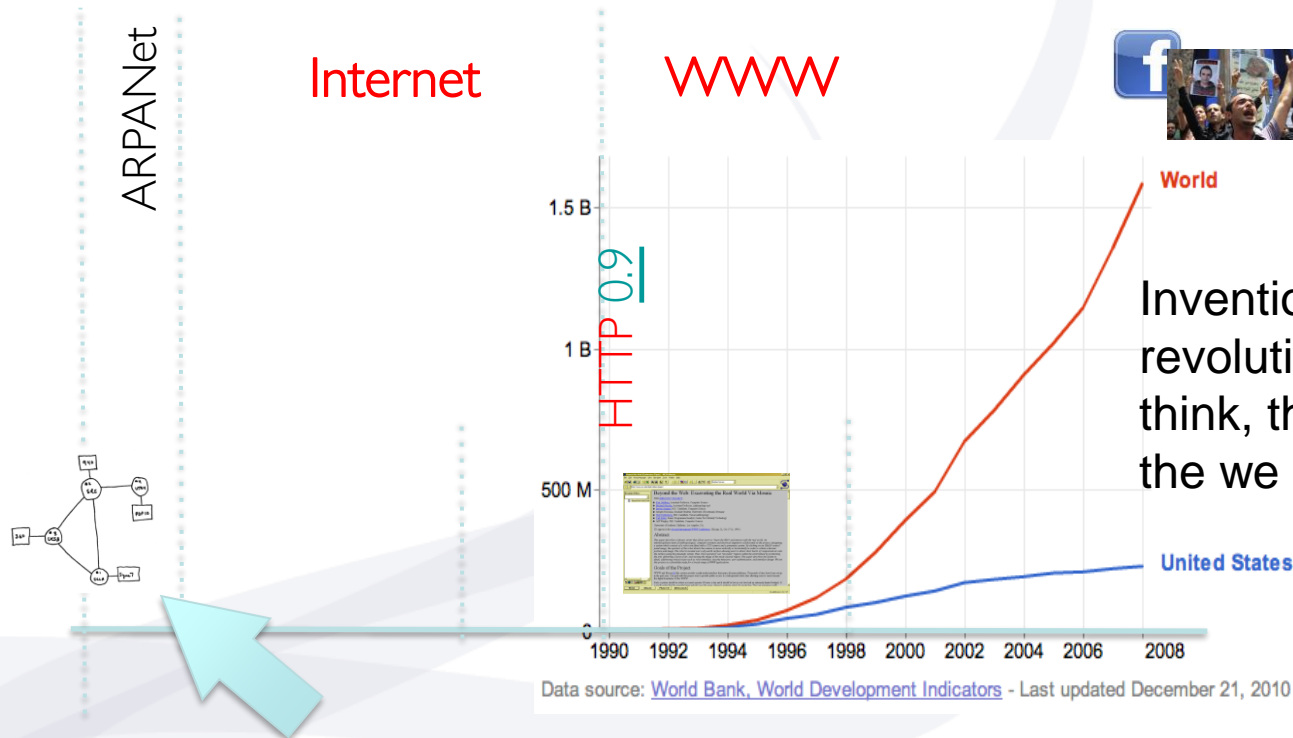
- System analysis and database design
- Unit contents and blackboard website
- Making expectations clear



# Introduction to the context

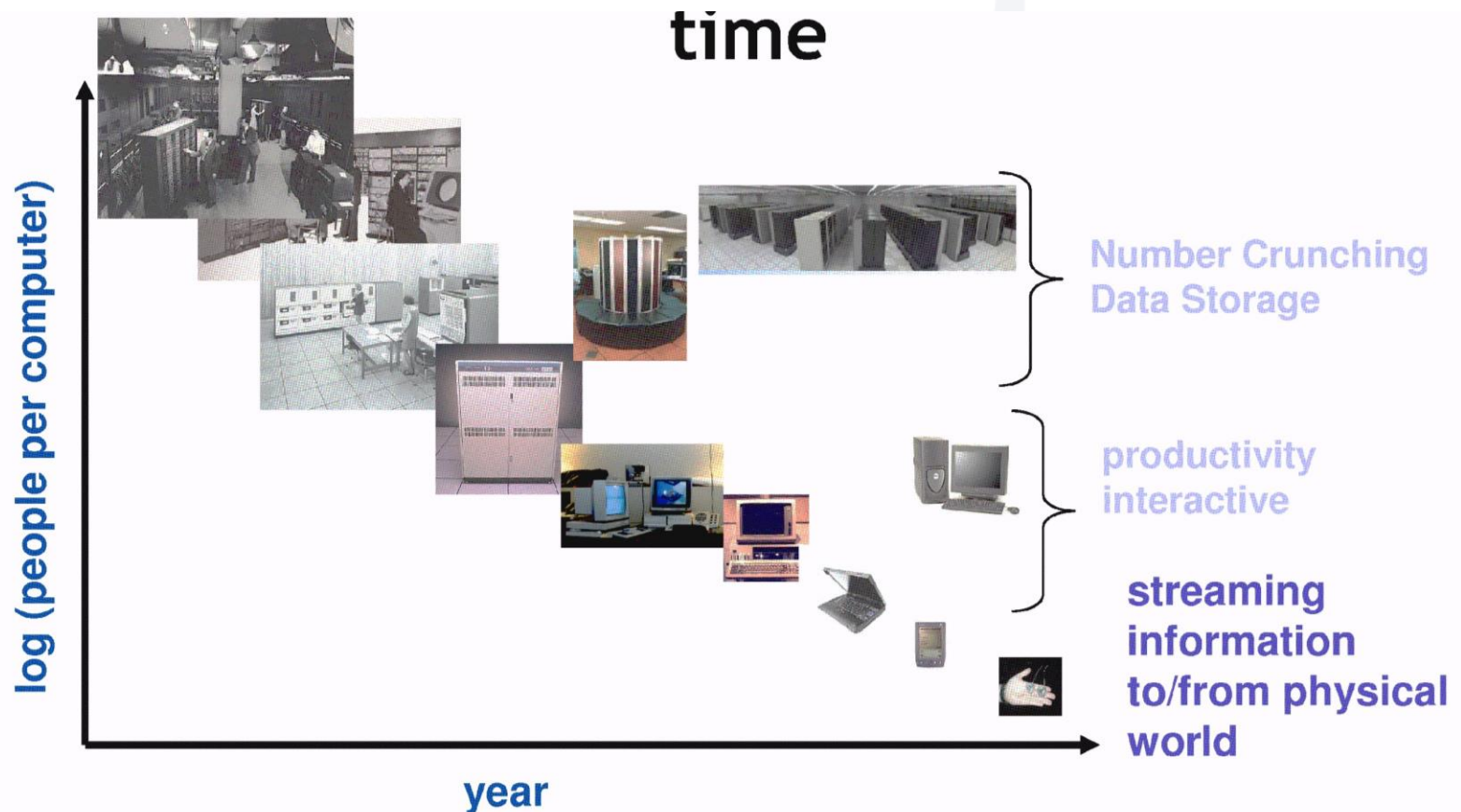


# Internet Growth and Systems Development



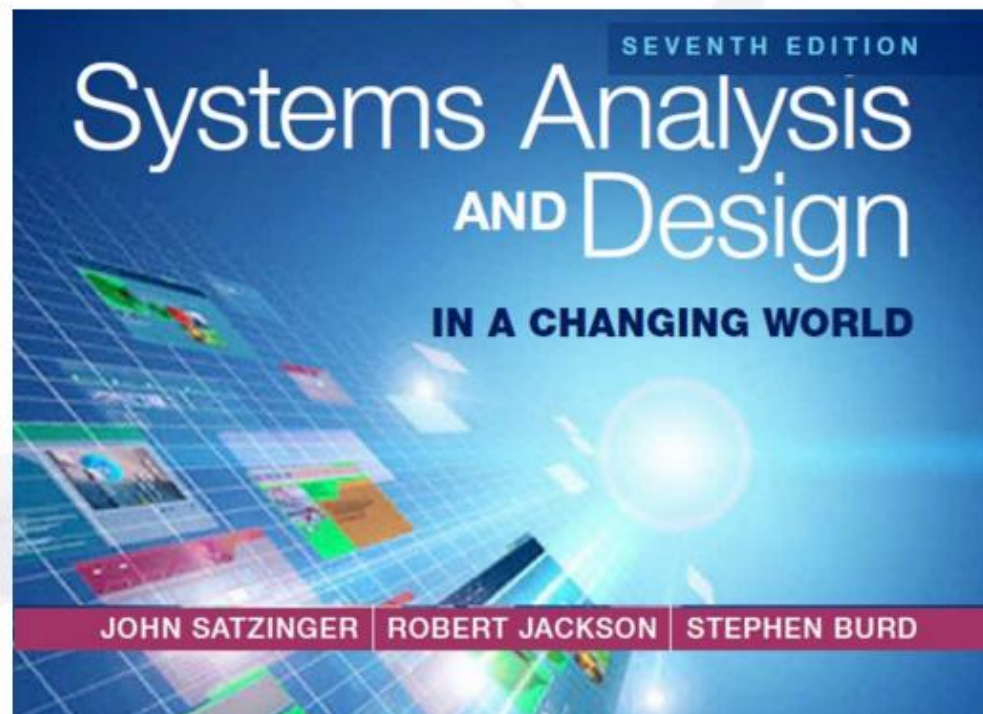
*"When I took office, only high energy physicists had ever heard of what is called the World Wide Web... Now even my cat has it's own page."*  
- [Bill Clinton](#)

# People-to-Computer Ratio Over Time



# Textbook


**Systems Analysis and Design in a Changing World. John W. Satzinger , Robert B. Jackson, Stephen D. Burd**





# Unit Canvas Website

<https://courses.ecu.edu.au/courses/43446>



Account

Dashboard

Courses

Calendar

Inbox

History

Commons

Search

Help

Library

Student Support

2024 Semester 1

Home

Announcements

Discussions

Syllabus

Modules

Assignments

Unit Outline

Reading List

People

Search

Microsoft Teams classes

Microsoft Teams meetings

Grades

Collaborations

Panopto Video

Participation Analytics

Rubrics

Quizzes

Outcomes

Pages

Files

Item Banks

Settings


CSI6207.2

Search this course


63 St

Edit

## Systems Analysis and Database Design - CSI6207.2





**SCHOOL OF SCIENCE**

 Edith Cowan University acknowledges and respects the Nyoongar people, who are the traditional custodians of the land upon which its campuses stand and its programs operate. In particular, ECU pays its respects to the Nyoongar Elders, past and present, and embraces their culture, wisdom and knowledge.

**Systems Analysis and Database Design**


**BUSINESS PROBLEMS AND IDEAS**






Functional and Non


**Agile – User Stories Modeling**




**Use Case Modeling**



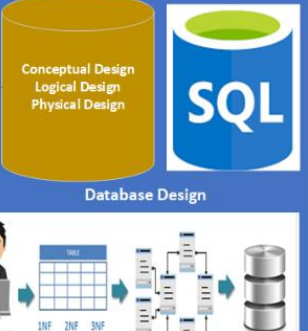
**Business Process Modeling**



**Object Oriented Design**



**Database Design**



8



# Assessment

Type	Description	Value
Test	Online Quizzes	20%
Report	Analysis and Modelling Report	35%
Project	Database Design and Implementation	35%
Presentation	Video Executive Summary	10%

# Getting Help

- See the 'where I can find help' block
- Academy Library – finding information
- Teaching & Learning Unit – references/citations, avoiding plagiarism, reviewing drafts
- Use discussion boards, contact me directly if specific to your circumstances (e.g. extension requests)



# My expectation of students

- Punctual and self motivated learners.
- Will follow up the set of resources and seek additional resources for areas that interest you.
- Will raise issues or problems as soon as they arise.
- Will display a questioning attitude and apply critical thinking to the areas under study.
- Remember, this is your unit which will help you develop and expand your skills.

# You can expect from me

- Be available for individual/group enquires via email or blackboard forums.
- Be open to alternative viewpoints and appreciate robust discussion.
- Value the experience and knowledge that each person add to the class and discussion forums.
- Will provide group level as well as individual level assessment feedback where appropriate.

# CSI6207

## Systems Analysis and Database Design

### Systems Analysis and the System Development Lifecycle

# Contents

- Systems Concepts and Components?
- Software Development and Systems Analysis and Design
- Systems Development Lifecycle (SDLC)
- Iterative Development
- Object-Oriented Design
- Databases

# Learning Objectives

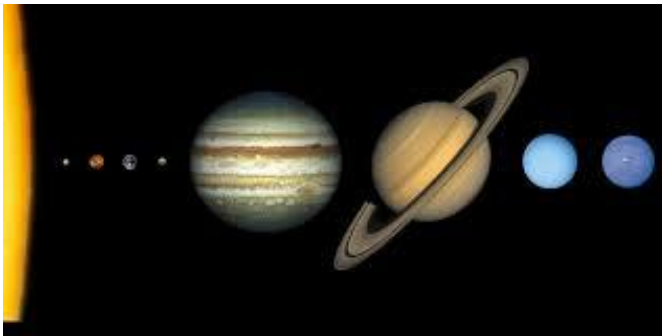
- After reading this chapter, you should be able to:
  - Understand **systems** and their **components**.
  - Describe the **purpose** of systems **analysis and design** when developing information systems
  - Explain the **purpose** of the **system development life cycle** and identify its six core processes
  - Explain how **information system methodologies** provide **guidelines** for completing the six core processes
  - Describe the characteristics of **Agile methodologies** and **iterative** system development
  - Describe the **role** of **databases** in systems development



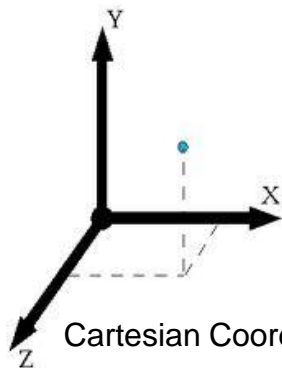
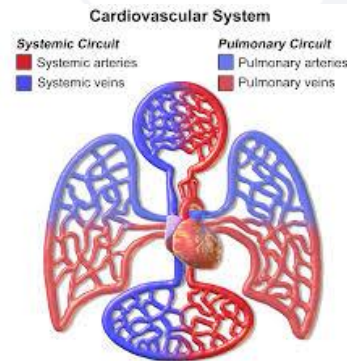
# Systems Concepts and Information Systems

# Systems Concepts

- System – a **collection** of interrelated **components** that function together to achieve some **outcome**



Solar system



Cartesian Coordinate system



Computer system

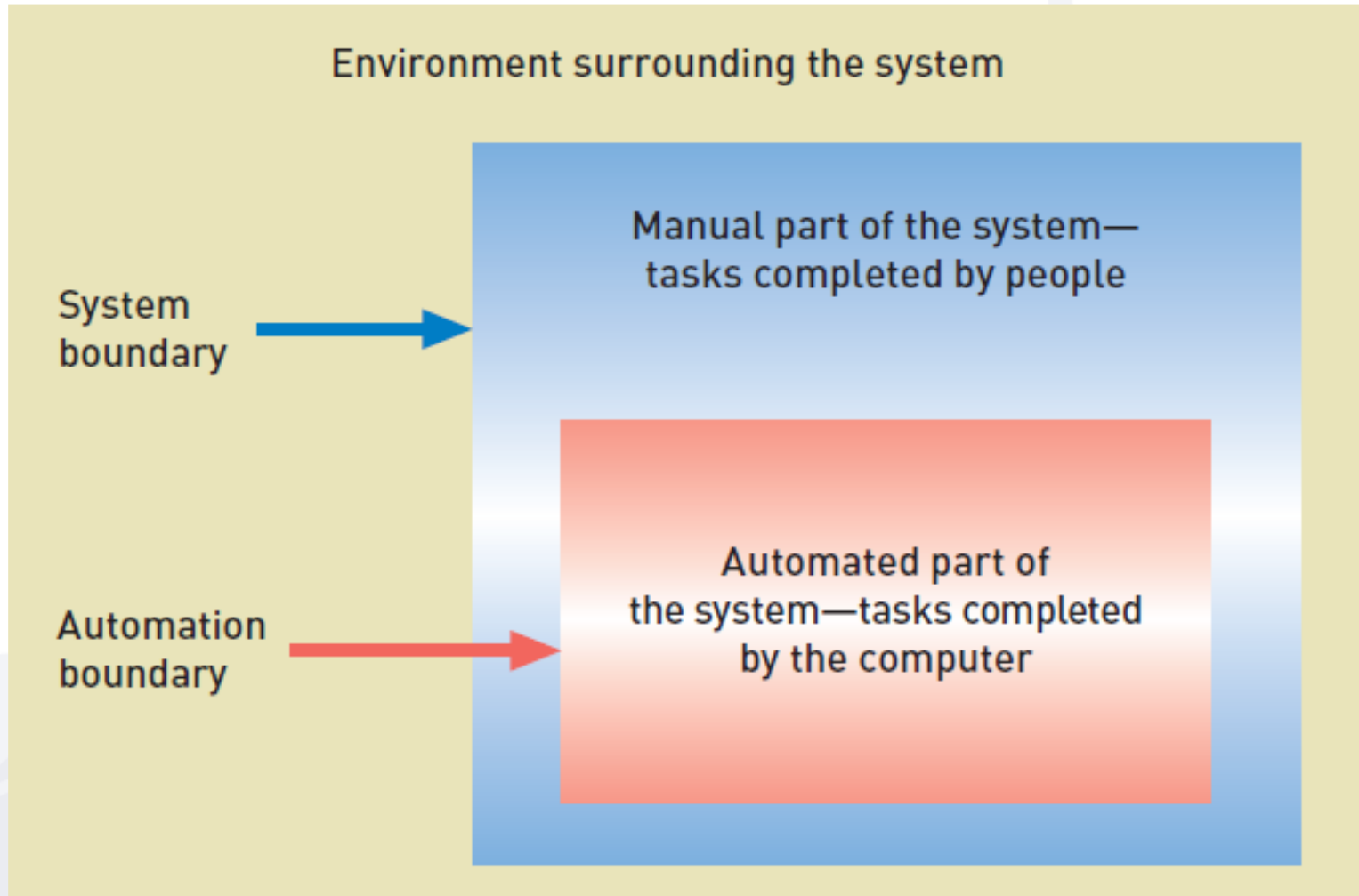


Transport system

# System Components



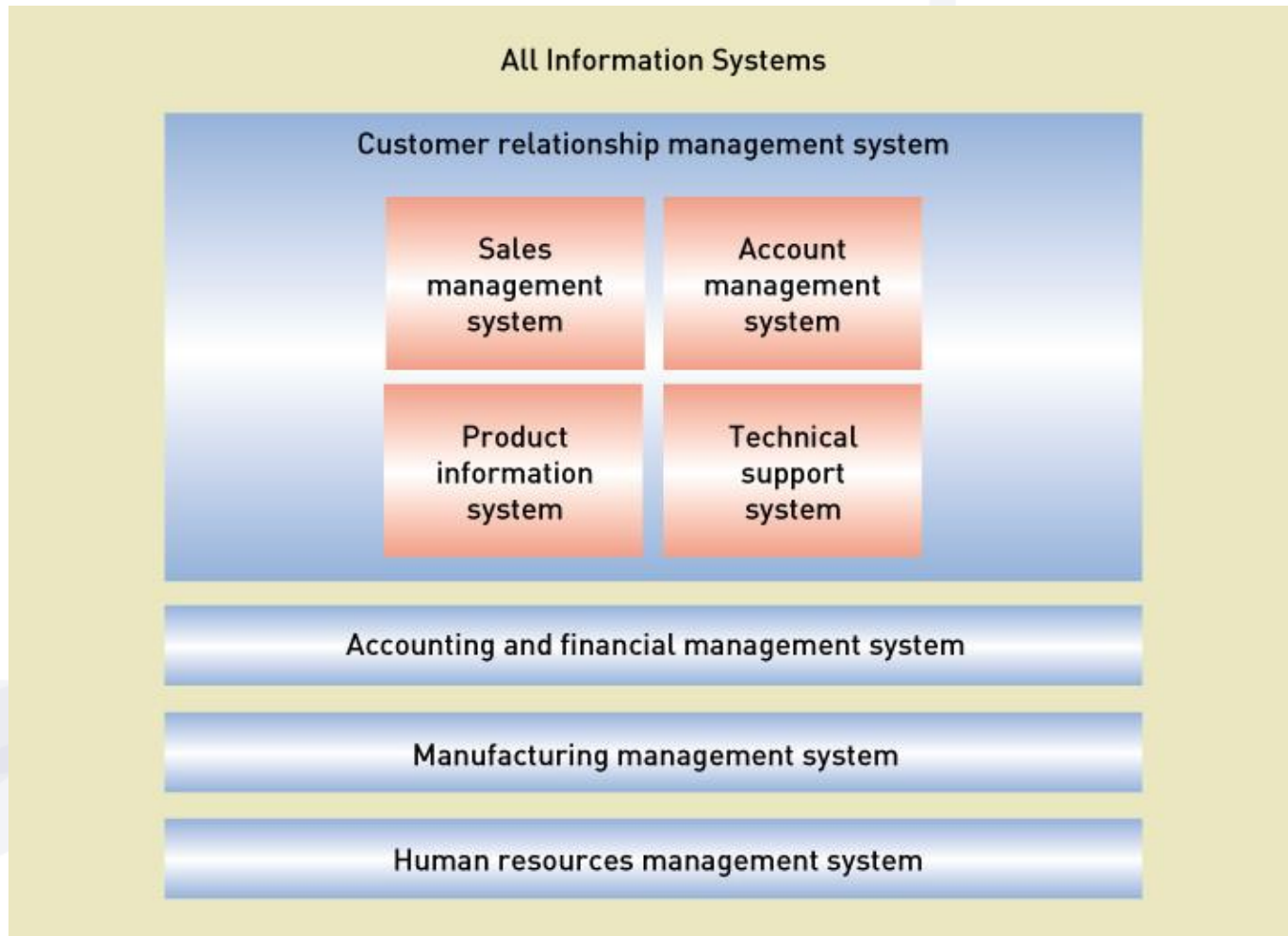
# Systems Concepts..



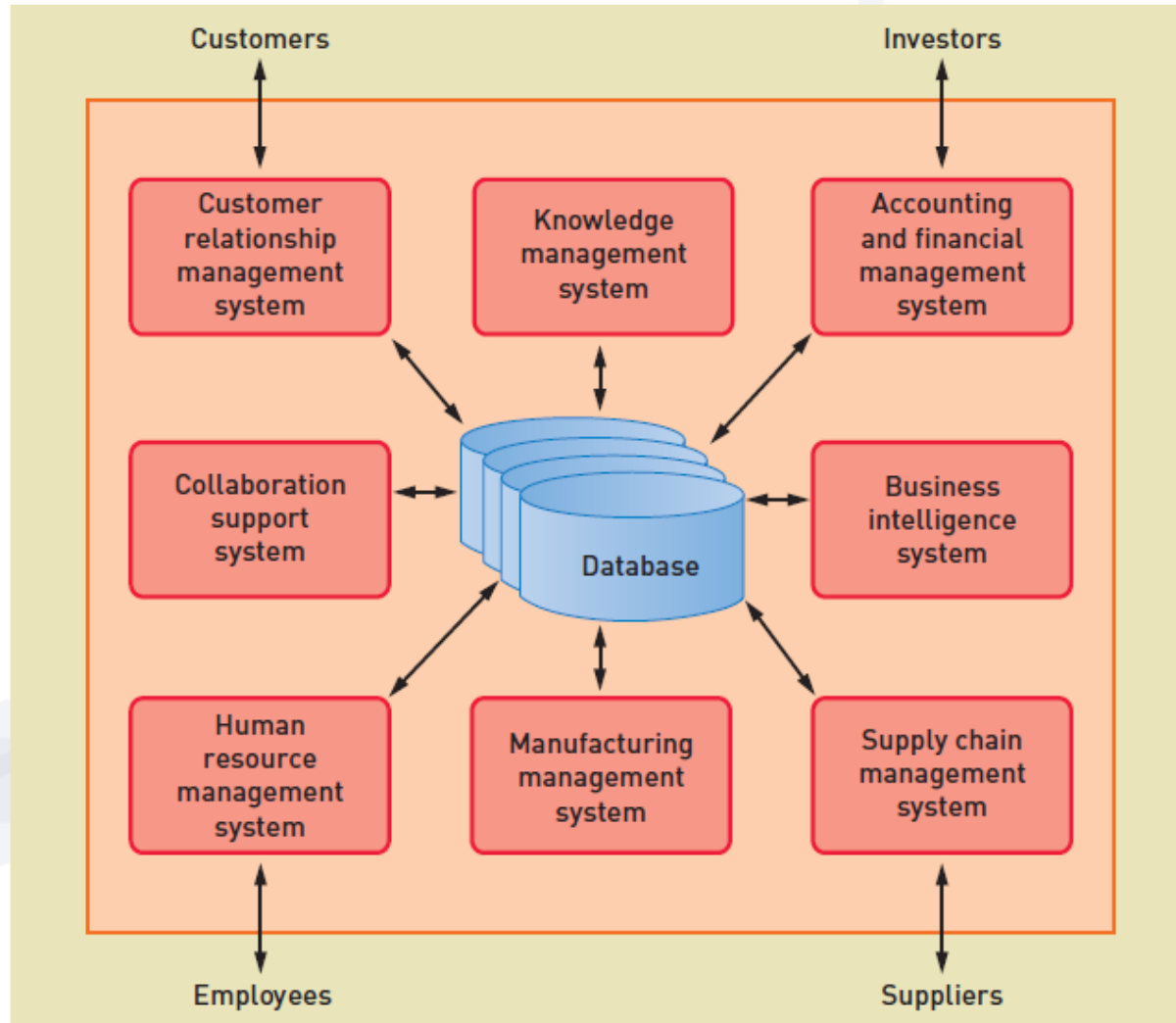
# Information Systems

- Information system – a collection of interrelated components that **collect**, **process**, **store**, and provide as **output** the **information** needed to complete **business tasks**
- Examples:
  - BlackBoard (MyECU)
  - A website
  - A payroll system
  - An order processing system
  - An expert system that diagnoses illnesses

# Information Systems and Subsystems



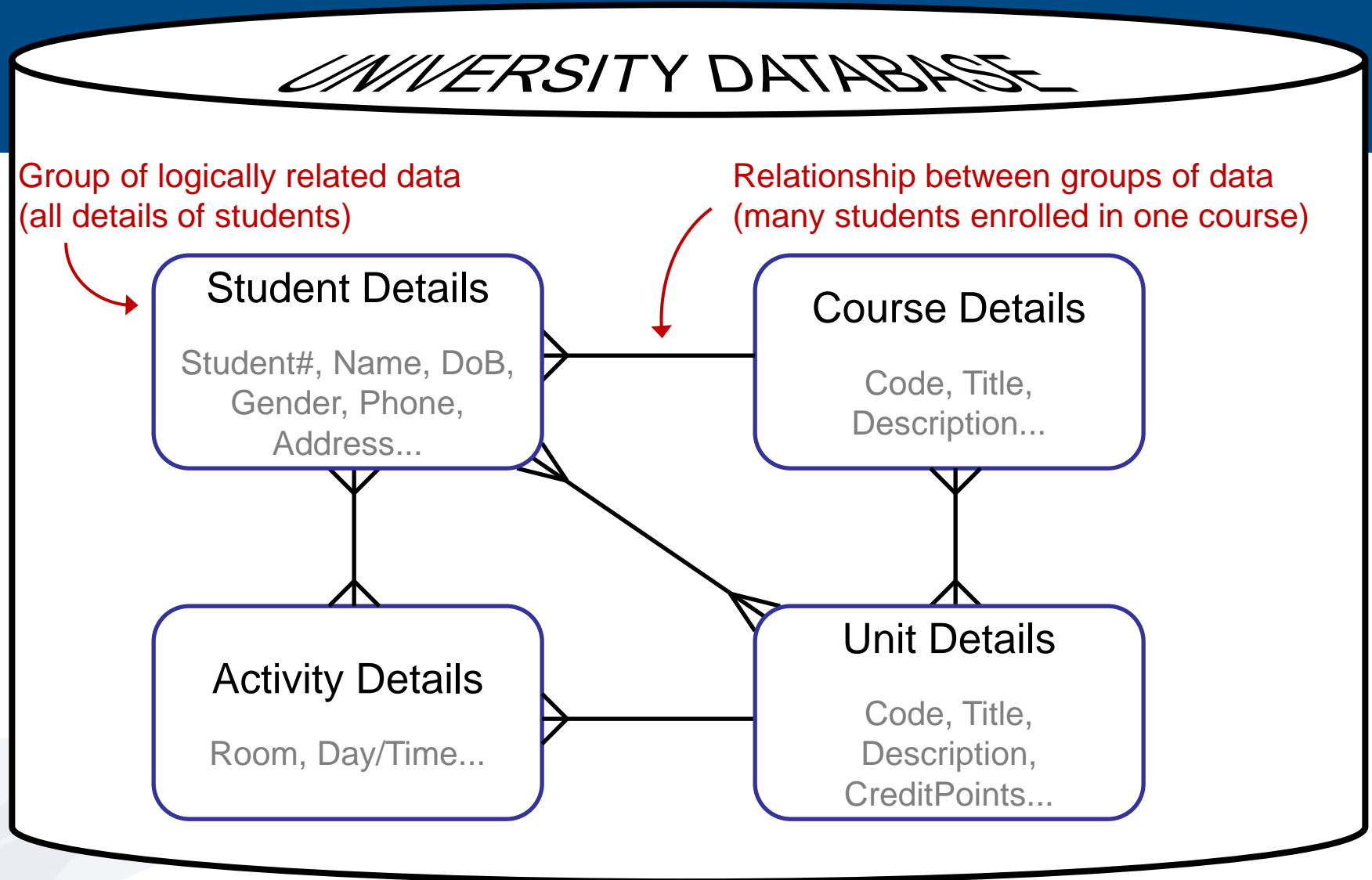
# Types of Information Systems





# What is a database?

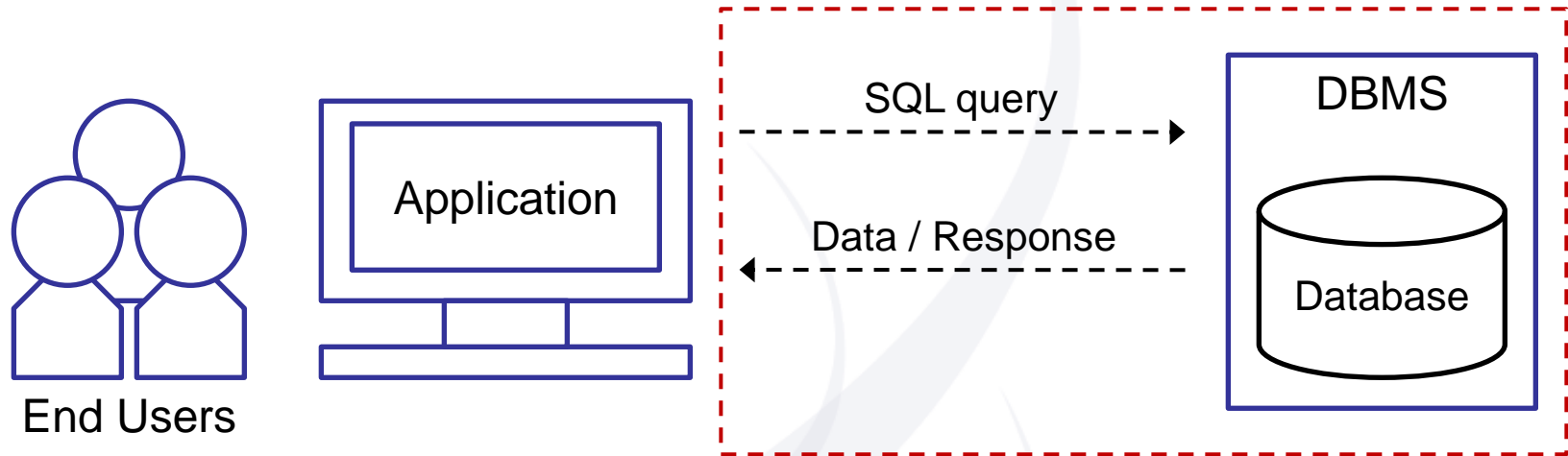
- **Shared** collection of logically related **data** (and a description of this data), designed to **meet** the information **needs** of an organisation
  - Shared data
  - Logically related data comprises of **entities**, **attributes**, and **relationships** of an organisation's information
  - Focus on storing data **efficiently**, without **redundancy**
- A database typically stores:
  - **Names** & **values** of data items in the database
  - **Constraints** on each data item
  - Details of authorised users
  - Data items accessible by a user and the **type of access**



# The Role of a Database

- The database is typically not accessed directly by users
  - A Database Management System (DBMS) **hosts databases**, making them available for applications to interact with as needed
  - Applications interact with databases; **requesting** data, **inserting** data, **updating** data, and **deleting** data
  - The application sends **SQL** (Structured Query Language, covered from week 7) queries to the DBMS, **which executes** them and sends the **resulting** data/response back to the application
  - **Users interact with applications**, not directly with databases

# The Role of a Database



- Notes:
  - The **DBMS** may contain **multiple databases**
  - **Multiple different applications** may interact with a DBMS/databases
  - The interaction, or application itself, may be over a network or the Internet – e.g. Most modern websites involve a **DBMS that a web application written in a server side scripting language** interacts with

# Software Development

# Problems — Software Development

1. **schedule and cost estimates** often grossly **inaccurate**
2. **productivity** of software developers **has not kept pace** with demand for their services
3. **quality** of software is sometimes **less than adequate**
  - unreliable                      Ariane 5 rocket
  - unsafe                              London Ambulance System; Therac-25
  - inflexible                        hard to change/maintain
  - abandoned                      London Stock Exchange

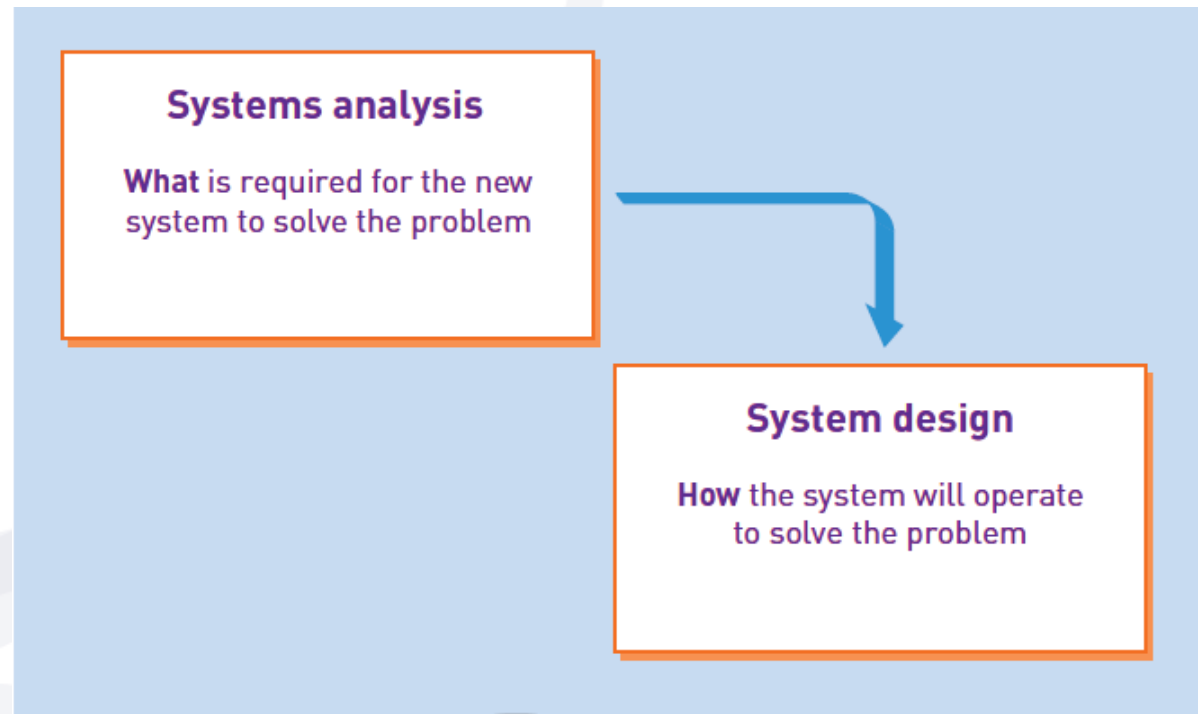
# Problems — Software Development

- **Ariane 501** whose maiden flight on June 4, 1996 ended in **the launcher being exploded** because of a **chain of software failures**. “The failure of the Ariane 501 was caused by the complete loss of guidance and attitude information 37 seconds after start of the main engine ignition sequence (30 seconds after lift-off). This loss of information was due to specification and design errors in the software of the inertial reference system.”
- **London Ambulance System** where because of a succession of software engineering failures, especially defects in project management, a system was introduced that failed twice in the autumn of 1992. Although the monetary cost, at “only” about £9m, was small by comparison with other examples, **it is believed that people died** who would not have died if ambulances had reached them as promptly as they would have done without this software failure.
- **Therac-25** where between 1985 and 1987 **six people (at least) suffered serious radiation overdoses** because of software-related malfunctions of the Therac-25 radiation therapy machine. Three of them are thought to have died of the overdoses. An important root cause was a lack of quality assurance, which led to an over-complex, inadequately tested, under-documented system being developed, and subsequently to the failure to take adequate corrective action.
- **Taurus** a planned automated transaction settlement system for the London Stock Exchange, The project was canceled in 1993 after having lasted more than five years. The project cost was around £75m; the **estimated loss to customers was around £450m**; and the damage to the reputation of the London Stock Exchange was incalculable.



# Software Development

- **Systems analysis** – those activities that enable a person to **understand** and **specify** what an information system **should accomplish**
- **Systems design** – those activities that enable a person to **define** and **describe** in detail the system that **solves the need**



# Software Development

Systems Analysis and design helps to:

1. Understand the need (business need)
2. Capture the vision
3. Define a solution
4. Communicate the vision and solution
5. Build the solution
6. Confirm that the solution meets the need
7. Launch the solution system

# System Development Life Cycle (SDLC)

- SDLC: The process consisting of **all activities** required to **build**, **launch**, and **maintain** an information system.
- Six core processes of SDLC are:
  1. Identify the problem or need and obtain approval
  2. Plan and monitor the project
  3. Discover and understand the details of the problem or need
  4. Design the system components that solve the problem
  5. Build, test, and integrate system components
  6. Complete system tests and then deploy the solution

# System Development Life Cycle (SDLC)

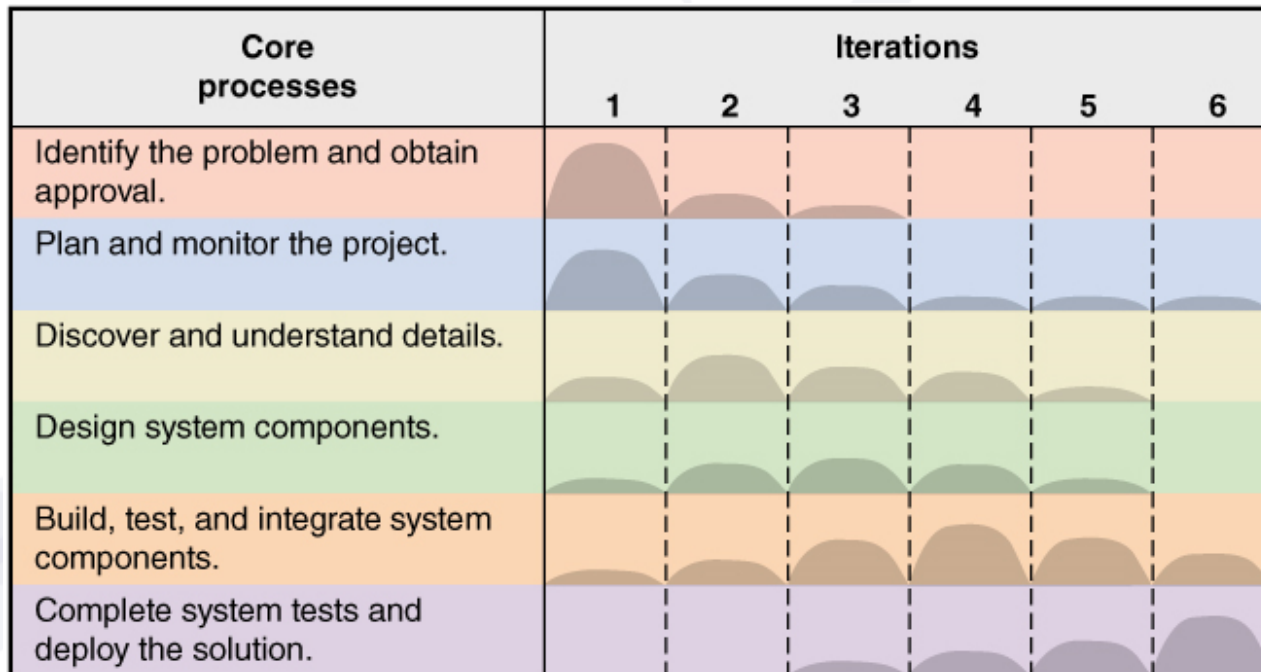
- **System development process** – the **actual approach** used to develop a particular information system (aka: ***methodology***)
- Most processes/methodologies now use **Agile** and **Iterative** development

# Agile Development approach

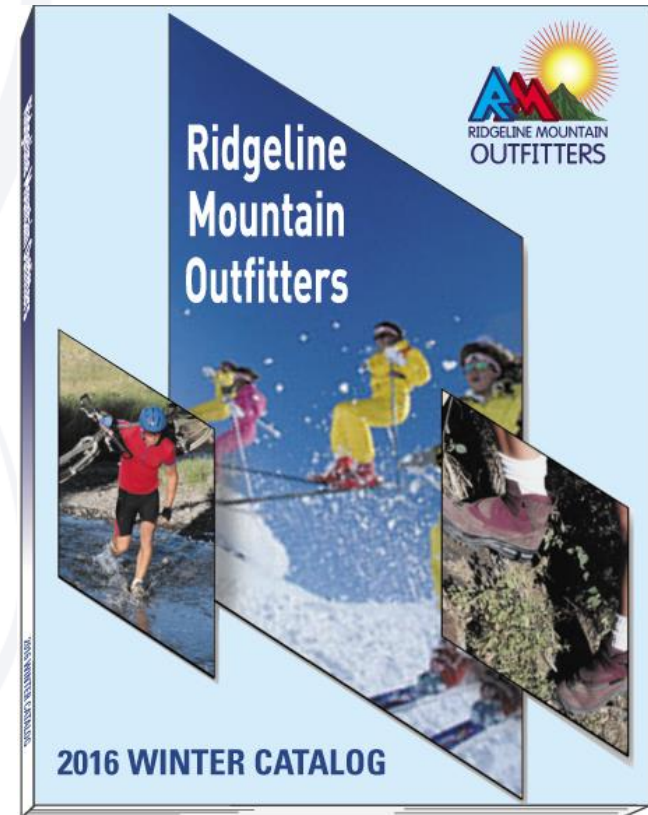
- It emphasises **flexibility** and **rapid** response to anticipate **new and changing requirements** during development
- Most software development processes or methodologies now use this approach
- Examples:
  - Unified process (UP)
  - Extreme programming (XP)
  - Scrum

# Iterative and Agile Systems Development Lifecycle (SDLC)

- In this approach, the **system** is **grown piece by piece** through multiple mini-projects called iterations



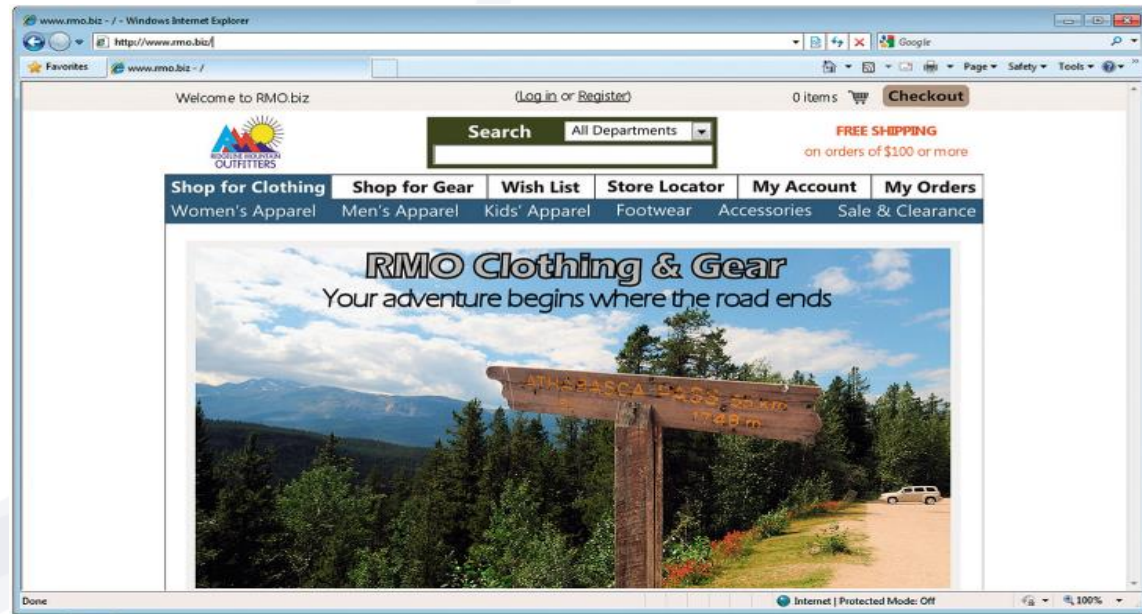
# Case Study: Ridgeline Mountain Outfitters (RMO)





# Ridgeline Mountain Outfitters (RMO)

- Large Retail Company
  - Outdoor and sporting clothing and accessories
  - Skiing, mountain biking, water sports
  - Hiking, camping, mountain climbing
- Rocky Mountain and Western States
  - Started mail order and phone order
  - Added retail stores
  - Added extensive E-business component



# RMO Tradeshow System

- **Problem--** purchasing agents attend apparel and fabric trade shows around the world to order **new products from suppliers**
- **Need—** information system (app) to **collect** and **track information** about **suppliers** and new **products** while at **tradeshows**
- **Tradeshow Project—** is proposed
  - Supplier information subsystem
  - Product information subsystem

# Initial Activities – pre-project

- Identify the **problem** and **document** the objective of the system (core process 1)
  - Preliminary investigation
  - System Vision Document
- Obtain **approval** to commence the project (core process 1)
  - Meet with key stakeholders, including executive management
  - Decision reached, approve plan and budget

# System Vision Document

Problem description

System capabilities

Business benefits

## RMO Tradeshow System



### Problem Description

Trade shows have become an important information source for new products, new fashions, and new fabrics. In addition to the large providers of outdoor clothing and fabrics, there are many smaller providers. It is important for RMO to capture information about these suppliers while the trade show is in progress. It is also important to obtain information about specific merchandise products that RMO plans to purchase. Additionally, if quality photographs of the products can be obtained while at the trade show, then the creation of online product pages is greatly facilitated.

It is recommended that a new system be developed and deployed so field purchasing agents can communicate more rapidly with the home office about suppliers and specific products of interest. This system should be deployed on portable equipment.

### System Capabilities

The new system should be capable of:

- Collecting and storing information about the manufacturer/wholesaler (suppliers)
- Collecting and storing information about sales representatives and other key personnel for each supplier
- Collecting information about products
- Taking pictures of products (and/or uploading stock images of products)
- Functioning as a stand-alone without connection
- Connecting via Wi-Fi (Internet) and transmitting data
- Connecting via telephone and transmitting data

### Business Benefits

It is anticipated that the deployment of this new system will provide the following business benefits to RMO:

- Increase timely communication between trade show attendees and home office, thereby improving the quality and speed of purchase order decisions
- Maintain correct and current information about suppliers and their key personnel, thereby facilitating rapid communication with suppliers
- Maintain correct and rapid information and images about new products, thereby facilitating the development of catalogs and Web pages
- Expedite the placing of purchase orders for new merchandise, thereby catching trends more rapidly and speeding up product availability

# Problem Description

Trade shows have become an important information source for new products, new fashions, and new fabrics. In addition to the large providers of outdoor clothing and fabrics, there are many smaller providers. It is important for RMO to capture information about these suppliers while the trade show is in progress. It is also important to obtain information about specific merchandise products that RMO plans to purchase.

Additionally, if quality photographs of the products can be obtained while at the trade show, then the creation of online product pages is greatly facilitated.

It is recommended that a new system be developed and deployed so field purchasing agents can communicate more rapidly with the home office about suppliers and specific products of interest. This system should be deployed on portable equipment.

# System Capabilities

The new system should be capable of:

- Collecting and storing information about the manufacturer/wholesaler (suppliers)
- Collecting and storing information about sales representatives and other key personnel for each supplier
- Collecting information about products
- Taking pictures of products (and/or uploading stock images of products)
- Functioning as a stand-alone without connection
- Connecting via Wi-Fi (Internet) and transmitting data
- Connecting via telephone and transmitting data



# Business Benefits

It is anticipated that the deployment of this new system will provide the following business benefits to RMO:

- Increase timely communication between trade show attendees and home office, thereby improving the quality and speed of purchase order decisions
- Maintain correct and current information about suppliers and their key personnel, thereby facilitating rapid communication with suppliers
- Maintain correct and rapid information and images about new products, thereby facilitating the development of catalogs and Web pages
- Expedite the placing of purchase orders for new merchandise, thereby catching trends more rapidly and speeding up product availability

# Day 1 Activities

- Core Process 2: Plan and monitor the Project
  - Determine the **major components** (functional areas) that are needed
    - Supplier information subsystem
    - Product information subsystem
  - Define the **iterations** and **assign** each **function to an iteration**
    - Decide to do Supplier subsystem first
    - Plan one iteration as it is small and straight forward
  - Determine **team members** and responsibilities



# Day 1: Work Breakdown Structure for Iteration

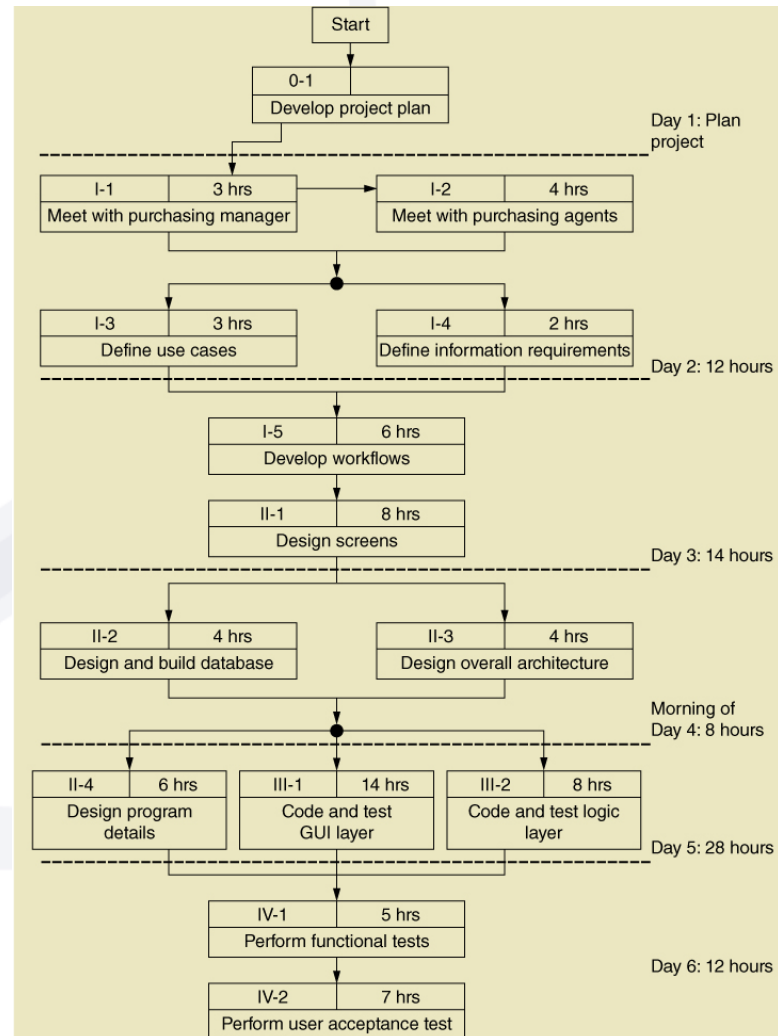
Describes the work and covers Core Processes 3, 4, 5, and 6

## Work Breakdown Structure

- I. Discover and understand the details of all aspects of the problem.
  1. Meet with the Purchasing Department manager. ~ 3 hours
  2. Meet with several purchasing agents. ~ 4 hours
  3. Identify and define use cases. ~ 3 hours
  4. Identify and define information requirements. ~ 2 hours
  5. Develop workflows and descriptions for the use cases. ~ 6 hours
- II. Design the components of the solution to the problem.
  1. Design (lay out) input screens, output screens, and reports. ~ 8 hours
  2. Design and build database (attributes, keys, indexes). ~ 4 hours
  3. Design overall architecture. ~ 4 hours
  4. Design program details. ~ 6 hours
- III. Build the components and integrate everything into the solution.
  1. Code and unit test GUI layer programs. ~ 14 hours
  2. Code and unit test Logic layer programs. ~ 8 hours
- IV. Perform all system-level tests and then deploy the solution.
  1. Perform system functionality tests. ~ 5 hours
  2. Perform user acceptance test. ~ 8 hours

# Day 1: Work Sequence Draft for Iteration

Elaborates the Work  
Breakdown Structure



# Day 2 Activities

- Core Process 3: Discover and Understand  
**Details**
  - Do preliminary fact-finding to understand **requirements**
  - Develop a preliminary list of **use cases** and a **use case diagram**
  - Develop a preliminary list of **classes** and a **class diagram**

# Day 2: Identify Use Cases

## Both subsystems

Use Case	Description
Look up supplier	Using supplier name, find supplier information and contacts
Enter/update supplier information	Enter (new) or update (existing) supplier information
Look up contact	Using contact name, find contact information
Enter/update contact information	Enter (new) or update (existing) contact information
Look up product information	Using description or supplier name, look up product information
Enter/update product information	Enter (new) or update (existing) product information
Upload product image	Upload images of the merchandise product

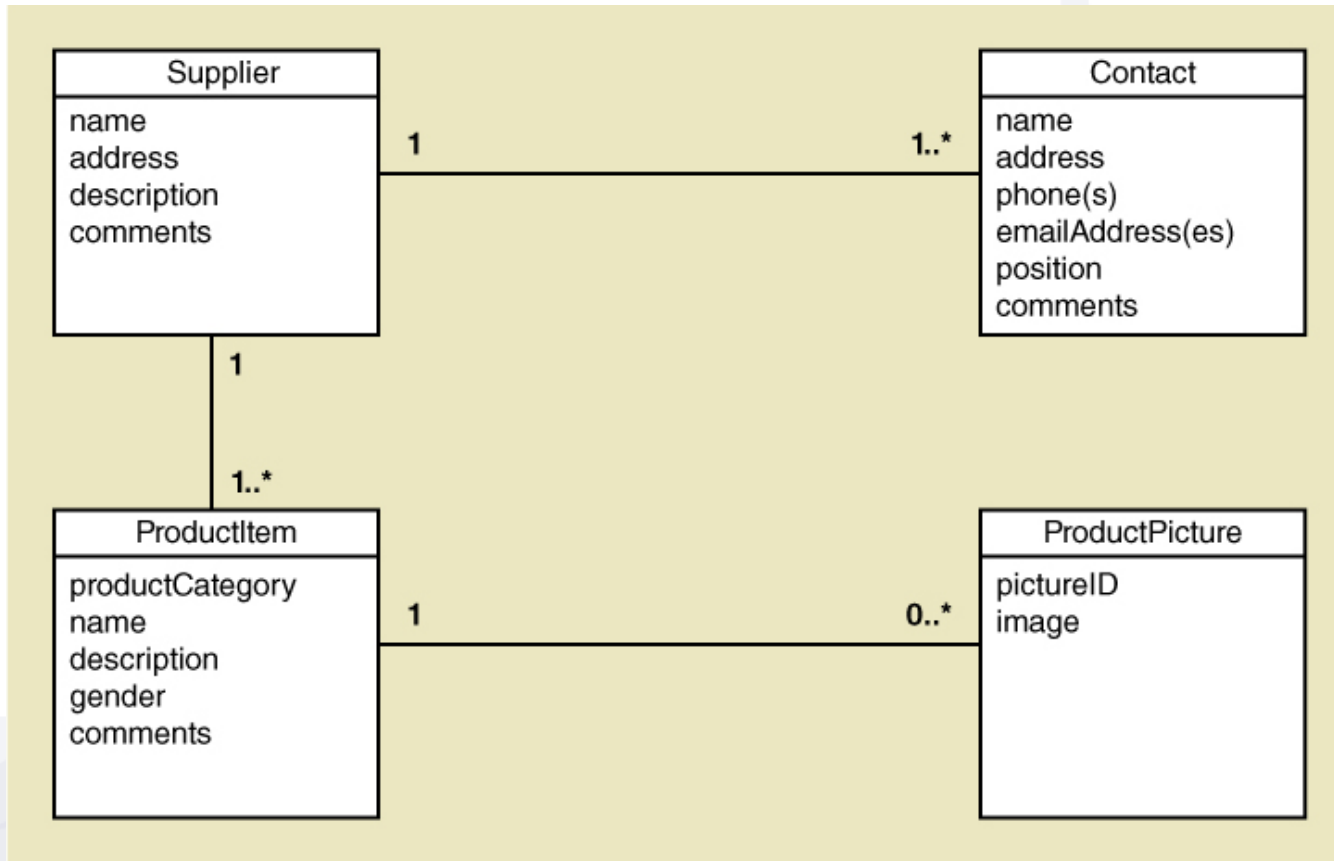
# Day 2: Identify Object Classes

## Both subsystems

Object Classes	Attributes
Supplier	supplier name, address, description, comments
Contact	name, address, phone(s), e-mail address(es), position, comments
Product	category, name, description, gender, comments
ProductPicture	ID, image

# Day 2: Preliminary Class Diagram

## Both subsystems

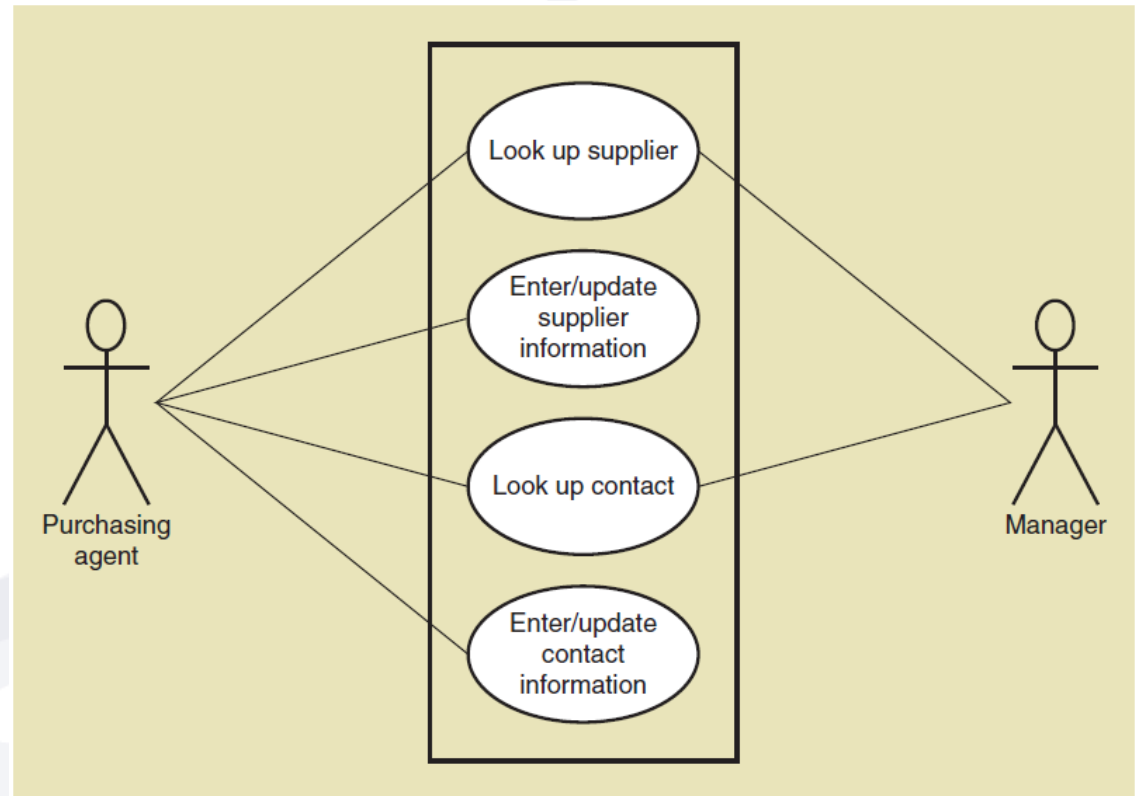


# Day 3 Activities

- Core Process 3: Discover and Understand Details
  - Do in-depth fact-finding to understand requirements
  - Understand and document the **detailed workflow of each use case**
- Core Process 4: Design System Components
  - Define the user experience with screens and report sketches

# Day 3: Supplier Information Subsystem

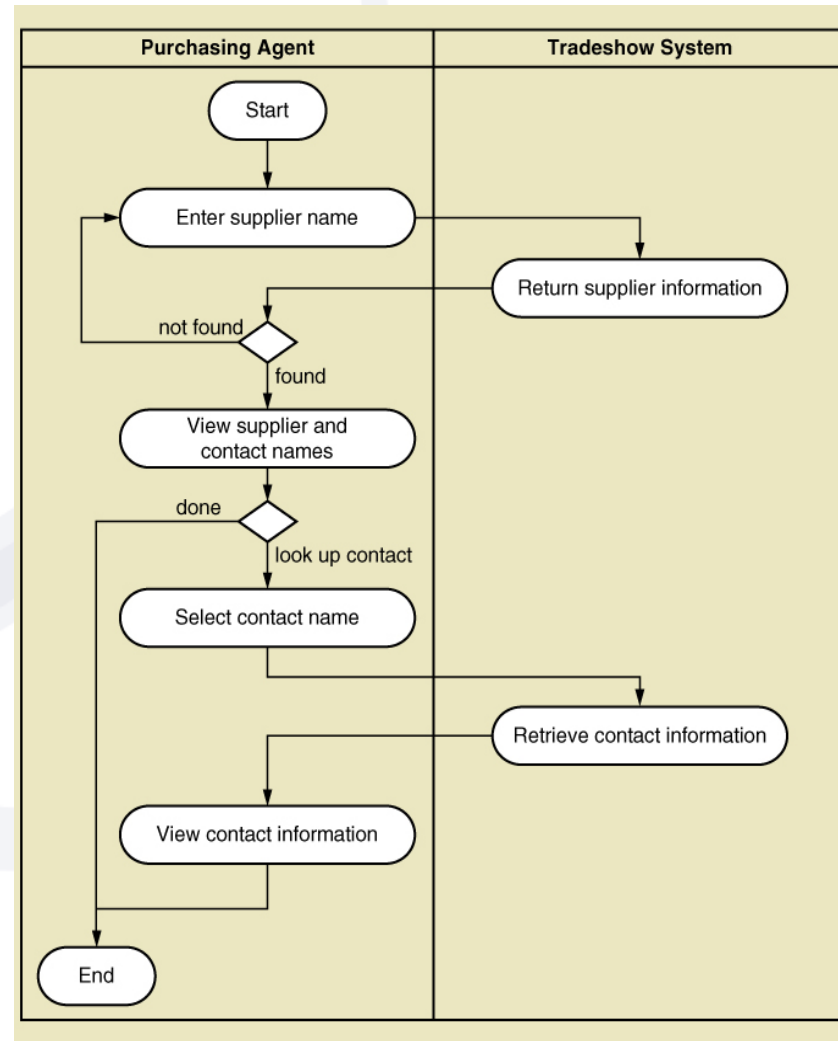
- Use cases:
  - Look up supplier
  - Enter/update supplier information
  - Lookup contact information
  - Enter/update contract information





# Day 3: Activity Diagram (Workflow)

***Look up supplier*** use case



# Day 3: Draft Screen Layout

*Look up supplier* use case

Logo

Web Search

GO

RMO Database Search

Supplier Name

Product Category

Product

Country

Contact Name

GO

Search Results

Supplier Name	Contact Name	Contact Position

# Day 4 Activities

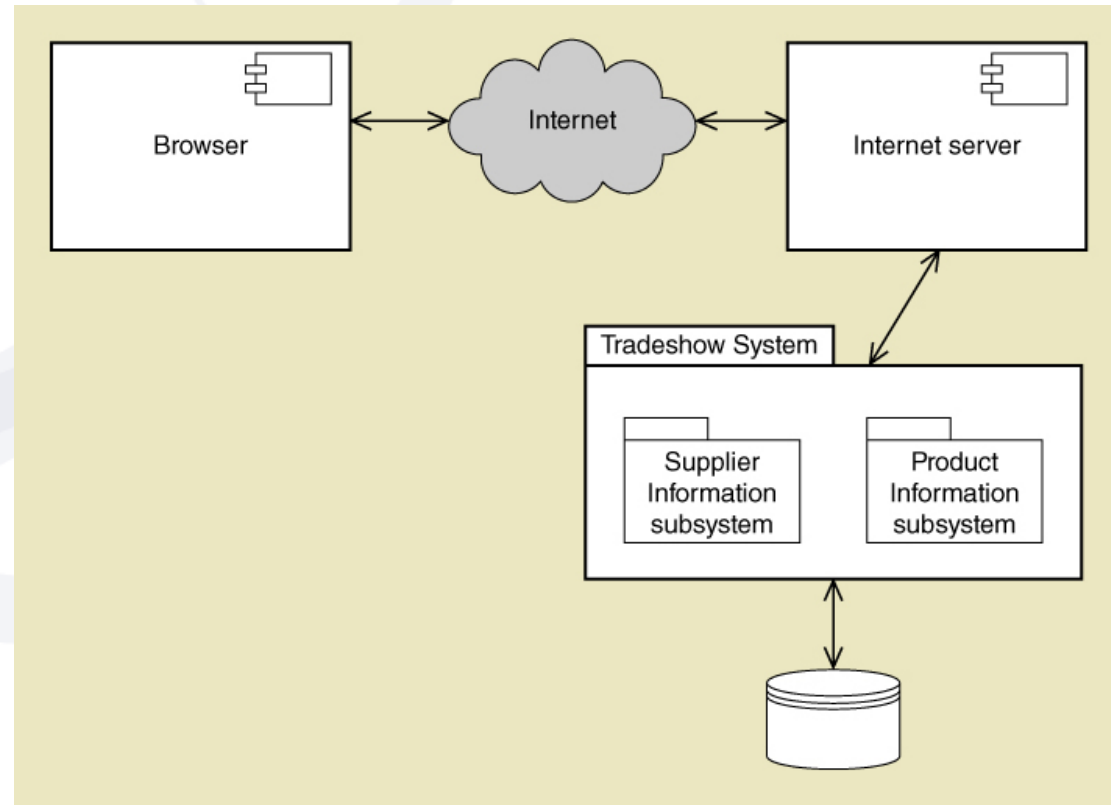
- Core Process 4: Design System Components
  - Design the **database** (schema)

Table Name	Attributes
Supplier	SupplierID: integer {key} Name: string {index} Address1: string Address1: string City: string State-province: string Postal-code: string Country: string SupplierWebURL: string Comments: string
Contact	ContactID: integer {key} SupplierID: integer {foreign key} Name: string {index} Title: string WorkAddress1: string WorkAddress2: string WorkCity: string WorkState: string WorkPostal-code: string WorkCountry: string WorkPhone: string MobilePhone: string EmailAddress1: string EmailAddress2: string Comments: string

# Day 4: Architectural Configuration Diagram

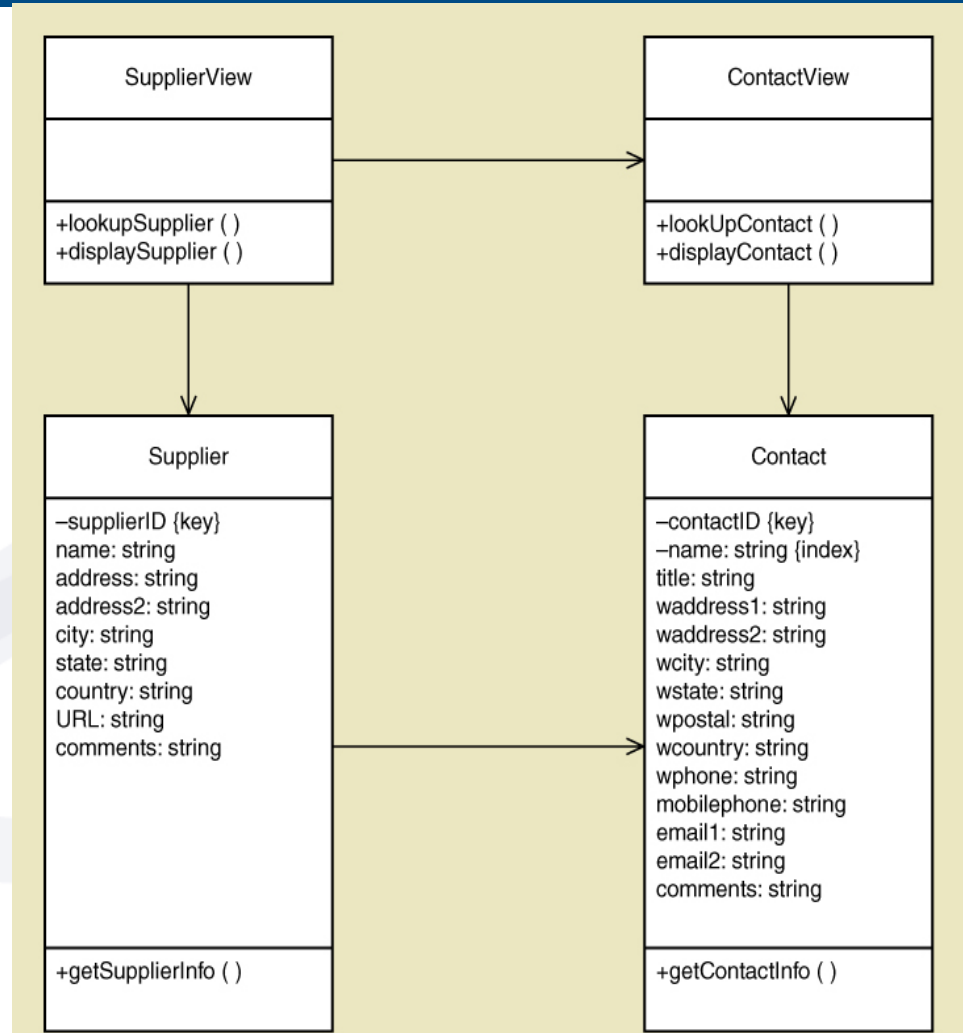
Design the system's **high level structure**

- Browser, Windows, or Smart phone
- Architectural configuration (components)
- Design class diagram
- Subsystem architectural design

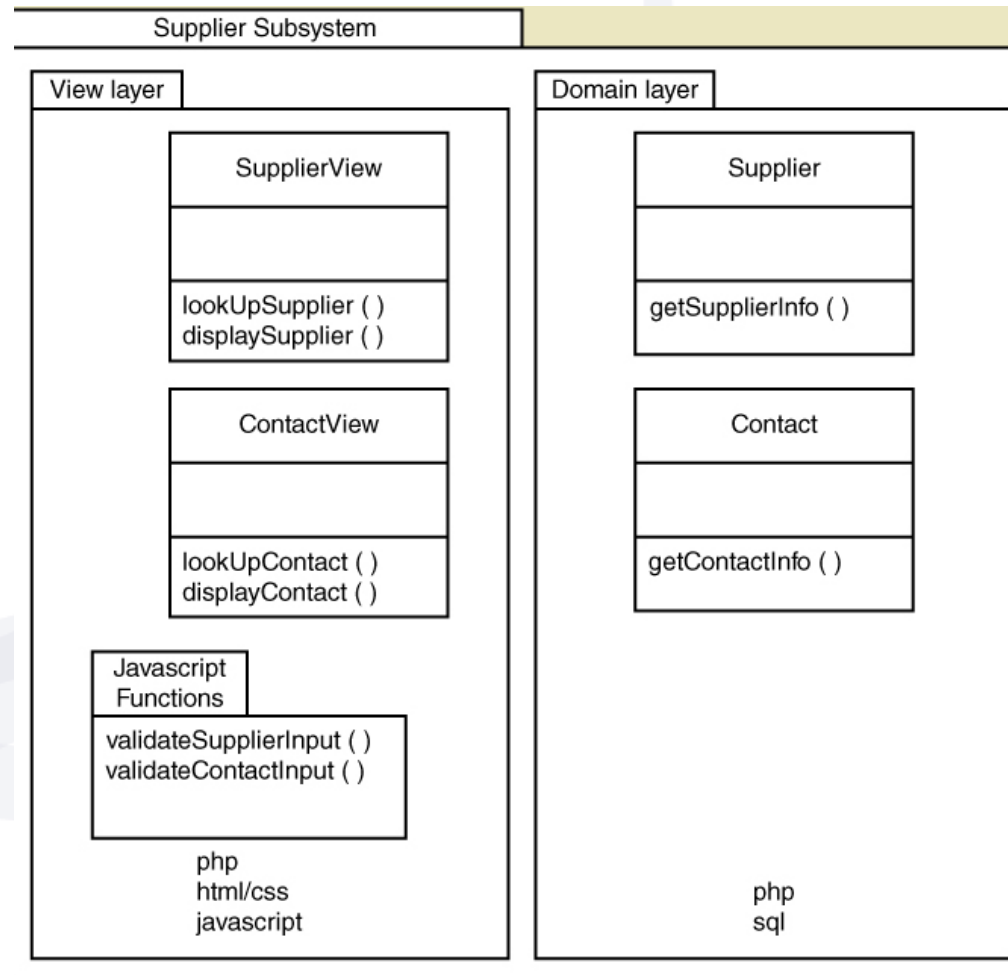


# Day 4: Preliminary Design Class Diagram

Includes View Layer  
Classes and Domain  
Layer Classes



# Day 4: Subsystem Architectural Design Diagram



# Notes on Managing the Project

- Lots of design diagrams shown
  - Design in a **complex** activity with **multiple levels**
  - **One diagram** builds on/complements another
  - Not everything is diagrammed, especially for a small project. **Pick and choose.**
- Programming is also done concurrently
  - You don't design everything then code
  - **You do some design, some coding, some design, some coding**

# Day 5 Activities

- Core Process 4: Design System Components
  - Continue with design details
  - Proceed use case by use case
- Core Process 5: Build, Test, and Integrate System Components
  - Continue programming (build)
  - Build use case by use case
  - Perform unit and integration tests



## Day 5: Code Example for One Class

```
<?php
class SupplierView
{
    private Supplier $theSupplier;

    function __construct()
    {
        $this->theSupplier = new Supplier();
    }

    function lookupSupplier()
    {
        include('lookupSupplier.inc.html');
    }

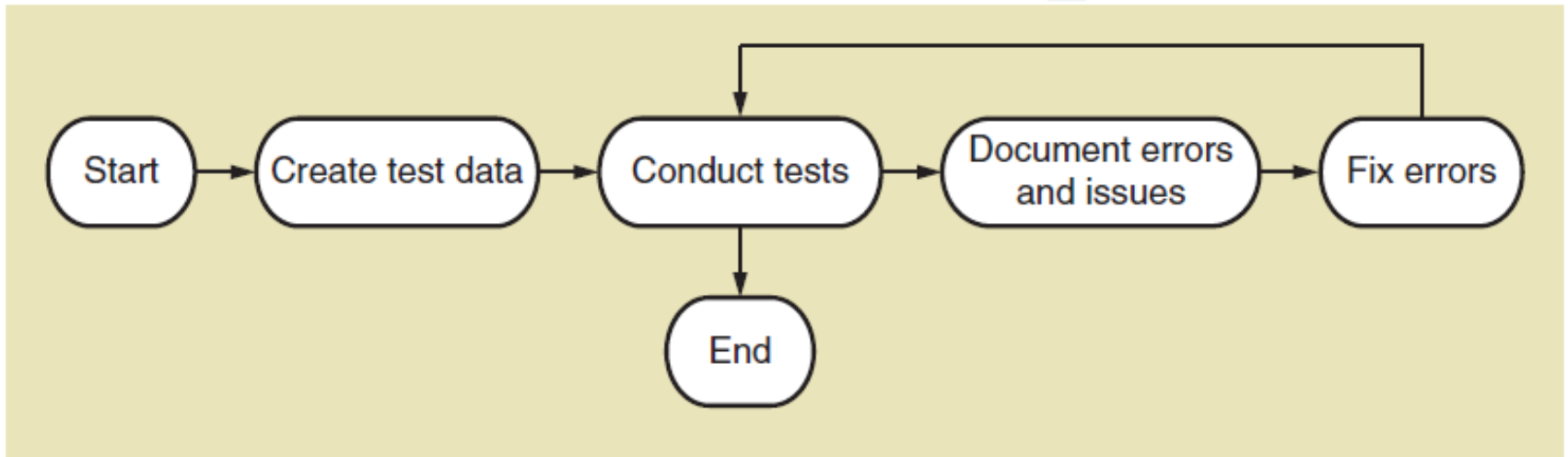
    function displaySupplier()
    {
        include('displaySupplierTop.inc.html');
        extract($_REQUEST); // get Form data
        //Call Supplier class to retrieve the data
        $results = $theSupplier->getSupplierInfo($supplier, $category,
                                                $product, $country, $contact);

        foreach ($results as $resultItem){
            ?>
                <tr>
                    <td style="border:1px solid black">
                        <?php echo $resultItem->supplierName?></td>
                    <td style="border:1px solid black">
                        <?php echo $resultItem->contactName?></td>
                    <td style="border:1px solid black">
                        <?php echo $resultItem->contactPosition?></td>
                </tr>
            <?php }
            include('displaySupplierFoot.inc.html');
        }
    }
?>
```

# Day 6 Activities

- Core Process 6: Complete System Testing and Deploy the System
  - Perform system functional testing
  - Perform user acceptance testing
  - Possibly deploy part of system

# Day 6: Workflow of Testing Tasks



# First Iteration Recap

- This was a 6 day iteration of a small project
  - Most iterations are longer (2 to 4 weeks)
  - This project might be of 2 iterations
  - Most projects have many more iterations
- End users need to be involved, particularly in day 1, 2, 3 and 6.
- Days 4 and 5 involved design and programming concurrently.

# Summary – Terms

- Terms to review and know include:
  - Computer application
  - Information system
  - Project
  - Systems analysis
  - System design
  - System development lifecycle (SDLC)
  - Information system development process (methodology)