

# Systems Design Fundamentals I

**CSIT883 System Analysis and Project Management**



UNIVERSITY  
OF WOLLONGONG  
AUSTRALIA



# Outline

What Is Systems Design?

Systems Design Activities

System Controls and Security

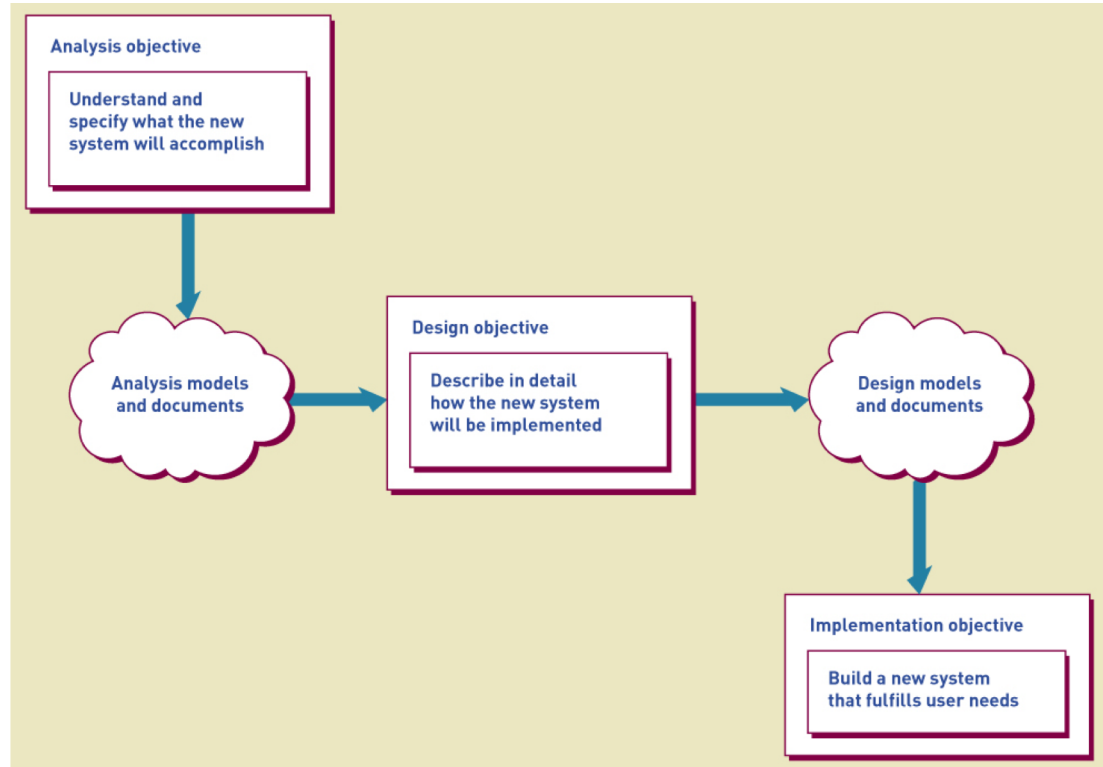
**Part I**

**Part II**



# Systems Analysis vs Systems Design

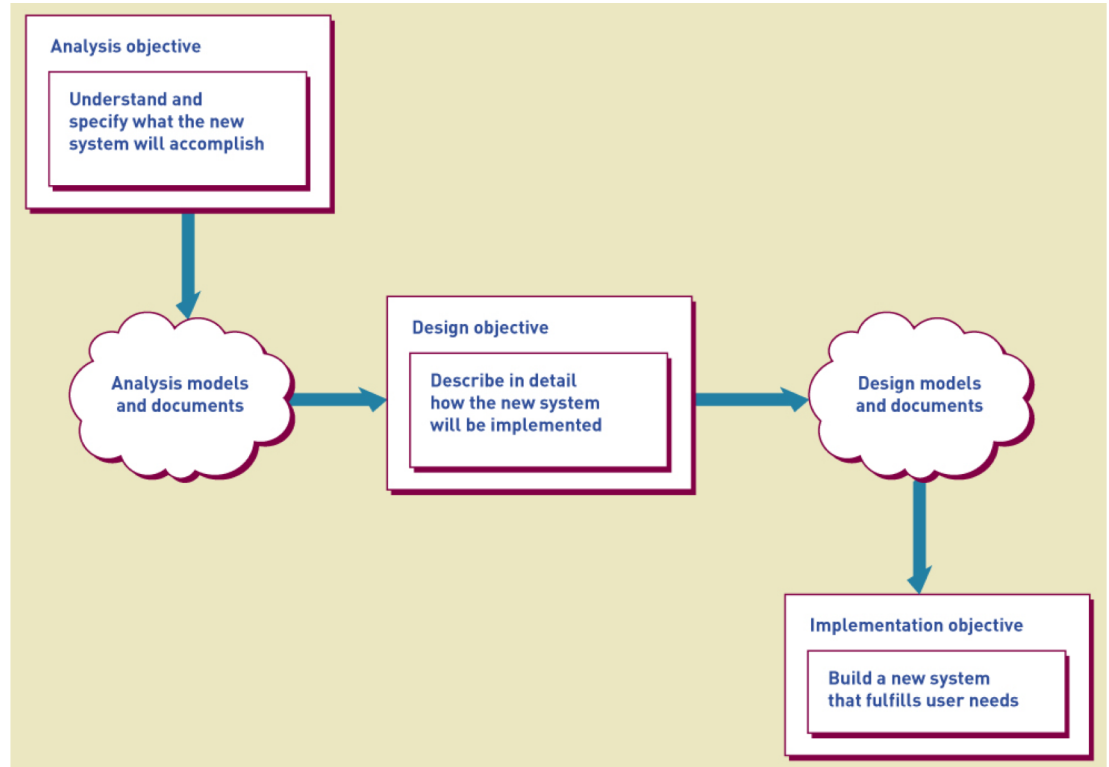
- **Systems Analysis:** System development activities that enable a person to *understand* and specify **what** the new system should accomplish.
- **Systems Design:** System development activities that enable a person to describe in detail **how** the resulting information system will actually be *implemented*.





# Systems Analysis vs Systems Design

- **Systems Analysis:** System development activities that enable a person to *understand* and specify **what** the new system should accomplish.
- **Systems Design:** System development activities that enable a person to describe in detail **how** the resulting information system will actually be *implemented*.





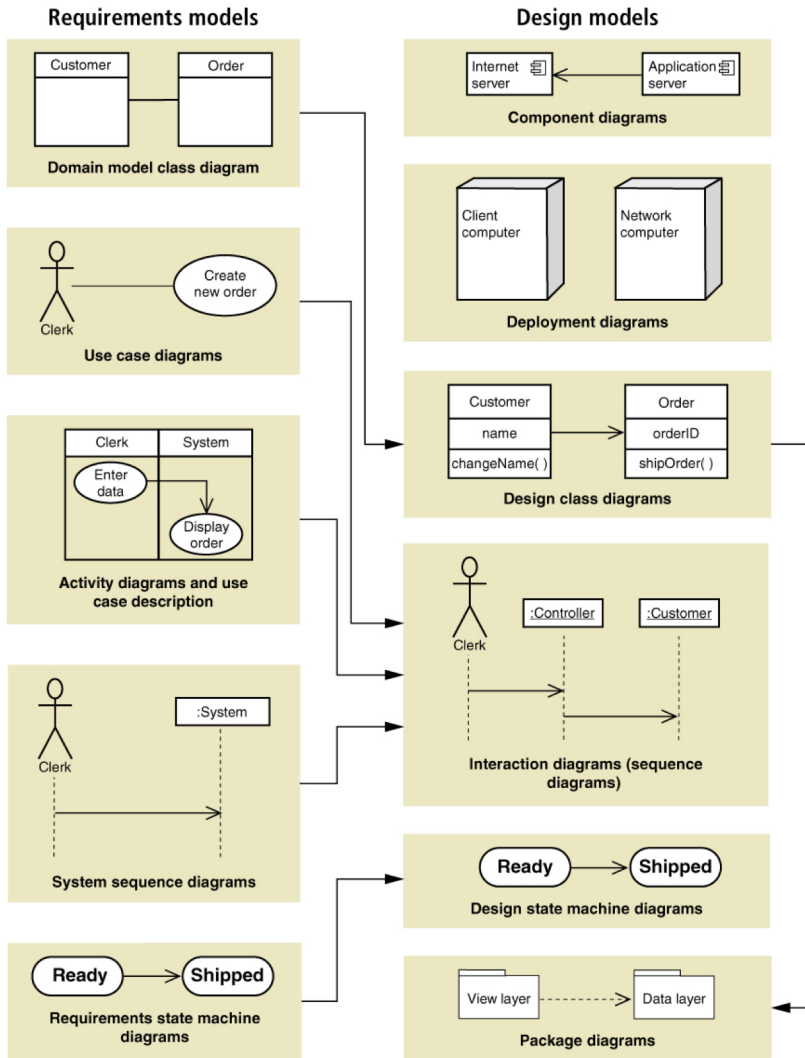


# Design Models

- Design begins with documents and models created during analysis activities.
  - Analysis and design may occur concurrently (e.g., in iterative projects).
- During analysis, requirements models are built to represent the real world and to understand business processes and the information used in those processes.
- Designers convert requirements models into models that represent the solution system.
- The formality of a project also affects how much design details are modeled.



# Analysis Models to Design Models





# Design Activities and Iterations

**Design activities**

Describe the environment.  
Design the application components.  
Design user interface.  
Design the database.  
Design the software classes and methods.

Core processes	Iterations					
	1	2	3	4	5	6
Identify the problem and obtain approval.						
Plan and monitor the project.						
Discover and understand details.						
Design system components.						
Build, test, and integrate system components.						
Complete system tests and deploy the solution.						



# Key Design Questions for each Activity

Design activity	Key question
Describe the environment	How will this system interact with other systems and with the organization's existing technologies?
Design the application components	What are the key parts of the information system and how will they interact when the system is deployed?
Design the user interface	How will users interact with the information system?
Design the database	How will data be captured, structured, and stored for later use by the information system?
Design the software classes and methods	What internal structure for each application component will ensure efficient construction, rapid deployment, and reliable operation?



# Describe the Environment

- *Two key elements* in the environment
  - Communications with External Systems
    - Message formats
    - Web and networks
    - Communication protocols
    - Security methods
    - Error detection and recovery
  - Conforming to an existing **Technology Architecture**
    - i.e., the set of computing hardware, network hardware and topology, and system software employed by an organization.



# Describe the Environment

- *Two key elements* in the environment
  - Communications with External Systems
    - Message formats
    - Web and networks
    - Communication protocols
    - Security methods
    - Error detection and recovery
  - Conforming to an existing **Technology Architecture**
    - i.e., the set of computing hardware, network hardware and topology, and system software employed by an organization.

Design activities
Describe the environment.
Design the application components.
Design user interface.
Design the database.
Design the software classes and methods.



# Design the Application Components

- **Application component** is *a well defined unit of software that performs some function(s)*.
- Issues involve how to package components including
  - Scope and size – what are the functions, boundaries, interfaces?
  - Programming language – what are the accepted languages?
  - Build or buy – is an acceptable version available to purchase?



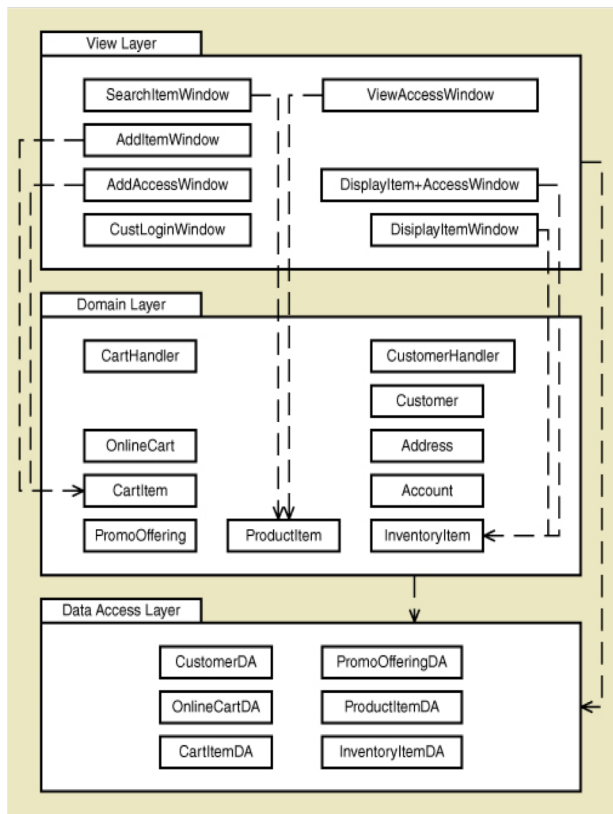
# Design the Application Components

- **Application component** is *a well defined unit of software that performs some function(s)*.
- Issues involve how to package components including
  - Scope and size – what are the functions, boundaries, interfaces?
  - Programming language – what are the accepted languages?
  - Build or buy – is an acceptable version available to purchase?
- Key decisions:
  - how functions of the system will be grouped or packaged
  - how they'll interact with one another once built (or acquired) and assembled
- Impact of technology architecture
- Followed by the software class design

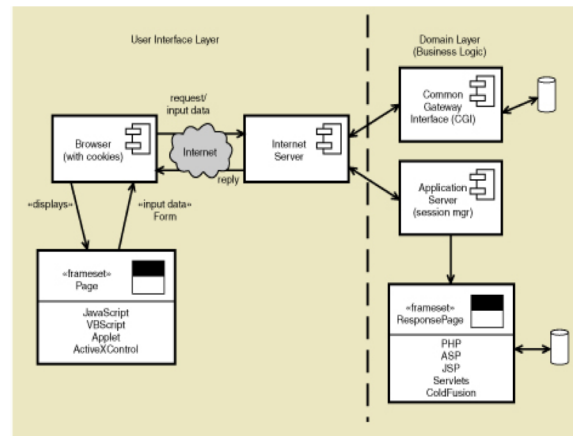




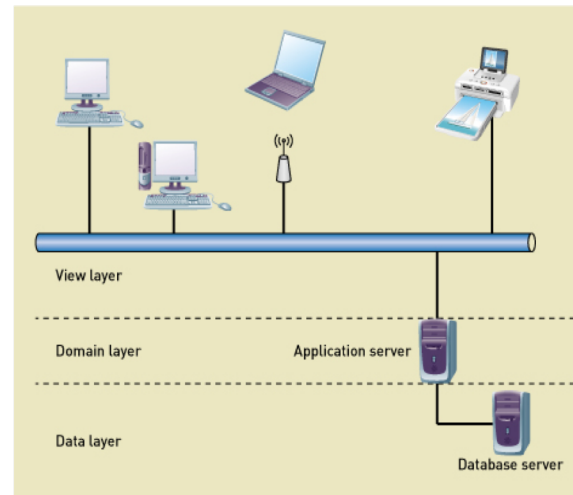
# Typical models for defining application components



Package diagram



Component diagram



Deployment diagram



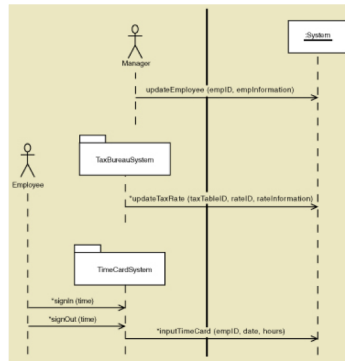
# Design the User Interface

- Importance of user interface design:  
*To the user, the UI is the system.*
- The user interface has large impact of user productivity
- Includes both Analysis and Design tasks
  - Requires heavy user involvement
- Current needs require multiple user interfaces
  - Many different devices and environments

8 numbered screenshots of the Ridgeline Mountain Outfitters user interface showing the checkout process:

1. Navigation menu with links: Browse, Share, Cart, Orders, Account. A sub-menu for 'Cart' shows 'View Empty' and 'Check out'.
2. Login prompt: 'You need to log in. Please enter your e-mail address or account number.' with an input field containing 'nwells22@gmail.com'.
3. Account confirmation: 'Please confirm account information' with fields for Name, Address, and City. Buttons: 'That's me', 'That's not me'.
4. Order summary: Table with columns Qty, SKU, Description, Price, Status. Items: 1 10967335 Toddler parka red 44.95 in-stock, 1 94462 Ladies parka blue 72.95 in-stock. Shipping options: Free - UPS ground (3-5 days), \$35.00 - UPS next day, \$20.00 - UPS two days, \$11.70 - USPS parcel post (5-7 days).
5. Shipping address confirmation: 'Please confirm shipping address' with fields for Name, Address, and City. Buttons: 'OK', 'Use another address'.
6. Shipping address input: 'Please enter shipping address' with fields for Name, Address, City, State, Zip Code. Buttons: 'OK', 'Cancel'.
7. Order summary and payment: Table with columns Qty, SKU, Description, Price, Ext. Items: 1 10967335 Toddler parka red 44.95 44.95, 1 94462 Ladies parka blue 72.95 72.95. Subtotal: 117.90, Shipping: 0.00, Sales Tax: 7.66, Total: \$125.56. Payment method: Visa \*\*\*\*-xxxx-xxxx-0899. Buttons: 'OK', 'Another method'.
8. Payment confirmation: 'Your payment has been approved. Your Visa credit card (xxxx-xxxx-xxxx-0899) has been charged for \$125.56. Your order number is 6773823. The order will be shipped today for delivery in 3-5 days. Thank you shopping with RMO!'.

Storyboard



System sequence diagram

Small screen menu prototype:

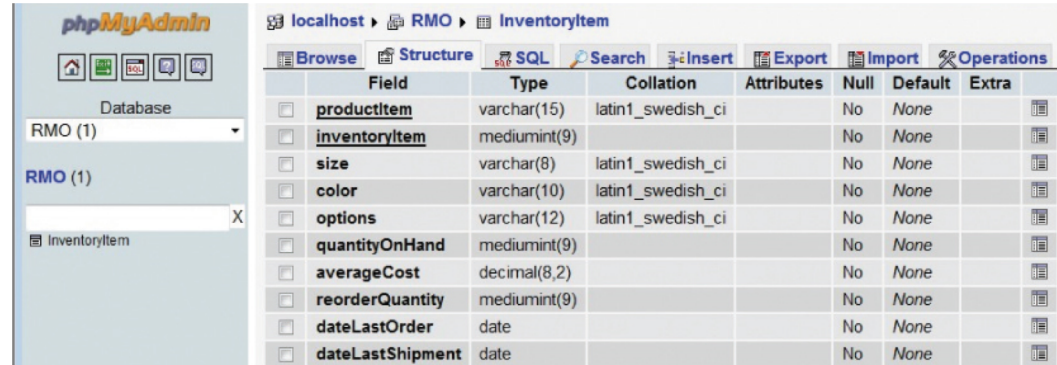
- Home | Stores | Log In | Cart
- Search
- Shop for Clothing
  - Women's Apparel >
  - Men's Apparel >
  - Kids' Apparel >
  - Footwear >
  - Accessories >
  - Sale & Clearance >
- Shop for Gear
- Wish List
- My Account

Small screen menu prototype



# Design the Database

- By definition, an Information System requires data – usually in a database
- Current technology frequently use Relational Database Management Systems (RDBMS)
- Converts the data model (i.e. domain model class diagram) to a relational database
- Requires addressing of many other technical issues
  - Throughput and response time
  - Security



The screenshot shows the phpMyAdmin interface. On the left, the 'Database' dropdown is set to 'RMO (1)', and the 'Table' dropdown is set to 'InventoryItem'. The main panel displays the 'Structure' tab for the 'InventoryItem' table. The table has the following fields:

	Field	Type	Collation	Attributes	Null	Default	Extra
<input type="checkbox"/>	<u>productItem</u>	varchar(15)	latin1_swedish_ci		No	None	
<input type="checkbox"/>	<u>inventoryItem</u>	mediumint(9)			No	None	
<input type="checkbox"/>	size	varchar(8)	latin1_swedish_ci		No	None	
<input type="checkbox"/>	color	varchar(10)	latin1_swedish_ci		No	None	
<input type="checkbox"/>	options	varchar(12)	latin1_swedish_ci		No	None	
<input type="checkbox"/>	quantityOnHand	mediumint(9)			No	None	
<input type="checkbox"/>	averageCost	decimal(8,2)			No	None	
<input type="checkbox"/>	reorderQuantity	mediumint(9)			No	None	
<input type="checkbox"/>	dateLastOrder	date			No	None	
<input type="checkbox"/>	dateLastShipment	date			No	None	



# Design Software Classes and Methods

- Extends the analysis models to incorporate software-specific elements
- Resulted models:
  - Design Class Diagram
  - Sequence Diagrams
  - State-Machine Diagrams

