

SYSTEMS DESIGN

Systems Analysis & Design

Author: Eng. Carlos Andrés Sierra, M.Sc.
cavirguezs@udistrital.edu.co

Lecturer
Computer Engineering
School of Engineering
Universidad Distrital Francisco José de Caldas

2025-I



UNIVERSIDAD DISTRITAL
FRANCISCO JOSÉ DE CALDAS

Outline

1 Requirements Engineering

2 Design & Process



Outline

1 Requirements Engineering

2 Design & Process



Stakeholders Vs. Shareholders

- **Stakeholders** are **individuals** or **groups** who have an **interest** in the **success** of a **project**.
- **Shareholders** are **individuals** or **groups** who have an **ownership** interest in a **company**.
- **Stakeholders** can be **internal** or **external** to a **company**. For example, **customers**, **employees**, **suppliers**, and **regulators** are **external** stakeholders.
- **Shareholders** are **internal** to a **company**. For example, **investors**, **owners**, and **managers** are **internal** stakeholders.



Stakeholders Vs. Shareholders

- **Stakeholders** are **individuals** or **groups** who have an **interest** in the **success** of a **project**.
- **Shareholders** are **individuals** or **groups** who have an **ownership interest** in a **company**.
- **Stakeholders** can be **internal** or **external** to a **company**. For example, **customers**, **employees**, **suppliers**, and **regulators** are **external stakeholders**.
- **Shareholders** are **internal** to a **company**. For example, **investors**, **owners**, and **managers** are **internal stakeholders**.



Stakeholders Vs. Shareholders

- **Stakeholders** are **individuals** or **groups** who have an **interest** in the **success** of a **project**.
- **Shareholders** are **individuals** or **groups** who have an **ownership interest** in a **company**.
- **Stakeholders** can be **internal** or **external** to a **company**. For example, **customers**, **employees**, **suppliers**, and **regulators** are **external** stakeholders.
- **Shareholders** are **internal** to a **company**. For example, **investors**, **owners**, and **managers** are **internal** stakeholders.



Stakeholders Vs. Shareholders

- **Stakeholders** are **individuals** or **groups** who have an **interest** in the **success** of a **project**.
- **Shareholders** are **individuals** or **groups** who have an **ownership interest** in a **company**.
- **Stakeholders** can be **internal** or **external** to a **company**. For example, **customers**, **employees**, **suppliers**, and **regulators** are **external** stakeholders.
- **Shareholders** are **internal** to a **company**. For example, **investors**, **owners**, and **managers** are **internal** stakeholders.



Requirements

- **Requirements** are **statements** that **describe** the **features**, **functions**, and **constraints** of a **system**.
- Requirements are used to communicate the needs of stakeholders to developers.
- Requirements are used to guide the design, development, and testing of a system.



Requirements

- **Requirements** are **statements** that **describe** the **features**, **functions**, and **constraints** of a **system**.
- **Requirements** are **used** to **communicate** the **needs** of **stakeholders** to **developers**.
- Requirements are used to guide the design, development, and testing of a system.



Requirements

- **Requirements** are **statements** that **describe** the **features**, **functions**, and **constraints** of a **system**.
- **Requirements** are **used** to **communicate** the **needs** of **stakeholders** to **developers**.
- **Requirements** are **used** to **guide** the **design**, **development**, and **testing** of a **system**.



User Stories

- **User stories** are short, simple descriptions of a feature or function of a system.
- They are written from the perspective of the user and describe what the user wants to achieve.
- They are used to capture the requirements of a system in a simple and understandable way.



User Stories: Format Example

User Story

Title:	Priority:	Estimate:
User Story: As a [description of user], I want [functionality] so that [benefit].		
Acceptance Criteria: Given [how things begin] When [action taken] Then [outcome of taking action]		

 ProductPlan

What is Requirements Engineering?

- **Requirements engineering** is the **process** of **eliciting**, **analyzing**, **specifying**, **validating**, and **managing** the **requirements** of a **system**.
- It is a **critical activity** in the **systems development lifecycle** that **ensures** that the **system** meets the **needs** of its **users**.
- It is a **collaborative process** that **involves stakeholders** from **different backgrounds** and **perspectives**.



What is Requirements Engineering?

- **Requirements engineering** is the **process** of **eliciting**, **analyzing**, **specifying**, **validating**, and **managing** the **requirements** of a **system**.
- It is a **critical activity** in the **systems development lifecycle** that **ensures** that the **system** meets the **needs** of its **users**.
- It is a **collaborative process** that **involves stakeholders** from **different backgrounds** and **perspectives**.



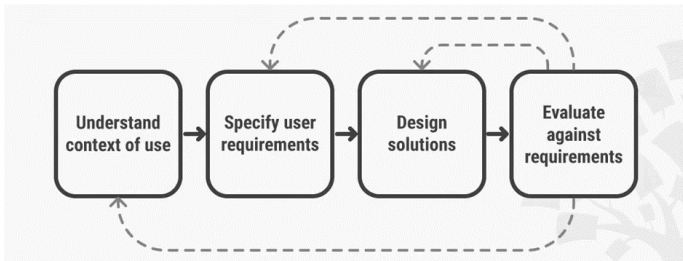
What is Requirements Engineering?

- **Requirements engineering** is the **process** of **eliciting**, **analyzing**, **specifying**, **validating**, and **managing** the **requirements** of a **system**.
- It is a **critical activity** in the **systems development lifecycle** that **ensures** that the **system** meets the **needs** of its **users**.
- It is a **collaborative process** that **involves stakeholders** from **different backgrounds** and **perspectives**.



User Centered Design(UCD)

- **User-centered design** (UCD) is an **iterative** design process that focuses on **understanding** the **needs**, **preferences**, and **behaviors** of users.
- UCD is a **collaborative** process that **involves** users in the design and development of a system.
- UCD is used to create systems that are **usable**, **efficient**, and **satisfying** to users.



Requirements Engineering Process

The **requirements engineering** process **consists** of the **following activities**:

- **Gathering** requirements.
- **Analyzing** requirements.
- **Validating** requirements.
- **Verifying** requirements.
- **Documenting** requirements.
- **Managing** requirements.
- **Communicating** requirements.



Gathering Requirements

- **Gathering** requirements is the **process** of **collecting** and **documenting** the **needs** of **stakeholders**.
- It involves **interviewing** stakeholders, **conducting** surveys, and **observing** users to understand their **requirements**.
- It is a **collaborative process** that **involves** stakeholders from **different backgrounds** and **perspectives**.



Gathering Requirements

- **Gathering** requirements is the **process** of **collecting** and **documenting** the **needs** of **stakeholders**.
- It involves **interviewing** stakeholders, **conducting** surveys, and **observing** users to **understand** their **requirements**.
- It is a **collaborative process** that **involves** stakeholders from **different backgrounds** and **perspectives**.



Gathering Requirements

- **Gathering** requirements is the **process** of **collecting** and **documenting** the **needs** of **stakeholders**.
- It involves **interviewing** stakeholders, **conducting** surveys, and **observing** users to **understand** their **requirements**.
- It is a **collaborative process** that **involves** stakeholders from **different backgrounds** and **perspectives**.



Clients are not always right

Dear Santa,
 How are you? I'm good.
 Here is what I want for
 Christmas.

http://www.amazon.com/gp/product/B0032HFG0M/ref=ssq_hps_bw_g21_ir03?pf_rd_m=ATVPDKIKXODER&pf_rd_s=center-3&pf_rd_e=gxwy42FH2K03Y7BMWQNM&pf_rd_t=101&pf_rd_p=1328901542&pf_rd_i=165379



Analyzing Requirements

- **Analyzing** requirements is the **process** of **examining** and **understanding** the **requirements** of a **system**.
- It involves identifying dependencies, conflicts, and inconsistencies in the requirements.
- It is a critical activity that ensures that the requirements are complete, consistent, and correct.



Analyzing Requirements

- **Analyzing** requirements is the **process** of **examining** and **understanding** the **requirements** of a **system**.
- It involves **identifying dependencies**, **conflicts**, and **inconsistencies** in the **requirements**.
- It is a **critical activity** that **ensures** that the **requirements** are **complete**, **consistent**, and **correct**.



Analyzing Requirements

- **Analyzing** requirements is the **process** of **examining** and **understanding** the **requirements** of a **system**.
- It involves **identifying** **dependencies**, **conflicts**, and **inconsistencies** in the **requirements**.
- It is a **critical activity** that **ensures** that the **requirements** are **complete**, **consistent**, and **correct**.



Documenting Requirements

- **Documenting** requirements is the **process** of **writing** and **organizing** the **requirements** of a **system**.
- It involves creating documents, diagrams, and models that describe the requirements in a clear and concise way.
- It is a collaborative process that involves stakeholders from different backgrounds and perspectives.



Documenting Requirements

- **Documenting** requirements is the **process** of **writing** and **organizing** the **requirements** of a **system**.
- It involves **creating documents**, **diagrams**, and **models** that **describe** the **requirements** in a **clear** and **concise** way.
- It is a **collaborative process** that **involves stakeholders** from **different backgrounds** and **perspectives**.



Documenting Requirements

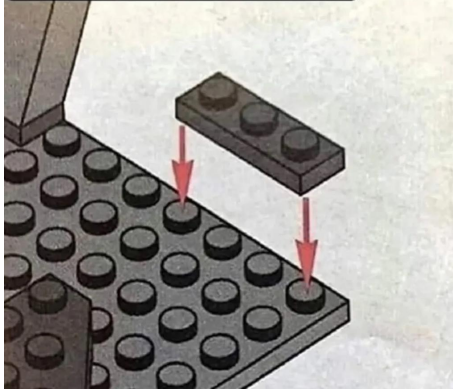
- **Documenting** requirements is the **process** of **writing** and **organizing** the **requirements** of a **system**.
- It involves **creating documents**, **diagrams**, and **models** that **describe** the **requirements** in a **clear** and **concise** way.
- It is a **collaborative process** that **involves** stakeholders from **different backgrounds** and **perspectives**.



Everyone hates to write Documentation

**En la documentación
está todo bien explicado**

La documentación:



Validating Requirements

- **Validating** requirements is the **process** of **ensuring** that the **requirements** are **correct** and **complete**.
- It involves **reviewing** the **requirements** with **stakeholders** to **verify** that they **meet** their **needs**.
- It is a **collaborative process** that **involves** stakeholders from **different backgrounds** and **perspectives**.



Validating Requirements

- **Validating** requirements is the **process** of **ensuring** that the **requirements** are **correct** and **complete**.
- It involves **reviewing** the **requirements** with **stakeholders** to **verify** that they **meet** their **needs**.
- It is a **collaborative process** that **involves stakeholders** from **different backgrounds** and **perspectives**.



Validating Requirements

- **Validating** requirements is the **process** of **ensuring** that the **requirements** are **correct** and **complete**.
- It involves **reviewing** the **requirements** with **stakeholders** to **verify** that they **meet** their **needs**.
- It is a **collaborative process** that **involves** stakeholders from **different backgrounds** and **perspectives**.



NOT Clear Understanding of Requirements



Dad Jokes
@Dadsaysjokes

..

My dad told me his password is:
MickeyMinnieGoofyDonaldPlutoHuey
LouieDeweyDublin.

Because he was told his password
had to contain 8 characters and at
least one Capital.



Verifying Requirements

- **Verifying** requirements is the **process** of **ensuring** that the **requirements** are **correctly implemented** in the **system**.
- It involves **testing** the **system** to **verify** that it **meets** the **requirements**.
- It is a **critical activity** that **ensures** that the **system** **meets** the **needs** of its **users**.



Verifying Requirements

- **Verifying** requirements is the **process** of **ensuring** that the **requirements** are **correctly implemented** in the **system**.
- It involves **testing** the **system** to **verify** that it **meets** the **requirements**.
- It is a **critical activity** that **ensures** that the **system** **meets** the **needs** of its **users**.



Verifying Requirements



- **Verifying** requirements is the **process** of **ensuring** that the **requirements** are **correctly implemented** in the **system**.
- It involves **testing** the **system** to **verify** that it **meets** the **requirements**.
- It is a **critical activity** that **ensures** that the **system meets** the **needs** of its **users**.



Typical Mistakes when Testing

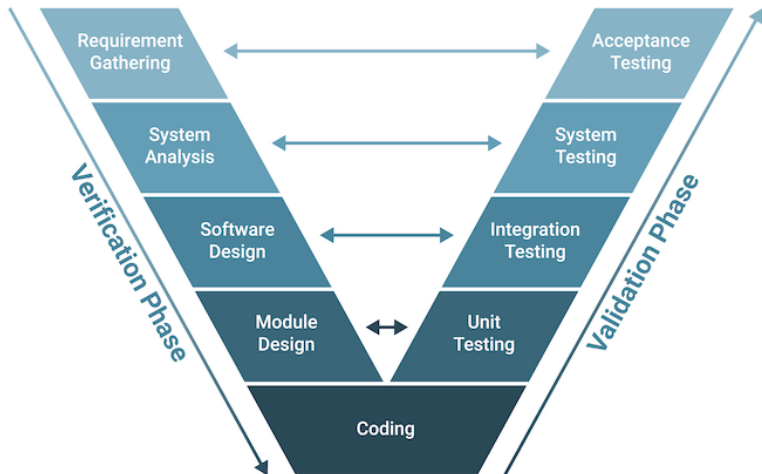
Disturbing Chinese calorie app...



	カシューナッツ (cashew)	1粒	9 kcal
	ジャムパン (Pomeranian)	1個	327 kcal



V-Model inn SDLC



Outline

1 Requirements Engineering

2 Design & Process



Conceptual Design

- Once the initial set of **requirements** are defined, the next step is to create a **conceptual design** of the system.
- **Conceptual Design** is a **high-level design** that defines the structure and behavior of the system. It is achieved by the recognition of the appropriate **components**, **connections**, and **responsibilities**.
- The **conceptual design** is used to **communicate the vision** of the system to **stakeholders** and to **guide the development** of the system.



Conceptual Design

- Once the initial set of **requirements** are defined, the next step is to create a **conceptual design** of the system.
- **Conceptual Design** is a **high-level design** that defines the structure and behavior of the system. It is achieved by the recognition of the appropriate **components**, **connections**, and **responsibilities**.
- The **conceptual design** is used to **communicate** the **vision** of the system to **stakeholders** and to **guide** the **development** of the system.



Process Definition

- A **Process** is a **series** of steps or actions taken to achieve a particular end.
- **Processes** are used to **organize** and **manage** work.



Workflows

- A **Workflow** is a **series** of tasks that are performed in a specific order to achieve a goal.
- **Workflows** are used to **automate** and **optimize** business processes.
- **Workflows** can be **sequential**, **parallel**, **conditional**, or **repetitive**.

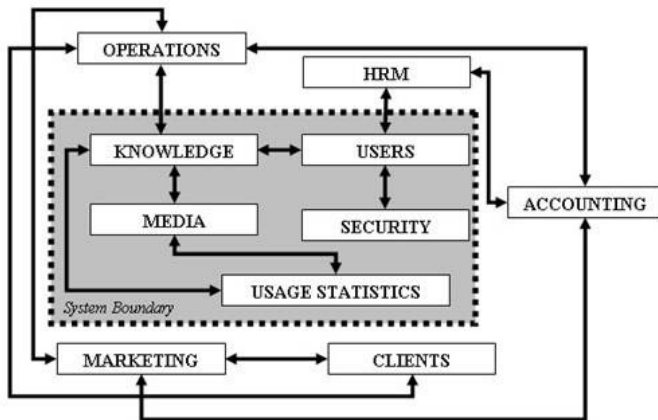


Process Models

- A **Process Model** is a representation of a **process** that shows the sequence of steps and the **relationships** between them.
- **Process models** are used to **analyze**, **design**, and **improve** processes.
- Examples of **process models** include flowcharts, data flow diagrams, activity diagrams, business process model and notation (BPMN), petri nets, state diagrams, among others.



System Schema Example: Company Structure



Causal Loops

- A **Causal Loop** is a **diagram** that shows the **relationships** between different variables in a system.
- Causal loops are used to **analyze** and **understand** the **dynamics** of a system.
- Causal loops can be **positive** or **negative**.

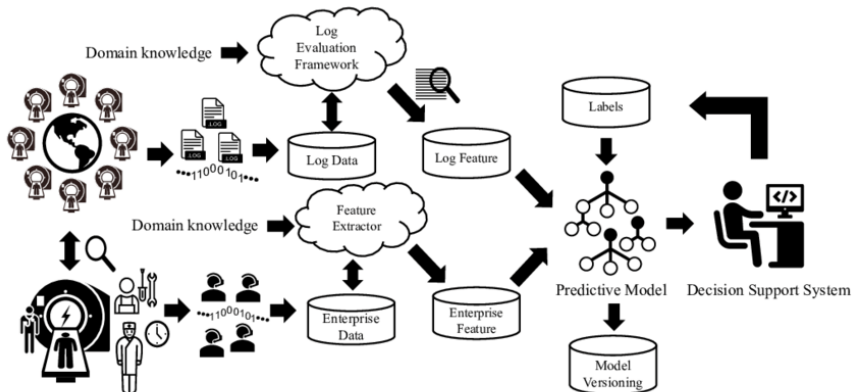


Causal Loops

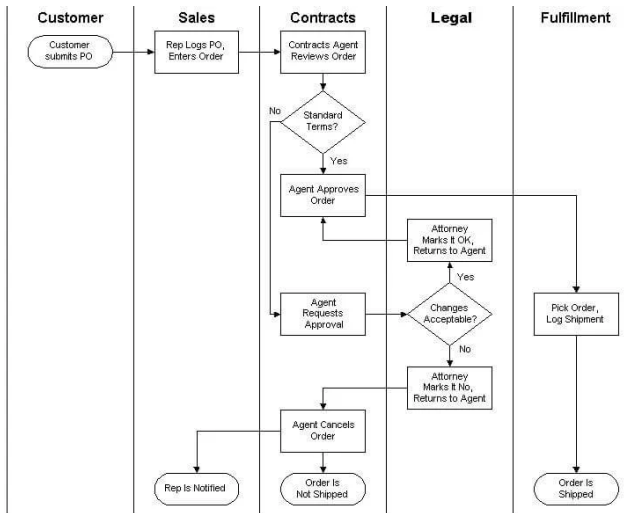
- A **Causal Loop** is a **diagram** that shows the **relationships** between different variables in a system.
- Causal loops are used to **analyze** and **understand** the **dynamics** of a system.
- Causal loops can be **positive** or **negative**.



System Schema Example: Processing Pipeline



Business Process Model and Notation (BPMN)



Technical Design

- Once the **conceptual design** of the system is defined, the next step is to create a **technical design** of the system.
- **Technical Design** is a **detailed design** that defines the architecture, components, and interfaces of the system.
- The **technical design** is used to **guide the development** of the system and to **communicate the implementation details to developers**.

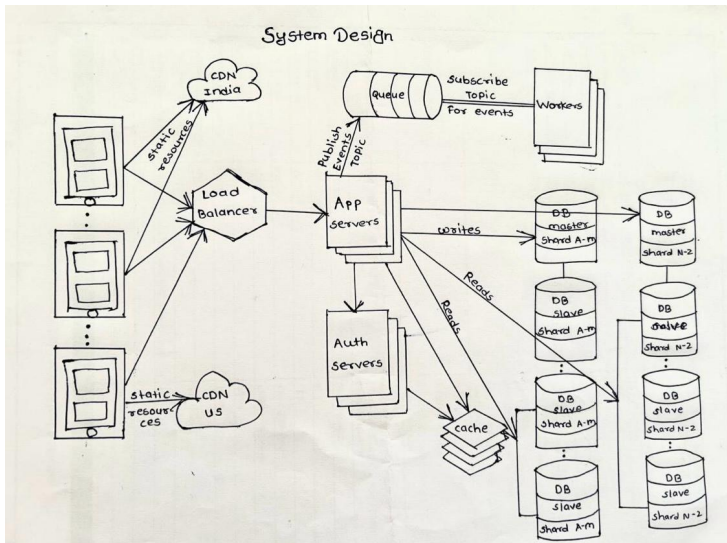


Technical Design

- Once the **conceptual design** of the system is defined, the next step is to create a **technical design** of the system.
- **Technical Design** is a **detailed design** that defines the architecture, components, and interfaces of the system.
- The **technical design** is used to **guide** the **development** of the system and to **communicate** the **implementation** details to **developers**.



Systems Design applied to Software Architectures



Outline

1 Requirements Engineering

2 Design & Process



Thanks!

Questions?



Repo: <https://github.com/EngAndres/ud-public/tree/main/courses/systems-analysis>

