Software Engineering

WEEK 02 LECTURE 01

TYPES OF REQUIREMENTS

WE CAN DIVIDE REQUIREMENT IN TO FIVE TYPES

- Functional requirements
  - Business requirements
  - Business rules
  - User requirements
- Non-functional requirements
- External interfaces
- Physical product Requirements
- Development constraints

FUNCTIONAL REQUIREMENTS

- **Functional requirements** are those behaviors that the developed product should do or support.
- These are usually expressed as inputs into the product, outputs of the product, or a description of the behavior itself.

FUNCTIONAL REQUIREMENTS

- For example, input might be data related to a user's online purchase, including name, address, item purchased, and payment information.
- The output in this scenario might be a notification email after the transaction is performed.

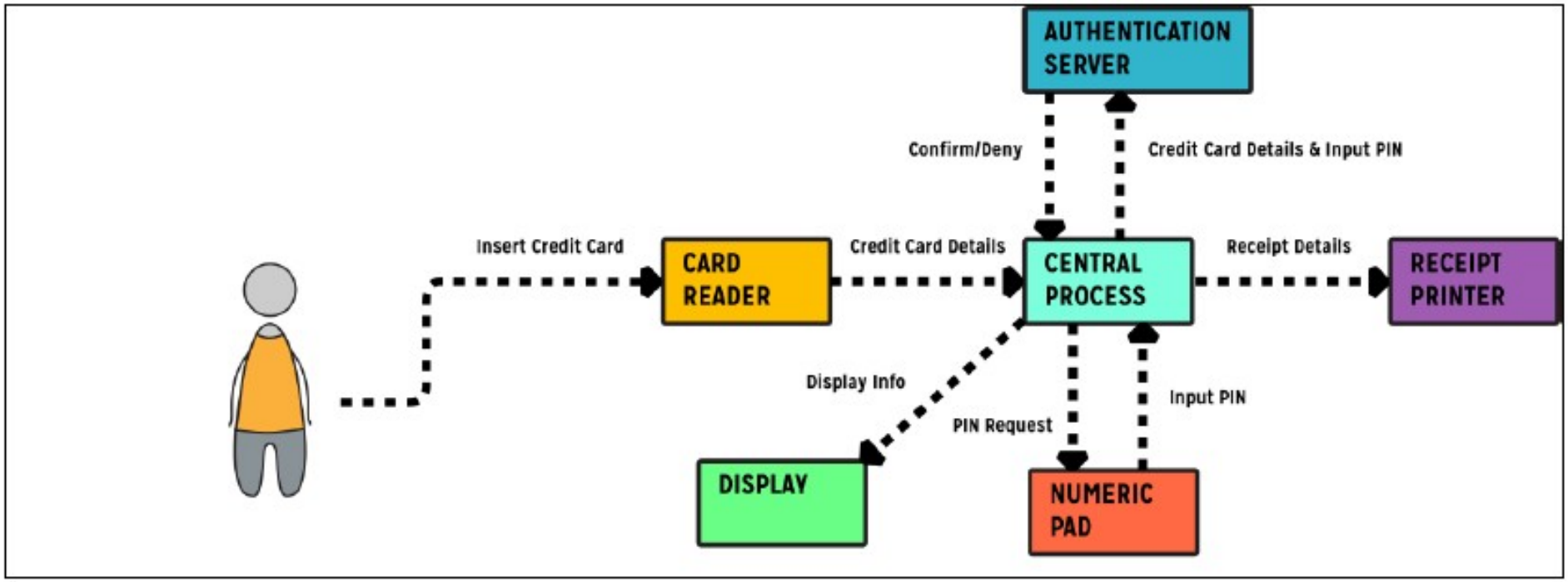
FUNCTIONAL REQUIREMENTS

- **Information flow diagrams** are a graphical technique commonly used to show how data flows throughout Client Needs and Software the system and the dependencies of all the system components.

FUNCTIONAL REQUIREMENT: **INFORMATION FLOW DIAGRAM**

- The following is an example of an information flow diagram for a customer using a credit card with a product, and where the credit card information is verified before a receipt is generated for the user.

FUNCTIONAL REQUIREMENT: INFORMATION FLOW DIAGRAM



- Functional requirements
  - Business requirements
  - Business rules
  - User requirements
- Non-functional requirements
- External interfaces
- Physical product Requirements
- Development constraints

BUSINESS REQUIREMENTS

- **Business Requirements** refer to those requirements that involve the purpose of the project.
- For Example, The client needs to reduce errors in orders made to their company by 25% by the end of next year to raise their yearly revenue by $10,000

BUSINESS RULES

- Business requirements should not be confused with **business rules**, although they are often associated.
- Business requirements deal with why the project was pursued, while **business rules are constraints** on how the product will function

BUSINESS RULE

- Business rules are sometimes necessary to make a project appropriate or successful.
- They are often budgets, policies, guidelines, or regulations

BUSINESS RULE

- Examples of business rules include:
  - Government or legal regulations
  - Privacy policies
  - Brand uniformity requirements
  - Student shall struck off from role if GPA Less than 2.0 is two continuous semester

USER REQUIREMENTS

- **Users** or **end-users** are the people who will use the software once it has been created.
- **User requirements** are the tasks that these users can accomplish with the product, or what the product can do for the user.

USER REQUIREMENT

- They are part of the core functionality of the product.
- For this reason, determining user requirements is usually very time consuming

USER RRQUIREMENTS

- There are many methods for expressing user requirements, including:
  - Use cases.
  - User stories.
  - Storyboards.
  - Scenarios.

- Functional requirements
  - Business requirements
  - Business rules
  - User requirements
- Non-functional requirements
- External interfaces
- Physical product Requirements
- Development constraints

NON FUNCTIONAL REQUIREMENTS

- In addition to functional requirements, there are **non-functional requirements** that describe how well a product must perform.
- They are also known as quality requirements for this reason.

NON-FUNCTIONAL REQUIREMENT

- Non-functional requirements address product issues including accuracy, dependability, security, usability, efficiency, performance, and maintainability.

NON-FUNCTIONAL REQUIREMENTS

- Drawing on the online purchasing example used in functional requirements, a non-functional requirement in the same scenario would be that emails should be delivered to users within two hours of purchase.

- Non-functional requirements are usually complementary to functional requirements.

EXTERNAL INTERFACES

- **External interfaces requirements** refer to those requirements related to how the product is situated within a larger system.

EXTERNAL INTERFACES

- In other words, external interfaces do not concern themselves with the physical environment of the product, but rather they are concerned with the relationship of the product to other entities outside the product.

EXTERNAL INTERFACES

- For example, a software application that retrieved information from a remote database to display to users sits between the entities of the database and the end-user.

EXTERNAL INTERFACES

- For example, a software application that retrieved information from a remote database to display to users sits between the entities of the database and the end-user.An external interface is used between each one.

PHYSICAL REQUIREMENTS

- **Physical setting requirements** refer to how the product needs to be designed in order to function in its physical environment.

PHYSICAL REQUIREMENTS

- For example, if a product was being developed for underwater exploration, it would need to be waterproof.

DEVELOPMENT CONSTRAINTS

- **Development constraints** affect everything from implementation technology, conventions, documentation, and the process used to develop the product.

DEVELOPMENT CONSTRAINTS

- They generally refer to constraints related to creating the product, such as which devices or platforms will be supported, or how much memory, bandwidth, or processing power the product is limited to using.

DEVELOPMENT CONSTRAINTS

- Similarly, if the product was designed for use in the desert, or in Antarctica, it would need to withstand and function in those environments.