

<b>Use case Name:</b>	Dealer set up auction
<b>Primary Actor:</b>	Dealer
<b>Stakeholders and Interests:</b>	Dealer-wants to set Up an auction
<b>Brief Description:</b>	This Use case describe how dealer set up an auction in system
<b>Importance Level:</b>	Essensial

**Normal Flow of Event:**

Dealer	System
The Use case begins when Dealer need start new auction	The system presents an option to create a new auction.
Dealer enters auction details	
Dealer adds items detail to the auction	
Dealer choose bidding rules	
Dealer reviews auction details	
Dealer confirms the setup or makes modifications.	The system activates the auction, making it visible to potential bidders.

**Alternative Flows**

If any required Auction details are missing, the system prompts the dealer to complete the required fields.

Auction Detail :	Auction title Start and end date/time Type of auction (e.g., open, sealed bid) Minimum bid amount (if applicable)
------------------	--

Item Detail :	description, images, starting price
Bidding rules :	bid increments, reserve price, and bid validation rules.

**1. What are the different elements of a use-case description?**

A use-case description typically includes the following elements:

**Use Case Name:** A descriptive title for the use case.

**Actor(s):** The user(s) or external systems that interact with the system.

**Goal:** What the actor wants to achieve.

**Trigger:** The event that starts the use case.

**Preconditions:** Conditions that must be true before the use case begins.

**Postconditions:** Conditions that will be true after the use case ends.

**Basic Flow (Main Flow):** The normal sequence of steps from start to successful completion.

**Alternative Flows:** Optional or different paths the interaction could take.

**Exception Flows:** Paths taken when an error or unexpected situation occurs.

## **2. What do we mean by the 'basic/normal flow' of a use case description?**

The basic flow (also called the normal or main flow) is the standard, successful sequence of steps that the system and user follow when everything goes as expected. It describes the ideal scenario where no errors or exceptions occur.

Example: A customer logs in → browses products → adds items to cart → makes payment → order confirmation.

## **3. Why might we need additional, alternative / exception flows?**

We need alternative and exception flows to represent different or unexpected situations that can occur during the use case.

**Alternative flows** handle variations in user behavior or choices.

Example: The user chooses to check out as a guest instead of logging in.

**Exception flows** deal with errors or problems.

Example: Payment fails due to insufficient funds.

Including these flows helps to build a complete, realistic, and robust system by preparing for both expected and unexpected scenarios.