Waterfall Model for a Sports Event

The Waterfall model is a sequential design process, where progress flows steadily downwards (like a waterfall) through the phases of conception, initiation, analysis, design, construction, testing, production/implementation, andmaintenance.

1. **Requirements Gathering & Analysis (Conception/Initiation)**

* **System Requirements (Technical):**
* **Performance:** Handle high user traffic and event data.
* **Security:** Secure data storage and transactions.
* **Integration:** Connect with payment and ticketing systems.
* **Database:** Manage event details, registrations, and results.
* **Authentication:** Secure user login and role management.
* **Cross-Platform Support:** Accessible via web and mobile.
* **Event Management:** Tools for creating and updating event schedules.
* **Notifications:** Inform users of event changes or updates.
* **User Requirements (End-User Needs):**
* **Registration:** Users can register and pay for events.
* **Event Search:** Users can browse and filter events.
* **Live Updates:** Real-time event scores and results.
* **Notifications:** Alerts for schedule changes or cancellations.
* **Profile Management:** Users can manage personal details and track performance.
* **Ticketing:** Spectators can book tickets for events.
* **Results:** Users can view detailed event results and statistics.
* **Action:** "Plan event"
  + **Description:** This is the initial stage where the club identifies the need for a sports event. They define the event's purpose, goals, and scope. This phase includes brainstorming and initial planning.
* **Action:** "Choose sport"
  + **Description:** Based on the club's resources, member interests, and available facilities, a specific sport is selected for the event.

**2. System Design (Analysis/Design)**

* **Action:** "Select date and time"
  + **Description:** The club determines the most suitable date and time for the event, considering member availability and facility schedules.
* **Action:** "Notify members"
  + **Description:** The club communicates the event details (sport, date, time) to its members, inviting them to participate.
* **Action:** "Arrange equipment"
  + **Description:** The club identifies and secures the necessary sports equipment for the event.
* **Action:** "Arrange refreshments"
  + **Description:** The club plans and organizes refreshments for participants, considering dietary needs and preferences.
* **Action:** "Set up venue"
  + **Description:** The club prepares the venue, including setting up the playing area, seating, and any other necessary arrangements.

**3. Implementation (Construction)**

* **Action:** "Sign up for event" (Member's Action)
  + **Description:** Members register their participation in the event. This is the member's "implementation" of their involvement.
* **Action:** "Prepare for event" (Member's Action)
  + **Description:** Members prepare themselves for the event, which might include practice, gathering personal equipment.

**4. Testing (Testing)**

* **Action:** "Start event"
  + **Description:** The event officially begins, and the organized activities commence.
* **Action:** "Conduct games"
  + **Description:** The actual sporting activities are carried out as planned. This is the core "testing" phase, where the event's execution is evaluated.

**5. Deployment (Production/Implementation)**

* **Action:** "Price and certificate distribution"
  + **Description:** This phase involves the distribution of prizes and certificates to the winners and participants, marking the completion of the event's formal activities.

**6. Maintenance (Maintenance)**

* **Action:** "Thank participants"
  + **Description:** The club expresses gratitude to the participants for their involvement and gathers feedback for future events. This is the "maintenance" phase, focusing on post-event activities and improvements.

**Important Considerations for the Waterfall Model in this Context:**

* **Strict Sequencing:** The Waterfall model emphasizes a strict sequence of phases. In reality, some overlap or iteration might occur, but the general flow remains linear.
* **Documentation:** Each phase should be documented, ensuring clarity and transparency throughout the process.
* **Feedback:** While the Waterfall model is less flexible to changes, feedback can be incorporated at the end of each phase to improve subsequent steps.

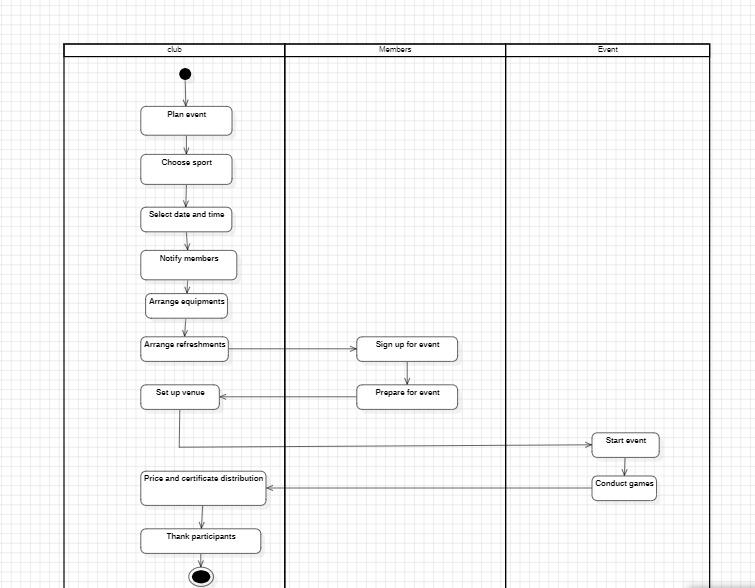
By structuring the sports event scenario using the Waterfall model, we gain a clear understanding of the project's progression and the tasks involved at each stage. This approach helps ensure a well-organized and successful event.

**Sports event explained using staruml diagrams:**

StarUML is a software design tool that uses UML (Unified Modelling Language) to visually represent the structure and behavior of a software system. This helps in understanding complex systems, identifying potential issues early on, and improving communication among developers and stakeholders.

**1.Activity diagram symbols and notations:**

| **Symbol** | **Name** | **Description** |
| --- | --- | --- |
| start Symbol | | Start symbol | Represents the beginning of a process or workflow in an activity diagram.  It can be used by  itself or with a note symbol that explains the starting point. |
| activity Symbol | Activity symbol | Indicates the activities that make up a modelled process. These symbols, which  include short descriptions within the shape, are the main building blocks of an  activity diagram. |
| connector Symbol | Connector symbol | Shows the directional flow, or control flow, of the activity. An incoming arrow  starts a step of an activity; once the step is completed, the flow continues with  the outgoing arrow. |
| joint Symbol | Joint symbol/ Synchronization bar | Combines two concurrent activities and re-introduces them to a flow where only  one activity occurs at a time. Represented with a thick vertical or horizontal line. |
| fork Symbol | Fork symbol | Splits a single activity flow into two concurrent activities. Symbolized with  multiple arrowed lines from a join. |
| decision Symbol | Decision symbol | Represents a decision and always has at least two paths branching  out with condition text to allow users to view options. This symbol represents  the branching or merging of various flows with the symbol acting as a  frame or container. |
| condition text | Condition text | Placed next to a decision marker to let you know under what condition  an activity flow should split off in that direction. |
| end symbol | End symbol | Marks the end state of an activity and represents the completion  of all flows of a process. |



## Use case diagram symbols and notation:

* **Use cases:** Horizontally shaped ovals that represent the different uses that a user might have.
* **Actors:** Stick figures that represent the people actually employing the use cases.
* **Associations:** A line between actors and use cases. In complex diagrams, it is important to know which actors are associated with which use cases.
* **System boundary boxes:** A box that sets a system scope to use cases. All use cases outside the box would be considered outside the scope of that system. For example, Psycho Killer is outside the scope of occupations in the chainsaw example found below.
* **Packages:** A UML shape that allows you to put different elements into groups. Just as with component diagrams, these groupings are represented as file folders.

## Use case diagram

B. MANASWINI HU22CSEN0100303 JYOTHI HU22CSEN0101042 N. ANJANA HU22CSEN0100218 VAIBHAV HU22CSEN0100831