ANIMAL SHELTER WEBSITE

SHIRISH SHETTY -2103164
MAITHALI SHINDE -2103166
ADARSH SHUKLA -2103167

EXPERIMENT NO 5

AIM: Develop Activity and State diagram for the project

THEORY:

State-Activity Diagram (SAD) is a graphical representation used in software engineering to depict the behavior and interactions within a system or software application. It provides insight into how different components of the system communicate and the sequence of activities that occur. SAD is particularly useful for communicating with developers, designers, and stakeholders involved in the software development process. It can be applied to both existing systems for analysis and proposed systems for design.

A State-Activity Diagram has four main components:

1. Activity: Activities represent processes or actions within the system. These are depicted as rectangular boxes, ovals, rectangles, or circles, just like in DFD. Each activity is named descriptively to convey its purpose or function within the system. Activities define what the system does and how it processes data.

2. State: States represent the different conditions or modes that the system can be in during its operation. States are often represented as circles or rounded rectangles. They describe the various situations or statuses that the system can transition between. States capture the current state of the system at a given point in time.

3. Transition: Transitions are depicted as arrows or lines and describe how the system moves from one state to another. Transitions indicate the flow of control or data between activities and states. They show the sequence in which activities and states are executed or entered.

4. Store: Similar to the data store in DFD, a store in the State-Activity Diagram represents where data or information is stored within the system. It can be temporary or permanent storage and is typically represented by two horizontal lines. Stores are used to depict data storage and retrieval operations within the system.

<u>Rules for creating a State-Activity Diagram:</u>

1. Clear Naming: Ensure that the names of activities and states are concise and easy to understand without additional explanations. This makes it easier for stakeholders to grasp the diagram's meaning.

2. Sequential Order: If needed, number activities or list them in a logical sequence to aid in understanding the flow of operations within the system.

3. Consistency: Maintain consistency in the representation of components across all levels of the diagram. This consistency helps in avoiding confusion and ensuring clarity.

4. Limit Processes: While there is no strict limit, it's generally advisable to keep the number of activities reasonable to maintain diagram readability. A single diagram might have several activities, but it's essential not to overwhelm the viewer.

Symbols Used in State-Activity Diagram:

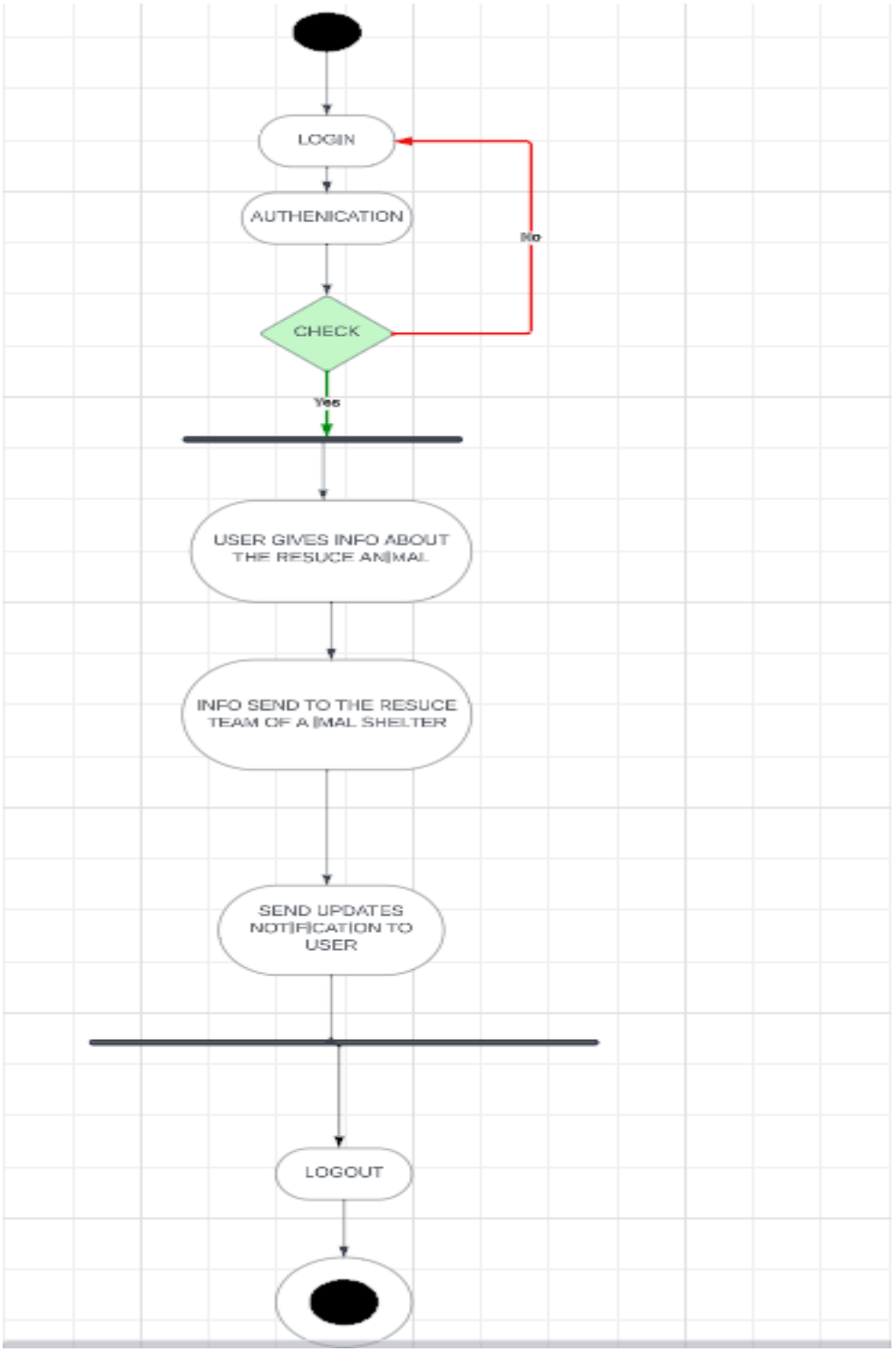
- Rectangle: Represents an activity or process within the system, similar to a DFD process symbol.

- Circle or Rounded Rectangle: Represents a state or condition that the system can be in, similar to DFD states like "data reading" or "data entry."

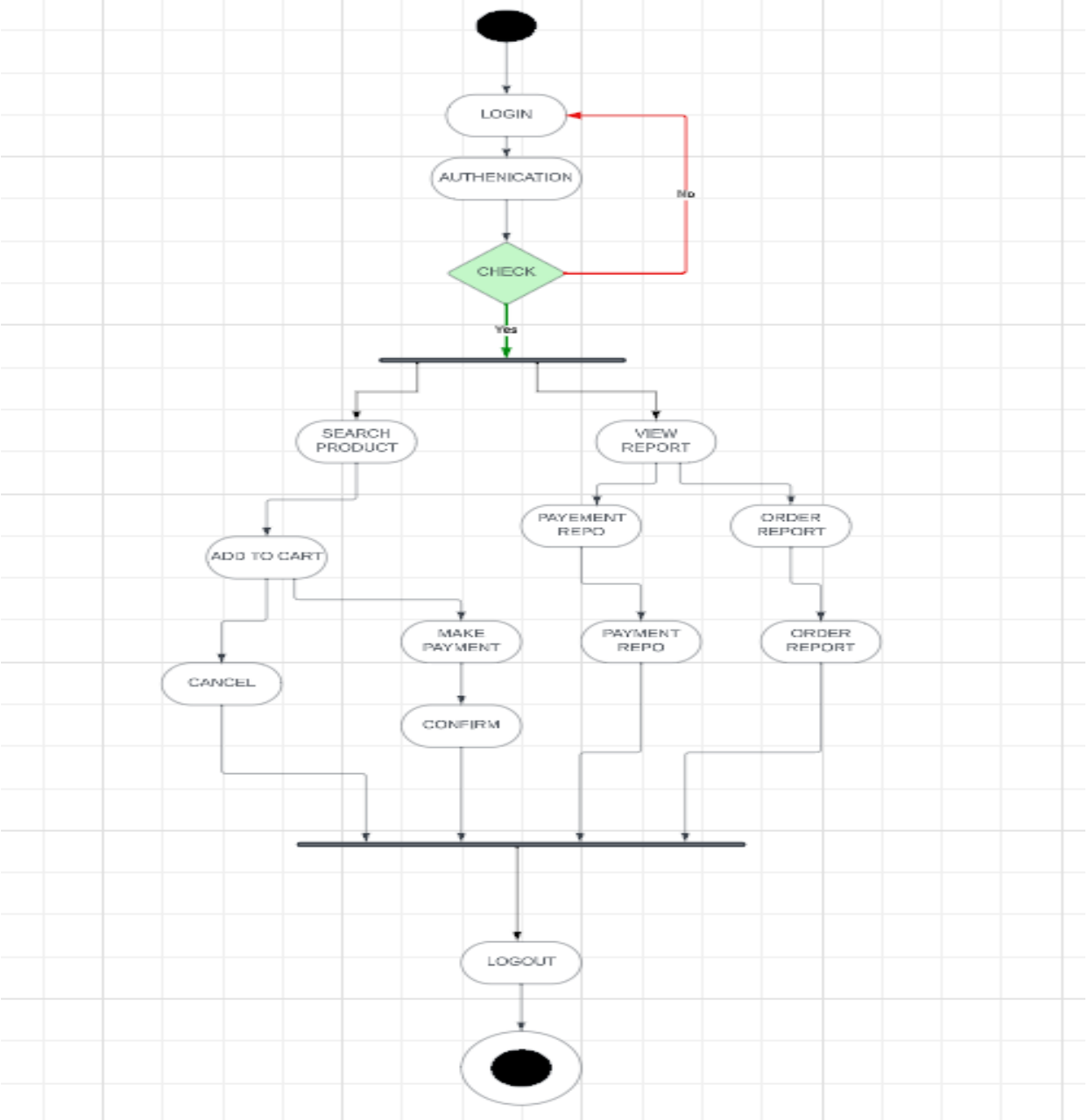
- Arrow or Line: Represents transitions or flow of control/data between activities and states, much like the arrow symbols in DFD.

- Two Horizontal Lines: Depicts a store where data can be temporarily or permanently stored, analogous to a DFD data store.

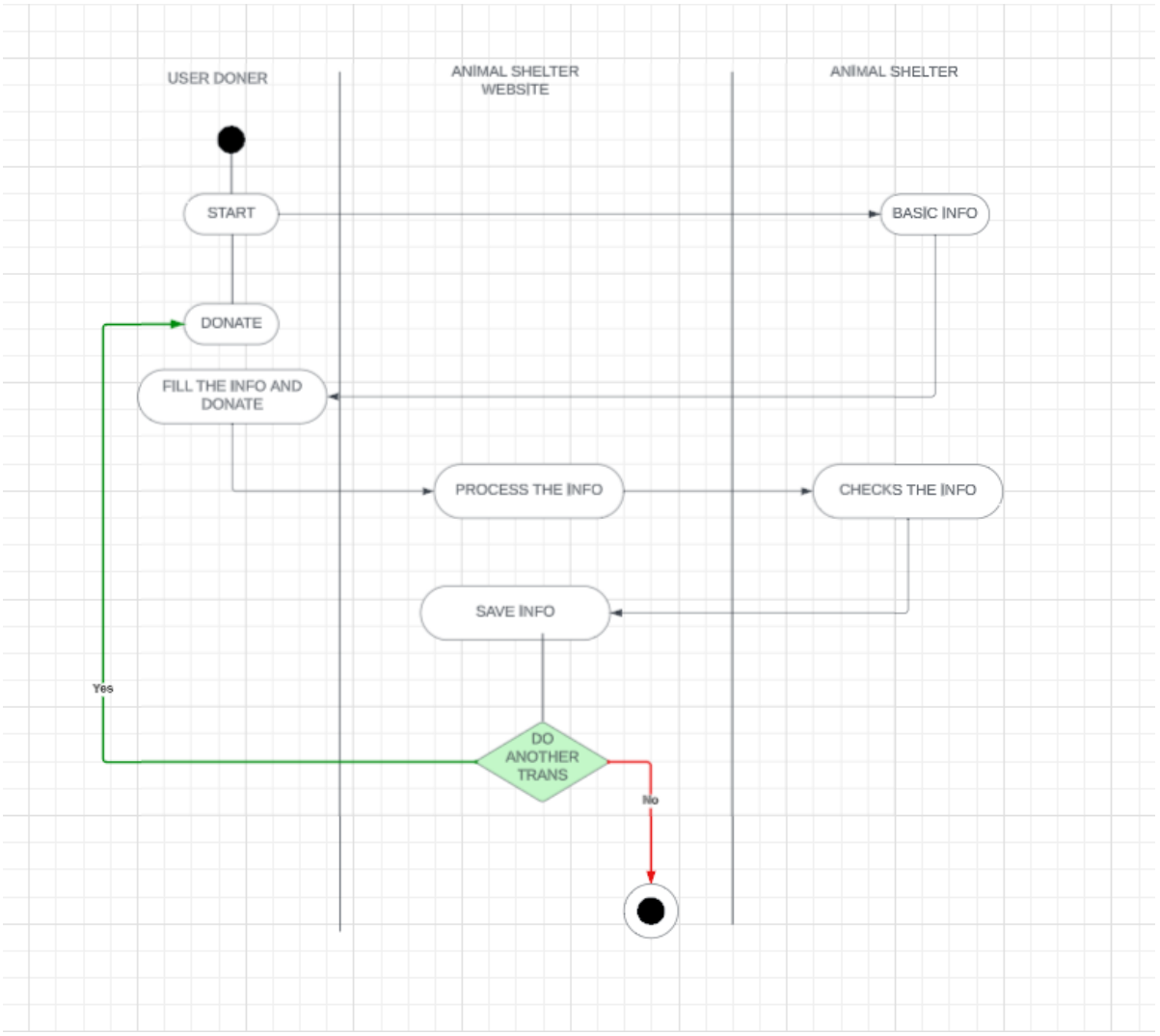
In summary, a State-Activity Diagram is a valuable tool in software engineering for modeling and visualizing the behavior and interactions of a software system. It helps stakeholders understand how data and control flow through the system and how it transitions between different states and activities.

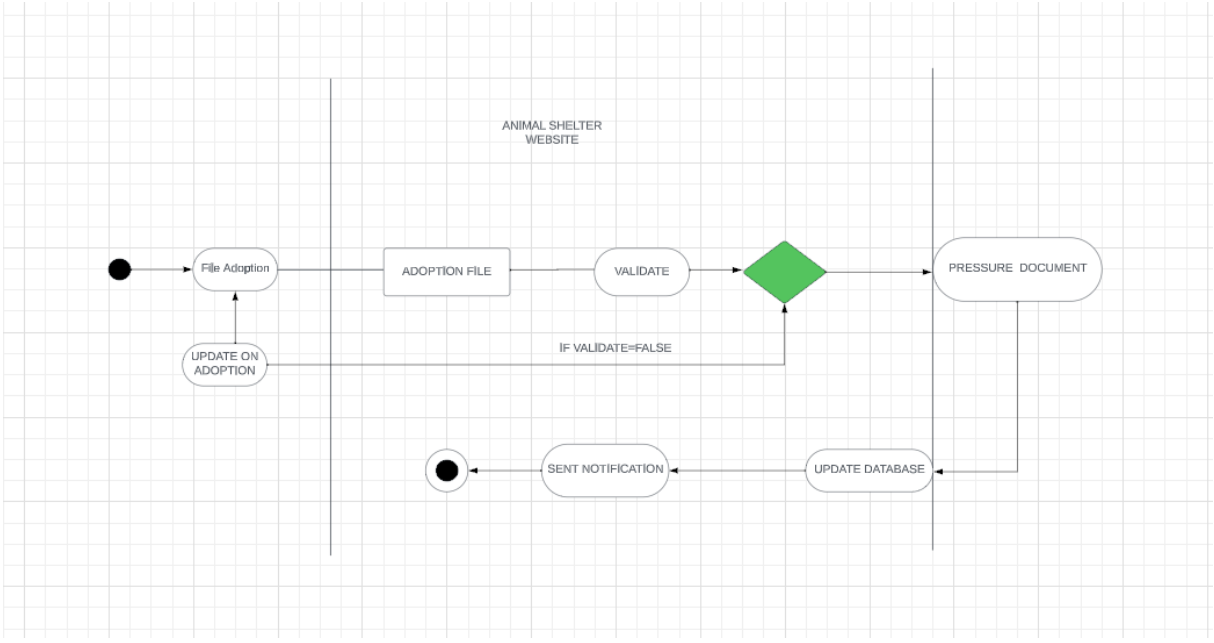


REPORT AND Rescue Activity Diagram

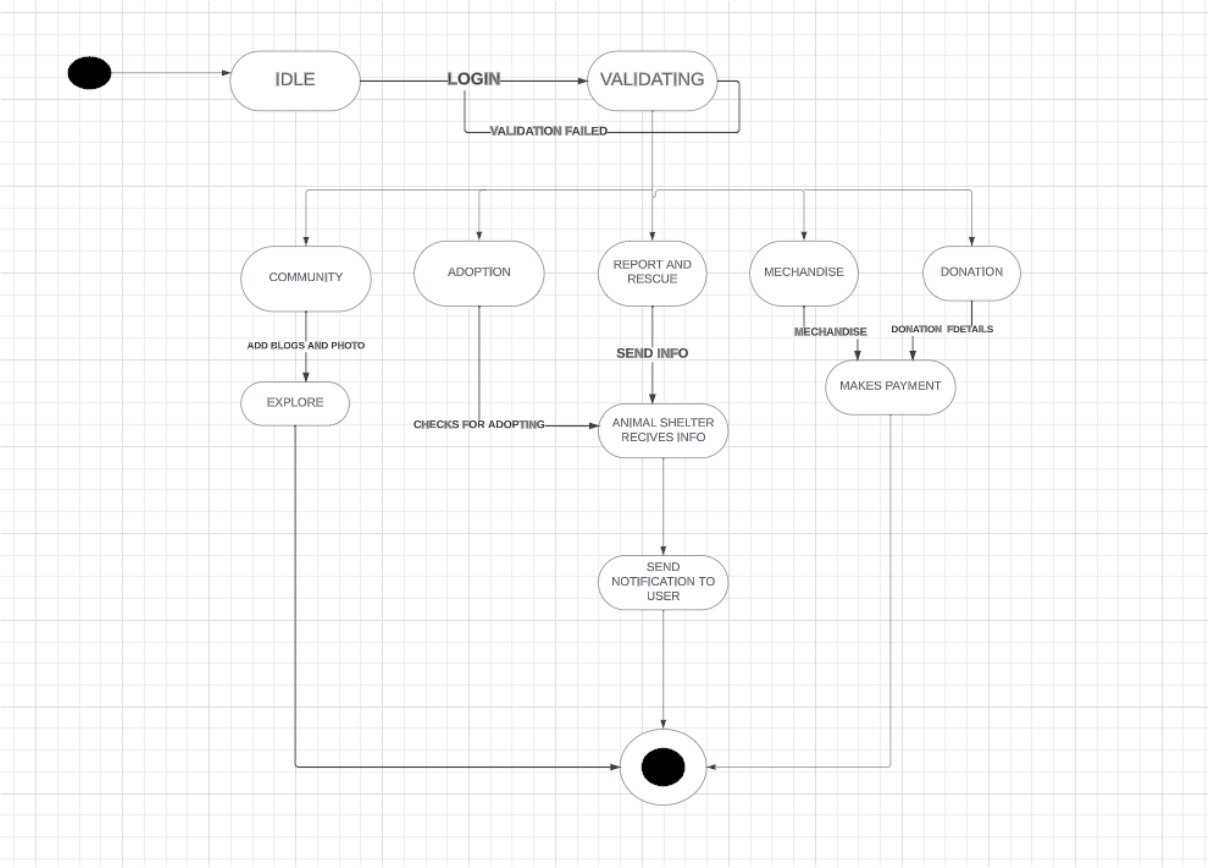


Merchandise ACTIVITY DIAGRAM





Donation and Adoption Activity Diagram



State Diagram