

## ABSTRACT

Currently, the network architecture that is commonly used is the IP address, where data is sent based on the host address on each device. The use of IP addresses in the process of sending data requires an additional process to map the destination IP address to the closest position to the user. However, a new paradigm emerged in data delivery called NDN (Named Data Networking), which changed the approach from host-centric to data-centric. While NDN offers these advantages, there are drawbacks in its operation, namely that there are limitations in storing cached data when implementing a caching strategy.

In this Final Project topic, the aim is to address the existing issues within the NDN network. In this Final Project, the author successfully created a Smart Home, which typically utilizes an IP network, but then transitioned to the emerging NDN network. Simulations were conducted using the Human Pose Estimation program with YOLOV7 and Fall Detection to identify falling individuals. This could be monitored through a created website, incorporating scenarios involving forwarding and caching strategies within the NDN network. The evaluation of the system's success encompasses testing delay/video streaming using Wireshark, website functionality, and FPS for optimal outcomes.

The results of this Final Project reveal that in the NDN network, there is a favorable delay during testing, with an average delay of 7.83 ms over 30 trials. In the FPS testing for Human Pose Estimation, 30 tests were conducted, resulting in averages of 3.2108 (for 1 person) and 2.9646 (for 2 persons), indicating the impact of increased detected objects on device performance and FPS. The testing of the Website and UI/UX Design involved a survey conducted through Google Form with 36 respondents, indicating that the website is well-implemented and user-friendly, although further enhancements are needed based on respondent feedback to enhance the overall user experience.

*Keywords : Named Data Network, Live Streaming, Human Pose Estimation, Caching, Forwarding.*