# Executive Summary of the *Lady Linux* Capstone Proposal

This document proposes **Lady Linux** as an ambitious, multidisciplinary **senior capstone project** centered on building a new Linux-based operating system that integrates a **baked-in Large Language Model (LLM)** to help users understand, manage, and control their data and system settings through natural language.

At its core, Lady Linux is designed to **restore agency, data literacy, and configurability** to everyday users, capabilities that currently require expert-level technical knowledge.

---

# Core Vision

Lady Linux is envisioned as:

- An **open-source Linux distribution**
- With a **native LLM companion**
- Capable of inspecting, explaining, and helping configure:
    - Operating system settings
    - Application behaviors
    - Data storage and data flows
- Using **human-centered, language-based interaction**

The system reframes the operating system as an *intelligent, teachable partner* rather than an opaque technical artifact.

---

# The Problem It Addresses

This proposal argues that:

- Most users lack **data literacy**
- Modern systems hide critical mechanisms (cookies, sessions, local storage, permissions)
- Current interfaces are too technical and fragmented
- Mobile and consumer devices are intentionally:
    - Locked down
    - Non-repairable
    - Designed for data extraction rather than user control

Lady Linux aims to counter this trend by providing **visibility, explanation, and consent-driven control**.

---

# Major Technical Pillars (Capstone Components)

The project identifies **eight to ten major areas**, each suitable for deep student involvement:

## 1. Operating System Construction

- Evaluate existing Linux distributions **or**
- Build directly from the kernel
- Strip bloatware
- Select libraries, utilities, and security fundamentals deliberately

## 2. Integrated LLM

- Pre-installed, local-first LLM (CPU or GPU)
- Fine-tuned on:
    - Apache documentation
    - Git repositories
    - OS internals
- Purpose-built for system inspection and explanation

## 3. Abstraction Layer

- Middleware (currently prototyped in Python + FastAPI)
- Provides **controlled, pseudo-access** to system functions
- Enforces:
    - Least privilege
    - Approval workflows
    - Reversibility (rollback of changes)

## 4. Security & Safety Layer

- Prevents autonomous agents from:
    - Making harmful changes
    - Acting without user consent
- Strong emphasis on **human-in-the-loop control**

## 5. Data Representation & Management

- Treats *all user activity* as data:
    - Files
    - Messages
    - Clicks, hovers, scrolls
- Questions addressed:
    - Encryption vs plaintext
    - Data ownership
    - Exporting and sharing

## 6. User Interface & HCI

- Graphical interface tightly integrated with the LLM
- Designed for novices
- Includes:
    - Interactive tutorials
    - Guided onboarding
    - Explainable system actions
- Identified as *the most critical success factor*

## 7. Hardware Platform Exploration

- Desktop and laptop prototyping first
- Long-term goal: mobile devices
- Explicit critique of:
    - Planned obsolescence
    - Non-repairable consumer hardware
- Aligns with right-to-repair philosophy

## 8. Mobile vs Desktop Adaptation

- Separate challenges and architectures
- Recognizes mobile Linux as a hard but important frontier

## 9. Project Management

- Coordination of multiple technical teams

- Integration across layers

### 10. Client / Stakeholder Role

- The author serves as domain expert and guiding client

---

# Educational Value as a Capstone

The proposal argues Lady Linux is **capstone-worthy** because students will:

- Work across **systems, AI, security, UX, and ethics**
- Engage in real-world problems:
    - Privacy
    - Automation risk
    - Hardware sustainability
- Gain experience from kernel-level design up through UI
- Produce a meaningful, extensible open-source artifact

---

# Overall Thesis

Lady Linux is not just an operating system, it is a **human-centered computing platform** designed to:

- Teach users about their systems
- Protect their data
- Restore configurability and understanding
- Push back against opaque, extractive consumer technology

As a senior capstone, it offers **depth, breadth, and societal relevance**, making it suitable for a large, collaborative, interdisciplinary student team.