Systems Analysis & Design
Semester 2025-III
Workshop No. 4 — Kaggle System Simulation

Eng. Carlos Andrés Sierra, M.Sc.
Full-time Adjunct Professor
Computer Engineering Program
School of Engineering
Universidad Distrital Francisco José de Caldas

Welcome to the fourth workshop of the *Systems Analysis & Design* course!

So far, you have:

- Conducted a **systems analysis** on an *open-priced Kaggle competition*[1] in **Workshop #1**.
- Developed a **system design** for implementing the requirements, addressing sensitivities, and mitigating chaos-related challenges in **Workshop #2**.
- Improved your system design and project management strategies in **Workshop #3**.

Now, you will move on to the realm of **computational simulation**, leveraging the insights from previous workshops. The central objective is to *simulate* key processes or interactions from the chosen Kaggle competition within the **system architecture** you designed.

Workshop Scope and Objectives

- **Data-driven Simulation:** Incorporate competition data (or a subset of it) into a simulation pipeline that mimics core processes (e.g., training, evaluation, or user interaction) using a classic machine learning model to analyze learning dynamics.
- **Event-based Simulation:** Develop an event-driven simulation using cellular automata adaptation to model spatial behaviors and emergent phenomena within your system architecture.

[1] https://www.kaggle.com/competitions?hostSegmentIdFilter=1

Carlos Andrés Sierra, Computer Engineer, M.Sc. in Computer Engineering, Titular Professor at Universidad Distrital Francisco José de Caldas.

Any comment or concern about this document can be sent to Carlos A. Sierra at: *cavirguezs@udistrital.edu.co*.

- **System Workflow Validation:** Validate your system's design by running both simulations and observing how data and events flow through its modules.
- **Complexity and Chaos Exploration:** Look for any emergent behaviors or chaotic patterns, especially in how sensitive variables impact the simulation outcomes.
- **Documentation & Reporting:** Generate a simulation report detailing the steps, tools, and results of your computational experiments.

Steps to Follow

1. **Data Preparation:**
   - Fetch or download the dataset from the same closed-priced Kaggle competition used in Workshop #1.
   - Clean, preprocess, and, if needed, reduce the dataset size to keep simulations feasible.
   - Summarize the data's main characteristics (e.g., number of features, missing values, distribution).
2. **Simulation Planning:**
   - Define two simulation scenarios:
     - **Scenario 1: Data-driven Simulation** — Use a classic ML model (e.g., neural network, random forest) to simulate learning and prediction processes.
     - **Scenario 2: Event-based Simulation** — Implement a cellular automata adaptation to simulate spatial or event-driven behaviors relevant to your system.
   - Align both scenarios with your system design from Workshop #2, specifying how each architectural component will be exercised.
   - Identify constraints, resource limits, and success metrics for each simulation (e.g., accuracy, throughput, emergent patterns).
3. **Simulation Implementation:**
   - For Scenario 1, implement and document the ML model, training process, and evaluation metrics.
   - For Scenario 2, design and code the cellular automata, defining rules, initial states, and event triggers.
   - Provide well-commented prototype code or stubs for relevant modules.
   - Consider chaos theory insights (random perturbations, feedback loops) in both simulations.

4. **Executing the Simulations:**
    - Run both simulations with different parameters or data slices to examine how system performance or outcomes vary.
    - Identify anomalous behaviors, bottlenecks, or emergent phenomena in each approach.
5. **Results and Discussion:**
    - Compile results into graphs, logs, or statistical summaries for both simulations.
    - Analyze differences and similarities between the outcomes of the ML-based and event-based simulations.
    - Suggest potential design improvements or next steps to refine your system.
6. **Documentation and Deliverables:**
    - Write a Simulation Report (PDF) describing both scenarios, methodology, code highlights, main results, and discussion of findings.
    - Store all simulation files in a dedicated folder named `Workshop_4_Simulation` within your GitHub repository.
    - Update your repository's `README.md` to reference this new folder, providing a brief summary of your simulation approaches and linking to the final PDF.

Deadline

Saturday, November 29th, 2025, at 8:00.
Late submissions may be subject to penalties as per course guidelines.

Notes

- The final report must be submitted in **English**, **PDF format**, and clearly reference your work from Workshops #1, #2, and #3.
- Emphasize any unexpected or chaotic outcomes, demonstrating how your design addresses potential system instabilities.
- If your simulation requires external libraries, specify them in a `requirements.txt` or other environment definition file.
- Treat this delivery as an incremental step. You will continue to expand and refine your simulation and system design for the final project.

Continue honing your skills as a systems engineer by exploring data-driven simulations, bridging design theory with practical implementations. Good luck!