Systems Analysis & Design
Semester 2025-I
Workshop No. 3 — Kaggle System Simulation

Eng. Carlos Andrés Sierra, M.Sc.
Computer Engineering
Universidad Distrital Francisco José de Caldas

Welcome to the third 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**.

Now, you will move on to the realm of **computational simulation**, leveraging the insights from those 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).
- **System Workflow Validation:** Validate your system's design by running the simulation and observing how data flows through its modules.
- **Complexity and Chaos Exploration:** Look for any emergent behaviors or chaotic patterns, especially in how sensitive variables impact the simulation outcome.

[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*.

- **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 open-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 the **scenario** to be simulated (e.g., model training performance over time, user-based data submissions, or predictive system responses).
   - Align the scenario with your *system design* from Workshop #2, specifying how each architectural component will be **exercised** by the simulation.
   - Incorporate **systems engineering** principles by identifying constraints, resource limits, and success metrics (e.g., throughput, latency, error rates).
3. **Simulation Implementation:**
   - Translate your simulation plan into code (e.g., Python scripts, Jupyter Notebooks) within the constraints of your system design.
   - Provide well-commented **prototype code** or *stubs* for relevant microservices or modules that handle data ingestion, processing, or inference.
   - Consider any **chaos theory** insights (such as random perturbations or feedback loops) to test how the system adapts under uncertain inputs.
4. **Executing the Simulation:**
   - Run your simulation with different parameters or data slices to examine how system performance or outcomes vary.
   - Identify any anomalous behaviors or bottlenecks; note if they are connected to high-sensitivity variables discovered earlier.
5. **Results and Discussion:**
   - Compile results into graphs, logs, or statistical summaries that highlight key findings.
   - Analyze any disagreements between expected outcomes (from design assumptions) and what the simulation reveals.
   - Suggest potential design improvements or next steps to refine your system.

6. **Documentation and Deliverables:**
   - Write a short *Simulation Report* (PDF) describing your scenario, methodology, code highlights, main results, and discussion of findings.
   - Store all simulation files in a dedicated folder named `Workshop_3_Simulation` within your GitHub repository.
   - Update your repository's `README.md` to reference this new folder, providing a brief summary of your simulation approach and linking to the final PDF.

Deadline

Saturday, June 28th, 2025, 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 and #2.
- 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.

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