EGN 6216 - AI Systems Icebreaker

A Living Padlet Icebreaker and Course Community Board

Instructor: Andrea Ramirez-Salgado | **Course:** AI Systems

PLATFORM

Padlet

TIMING

First week + course checkpoints

Assignment Description

Welcome to AI Systems. Our first activity will help us learn who is in the room while beginning the kind of systems-thinking we will use throughout the course. Each student will create an AI Systems Profile Card on a shared Padlet called the AI Systems Constellation.

The Padlet will become a living class resource. Students will use it to connect with classmates, identify complementary knowledge and interests, and revisit a real AI system as they learn about data, models, computing infrastructure, evaluation, deployment, and societal consequences. No prior AI experience is required. The activity values technical experience, disciplinary knowledge, lived experience, creativity, and thoughtful questions as different forms of expertise.

Purpose of the Activity

- Create early, meaningful connections among students rather than relying on unstructured introductions.
- Introduce AI systems as combinations of technical components, people, purposes, and consequences.
- Help students identify classmates with shared interests and classmates who bring complementary perspectives.
- Provide the instructor with information that can be used to select examples, support students, and form balanced teams.

Part 1: Build Your AI Systems Profile Card

Create one Padlet post using your name as the title. You may include a photograph, an avatar, or no image. In your post, respond to the prompts below:

1. **Introduce yourself.** Share your preferred name and pronunciation, your academic or professional background, and one interest outside of class.
2. **Choose an AI system that matters to you.** Select a system you use, study, question, or hope to design. Examples could include a recommendation system, medical diagnostic tool, autonomous vehicle, chatbot, smart agricultural system, hiring system, computer vision application, or another AI-enabled system. Explain in two or three sentences why it is meaningful to you.
3. **Name the perspective you bring.** Choose one or two labels that best describe how you might contribute to our learning community: model builder, data detective, domain expert, hardware tinkerer, interface designer, ethics advocate, systems integrator, communicator, or curious beginner. You may create your own label.
4. **Share one strength and one growth goal.** Identify something you could help a classmate understand and something you hope another classmate might help you learn.
5. **Ask a human-centered question.** End your post with one question about the people affected by your chosen AI system. For example: Who benefits? Who could be overlooked? What would make people trust or reject this system?

Part 2: Make Three Meaningful Connections

After posting, explore the Padlet and respond to at least three different classmates. Each response should be two or three sentences and serve a distinct purpose:

- **Common ground:** Identify an interest, experience, or concern you share with one classmate.
- **Complementary expertise:** Respond to someone whose strengths or perspective differ from yours. Explain what you might learn from each other.
- **Curious connection:** Ask a substantive question about a classmate's selected AI system or offer a related example they may not have considered.

A useful response goes beyond "I agree" or "That is interesting." It should help the other person feel recognized and open a possible path for future conversation or collaboration.

Part 3: Create a First-Week Systems Snapshot

During the first class meeting, or asynchronously in assigned groups, students will work in groups of four. Each group will select one AI system represented on its members' profile cards and create a short collaborative "Systems Snapshot" post. The snapshot will identify:

- the system's intended purpose;
- one likely source of data;
- one model, algorithm, or automated decision the system may use;
- one person or group affected by the system; and
- one failure, tradeoff, or unanswered question.

Each student will initial the component they contributed. The task is intentionally small and low stakes; its purpose is to reveal that an AI system cannot be understood from a single perspective and that the group's combined knowledge is more useful than any one person's initial answer.

How the Padlet Will Be Used Throughout the Course

The AI Systems Constellation will remain active throughout the semester. At selected checkpoints, students will return to their original system and add a brief update using the current module's lens:

- **Data lens:** What data would this system need, and what quality, representation, or bias concerns might arise?
- **Model lens:** What type of model or decision process could support the system, and what assumptions would it make?
- **Infrastructure lens:** Where should computation occur: device, edge, cloud, or a combination? What constraints matter?
- **Evaluation lens:** What metric would indicate success, and what important outcome might that metric miss?
- **Human and ethical lens:** Who gains power, convenience, risk, or responsibility when the system is deployed?
- **Final reflection:** How has your understanding of the system changed, and which classmate's perspective influenced your thinking?

By the end of the course, the Padlet will function as both a record of individual growth and a collective gallery of AI systems analyzed from multiple technical and human perspectives.

How I Incorporated Principles from the Module

This activity is grounded in the module's emphasis on relationship-rich classrooms. Felten and Lambert (2020) explain that instructors shape both student-faculty and student-student relationships through course design, and that peer interaction is most effective when it is purposeful, structured, and connected to learning. The activity incorporates those principles in the following ways:

Personal connection with academic purpose

Students share aspects of who they are, but each introduction is anchored in an AI system and a question they will continue investigating. The personal and disciplinary elements therefore reinforce one another.

Interaction with multiple classmates

Students engage with at least three peers for different reasons rather than responding only to a friend or to the first posts they see. The prompts encourage both similarity-based and cross-difference connections.

Meaningful peer work

The first-week Systems Snapshot requires students to combine technical, contextual, and human perspectives to produce something that no single introduction provides.

Positive interdependence and individual accountability

The group needs several perspectives to construct the snapshot, while initials make each student's contribution visible. The individual Padlet post and comments also remain independently accountable.

Validation and belonging

The prompt explicitly recognizes that students enter with different forms and levels of expertise. "Curious beginner" is presented as a legitimate starting point, and students identify both something they can contribute and something they want to learn.

Intentional student-faculty connection

I will respond briefly to every profile card during the first week, use students' names and interests in class examples, and draw on the Padlet when forming teams or connecting students to resources. This communicates that I notice students and expect them to succeed.

Sustained relationships rather than a one-time icebreaker

Because the Padlet is revisited across modules, early introductions become a foundation for later intellectual exchange, peer support, and reflection.

Participation, Accessibility, and Privacy

The activity will be graded for thoughtful completion rather than for prior technical knowledge. Students may use a photo, avatar, or text-only post and should not share private information. Captions or transcripts should accompany audio or video, and images should include a brief text description. A student who cannot access Padlet may submit the same content in the learning management system and receive a text-based version of classmates' posts for the interaction component.

Reference

Felten, P., & Lambert, L. M. (2020). *Relationship-rich education: How human connections drive success in college*. Johns Hopkins University Press.

Modular Learning Experience

Instructor Andrea Ramirez-Salgado	**Course** AI Systems	**Unit** Foundational Pieces of CNNs
Format 75-minute class + pre/post work	**Learners** Upper-level undergraduate / graduate	**Prerequisites** Basic neural networks and Python

Essential question: How can a neural network preserve and learn spatial patterns in an image?

Learning Objectives

1. **LO1:** Explain why fully connected networks are poorly matched to image data and why local connectivity and parameter sharing matter.
2. **LO2:** Trace how a convolutional filter produces a feature map, including the effects of kernel values, stride, and padding.
3. **LO3:** Interpret the distinct roles of convolution, activation, pooling, and classification layers in a CNN.
4. **LO4:** Diagnose and improve a simple CNN architecture, then use model evidence to justify design choices.

Design logic. The module follows a predict-observe-explain-design sequence. Students first surface their assumptions, then inspect visible effects of convolution, formalize the underlying concepts with targeted instruction, and finally use those concepts to make and defend engineering decisions. Instruction is deliberately inserted after students have a concrete problem or observation to explain.

Modular Learning Experience

Before class | The “same object, different pixels” provocation

INSTRUCTIONAL + INDIVIDUAL ACTIVE LEARNING | 10-15 minutes before class

What students do. Students complete a short, low-stakes interactive page containing three pairs of images: the same object shifted within a frame, the same object photographed under different lighting, and two visually similar objects from different classes. For each pair, students predict whether a basic fully connected network would treat the images as similar and explain their reasoning in two or three sentences. A brief multimedia explanation then reviews image tensors and dense-layer parameter growth, but it does not yet introduce convolution.

Instructor role and support. The instructor reviews responses before class and selects two contrasting explanations to display anonymously. These responses become the intellectual starting point for the class rather than a separate “homework check.”

Why this component is here. Targets LO1. The prediction task exposes the gap between human perception and raw pixel comparison. Delaying the CNN solution creates a need for the new architecture and gives the instructor evidence about students’ prior conceptions.

1. Launch | What changed, and what stayed the same?

ACTIVE LEARNING | 8 minutes

What students do. Students examine the two anonymous pre-class explanations and vote on which is more convincing. In pairs, they identify what information a useful image model should preserve when an object shifts, rotates slightly, or appears in a new location. Each pair writes one design requirement for an image-aware neural network on a shared board.

Instructor role and support. The instructor clusters the class requirements into three emerging ideas: local patterns, reusable detectors, and tolerance to small spatial changes. The instructor does not label these as convolution, parameter sharing, or invariance yet.

Why this component is here. Targets LO1 and prepares LO3. This opening makes students generate the functional requirements that CNN components will later satisfy. It also turns the formal vocabulary into answers to a problem students have already articulated.

2. Visual investigation | The filter lab

INSTRUCTIONAL SCAFFOLDING + COLLABORATIVE ACTIVE LEARNING | 15 minutes

What students do. Pairs use a prepared Jupyter notebook with a small grayscale image and four kernels: vertical-edge, horizontal-edge, blur, and sharpen. Students run each kernel, inspect the resulting feature map, and record two observations: what became more visible and what information was suppressed. They then modify one kernel value and predict the effect before running it.

Instructor role and support. The notebook provides only essential scaffolds: the image matrix, a reusable convolution function, and prompts. The instructor circulates with three questions: "What pattern activates this filter?", "Where is spatial information preserved?", and "What evidence supports your interpretation?"

Why this component is here. Targets LO2. The activity makes convolution observable before it becomes symbolic. Prediction before execution prevents trial-and-error clicking from replacing reasoning, while the visual output supports learners who may not yet be fluent with the matrix operation.

3. Sensemaking pause | From visual effects to the convolution operation

INSTRUCTIONAL COMPONENT | 10 minutes

What students do. Using one student-generated feature map, the instructor models a single convolution window by hand and connects each calculation to what students observed in the notebook. The explanation introduces kernel, receptive field, feature map, stride, padding, and parameter sharing. Two strategically chosen nonexamples are included: a stride that skips important details and no padding that rapidly shrinks spatial dimensions.

Instructor role and support. Students annotate a provided diagram rather than copying notes. Twice during the explanation, they complete a 30-second "predict the output shape" prompt and compare with a neighbor.

Why this component is here. Targets LO1 and LO2. This instruction is intentionally brief and responsive: it formalizes patterns students have already seen, addresses likely misconceptions, and immediately checks whether students can connect the terminology to dimensions and behavior.

4. Human CNN | Build the feature hierarchy

ACTIVE LEARNING | 12 minutes

What students do. Teams receive a set of cards representing an input image, low-level feature maps (edges and corners), intermediate patterns (textures and shapes), and possible classes. Their task is to arrange the cards into a defensible processing hierarchy and assign convolution, activation, pooling, or classification to each transition. One card is deliberately misleading, such as placing pooling before any feature extraction.

Instructor role and support. Teams conduct a gallery walk, leaving one question or challenge on another team's model. They then revise one element of their own representation.

Why this component is here. Targets LO3. The activity helps students distinguish layer functions and understand hierarchical representation without requiring a full implementation. The misleading card and peer challenge reveal whether students are reasoning about function rather than reproducing a memorized sequence.

5. Architecture clinic | Diagnose before you design

INSTRUCTIONAL FEEDBACK + ACTIVE LEARNING | 20 minutes

What students do. Each team receives one "patient chart" describing a flawed CNN and its symptoms. Examples include: training accuracy rises while validation performance stalls; feature maps collapse too quickly; the network contains millions of unnecessary parameters; or output dimensions are incompatible with the classifier. Teams identify the likely architectural cause, propose a targeted revision, and prepare a two-minute evidence-based consultation.

Instructor role and support. Before teams report, the instructor gives a three-minute decision guide connecting common symptoms to design levers such as kernel size, number of filters, pooling frequency, padding, and model capacity. During reports, classmates use a "support or challenge" protocol: they must either cite evidence supporting the diagnosis or identify a competing explanation.

Why this component is here. Targets LO3 and LO4. Diagnosing a flawed model is more intellectually demanding than assembling a standard architecture from a recipe. The short decision guide supplies just-in-time instruction, while the consultation format requires students to connect design decisions to observable evidence.

6. Individual transfer | One change, one claim, one piece of evidence

INDIVIDUAL ACTIVE LEARNING | 7 minutes

What students do. Students independently receive a compact CNN diagram and one constraint, such as limited computation, small objects in the images, or risk of overfitting. They mark one architectural change, write a one-sentence claim explaining why it should help, and name the evidence they would examine after training.

Instructor role and support. The instructor collects responses as an exit ticket and sorts them into three categories: conceptually sound, partially sound, and requiring follow-up. The next class begins with one anonymous example from each category.

Why this component is here. Targets LO4 and provides individual evidence after collaborative work. Requiring an anticipated evaluation metric or diagnostic prevents unsupported architecture preferences and reinforces that CNN design is an evidence-driven process.

After class | Controlled CNN experiment

ACTIVE LEARNING WITH INSTRUCTOR FEEDBACK | 30-45 minutes

What students do. Students complete a guided notebook using a small image dataset. They train a provided baseline CNN, alter exactly one architectural feature, and compare the baseline and revised models using training curves, validation performance, parameter count, and at least one feature-map visualization. They submit a short "claim-evidence-reasoning" reflection rather than a conventional lab report.

Instructor role and support. The instructor provides a rubric emphasizing interpretation and causal caution. Students are explicitly reminded that one run cannot prove a universal architectural rule; they must frame conclusions as evidence from this experiment.

Why this component is here. Targets LO2-LO4. Restricting students to one change supports meaningful comparison, while the claim-evidence-reasoning format aligns technical experimentation with the explanatory thinking expected in AI systems work.

UNIVERSAL DESIGN FOR LEARNING IN HIGHER EDUCATION

Syllabus Redesign Portfolio

EGN 6216 | Artificial Intelligence Systems | Fall 2026

Andrea Ramirez-Salgado

Teaching Excellence Academy | Module 3: Quality Learning Environment

Prepared from the existing Fall 2026 Simple Syllabus and the Activity 3.1 syllabus critique.

Portfolio Overview

This submission presents a complete UDLHE redesign of the Fall 2026 syllabus for EGN 6216, Artificial Intelligence Systems. It preserves the course's technical rigor, project-based structure, grading weights, and AI-system life-cycle sequence while making accessibility, purpose, choice, feedback, and support visible to students.

The redesign is intentionally more course-specific than a generic UDL checklist. Every access or choice feature is tied to a real learning demand in AI systems: interpreting complex material, managing technical uncertainty, collaborating across perspectives, producing evidence, communicating decisions, and revising a system as new information becomes available.

Research-Informed Design Foundations

The redesign is based first on the principles emphasized in Activity 3.1 and then strengthened with established UDL, higher-education, accessibility, and relational-pedagogy sources. The central design move is proactive: recurring barriers are anticipated in the syllabus, while individual accommodations remain available when universal design alone is not sufficient.

Research or framework foundation	Application in the EGN 6216 syllabus
Learner variability and proactive design (CAST, 2024)	The syllabus assumes that students will differ in prior preparation, language, confidence, sensory access, technology access, project experience, and ways of communicating. It therefore provides consistent Canvas organization, multiple entry points, advance rubrics, minimum and extension pathways, and early access checks.
Multiple means of engagement (CAST, 2024)	Students choose consequential project domains, connect technical decisions to stakeholder needs, participate through several channels, and use milestones, progress checks, and recovery plans to sustain effort. Choice is meaningful but bounded by course outcomes.
Multiple means of representation (CAST, 2024)	Complex ideas are represented through concise written explanations, system maps, diagrams, annotated code, worked examples, captioned demonstrations, transcripts, and professional cases. Representations are paired with specific barriers rather than with fixed “learning style” labels.

Research or framework foundation	Application in the EGN 6216 syllabus
Multiple means of action and expression (CAST, 2024)	Students may use written, oral, recorded, visual, or annotated modes when the communication format is not the construct being assessed. Required technical evidence - such as code, experiment records, metrics, risk analysis, and deployment reasoning - remains constant.
Executive function, feedback, and iteration (CAST, 2024)	The cumulative project is divided into visible stages. Templates, checklists, milestone dates, peer-feedback criteria, decision logs, and revision expectations help students plan, monitor, and improve their work rather than treating feedback as an endpoint.
Accessibility and POUR (W3C, 2023)	The document uses a logical heading hierarchy, readable fonts, strong contrast, descriptive links, text alternatives to visual meaning, table headers, predictable organization, and commitments to captions/transcripts. These decisions support content that is perceivable, operable, understandable, and robust.
Warm-demander stance (Bondy & Ross, 2008)	The syllabus combines non-negotiable intellectual and ethical standards with explicit confidence, explanation, support, and revision. The learning agreement communicates: the work matters, students are capable of growth, and help-seeking is part of serious engineering practice.
UDL in higher education (Tobin & Behling, 2018)	Access is framed as a course-design responsibility rather than only a disability-services referral. The redesign reduces high-frequency barriers for the whole class while preserving individualized accommodations and direct connections to campus resources.

References

Bondy, E., & Ross, D. D. (2008). The teacher as warm demander. Educational Leadership, 66(1), 54-58.

CAST. (2024). CAST Universal Design for Learning Guidelines, version 3.0.

Meyer, A., Rose, D. H., & Gordon, D. (Ed.). (2025). Universal Design for Learning: Principles, Framework, and Practice (3rd ed.). CAST Professional Publishing.

Tobin, T. J., & Behling, K. T. (2018). Reach Everyone, Teach Everyone: Universal Design for Learning in Higher Education. West Virginia University Press.

World Wide Web Consortium. (2023). Web Content Accessibility Guidelines (WCAG) 2.2.

PART I | Redesigned UDLHE Syllabus

Welcome to Artificial Intelligence Systems

Welcome to EGN 6216. In this course, we will move beyond asking whether a machine-learning model performs well in a notebook. We will ask a more demanding question: How do we design an AI system that is useful, trustworthy, scalable, secure, monitorable, and defensible in the context where it will operate?

AI systems can fail even when a model is technically accurate. The wrong problem may be selected. Data may be incomplete or harmful. Infrastructure may not support deployment. Users may not trust or understand the system. A model may drift after launch, or a team may not know when to intervene. You will learn to reason across these interconnected decisions rather than treating model training as the end of the work.

This is a rigorous, project-based course. Rigor here does not mean that you are expected to arrive knowing everything. It means that you will be expected to make evidence-based decisions, explain tradeoffs, test assumptions, respond to critique, and revise your work. Questions, help-seeking, and revision are not signs that you do not belong in the course. They are practices of responsible engineering.

Our learning agreement
The work matters. I believe you can do it. You will be held to high standards, and you will also receive clear expectations, meaningful feedback, multiple pathways to engage, and support for learning how to meet those standards.

Course at a Glance

Item	Information
Course	EGN 6216 | Artificial Intelligence Systems
Semester and credits	Fall 2026 | 3 credit hours | Class number 21125
Delivery method	Primarily classroom-based with Canvas-supported learning materials and project work
Meeting pattern	Tuesday, Periods 2-3 (8:30-10:25 a.m.), McCarty Hall A 1142; Thursday, Period 3 (9:35-10:25 a.m.), Weimer Hall 1094
Instructor	Andrea Ramirez Salgado
Email	aramirezsalgado@ufl.edu
Telephone	(352) 392-0951
Office	MALA 4112
Student office hours	Thursdays, 11:45 a.m.-1:45 p.m.; alternative appointment by request
Prerequisite	One machine-learning course: EGN 5216 or ABE 6933
Course fee	$0.00

Working With Your Instructor

Office hours are a learning resource, not a last resort. You may use them to test an idea, interpret feedback, debug a conceptual problem, narrow a project, rehearse a presentation, discuss a team concern, or make a recovery plan when you are behind. You do not need to prepare a polished question before coming.

Email is the best method for private or individual questions. For questions that may help the whole class, use the Canvas course Q&A so everyone can benefit from the response. I generally respond within two business days. When a time-sensitive issue affects a deadline or access to the course, put “EGN 6216 - time sensitive” in the subject line and explain the immediate need.

You are welcome to request an alternative meeting time when the posted office hours are inaccessible because of another course, work, caregiving, transportation, time-zone, or disability-related need.

Course Purpose, Description, and Audience

Catalog Description

Apply the concepts, frameworks, and tools used for building artificial intelligence systems in the real world. Examine the life cycle of AI systems and how such systems can be successfully deployed at scale and monitored in production.

Student-Facing Course Description

This course treats AI as a socio-technical system rather than an isolated model. You will work through the full system life cycle: problem framing, stakeholder and user analysis, infrastructure planning, data collection and governance, model experimentation, testing, deployment, security, monitoring, evaluation, and communication. A semester project provides the organizing context. Each deliverable becomes evidence for the next decision, and feedback is used to improve later work.

Target Audience

The course is designed for graduate students and advanced learners in engineering, data science, computing, or related fields who have prior exposure to machine learning and want to learn how AI is designed, deployed, governed, and sustained in real-world environments. Students may enter with different disciplinary strengths. Some will be stronger in modeling, others in systems, design, policy, communication, or domain knowledge. The course uses those differences as a resource while ensuring that every student develops core systems-level judgment.

The Big Questions We Will Revisit

- Should this problem be addressed with AI, and what evidence would justify that decision?
- Whose needs, risks, knowledge, and constraints shape the system requirements?
- What data, infrastructure, model, and deployment choices are appropriate for this context?
- How will we know whether the system is useful, trustworthy, secure, fair, and safe enough to deploy?
- What will be monitored after deployment, who is responsible for responding, and when should the system be changed or stopped?
- How can a team communicate technical evidence and limitations to audiences who will live with the consequences?

Student Learning Outcomes

By the end of the course, you will be able to:

1. Formulate and justify an AI-system development plan that integrates problem definition, stakeholder needs, computing requirements, human-centered design, trustworthiness, and risk considerations.
2. Plan and carry out data collection and management processes that align with project requirements and support valid testing, responsible use, reproducibility, and governance.
3. Design, train, test, and evaluate AI models through reproducible experiments, and defend model and metric choices in relation to system goals and tradeoffs.
4. Develop a deployment strategy that addresses scalability, security, compliance, user interaction, system dependencies, and plausible failure modes.
5. Establish an operating and monitoring framework that tracks model and system performance, data quality, drift, risk, and conditions requiring human intervention or redesign.
6. Communicate system rationale, evidence, limitations, risks, and recommendations to technical and nontechnical audiences, and use feedback to improve subsequent work.

How the Course Is Organized

The AI-system learning cycle
FRAME the problem -> PLAN the system -> GOVERN and PREPARE data -> EXPERIMENT with models -> TEST and DEPLOY -> MONITOR operations -> EVALUATE consequences -> COMMUNICATE and REVISE.

The sequence is cumulative. The proposal is not discarded after it is graded; it becomes the starting point for the development plan. The development plan informs data decisions. Data constraints reshape the experiment. Testing may require changes to deployment. Monitoring reveals what the team must revisit. Your final work should therefore show not only what you built, but how your reasoning changed across the semester.

Learning Materials and Required Readings

Required Materials

There is no required textbook purchase. Required readings, case materials, code examples, templates, and demonstrations will be provided through Canvas. Each Canvas item will be labeled as Required, Recommended, or Reference so students can distinguish essential preparation from optional extension.

The required Canvas reading packet is organized around five recurring streams:

- AI-system life-cycle and architecture: framing, requirements, infrastructure, dependencies, and deployment decisions.
- Human-centered and responsible design: stakeholders, usability, accessibility, security, privacy, ethics, trustworthiness, and risk.
- Data and experimentation: data collection, governance, reproducibility, validation, testing, and metric selection.
- Operations: deployment patterns, monitoring, drift, incident response, maintenance, and evaluation.
- Professional cases: short accounts of AI-system successes, failures, tradeoffs, and organizational decisions.

For each major topic, Canvas will include a “minimum pathway” with the essential reading or viewing and an “extension pathway” for students who want additional depth. When possible, core information will be available in at least two complementary forms, such as text plus an annotated diagram, or a demonstration plus a transcript and code notebook.

Recommended References

Author	Title	Publisher/year	Identifier
Martinez, David R., and Bruke M. Kifle	Artificial Intelligence: A Systems Approach from Architecture Principles to Deployment	MIT Press, 2024	ISBN 9780262048989
Ping, David	The Machine Learning Solutions Architect Handbook	Packt Publishing, 2022	ISBN 9781801070416
Huyen, Chip	Designing Machine Learning Systems: An Iterative Process for Production-Ready Applications	O'Reilly Media, 2022	ISBN 9781098107963

Technology and Computing Access

- Core work uses Python-based notebooks or scripts, a version-controlled code repository, and Canvas. Project teams may use additional tools when they can justify the choice and maintain reproducibility.
- HiPerGator or other high-performance computing resources may be used when project scope requires them. Training and starter materials will be linked in Canvas; advanced infrastructure should serve the learning goal rather than become an avoidable barrier.
- If reliable internet, hardware, software, computing capacity, or a platform is preventing participation, contact the instructor early. We will distinguish a temporary technical problem from a course-readiness issue and identify a feasible next step.
- Do not place restricted, private, proprietary, or personally identifiable data in an unapproved system or generative AI tool.

Assignments, Evidence, and Grading

The course grade is built around an iterative AI-system project. The assignment system is intentionally cumulative: each artifact records a decision, receives feedback, and provides evidence for a later artifact. Rubrics assess the quality of reasoning and evidence, not merely the completion of a template.

Assignment system, purpose, student choice, and grade weight

Assignment and weight	Purpose and required evidence	Choice or mode options
Project proposal (5%)	Define the problem, need, intended users or stakeholders, preliminary evidence, AI suitability, scope, and initial risks.	Choose domain, problem, stakeholder context, and concise written or narrated proposal format; required evidence must remain identifiable.
Elevator pitch (10%)	Explain why the problem matters, why AI may be appropriate, who is affected, and what decision or support is requested.	Live pitch, prerecorded pitch with live Q&A, or narrated visual pitch when approved. Same time and evidence criteria.
Elevator-pitch peer feedback (5%)	Give specific, evidence-based, respectful, and actionable feedback that the receiving team can use.	Written, audio, or time-stamped/annotated feedback using the common criteria.
AI-system development plan (10%)	Document requirements, architecture, computing needs, human-centered design, risk, security, privacy, team responsibilities, and decision points.	Choose appropriate architecture and planning tools; may submit as a structured report or technical design dossier.

Assignment and weight	Purpose and required evidence	Choice or mode options
Data collection and management plan/report (10%)	Justify data sources, collection, quality, representativeness, governance, documentation, validation, and limitations.	Choose data and documentation approach. Visual data maps may accompany, but not replace, required analysis.
Project-plan peer feedback (5%)	Audit a plan for hidden assumptions, feasibility, risk, accessibility, and alignment; propose actionable revisions.	Written, audio, or annotated plan review; feedback must point to evidence and course criteria.
Experiment design and report (10%)	Define baselines, variables, metrics, validation, reproducibility, results, uncertainty, and model-selection rationale.	Choose models, tools, and experimental scope within project constraints; report may be notebook-plus-brief or formal report.
Performance metrics and monitoring report (10%)	Define model, system, user, data-quality, drift, reliability, and risk indicators; specify thresholds and response actions.	Choose a dashboard mock-up, structured monitoring plan, or operations memo with equivalent required evidence.
Final code repository (10%)	Provide runnable, organized, documented, attributed, and responsibly handled code and artifacts.	Choose repository structure and reproducibility mechanism; core technical requirements are fixed.
Final project report (10%)	Integrate the system rationale, evidence, design changes, limitations, consequences, and recommendations across the life cycle.	Traditional technical report or equivalent design dossier with all required sections and references.
Final presentation (10%)	Communicate the system, evidence, limitations, risks, and recommendation to a defined audience; respond to questions.	Live, prerecorded with live Q&A, or accessible demonstration/poster format when approved. Same criteria.
Final-presentation peer feedback (5%)	Evaluate communication, evidence, limitations, and recommendations; identify one transferable lesson.	Written, audio, or structured annotation.

Grade Categories

Category	Percentage of final grade
Project deliverables	65%
Presentations	20%
Peer feedback	15%
Total	100%

Grading Scale

Grade	Range	Grade	Range
A	94-100%	A-	90-93%
B+	87-89%	B	83-86%
B-	80-82%	C+	77-79%

Grade	Range	Grade	Range
C	73-76%	C-	70-72%
D+	67-69%	D	63-66%
D-	60-62%	E	59% and below

Choice Architecture: Where You Have Agency

Student choice is most useful when it is real, visible, and connected to the learning goals. In this course, you will have agency within defined professional constraints.

Type of agency	What it means in EGN 6216
You may choose	The project domain and problem; stakeholder focus; data and model approach; system architecture; team role distribution; examples used to demonstrate learning; approved mode for selected presentations, feedback, and reflections.
You must justify	Why AI is appropriate; why the selected data, model, metrics, infrastructure, and deployment choices fit the context; how risks and accessibility are addressed; what evidence supports the recommendation.
You may request	An alternative participation pathway, an accessible submission format, clarification of assignment-level AI-use boundaries, a project-scope consultation, or an alternative office-hour appointment.
You may not replace	Required technical evidence, reproducible code, validation, attribution, risk analysis, or individual accountability with a different medium or with unverified AI-generated work.

Feedback, Revision, and Flexible Pacing

Feedback is part of the work, not commentary after the work is finished. Major rubrics and required components will be available before the assignment. Peer feedback will be taught and evaluated using five criteria: specific, evidence-based, respectful, actionable, and connected to a course outcome.

- Revision expectation: each major project artifact should visibly respond to earlier instructor, peer, user, or testing feedback. Final submissions include a brief decision log showing what changed, what did not change, and why.
- Flex passes: each student may use two 48-hour flex passes on eligible non-presentation deliverables without providing personal details. A flex pass cannot be used for scheduled peer-review exchanges, live presentations, or the final end-of-semester deadline unless an accommodation or approved exception applies.
- Revision pass: one major project artifact may be revised once after feedback within the announced revision window. The new submission must identify the changes and evidence used. The revised score replaces the earlier score when it demonstrates improvement.
- Alternative feedback mode: students may ask to receive or discuss feedback through written comments, an audio explanation, or a short conference when one mode creates an access barrier. The evaluative criteria remain the same.

Participation and Attendance: Why Presence Matters

Class meetings are not primarily used to repeat readings or slides. They include project studios, design decisions, critique, troubleshooting, case analysis, and team work that cannot always be reconstructed from a posted file. Consistent participation therefore matters because your decisions affect both your learning and the work of others.

Participation does not require speaking first or speaking most. You may contribute through whole-class discussion, a small group, an anonymous or digital question, an annotated example, a shared decision document, a written exit reflection, a project demonstration, or a scheduled follow-up. The appropriate mode may vary by activity.

For planned excused absences, contact the instructor as early as possible. For illness, emergencies, or unexpected barriers, contact the instructor when conditions permit. You will receive a reasonable opportunity to make up eligible work consistent with UF attendance policy. For a missed collaborative activity, the make-up may require an alternative individual analysis rather than recreating the original interaction.

Respectful Learning, Critique, and Team Responsibility

AI systems involve uncertainty, competing values, and consequences for people. Disagreement is expected and can improve technical decisions when it is handled responsibly. In this course, respectful critique is a professional competency.

- Critique claims, assumptions, evidence, and designs rather than the person presenting them.
- Ask clarifying questions before assigning intent; distinguish “I disagree” from “you are wrong.”
- Share airtime and technical responsibility. Teams should not assign one student all coding, all writing, or all coordination by default.
- Use names and pronunciations respectfully; do not ask classmates to disclose identities or experiences to validate a point.
- Make meetings, files, visuals, and presentations accessible to team members; accessibility is a shared design responsibility.
- Raise team concerns early. The instructor can facilitate a role reset, decision process, or accountability plan before conflict becomes a final-week crisis.

Accessibility by Design

The course is designed around the expectation that students vary in prior knowledge, language, processing speed, sensory access, executive-function needs, confidence, and ways of engaging. The goal is not to create a separate course for each learner. The goal is to make the core pathway clear and accessible while preserving meaningful challenge.

- Canvas modules follow the same sequence: orient, prepare, examine, practice, apply, submit, reflect.
- Documents use meaningful headings, readable fonts, sufficient contrast, descriptive links, accessible tables, and selectable text.
- Videos and demonstrations include captions or transcripts. Important visual information is also explained in words.
- Code examples include comments, a plain-language purpose statement, expected inputs and outputs, and common failure points.
- Instructions separate purpose, task, evidence, criteria, and submission steps so students can see what matters and plan their work.
- Rubrics, templates, and examples are provided early enough to guide performance, not only to judge it afterward.
- Students may report an access barrier in person, by email, through Canvas, or through a private course access form.

Disability-Related Accommodations

Students with disabilities who experience learning barriers and would like to request academic accommodations should connect with the University of Florida Disability Resource Center (DRC). Please share your accommodation letter and discuss your access needs as early as possible so the approved accommodations can be implemented effectively.

Start with the UF Disability Resource Center: Get Started page. You may also review the DRC accommodations information.

You do not need to disclose a diagnosis to the instructor. We can discuss the functional access need and how the approved accommodation applies to course activities. Built-in UDLHE options do not replace formal accommodations. If a course design feature or technology creates a barrier even when you do not use DRC services, please tell me so I can address the barrier when feasible.

Responsible Use of AI and Code Assistance

Because this is an AI course, the goal is not to pretend that AI-assisted tools do not exist. The goal is to learn when assistance supports learning, when it substitutes for learning, how to verify it, and how to disclose it. Every assignment will carry one of the three text labels below. The label, not color alone, defines the boundary.

Assignment label	Meaning
OPEN USE - disclose and verify	AI tools may be used for brainstorming, code explanation, debugging, test generation, editing, or alternative representations. You must verify the output, retain responsibility, and include the required disclosure.
LIMITED USE - follow listed purposes	The assignment identifies exactly what is permitted, such as syntax help but not generated architecture, or editing but not generated analysis. Uses outside the listed boundary require written permission.
INDEPENDENT REASONING - no generative production	Generative AI may not produce the submitted reasoning, code, analysis, or prose. Accessibility tools such as screen readers, speech-to-text, spelling support, or approved accommodations remain permitted unless they perform the prohibited intellectual work.

Uses That Are Generally Permitted When the Assignment Label Allows Them

- Explaining an error message or unfamiliar code at a conceptual level.
- Generating practice questions, edge cases, unit-test ideas, or alternative explanations.
- Brainstorming possible approaches before the student or team makes and documents the decision.
- Improving clarity, grammar, organization, or accessibility after the student has produced the substantive content.
- Translating or reformatting a student-created explanation while preserving meaning and attribution.

Uses That Are Prohibited Unless an Assignment Explicitly Authorizes Them

- Submitting generated code, analysis, architecture, prose, citations, data, results, or visuals as though they were your own work.
- Using AI to replace the individual reasoning, oral defense, reflection, or contribution that the assignment is designed to assess.
- Entering restricted, private, proprietary, personal, or project-confidential information into an unapproved tool.
- Fabricating references, experiments, metrics, stakeholder evidence, test results, or peer feedback.

- Using a tool in a way that prevents you from explaining, testing, or defending the submitted work.

AI-Use Disclosure Template

Include this statement when disclosure is required
Tool and version (if known): _____. Purpose: _____. What I entered or asked: _____. What I used, changed, or rejected: _____. How I verified the result: _____. Where the use appears in the submission: _____.

When you are uncertain, ask before submitting. A question about the boundary is evidence of responsible judgment, not evidence of misconduct.

Academic Integrity

Independent capacity matters in this course because engineers must be able to evaluate evidence, recognize failure, and accept responsibility for system decisions. Collaboration and tools can support that capacity, but they cannot replace it. Submit work that accurately represents your contribution, sources, process, and use of assistance.

UF students are bound by the University Honor Pledge and Student Honor Code. Potential violations will be handled through the established university process. The course will not use pre-announced punishment language as a substitute for clear expectations. Examples, assignment labels, disclosure requirements, and opportunities to ask questions are provided so that students can make responsible choices before submitting work.

Review the UF Honor Code syllabus guidance

Weekly Learning Pathway

All major required readings and preparation materials will be posted in the corresponding Canvas module. Exact assignment dates and any approved schedule adjustments will appear in Canvas and the official Simple Syllabus. Fall 2026 classes begin August 20, Thanksgiving break is November 23-28, and classes end December 2.

Fifteen-week learning pathway: preparation, engagement, milestones, and support

Week, dates, and focus	Required preparation, learning activities, and milestone
1 | Aug. 20 - AI systems introduction and life-cycle overview	Complete the course access and readiness orientation. Analyze an AI-system failure, map project interests, and submit a low-stakes readiness reflection.
2 | Aug. 25-27 - Use cases, project life cycle, and problem framing	Prepare with the required system-framing materials. Participate in a proposal clinic. Project proposal due (5%).
3 | Sept. 1-3 - Computing infrastructure and high-performance computing	Complete an architecture scenario analysis and resource-estimation practice.
4 | Sept. 8-10 - Human-computer interaction, stakeholders, usability, and accessibility	Prepare from the HCI case packet. Elevator pitch due (10%) and peer feedback due (5%).
5 | Sept. 15-17 - Security, privacy, ethics, and trustworthiness	Analyze a case through threat, stakeholder-impact, and trustworthiness lenses.
6 | Sept. 22-24 - Risk management, requirements, and organizational constraints	Complete the risk and requirements workshop. AI-system development plan due (10%).

Week, dates, and focus	Required preparation, learning activities, and milestone
7 | Sept. 29-Oct. 1 - Data collection and management I	Prepare on data sources, quality, governance, and documentation. Build a data map and conduct an evidence audit.
8 | Oct. 6-8 - Data collection and management II	Prepare on representativeness, validation, limitations, and access. Data collection/management plan and report due (10%).
9 | Oct. 13-15 - Mid-semester project review and design revision	Complete project-plan peer feedback (5%), then produce a revision and decision log.
10 | Oct. 20-22 - ML experiment design I	Prepare on baselines, variables, metrics, and reproducibility. Build a minimum reproducible experiment.
11 | Oct. 27-29 - Experiment design II, testing, and deployment I	Complete experiment testing and interpretation. Experiment design/report due (10%).
12 | Nov. 3-5 - Testing and deployment II	Analyze scalability, security, interfaces, dependencies, and failure modes in a deployment failure clinic and go/no-go review.
13 | Nov. 10-12 - Operating and monitoring	Prepare on performance, drift, reliability, and incident response. Performance metrics/monitoring report due (10%).
14 | Nov. 17-19 - AI-system evaluation and integration	Conduct a system audit and presentation rehearsal. Final code repository due (10%).
Break | Nov. 23-28 - Thanksgiving break	No classes and no new instructional content. Use the break to rest, recover, and protect the final-week scope.
15 | Dec. 1-2 - Final integration and communication	Submit the final report (10%), final presentation (10%), and final peer feedback (5%). Complete the final decision log.

University Policies and Student Resources

Current UF policies on grading, attendance, disability-related accommodations, course evaluations, academic integrity, and related resources are maintained on the official UF syllabus policy page and are embedded in the Simple Syllabus platform. Students should use the current university language when a policy link and this course document differ.

[Open current UF syllabus policy links]

[Open current UF attendance policies]

Course-specific expectations in this syllabus explain the learning purpose and how the university policies apply in EGN 6216. They do not remove or narrow student rights under university policy or approved accommodations.

PART II | Design Reflection and Change Analysis

Why I Redesigned the Syllabus

Activity 3.1 revealed that the original syllabus was organized and technically coherent, but it did not sound like the course I intended to teach. The course is project-based, iterative, human-centered, and grounded in consequential decisions about data, deployment, risk, monitoring, and responsibility. The original document largely presented logistics, deliverables, attendance rules, and enforcement language. Students could see what to do, but they had less help understanding why the work mattered, what professional habits the course would cultivate, where they had agency, or how to recover when learning became difficult.

The redesign therefore did not begin by adding generic statements about inclusion. It began by asking what barriers a student might encounter at each point in the AI-system learning cycle and what syllabus design could do before the student needed an exception. I retained the original life-cycle structure and grade weights because they are pedagogically strong. I changed the story, support architecture, assignment transparency, and boundaries around choice.

The Central Shift: From Administrative Contract to Learning Interface

The original syllabus functioned primarily as a record of course information. The redesigned syllabus functions as a learning interface. It helps students orient to the purpose, anticipate the sequence, interpret standards, select appropriate pathways, and know what to do when a barrier appears. This does not make the document less rigorous. It makes rigor more teachable.

Warm-demander stance
The redesigned voice communicates three ideas together: the work is demanding and consequential; students are capable of developing the required judgment; and support, feedback, and revision are built into the process rather than offered only after failure.

Major Changes and Their Rationale

Change analysis: original limitation, redesign decision, UDLHE principle, and intended effect

Area	What the original document communicated	Redesign decision and intended effect
Opening and instructor presence	The course began with course number, delivery, contact information, and requirements. Instructor presence was mainly administrative.	Added a student-facing welcome, the course's central challenge, a learning agreement, and an explanation of office hours. Students meet the intellectual purpose and instructor commitment before policy language.
Course description and audience	The catalog description described real-world deployment but did not show the socio-technical experience or acknowledge varied disciplinary strengths.	Added a target audience, expanded description, big questions, and explicit use of disciplinary variability. This supports relevance, activates prior knowledge, and clarifies readiness.
Learning outcomes	The five outcomes were measurable but emphasized production stages more than justification, tradeoffs, consequences, and communication.	Revised the outcomes to include defensible choices, governance, reproducibility, risk, monitoring response, communication, and use of feedback. Assessment now foregrounds professional judgment.
Course sequence	The schedule listed deliverables as separate graded products even though the project was intended to be iterative.	Made the development cycle explicit, required decision logs, and described how each artifact becomes evidence for the next. Students can see why feedback and revision matter.

Area	What the original document communicated	Redesign decision and intended effect
Accessibility	Access was largely delegated to university policy links.	Added accessible document commitments, consistent Canvas architecture, captions/transcripts, descriptive links, alternative representations, barrier reporting, and a distinct accommodation statement. Access becomes proactive design while accommodations remain individualized.
Student support	Resources were available through broad institutional policy links but students were not guided by the type of barrier they might face.	Created a no-shame recovery pathway that begins with a specific course action and connects to multimodal course and university supports. This reduces the executive-function burden of finding help.
Student agency	Choice existed in project work but was implicit.	Named what students may choose, must justify, may request, and may not replace. Agency becomes visible and bounded by learning outcomes rather than becoming unstructured choice overload.
Action and expression	Most assignments appeared to have a single submission pathway.	Created equivalent options for selected pitches, reports, monitoring artifacts, peer feedback, and participation while protecting non-negotiable technical evidence.
Peer feedback and teamwork	Peer feedback carried 15% of the grade, but quality criteria and constructive disagreement were not taught in the syllabus.	Defined feedback as specific, evidence-based, respectful, actionable, and outcome-connected; added equitable teamwork and conflict-recovery expectations. Community becomes a taught professional practice.
Attendance	Several pages of institutional attendance language overwhelmed the course-specific reason for being present.	Led with the learning rationale, named multiple participation modes, and then connected to UF attendance policy. Students encounter purpose before consequences and exceptions.
AI and code policy	AI was framed mainly as a threat, with unclear distinctions between help, substitution, debugging, editing, and generation.	Created assignment-level OPEN, LIMITED, and INDEPENDENT labels; added permitted/prohibited examples and a disclosure template. The policy becomes instruction in verification, attribution, privacy, and accountability.
Integrity tone	Predetermined sanctions and absolute language created a prosecutorial tone before relationship or clarification.	Explained why independent capacity matters, invited questions before submission, and deferred allegations and sanctions to the established university process. Accountability remains without unnecessary fear.
Flexibility and revision	Deadlines and graded products were fixed without an explicit recovery mechanism.	Added limited flex passes, one revision pass, milestone support, and recovery planning with clear exclusions. Flexibility is predictable without making deadlines arbitrary or shifting work to the end.

How the Redesign Uses Multiple Means of Representation

The strongest change is not the number of formats listed; it is the deliberate pairing of representations with a learning barrier. A system architecture diagram reduces the burden of holding every dependency in working memory. An annotated notebook reveals the relationship between code and purpose. A captioned demonstration helps a student see a process unfold, while the transcript supports review, search, translation, and screen-reader access. A checklist externalizes a multi-stage professional process. The syllabus now promises these supports where they are most useful rather than making a generic claim that "visual and auditory materials" will be offered.

I also avoided assuming that a preferred sensory format is a fixed learner identity. Students are not categorized as visual, auditory, or kinesthetic learners. Instead, learners can use complementary forms depending on the task, context, access need, and stage of learning.

How the Redesign Uses Multiple Means of Engagement

Engagement is built through relevance, agency, collaboration, and manageable challenge. Students choose a project domain and must connect technical decisions to people and consequences. They encounter real failure cases rather than only ideal workflows. They can participate through discussion, small groups, digital questions, shared documents, demonstrations, and reflections. At the same time, the design avoids unlimited choice. Project decisions must be justified, and the common outcomes define the boundaries.

The welcome and support language also change the emotional meaning of difficulty. The previous syllabus could be read as “succeed independently or face consequences.” The redesigned syllabus communicates “develop independent judgment, use support responsibly, and ask before a problem becomes unrecoverable.” That is a warmer message, but it remains demanding.

How the Redesign Uses Multiple Means of Action and Expression

I distinguished between the construct being assessed and the mode used to communicate it. A student can provide peer feedback in writing, audio, or annotation because the construct is the quality and usefulness of the feedback. A pitch can be live or prerecorded with live questions because the construct is audience-centered communication and defense. By contrast, the code repository cannot be replaced by a poster because reproducible code is part of the construct. This distinction allows meaningful flexibility without weakening alignment.

Accessible Format Decisions

The document itself was redesigned as an accessible artifact, not merely as text about accessibility. It uses:

- A logical heading hierarchy that supports navigation and screen-reader structure.
- High-contrast dark blue headings and orange rules used as accents rather than as the only signal of meaning.
- Descriptive link text instead of “click here” or long raw web addresses.
- Simple tables with identified header rows, repeated headers, concise captions, and no information communicated by color alone.
- Consistent labels for purpose, evidence, choice, support, and policy so students can predict where information will appear.

What I Intentionally Did Not Do

UDLHE does not require making every assignment available in every imaginable format, eliminating deadlines, or allowing students to avoid collaborative or technical work. Those choices could undermine the course outcomes. Instead, I used bounded flexibility: options are provided where the mode is not the construct, and requirements remain fixed where the evidence itself must be demonstrated.

I also did not rely on the language of learning styles. The design offers multiple representations because varied representations can clarify different aspects of a complex system and improve access, not because students should be placed into fixed sensory categories.

What Students Should Now Infer

After the redesign, I want students to infer the following before the first project challenge appears:

- This course is demanding because the decisions are consequential, not because confusion is treated as failure.
- The instructor expects students to develop, not already possess, systems-level judgment.
- Student perspectives and choices matter, but choices must be supported by evidence and responsibility.
- Feedback is supposed to change later work, and revision is a normal part of engineering.
- Accessibility, respectful critique, and equitable teamwork are part of technical quality.
- Using AI tools responsibly requires boundaries, verification, disclosure, and the ability to defend the work.
- When a barrier appears, there is a visible next action and more than one way to seek support.

Implementation in UF Simple Syllabus

The existing syllabus was produced through UF's Simple Syllabus format. The redesigned content can be entered into that official platform without abandoning the required structure. The welcome and target-audience text can be placed in the additional course description; the revised outcomes in the learning-outcomes field; assignment purposes and choice notes in methods of evaluation; the course-specific attendance rationale before the embedded policy; and the accessibility, support, AI-use, and respectful-learning statements in course policies and resources.

This portfolio document remains useful even when the official platform controls the visual template. It functions as the complete source text and design blueprint. It also demonstrates that UDLHE is not dependent on decorative formatting. The most consequential redesign decisions are in the language, sequence, transparency, choice architecture, and support pathways.

EGN 6216 - AI SYSTEMS
Transparent Assignment (TILT)
Project 2: Designing a Trustworthy Convolutional Neural Network

Module: Convolutional Neural Networks (CNNs)
Format: Individual | 10–12 hours | 15% of course grade

Project Overview

Artificial intelligence engineers rarely receive perfectly defined problems. They must evaluate data, design models, interpret results, and recommend solutions that balance performance, efficiency, robustness, and explainability. In this project, you will assume the role of an AI engineer developing a computer vision system for environmental conservation. Rather than maximizing accuracy alone, you will determine which CNN should be deployed and defend your recommendation with evidence.

Purpose (Why are you completing this assignment?)

This assignment is intentionally designed so you understand why each activity matters and how it connects to your learning.

- Develop a conceptual understanding of how CNNs extract hierarchical visual features.
- Practice authentic engineering decision-making through iterative experimentation.
- Strengthen your ability to analyze evidence, justify design choices, and communicate technical recommendations.
- Gain experience with explainable AI techniques used in modern AI practice.

Course learning outcomes supported

- Design and implement convolutional neural networks.
- Evaluate AI models using appropriate metrics and engineering judgment.
- Communicate reproducible, evidence-based AI solutions.
- Apply responsible AI principles when recommending deployment.

Beyond this course

Graduates working in robotics, autonomous systems, manufacturing, agriculture, healthcare, and environmental monitoring are expected to justify why an AI system should be trusted. This project mirrors that professional responsibility.

Engineering Scenario

A conservation agency uses drones to monitor invasive plant species. Computing resources are limited, field scientists need understandable predictions, and false negatives are costly. Your team must recommend the CNN architecture that should be deployed. Your recommendation must balance predictive performance, computational efficiency, explainability, and practical deployment constraints.

Your Tasks

1. **Investigate:** Explore the dataset, identify potential bias, imbalance, and preprocessing needs.
2. **Build:** Develop one baseline CNN and document its assumptions.
3. **Improve:** Create two substantially different CNN solutions. Every architectural choice must be justified.
4. **Experiment:** Conduct controlled experiments by changing one meaningful factor at a time.
5. **Evaluate:** Compare models using multiple metrics (accuracy, precision, recall, F1-score, confusion matrix, learning curves, inference time).
6. **Explain:** Use Grad-CAM (or similar) to interpret model attention.
7. **Reflect:** Analyze at least five incorrect predictions and discuss likely causes.
8. **Recommend:** Select one model for deployment and defend your decision for a technical and non-technical audience.

What Success Looks Like

Before beginning, review the sample notebook, sample report excerpt, and model comparison example. During the project you may use office hours, the discussion board, peer feedback, and the CNN troubleshooting guide. These resources illustrate expectations without providing the solution.

Deliverables

- Reproducible Jupyter Notebook with comments.
- Technical Report (4–5 pages) including methods, results, discussion, and deployment recommendation.
- One-page experiment log documenting architectural decisions and hyperparameters.
- 250-word reflection describing the most important engineering insight gained.

Evaluation Rubric

Criterion	Excellent	Developing	Weight
Engineering reasoning	Every design decision is justified with evidence and CNN concepts.	Limited or unsupported justification.	25%
Experimental design	Experiments are systematic, reproducible, and well documented.	Experiments lack control or documentation.	20%
Analysis & Explainability	Results, errors, and Grad-CAM are interpreted thoughtfully.	Mostly descriptive with limited interpretation.	25%

Recommendation	Recommendation balances accuracy, efficiency, robustness, and explainability.	Recommendation based mainly on accuracy.	15%
Professional communication	Clear, polished report with publication-quality figures.	Organization or clarity needs improvement.	15%

Common Mistakes to Avoid

- Reporting only accuracy.
- Changing several hyperparameters simultaneously.
- Copying architectures without justification.
- Presenting figures without interpretation.
- Ignoring computational cost or ethical considerations.
- Submitting code that cannot be reproduced.

Self-Assessment Checklist

☐ I can explain every architectural decision.

☐ I justified changes using evidence rather than intuition.

☐ I evaluated models fairly with multiple metrics.

☐ I interpreted model failures.

☐ I included explainability results.

☐ I defended my deployment recommendation.

☐ My notebook executes successfully.

☐ My report is professional and free of major writing errors.

Note

This assignment is intentionally designed using the Transparency in Learning and Teaching (TILT) framework. The purpose, task, available support, and evaluation criteria are made explicit so that students can devote their effort to developing deep engineering understanding rather than guessing instructor expectations.

Authentic Assessment
EGN 6216 – AI Systems
CNN Module

Assignment Title

AI Vision Consulting Challenge: Developing a Convolutional Neural Network for a Real-World Client

Assignment Context

You are a newly hired Machine Learning Engineer at VisionAI Solutions, an artificial intelligence consulting firm. A client has approached your team seeking an image-classification system that will improve an important real-world process. The client may represent healthcare, agriculture, manufacturing, transportation, environmental conservation, disaster response, or another application approved by the instructor. Your responsibility is not simply to build an accurate CNN, but to determine whether a CNN is the right solution, justify every engineering decision, communicate findings to stakeholders, and evaluate whether the model is suitable for responsible deployment.

In professional practice, AI engineers rarely receive step-by-step instructions. Instead, they analyze ambiguous problems, balance technical and societal considerations, and defend their recommendations using evidence. This assignment mirrors that process by asking you to think and act as a practicing AI engineer.

Purpose

This assessment is designed to evaluate your ability to apply concepts from the CNN module in an authentic engineering context. Success depends on your ability to analyze, design, justify, evaluate, and communicate. The idea is not simply reproduce code or definitions.

Student Instructions

1. Select a meaningful computer vision problem and identify the stakeholder who would benefit from your solution.

2. Locate an appropriate public dataset (or use the provided one). Briefly describe the data, its strengths, and its limitations.

3. Explain why a Convolutional Neural Network is an appropriate approach for this problem. Compare it with at least one alternative approach.

4. Prepare and preprocess the dataset. Clearly explain cleaning, resizing, normalization, augmentation, or other preprocessing decisions.

5. Design, implement, and train your CNN. Justify your architecture, optimizer, loss function, learning rate, number of epochs, and other key hyperparameters.

6. Evaluate your model using appropriate performance metrics (accuracy plus at least two additional metrics). Include visualizations such as a confusion matrix and training curves.

7. Perform an error analysis by examining at least five incorrect predictions. Discuss why the model failed and propose evidence-based improvements.

8. Discuss one ethical, societal, or deployment challenge relevant to your application, such as bias, privacy, accessibility, safety, or reliability.

9. Write an executive summary for the client explaining your recommendation in language understandable to non-technical stakeholders.

Deliverables

- Jupyter Notebook with well-documented code.
- Technical Report (4–6 pages).
- One-page Executive Summary for the client.

Why This Assessment Is Relevant and Engaging

Students complete the type of work expected from AI engineers in industry by solving an authentic problem for a realistic stakeholder. The assignment promotes ownership through student choice, requires engineering judgment instead of memorization, and integrates technical, ethical, and communication skills into a single performance task.

Evaluation Criteria

Criterion	Evidence of Excellence	Points
Problem Selection & Context	Problem is authentic, clearly justified, and stakeholder needs are well defined.	10
Application of CNN Knowledge	Engineering decisions demonstrate deep understanding and are supported by evidence.	25
Model Evaluation	Appropriate metrics,	20

	visualizations, and insightful error analysis.	
Responsible AI	Thoughtful discussion of ethics, bias, deployment, and limitations.	10
Executive Communication	Professional summary appropriate for non-technical stakeholders.	10
Technical Report & Notebook	Clear organization, reproducibility, documentation, and professionalism.	15
Overall Reflection	Insightful reflection on lessons learned and future improvements.	10

How Authentic Assessment Principles Were Applied

This assignment places students in a realistic professional role, requiring them to apply knowledge in a meaningful context rather than recall isolated facts. It emphasizes higher-order thinking, problem solving, evidence-based decision making, communication with multiple audiences, and reflection. Students produce artifacts similar to those created by practicing AI professionals, making the assessment relevant, transferable, and engaging.

Peer Review & Self-Reflection Worksheet
EGN 6216 – AI Systems
CNN Assignment Assessment

Assignment Context

This worksheet accompanies the assessment 'AI Vision Consulting Challenge' completed during the Convolutional Neural Networks (CNN) module. In that assignment, you assumed the role of Machine Learning Engineers working for an AI consulting company hired by a real-world stakeholder. Rather than simply training a CNN, you had to define the engineering problem, justify why a CNN is an appropriate solution, prepare and analyze a dataset, design and evaluate a model, perform error analysis, consider ethical implications, and communicate your recommendations through a professional technical report, Jupyter notebook, and executive summary.

Purpose of This Worksheet

Imagine you are the Senior Machine Learning Engineer responsible for reviewing your colleague's work before it is delivered to the client. Your goal is to verify that the project meets professional engineering standards while providing constructive feedback that will strengthen the final submission. After reviewing your peer's work, complete the reflection section to evaluate your own learning and engineering practices.

Instructions

1. Read your peer's report, executive summary, and notebook.
2. For each criterion, indicate whether it is Complete, Needs Revision, or Not Evident.
3. Provide specific, actionable feedback rather than general comments.
4. Discuss your feedback with your peer.
5. Complete the self-reflection before submitting your final project.

Review Criterion	Complete	Needs Revision	Comments / Evidence
PROJECT CONTEXT			
The stakeholder and real-world problem are clearly defined.			
The engineering objective addresses the stakeholder's needs.			
CNN DESIGN			
The choice of using			

a CNN is well justified.			
Dataset selection and preprocessing decisions are clearly explained.			
Architecture and hyperparameter choices are supported by evidence.			
MODEL EVALUATION			
Performance metrics extend beyond accuracy.			
Visualizations (e.g., confusion matrix, learning curves) support conclusions.			
Error analysis identifies meaningful failure patterns and improvements.			
RESPONSIBLE AI			
Ethical, fairness, privacy, accessibility, or deployment issues are considered.			
COMMUNICATION			
The executive summary is understandable for non-technical stakeholders.			
The report is professional, organized, and well written.			
The notebook is reproducible and well documented.			

Self-Reflection

- What aspect of your project demonstrates your strongest understanding of CNNs?

Response: ____

- What feedback from your peer will have the greatest impact on your final submission?

Response: ____

- What engineering decision are you most confident in? Why?

Response: ____

- What engineering decision would you revise after this review?

Response: ____

- What strategy did you use when your model underperformed?

Response: ____

- What is one concept from the CNN module you understand more deeply because of this project?

Response: ____

- Describe one specific improvement you will make before submitting your final assignment.

Response: ____

Peer Review Summary

Strengths identified:

- ___________________________________
- ___________________________________

Priority revisions:

- ___________________________________
- ___________________________________

Overall recommendation:

☐ Ready for submission ☐ Minor revisions recommended ☐ Major revisions recommended

Connection to Authentic Assessment

This worksheet supports authentic assessment by helping students evaluate technical quality, engineering reasoning, professional communication, and responsible AI practices. The reflection prompts encourage metacognition and self-regulated learning, while the peer review process provides formative feedback that students can immediately apply to improve their final submission.