INNOWORKS PRESS

CHATAKE
INNOWORKS

VOLUME V · 2026 COHORT EDITION

Applied AI Project Engineering Handbook

Fourteen Capstone Field Guides

Team structures, project gates, research directions, data plans, architecture, evaluation, deployment, reporting and publication pathways.

Innoworks Press
Chatake Innoworks Private Limited
MindForgeAI Internship 1.0
Solapur, Maharashtra, India

FOREWORD

A controlled learning system, not a collection of disconnected handouts.

This publication series was created so that classroom explanation, laboratory execution, home practice, assessment and project work refer to one controlled curriculum. Each volume serves a different purpose, but all volumes use the same weekly and daily index.

> The material follows the 2026 MindForgeAI Internship 1.0 cohort. Weeks 1 and 2 document the delivered foundation and workspace sequence, Week 3 reflects the current data-analytics stage, and later weeks provide the controlled continuation through machine learning, deep learning, applied AI, deployment and capstone defense.

How to use this volume

- Read the theory chapter before or immediately after the lecture.
- Execute every practical from a clean environment and compare the observed result with the expected behaviour.
- Complete the workbook without copying the model solution.
- Record evidence, errors, corrections and conclusions in the engineering diary.
- Use project handbooks to connect weekly concepts to the group capstone.

Publication notice

Do not claim ISBN, indexing, accreditation or formal academic affiliation until officially obtained.

PROJECT INDEX

13 field guides

01 **CIV-AI-01 · CivicVision AI**

AI for civil infrastructure inspection and site intelligence

02 **CIV-AI-02 · InfraSense AI**

AI for structural safety, planning and smart campus utilities

03 **ML-ENERGY-03 · GridMinds**

Machine learning for energy forecasting and sustainable operations

04 **ML-HEALTH-04 · CarePulse AI**

ML for healthcare triage, diagnostics support and wellness analytics

05 **ML-AGRI-05 · AgroLens AI**

Computer vision and predictive ML for agriculture and crop intelligence

06 **ML-EDU-06 · LearnSphere AI**

AI learning assistant, quiz analytics and personalized study support

07 **ML-FIN-07 · FinGuard AI**

Financial risk, fraud signal detection and responsible fintech automation

08 **ML-MANUF-08 · FactoryFlow AI**

Industrial quality prediction, maintenance and production analytics

09 **ML-ROBOT-09 · RoboPath AI**

Robotics perception, path planning and automation workflow intelligence

10 **ML-SAFETY-10 · SafeCity AI**

Public safety analytics, incident response and community intelligence

11 **ML-WATER-11 · AquaMind AI**

Water quality, distribution and resource monitoring using ML

12 **ML-SAAS-12 · NexoraOps AI**

AI-enabled SaaS dashboard for operations, analytics and workflow automation

13 **ML-IOT-13 · EdgePulse AI**

IoT, sensor data and edge intelligence for smart monitoring systems

REUSABLE PROJECT RECORDS

Controlled templates

Team charter

Member	Role	Primary deliverable	Backup responsibility

Weekly research log

Question	Work completed	Evidence	Decision	Owner / next action

Risk register

Risk	Likelihood	Impact	Mitigation	Owner	Residual risk

CIV-AI-01

CivicVision AI

AI for civil infrastructure inspection and site intelligence

Project intent

This project must produce a demonstrable engineering workflow, not only a model notebook. The team will define the user and decision, establish a data and evaluation protocol, build a reproducible baseline, add the approved intelligence layer, review risk, integrate a usable interface or service and defend the final evidence.

Minimum viable product

Upload an infrastructure image, classify visible condition, record confidence and generate an inspection-priority report with a mandatory human-review note.

Recommended research questions

- How can crack and defect image classification be designed, measured and validated within the available data and 12-week scope?
- How can inspection workflow design be designed, measured and validated within the available data and 12-week scope?
- How can risk-prioritized reporting be designed, measured and validated within the available data and 12-week scope?

Candidate data

- Public concrete-crack or surface-defect images
- Team-created labelled demonstration images
- Inspection metadata: location, date, asset type and reviewer

Candidate features

- Image pixels or embeddings
- Asset type
- Visible defect class
- Severity band
- Inspection context

Baselines to compare

- Manual rule checklist
- Simple image-feature classifier
- Pretrained image classifier

Primary evaluation

- Macro F1
- Per-class recall
- False-negative review
- Inference time

Recommended system architecture

Data layer

Source register, permission record, schema, validation, raw and processed versions.

Intelligence layer

Baseline, approved model or rules, evaluation, error analysis and model card.

Application layer

User workflow, API or notebook interface, logging, evidence and deployment plan.

Minimum data plan

Unit of analysis	Define one row, image, document, event or sensor window.
Target / output	Define the exact prediction, classification, retrieval or recommendation.
Data quality	Record missingness, duplicates, imbalance, annotation quality and drift risk.
Privacy	Exclude unnecessary personal or sensitive data and document access.

Twelve-week execution path

Week	Project action	Required evidence
1	Apply Computational Thinking and Engineering Discipline to CivicVision AI.	Architecture diagram, Problem statement, State or flow model
2	Apply Python Workspaces and Programming Foundations to CivicVision AI.	Executed notebook, Test cases, Flowchart
3	Apply Data Science and Analytics Foundations to CivicVision AI.	Raw and cleaned data, Data dictionary, Notebook
4	Apply Statistical Intuition for Engineers to CivicVision AI.	Statistical question, Method justification, Notebook
5	Apply Machine Learning Pipelines to CivicVision AI.	Problem framing, Data audit, Pipeline
6	Apply Neural Networks and Deep Learning Foundations to CivicVision AI.	Architecture diagram, Training notebook, Learning curves
7	Apply Computer Vision and Image Intelligence to CivicVision AI.	Dataset card, Training notebook, Error gallery
8	Apply Natural Language Processing and Generative AI to CivicVision AI.	Corpus card, Baseline, Retrieval workflow
9	Apply Responsible AI and Reproducible Engineering to CivicVision AI.	Stakeholder map, Bias analysis, Privacy plan
10	Apply MLOps, APIs, Docker and Cloud Foundations to CivicVision AI.	Model artifact, API, Dockerfile
11	Apply Specialization and Capstone Sprint to CivicVision AI.	Sprint board, Specialization evidence, Integrated system
12	Apply Capstone Completion and Professional Defense to CivicVision AI.	Release archive, Final report, Evaluation pack

Evaluation criteria

- Problem and user clarity
- Data provenance and quality
- Baseline and experiment discipline

- Error analysis and responsible AI
- Integration and reproducibility
- Documentation, demonstration and individual contribution

Suggested literature-search strings

- "Crack and defect image classification" dataset model evaluation review
- "Inspection workflow design" dataset model evaluation review
- "Risk-prioritized reporting" dataset model evaluation review

> **Project-specific risks**
>
> - Unsafe structural claims
> - Poor image quality
> - Background leakage
> - Unverified severity labels
>
> Do not promise real-time, medical, financial, safety-critical or autonomous use unless the system and evidence genuinely support it. Classroom prototypes must be labelled as prototypes.

Abstract worksheet

Problem and context	
Proposed method	
Data	
Expected contribution	
Limitations	

Final defense prompts

- What exact problem was solved?
- Why was this method chosen over the baseline?
- Which failure mode matters most?
- What did each member contribute?

- What evidence supports the result?
- What would be required before real deployment?

CIV-AI-02

InfraSense AI

AI for structural safety, planning and smart campus utilities

Project intent

This project must produce a demonstrable engineering workflow, not only a model notebook. The team will define the user and decision, establish a data and evaluation protocol, build a reproducible baseline, add the approved intelligence layer, review risk, integrate a usable interface or service and defend the final evidence.

Minimum viable product

Analyse a stream or table of utility and condition readings, detect abnormal behaviour and display maintenance alerts with traceable thresholds.

Recommended research questions

- How can sensor and utility analytics be designed, measured and validated within the available data and 12-week scope?
- How can structural condition indicators be designed, measured and validated within the available data and 12-week scope?
- How can anomaly and maintenance alerts be designed, measured and validated within the available data and 12-week scope?

Candidate data

- Synthetic or public sensor time series
- Campus utility readings where authorized
- Asset and maintenance event table

Candidate features

- Rolling mean and variance
- Rate of change
- Time-of-day
- Asset identifier
- Threshold breaches

Baselines to compare

- Static engineering thresholds
- Rolling z-score anomaly rule
- Isolation-based anomaly model

Primary evaluation

- Alert precision
- Event recall
- False alerts per day
- Detection delay

Recommended system architecture

Data layer

Source register, permission record, schema, validation, raw and processed versions.

Intelligence layer

Baseline, approved model or rules, evaluation, error analysis and model card.

Application layer

User workflow, API or notebook interface, logging, evidence and deployment plan.

Minimum data plan

Unit of analysis	Define one row, image, document, event or sensor window.
Target / output	Define the exact prediction, classification, retrieval or recommendation.
Data quality	Record missingness, duplicates, imbalance, annotation quality and drift risk.
Privacy	Exclude unnecessary personal or sensitive data and document access.

Twelve-week execution path

Week	Project action	Required evidence
1	Apply Computational Thinking and Engineering Discipline to InfraSense AI.	Architecture diagram, Problem statement, State or flow model
2	Apply Python Workspaces and Programming Foundations to InfraSense AI.	Executed notebook, Test cases, Flowchart
3	Apply Data Science and Analytics Foundations to InfraSense AI.	Raw and cleaned data, Data dictionary, Notebook
4	Apply Statistical Intuition for Engineers to InfraSense AI.	Statistical question, Method justification, Notebook
5	Apply Machine Learning Pipelines to InfraSense AI.	Problem framing, Data audit, Pipeline
6	Apply Neural Networks and Deep Learning Foundations to InfraSense AI.	Architecture diagram, Training notebook, Learning curves
7	Apply Computer Vision and Image Intelligence to InfraSense AI.	Dataset card, Training notebook, Error gallery
8	Apply Natural Language Processing and Generative AI to InfraSense AI.	Corpus card, Baseline, Retrieval workflow
9	Apply Responsible AI and Reproducible Engineering to InfraSense AI.	Stakeholder map, Bias analysis, Privacy plan
10	Apply MLOps, APIs, Docker and Cloud Foundations to InfraSense AI.	Model artifact, API, Dockerfile
11	Apply Specialization and Capstone Sprint to InfraSense AI.	Sprint board, Specialization evidence, Integrated system
12	Apply Capstone Completion and Professional Defense to InfraSense AI.	Release archive, Final report, Evaluation pack

Evaluation criteria

- Problem and user clarity
- Data provenance and quality
- Baseline and experiment discipline

- Error analysis and responsible AI
- Integration and reproducibility
- Documentation, demonstration and individual contribution

Suggested literature-search strings

- "Sensor and utility analytics" dataset model evaluation review
- "Structural condition indicators" dataset model evaluation review
- "Anomaly and maintenance alerts" dataset model evaluation review

> **Project-specific risks**
>
> - Missing sensors
> - Concept drift
> - False alarms
> - Confusing anomaly with structural failure
>
> Do not promise real-time, medical, financial, safety-critical or autonomous use unless the system and evidence genuinely support it. Classroom prototypes must be labelled as prototypes.

Abstract worksheet

Problem and context	
Proposed method	
Data	
Expected contribution	
Limitations	

Final defense prompts

- What exact problem was solved?
- Why was this method chosen over the baseline?
- Which failure mode matters most?
- What did each member contribute?

- What evidence supports the result?
- What would be required before real deployment?

ML-ENERGY-03

GridMinds

Machine learning for energy forecasting and sustainable operations

Project intent

This project must produce a demonstrable engineering workflow, not only a model notebook. The team will define the user and decision, establish a data and evaluation protocol, build a reproducible baseline, add the approved intelligence layer, review risk, integrate a usable interface or service and defend the final evidence.

Minimum viable product

Forecast short-horizon energy demand and show actual-versus-predicted load, uncertainty and operational recommendations.

Recommended research questions

- How can load forecasting be designed, measured and validated within the available data and 12-week scope?
- How can energy anomaly detection be designed, measured and validated within the available data and 12-week scope?
- How can operational recommendations be designed, measured and validated within the available data and 12-week scope?

Candidate data

- Public household or building energy datasets
- Weather and calendar features
- Synthetic campus-load series

Candidate features

- Lagged load
- Rolling statistics
- Hour and weekday
- Temperature
- Holiday indicator

Baselines to compare

- Last-value forecast
- Same-hour previous-day forecast
- Linear regression

Primary evaluation

- MAE
- RMSE
- MAPE with zero handling
- Peak-period error

Recommended system architecture

Data layer

Source register, permission record, schema, validation, raw and processed versions.

Intelligence layer

Baseline, approved model or rules, evaluation, error analysis and model card.

Application layer

User workflow, API or notebook interface, logging, evidence and deployment plan.

Minimum data plan

Unit of analysis	Define one row, image, document, event or sensor window.
Target / output	Define the exact prediction, classification, retrieval or recommendation.
Data quality	Record missingness, duplicates, imbalance, annotation quality and drift risk.
Privacy	Exclude unnecessary personal or sensitive data and document access.

Twelve-week execution path

Week	Project action	Required evidence
1	Apply Computational Thinking and Engineering Discipline to GridMinds.	Architecture diagram, Problem statement, State or flow model
2	Apply Python Workspaces and Programming Foundations to GridMinds.	Executed notebook, Test cases, Flowchart
3	Apply Data Science and Analytics Foundations to GridMinds.	Raw and cleaned data, Data dictionary, Notebook
4	Apply Statistical Intuition for Engineers to GridMinds.	Statistical question, Method justification, Notebook
5	Apply Machine Learning Pipelines to GridMinds.	Problem framing, Data audit, Pipeline
6	Apply Neural Networks and Deep Learning Foundations to GridMinds.	Architecture diagram, Training notebook, Learning curves
7	Apply Computer Vision and Image Intelligence to GridMinds.	Dataset card, Training notebook, Error gallery
8	Apply Natural Language Processing and Generative AI to GridMinds.	Corpus card, Baseline, Retrieval workflow
9	Apply Responsible AI and Reproducible Engineering to GridMinds.	Stakeholder map, Bias analysis, Privacy plan
10	Apply MLOps, APIs, Docker and Cloud Foundations to GridMinds.	Model artifact, API, Dockerfile
11	Apply Specialization and Capstone Sprint to GridMinds.	Sprint board, Specialization evidence, Integrated system
12	Apply Capstone Completion and Professional Defense to GridMinds.	Release archive, Final report, Evaluation pack

Evaluation criteria

- Problem and user clarity
- Data provenance and quality
- Baseline and experiment discipline

- Error analysis and responsible AI
- Integration and reproducibility
- Documentation, demonstration and individual contribution

Suggested literature-search strings

- "Load forecasting" dataset model evaluation review
- "Energy anomaly detection" dataset model evaluation review
- "Operational recommendations" dataset model evaluation review

> **Project-specific risks**
>
> - Future-data leakage
> - Seasonal drift
> - Missing readings
> - Overstated savings
>
> Do not promise real-time, medical, financial, safety-critical or autonomous use unless the system and evidence genuinely support it. Classroom prototypes must be labelled as prototypes.

Abstract worksheet

Problem and context	
Proposed method	
Data	
Expected contribution	
Limitations	

Final defense prompts

- What exact problem was solved?
- Why was this method chosen over the baseline?
- Which failure mode matters most?
- What did each member contribute?

- What evidence supports the result?
- What would be required before real deployment?

ML-HEALTH-04

CarePulse AI

ML for healthcare triage, diagnostics support and wellness analytics

Project intent

This project must produce a demonstrable engineering workflow, not only a model notebook. The team will define the user and decision, establish a data and evaluation protocol, build a reproducible baseline, add the approved intelligence layer, review risk, integrate a usable interface or service and defend the final evidence.

Minimum viable product

Demonstrate an educational risk-screening workflow using public or synthetic data, with explanations and explicit prohibition on clinical use.

Recommended research questions

- How can risk classification be designed, measured and validated within the available data and 12-week scope?
- How can responsible triage workflow be designed, measured and validated within the available data and 12-week scope?
- How can privacy and uncertainty be designed, measured and validated within the available data and 12-week scope?

Candidate data

- Public teaching datasets
- Synthetic patient scenarios
- Published variable definitions

Candidate features

- Age band
- Symptoms or measurements
- History indicators
- Missingness flags

Baselines to compare

- Transparent scorecard
- Logistic regression
- Decision tree

Primary evaluation

- Recall
- Precision
- Calibration
- Subgroup error review

Recommended system architecture

Data layer

Source register, permission record, schema, validation, raw and processed versions.

Intelligence layer

Baseline, approved model or rules, evaluation, error analysis and model card.

Application layer

User workflow, API or notebook interface, logging, evidence and deployment plan.

Minimum data plan

Unit of analysis	Define one row, image, document, event or sensor window.
Target / output	Define the exact prediction, classification, retrieval or recommendation.
Data quality	Record missingness, duplicates, imbalance, annotation quality and drift risk.
Privacy	Exclude unnecessary personal or sensitive data and document access.

Twelve-week execution path

Week	Project action	Required evidence
1	Apply Computational Thinking and Engineering Discipline to CarePulse AI.	Architecture diagram, Problem statement, State or flow model
2	Apply Python Workspaces and Programming Foundations to CarePulse AI.	Executed notebook, Test cases, Flowchart
3	Apply Data Science and Analytics Foundations to CarePulse AI.	Raw and cleaned data, Data dictionary, Notebook
4	Apply Statistical Intuition for Engineers to CarePulse AI.	Statistical question, Method justification, Notebook
5	Apply Machine Learning Pipelines to CarePulse AI.	Problem framing, Data audit, Pipeline
6	Apply Neural Networks and Deep Learning Foundations to CarePulse AI.	Architecture diagram, Training notebook, Learning curves
7	Apply Computer Vision and Image Intelligence to CarePulse AI.	Dataset card, Training notebook, Error gallery
8	Apply Natural Language Processing and Generative AI to CarePulse AI.	Corpus card, Baseline, Retrieval workflow
9	Apply Responsible AI and Reproducible Engineering to CarePulse AI.	Stakeholder map, Bias analysis, Privacy plan
10	Apply MLOps, APIs, Docker and Cloud Foundations to CarePulse AI.	Model artifact, API, Dockerfile
11	Apply Specialization and Capstone Sprint to CarePulse AI.	Sprint board, Specialization evidence, Integrated system
12	Apply Capstone Completion and Professional Defense to CarePulse AI.	Release archive, Final report, Evaluation pack

Evaluation criteria

- Problem and user clarity
- Data provenance and quality
- Baseline and experiment discipline

- Error analysis and responsible AI
- Integration and reproducibility
- Documentation, demonstration and individual contribution

Suggested literature-search strings

- "Risk classification" dataset model evaluation review
- "Responsible triage workflow" dataset model evaluation review
- "Privacy and uncertainty" dataset model evaluation review

> **Project-specific risks**
>
> - Clinical misuse
> - Sensitive-data exposure
> - Biased labels
> - False reassurance
>
> Do not promise real-time, medical, financial, safety-critical or autonomous use unless the system and evidence genuinely support it. Classroom prototypes must be labelled as prototypes.

Abstract worksheet

Problem and context	
Proposed method	
Data	
Expected contribution	
Limitations	

Final defense prompts

- What exact problem was solved?
- Why was this method chosen over the baseline?
- Which failure mode matters most?
- What did each member contribute?

- What evidence supports the result?
- What would be required before real deployment?

ML-AGRI-05

AgroLens AI

Computer vision and predictive ML for agriculture and crop intelligence

Project intent

This project must produce a demonstrable engineering workflow, not only a model notebook. The team will define the user and decision, establish a data and evaluation protocol, build a reproducible baseline, add the approved intelligence layer, review risk, integrate a usable interface or service and defend the final evidence.

Minimum viable product

Classify a crop or leaf condition from an image and combine it with field observations to produce a cautious advisory summary.

Recommended research questions

- How can crop image classification be designed, measured and validated within the available data and 12-week scope?
- How can field-data analytics be designed, measured and validated within the available data and 12-week scope?
- How can farmer-facing decision support be designed, measured and validated within the available data and 12-week scope?

Candidate data

- PlantVillage or another approved public image dataset
- Team-captured non-sensitive crop images
- Weather or field metadata

Candidate features

- Image tensor
- Crop type
- Color and texture
- Humidity or season where available

Baselines to compare

- Color histogram classifier
- Small CNN
- Transfer-learning classifier

Primary evaluation

- Macro F1
- Disease-class recall
- Confusion matrix
- Confidence under poor lighting

Recommended system architecture

Data layer

Source register, permission record, schema, validation, raw and processed versions.

Intelligence layer

Baseline, approved model or rules, evaluation, error analysis and model card.

Application layer

User workflow, API or notebook interface, logging, evidence and deployment plan.

Minimum data plan

Unit of analysis	Define one row, image, document, event or sensor window.
Target / output	Define the exact prediction, classification, retrieval or recommendation.
Data quality	Record missingness, duplicates, imbalance, annotation quality and drift risk.
Privacy	Exclude unnecessary personal or sensitive data and document access.

Twelve-week execution path

Week	Project action	Required evidence
1	Apply Computational Thinking and Engineering Discipline to AgroLens AI.	Architecture diagram, Problem statement, State or flow model
2	Apply Python Workspaces and Programming Foundations to AgroLens AI.	Executed notebook, Test cases, Flowchart
3	Apply Data Science and Analytics Foundations to AgroLens AI.	Raw and cleaned data, Data dictionary, Notebook
4	Apply Statistical Intuition for Engineers to AgroLens AI.	Statistical question, Method justification, Notebook
5	Apply Machine Learning Pipelines to AgroLens AI.	Problem framing, Data audit, Pipeline
6	Apply Neural Networks and Deep Learning Foundations to AgroLens AI.	Architecture diagram, Training notebook, Learning curves
7	Apply Computer Vision and Image Intelligence to AgroLens AI.	Dataset card, Training notebook, Error gallery
8	Apply Natural Language Processing and Generative AI to AgroLens AI.	Corpus card, Baseline, Retrieval workflow
9	Apply Responsible AI and Reproducible Engineering to AgroLens AI.	Stakeholder map, Bias analysis, Privacy plan
10	Apply MLOps, APIs, Docker and Cloud Foundations to AgroLens AI.	Model artifact, API, Dockerfile
11	Apply Specialization and Capstone Sprint to AgroLens AI.	Sprint board, Specialization evidence, Integrated system
12	Apply Capstone Completion and Professional Defense to AgroLens AI.	Release archive, Final report, Evaluation pack

Evaluation criteria

- Problem and user clarity
- Data provenance and quality
- Baseline and experiment discipline

- Error analysis and responsible AI
- Integration and reproducibility
- Documentation, demonstration and individual contribution

Suggested literature-search strings

- "Crop image classification" dataset model evaluation review
- "Field-data analytics" dataset model evaluation review
- "Farmer-facing decision support" dataset model evaluation review

> **Project-specific risks**
>
> - Dataset background shortcuts
> - Wrong crop variety
> - Unsafe treatment advice
> - Domain shift to field images
>
> Do not promise real-time, medical, financial, safety-critical or autonomous use unless the system and evidence genuinely support it. Classroom prototypes must be labelled as prototypes.

Abstract worksheet

Problem and context	
Proposed method	
Data	
Expected contribution	
Limitations	

Final defense prompts

- What exact problem was solved?
- Why was this method chosen over the baseline?
- Which failure mode matters most?
- What did each member contribute?

- What evidence supports the result?
- What would be required before real deployment?

ML-EDU-06

LearnSphere AI

AI learning assistant, quiz analytics and personalized study support

Project intent

This project must produce a demonstrable engineering workflow, not only a model notebook. The team will define the user and decision, establish a data and evaluation protocol, build a reproducible baseline, add the approved intelligence layer, review risk, integrate a usable interface or service and defend the final evidence.

Minimum viable product

Provide a student-facing learning dashboard that identifies completed work, missing evidence and recommended revision topics without making final academic decisions.

Recommended research questions

- How can learning analytics be designed, measured and validated within the available data and 12-week scope?
- How can question recommendation be designed, measured and validated within the available data and 12-week scope?
- How can grounded educational assistance be designed, measured and validated within the available data and 12-week scope?

Candidate data

- Portal task and diary completion records
- Synthetic quiz results
- Approved learning-resource metadata

Candidate features

- Completion rate
- Attempt version
- Topic scores
- Diary consistency
- Days since activity

Baselines to compare

- Rule-based reminders
- Topic-frequency summary
- Simple risk classifier

Primary evaluation

- Reminder precision
- Topic recommendation relevance
- False-alert rate
- Student feedback

Recommended system architecture

Data layer

Source register, permission record, schema, validation, raw and processed versions.

Intelligence layer

Baseline, approved model or rules, evaluation, error analysis and model card.

Application layer

User workflow, API or notebook interface, logging, evidence and deployment plan.

Minimum data plan

Unit of analysis	Define one row, image, document, event or sensor window.
Target / output	Define the exact prediction, classification, retrieval or recommendation.
Data quality	Record missingness, duplicates, imbalance, annotation quality and drift risk.
Privacy	Exclude unnecessary personal or sensitive data and document access.

Twelve-week execution path

Week	Project action	Required evidence
1	Apply Computational Thinking and Engineering Discipline to LearnSphere AI.	Architecture diagram, Problem statement, State or flow model
2	Apply Python Workspaces and Programming Foundations to LearnSphere AI.	Executed notebook, Test cases, Flowchart
3	Apply Data Science and Analytics Foundations to LearnSphere AI.	Raw and cleaned data, Data dictionary, Notebook
4	Apply Statistical Intuition for Engineers to LearnSphere AI.	Statistical question, Method justification, Notebook
5	Apply Machine Learning Pipelines to LearnSphere AI.	Problem framing, Data audit, Pipeline
6	Apply Neural Networks and Deep Learning Foundations to LearnSphere AI.	Architecture diagram, Training notebook, Learning curves
7	Apply Computer Vision and Image Intelligence to LearnSphere AI.	Dataset card, Training notebook, Error gallery
8	Apply Natural Language Processing and Generative AI to LearnSphere AI.	Corpus card, Baseline, Retrieval workflow
9	Apply Responsible AI and Reproducible Engineering to LearnSphere AI.	Stakeholder map, Bias analysis, Privacy plan
10	Apply MLOps, APIs, Docker and Cloud Foundations to LearnSphere AI.	Model artifact, API, Dockerfile
11	Apply Specialization and Capstone Sprint to LearnSphere AI.	Sprint board, Specialization evidence, Integrated system
12	Apply Capstone Completion and Professional Defense to LearnSphere AI.	Release archive, Final report, Evaluation pack

Evaluation criteria

- Problem and user clarity
- Data provenance and quality
- Baseline and experiment discipline

- Error analysis and responsible AI
- Integration and reproducibility
- Documentation, demonstration and individual contribution

Suggested literature-search strings

- "Learning analytics" dataset model evaluation review
- "Question recommendation" dataset model evaluation review
- "Grounded educational assistance" dataset model evaluation review

> **Project-specific risks**
>
> - Student profiling
> - Privacy
> - Punitive automation
> - Incomplete records
>
> Do not promise real-time, medical, financial, safety-critical or autonomous use unless the system and evidence genuinely support it. Classroom prototypes must be labelled as prototypes.

Abstract worksheet

Problem and context	
Proposed method	
Data	
Expected contribution	
Limitations	

Final defense prompts

- What exact problem was solved?
- Why was this method chosen over the baseline?
- Which failure mode matters most?
- What did each member contribute?

- What evidence supports the result?
- What would be required before real deployment?

ML-FIN-07

FinGuard AI

Financial risk, fraud signal detection and responsible fintech automation

Project intent

This project must produce a demonstrable engineering workflow, not only a model notebook. The team will define the user and decision, establish a data and evaluation protocol, build a reproducible baseline, add the approved intelligence layer, review risk, integrate a usable interface or service and defend the final evidence.

Minimum viable product

Flag unusual financial transactions in a public or synthetic dataset and provide the factors that triggered review.

Recommended research questions

- How can anomaly detection be designed, measured and validated within the available data and 12-week scope?
- How can risk scoring be designed, measured and validated within the available data and 12-week scope?
- How can explainability and false-positive control be designed, measured and validated within the available data and 12-week scope?

Candidate data

- Public fraud-detection teaching data
- Synthetic transactions
- Merchant and time metadata

Candidate features

- Amount deviation
- Transaction frequency
- Time
- Merchant category
- Location change

Baselines to compare

- Threshold rules
- Logistic regression
- Isolation forest

Primary evaluation

- Precision-recall AUC
- Recall at review capacity
- False positives
- Cost-weighted error

Recommended system architecture

Data layer

Source register, permission record, schema, validation, raw and processed versions.

Intelligence layer

Baseline, approved model or rules, evaluation, error analysis and model card.

Application layer

User workflow, API or notebook interface, logging, evidence and deployment plan.

Minimum data plan

Unit of analysis	Define one row, image, document, event or sensor window.
Target / output	Define the exact prediction, classification, retrieval or recommendation.
Data quality	Record missingness, duplicates, imbalance, annotation quality and drift risk.
Privacy	Exclude unnecessary personal or sensitive data and document access.

Twelve-week execution path

Week	Project action	Required evidence
1	Apply Computational Thinking and Engineering Discipline to FinGuard AI.	Architecture diagram, Problem statement, State or flow model
2	Apply Python Workspaces and Programming Foundations to FinGuard AI.	Executed notebook, Test cases, Flowchart
3	Apply Data Science and Analytics Foundations to FinGuard AI.	Raw and cleaned data, Data dictionary, Notebook
4	Apply Statistical Intuition for Engineers to FinGuard AI.	Statistical question, Method justification, Notebook
5	Apply Machine Learning Pipelines to FinGuard AI.	Problem framing, Data audit, Pipeline
6	Apply Neural Networks and Deep Learning Foundations to FinGuard AI.	Architecture diagram, Training notebook, Learning curves
7	Apply Computer Vision and Image Intelligence to FinGuard AI.	Dataset card, Training notebook, Error gallery
8	Apply Natural Language Processing and Generative AI to FinGuard AI.	Corpus card, Baseline, Retrieval workflow
9	Apply Responsible AI and Reproducible Engineering to FinGuard AI.	Stakeholder map, Bias analysis, Privacy plan
10	Apply MLOps, APIs, Docker and Cloud Foundations to FinGuard AI.	Model artifact, API, Dockerfile
11	Apply Specialization and Capstone Sprint to FinGuard AI.	Sprint board, Specialization evidence, Integrated system
12	Apply Capstone Completion and Professional Defense to FinGuard AI.	Release archive, Final report, Evaluation pack

Evaluation criteria

- Problem and user clarity
- Data provenance and quality
- Baseline and experiment discipline

- Error analysis and responsible AI
- Integration and reproducibility
- Documentation, demonstration and individual contribution

Suggested literature-search strings

- "Anomaly detection" dataset model evaluation review
- "Risk scoring" dataset model evaluation review
- "Explainability and false-positive control" dataset model evaluation review

> **Project-specific risks**
>
> - False accusations
> - Class imbalance
> - Sensitive financial data
> - Unexplainable flags
>
> Do not promise real-time, medical, financial, safety-critical or autonomous use unless the system and evidence genuinely support it. Classroom prototypes must be labelled as prototypes.

Abstract worksheet

Problem and context	
Proposed method	
Data	
Expected contribution	
Limitations	

Final defense prompts

- What exact problem was solved?
- Why was this method chosen over the baseline?
- Which failure mode matters most?
- What did each member contribute?

- What evidence supports the result?
- What would be required before real deployment?

ML-MANUF-08

FactoryFlow AI

Industrial quality prediction, maintenance and production analytics

Project intent

This project must produce a demonstrable engineering workflow, not only a model notebook. The team will define the user and decision, establish a data and evaluation protocol, build a reproducible baseline, add the approved intelligence layer, review risk, integrate a usable interface or service and defend the final evidence.

Minimum viable product

Predict a maintenance or quality-review priority from machine readings and display contributing signals and maintenance history.

Recommended research questions

- How can predictive maintenance be designed, measured and validated within the available data and 12-week scope?
- How can defect analytics be designed, measured and validated within the available data and 12-week scope?
- How can production bottleneck analysis be designed, measured and validated within the available data and 12-week scope?

Candidate data

- Public predictive-maintenance datasets
- Synthetic sensor sequences
- Maintenance event logs

Candidate features

- Temperature
- Vibration
- Runtime
- Rolling trends
- Time since maintenance

Baselines to compare

- Threshold rule
- Logistic regression
- Random forest

Primary evaluation

- Failure recall
- Lead time
- False alarms
- MAE for remaining-life variants

Recommended system architecture

Data layer

Source register, permission record, schema, validation, raw and processed versions.

Intelligence layer

Baseline, approved model or rules, evaluation, error analysis and model card.

Application layer

User workflow, API or notebook interface, logging, evidence and deployment plan.

Minimum data plan

Unit of analysis	Define one row, image, document, event or sensor window.
Target / output	Define the exact prediction, classification, retrieval or recommendation.
Data quality	Record missingness, duplicates, imbalance, annotation quality and drift risk.
Privacy	Exclude unnecessary personal or sensitive data and document access.

Twelve-week execution path

Week	Project action	Required evidence
1	Apply Computational Thinking and Engineering Discipline to FactoryFlow AI.	Architecture diagram, Problem statement, State or flow model
2	Apply Python Workspaces and Programming Foundations to FactoryFlow AI.	Executed notebook, Test cases, Flowchart
3	Apply Data Science and Analytics Foundations to FactoryFlow AI.	Raw and cleaned data, Data dictionary, Notebook
4	Apply Statistical Intuition for Engineers to FactoryFlow AI.	Statistical question, Method justification, Notebook
5	Apply Machine Learning Pipelines to FactoryFlow AI.	Problem framing, Data audit, Pipeline
6	Apply Neural Networks and Deep Learning Foundations to FactoryFlow AI.	Architecture diagram, Training notebook, Learning curves
7	Apply Computer Vision and Image Intelligence to FactoryFlow AI.	Dataset card, Training notebook, Error gallery
8	Apply Natural Language Processing and Generative AI to FactoryFlow AI.	Corpus card, Baseline, Retrieval workflow
9	Apply Responsible AI and Reproducible Engineering to FactoryFlow AI.	Stakeholder map, Bias analysis, Privacy plan
10	Apply MLOps, APIs, Docker and Cloud Foundations to FactoryFlow AI.	Model artifact, API, Dockerfile
11	Apply Specialization and Capstone Sprint to FactoryFlow AI.	Sprint board, Specialization evidence, Integrated system
12	Apply Capstone Completion and Professional Defense to FactoryFlow AI.	Release archive, Final report, Evaluation pack

Evaluation criteria

- Problem and user clarity
- Data provenance and quality
- Baseline and experiment discipline

- Error analysis and responsible AI
- Integration and reproducibility
- Documentation, demonstration and individual contribution

Suggested literature-search strings

- "Predictive maintenance" dataset model evaluation review
- "Defect analytics" dataset model evaluation review
- "Production bottleneck analysis" dataset model evaluation review

> **Project-specific risks**
>
> - Rare failures
> - Sensor drift
> - Maintenance label quality
> - Safety-critical misuse
>
> Do not promise real-time, medical, financial, safety-critical or autonomous use unless the system and evidence genuinely support it. Classroom prototypes must be labelled as prototypes.

Abstract worksheet

Problem and context	
Proposed method	
Data	
Expected contribution	
Limitations	

Final defense prompts

- What exact problem was solved?
- Why was this method chosen over the baseline?
- Which failure mode matters most?
- What did each member contribute?

- What evidence supports the result?
- What would be required before real deployment?

ML-ROBOT-09

RoboPath AI

Robotics perception, path planning and automation workflow intelligence

Project intent

This project must produce a demonstrable engineering workflow, not only a model notebook. The team will define the user and decision, establish a data and evaluation protocol, build a reproducible baseline, add the approved intelligence layer, review risk, integrate a usable interface or service and defend the final evidence.

Minimum viable product

Simulate a grid-based robot that perceives obstacles, computes a route and enters a safe-stop state when sensor or path information is invalid.

Recommended research questions

- How can path-planning simulation be designed, measured and validated within the available data and 12-week scope?
- How can perception-to-control workflow be designed, measured and validated within the available data and 12-week scope?
- How can safe fallback states be designed, measured and validated within the available data and 12-week scope?

Candidate data

- Generated occupancy grids
- Simulated sensor readings
- Path and collision logs

Candidate features

- Grid position
- Obstacle map
- Target position
- Sensor validity
- Path cost

Baselines to compare

- Breadth-first search
- Dijkstra
- A* heuristic search

Primary evaluation

- Path length
- Planning time
- Collision count
- Safe-stop success

Recommended system architecture

Data layer

Source register, permission record, schema, validation, raw and processed versions.

Intelligence layer

Baseline, approved model or rules, evaluation, error analysis and model card.

Application layer

User workflow, API or notebook interface, logging, evidence and deployment plan.

Minimum data plan

Unit of analysis	Define one row, image, document, event or sensor window.
Target / output	Define the exact prediction, classification, retrieval or recommendation.
Data quality	Record missingness, duplicates, imbalance, annotation quality and drift risk.
Privacy	Exclude unnecessary personal or sensitive data and document access.

Twelve-week execution path

Week	Project action	Required evidence
1	Apply Computational Thinking and Engineering Discipline to RoboPath AI.	Architecture diagram, Problem statement, State or flow model
2	Apply Python Workspaces and Programming Foundations to RoboPath AI.	Executed notebook, Test cases, Flowchart
3	Apply Data Science and Analytics Foundations to RoboPath AI.	Raw and cleaned data, Data dictionary, Notebook
4	Apply Statistical Intuition for Engineers to RoboPath AI.	Statistical question, Method justification, Notebook
5	Apply Machine Learning Pipelines to RoboPath AI.	Problem framing, Data audit, Pipeline
6	Apply Neural Networks and Deep Learning Foundations to RoboPath AI.	Architecture diagram, Training notebook, Learning curves
7	Apply Computer Vision and Image Intelligence to RoboPath AI.	Dataset card, Training notebook, Error gallery
8	Apply Natural Language Processing and Generative AI to RoboPath AI.	Corpus card, Baseline, Retrieval workflow
9	Apply Responsible AI and Reproducible Engineering to RoboPath AI.	Stakeholder map, Bias analysis, Privacy plan
10	Apply MLOps, APIs, Docker and Cloud Foundations to RoboPath AI.	Model artifact, API, Dockerfile
11	Apply Specialization and Capstone Sprint to RoboPath AI.	Sprint board, Specialization evidence, Integrated system
12	Apply Capstone Completion and Professional Defense to RoboPath AI.	Release archive, Final report, Evaluation pack

Evaluation criteria

- Problem and user clarity
- Data provenance and quality
- Baseline and experiment discipline

- Error analysis and responsible AI
- Integration and reproducibility
- Documentation, demonstration and individual contribution

Suggested literature-search strings

- "Path-planning simulation" dataset model evaluation review
- "Perception-to-control workflow" dataset model evaluation review
- "Safe fallback states" dataset model evaluation review

> **Project-specific risks**
>
> - Unsafe physical deployment
> - Sensor failure
> - Dynamic obstacles
> - Simulation-to-reality gap
>
> Do not promise real-time, medical, financial, safety-critical or autonomous use unless the system and evidence genuinely support it. Classroom prototypes must be labelled as prototypes.

Abstract worksheet

Problem and context	
Proposed method	
Data	
Expected contribution	
Limitations	

Final defense prompts

- What exact problem was solved?
- Why was this method chosen over the baseline?
- Which failure mode matters most?
- What did each member contribute?

- What evidence supports the result?
- What would be required before real deployment?

ML-SAFETY-10

SafeCity AI

Public safety analytics, incident response and community intelligence

Project intent

This project must produce a demonstrable engineering workflow, not only a model notebook. The team will define the user and decision, establish a data and evaluation protocol, build a reproducible baseline, add the approved intelligence layer, review risk, integrate a usable interface or service and defend the final evidence.

Minimum viable product

Analyse anonymized or public incident records, display patterns and support resource planning without predicting individual criminality.

Recommended research questions

- How can incident pattern analysis be designed, measured and validated within the available data and 12-week scope?
- How can resource prioritization be designed, measured and validated within the available data and 12-week scope?
- How can bias-aware public-safety reporting be designed, measured and validated within the available data and 12-week scope?

Candidate data

- Public incident datasets
- Synthetic community-safety records
- Time and location aggregates

Candidate features

- Incident category
- Time window
- Area aggregate
- Response time
- Resource load

Baselines to compare

- Frequency heat map
- Moving average
- Cluster or trend analysis

Primary evaluation

- Forecast MAE where applicable
- Pattern stability
- Coverage
- Bias review

Recommended system architecture

Data layer

Source register, permission record, schema, validation, raw and processed versions.

Intelligence layer

Baseline, approved model or rules, evaluation, error analysis and model card.

Application layer

User workflow, API or notebook interface, logging, evidence and deployment plan.

Minimum data plan

Unit of analysis	Define one row, image, document, event or sensor window.
Target / output	Define the exact prediction, classification, retrieval or recommendation.
Data quality	Record missingness, duplicates, imbalance, annotation quality and drift risk.
Privacy	Exclude unnecessary personal or sensitive data and document access.

Twelve-week execution path

Week	Project action	Required evidence
1	Apply Computational Thinking and Engineering Discipline to SafeCity AI.	Architecture diagram, Problem statement, State or flow model
2	Apply Python Workspaces and Programming Foundations to SafeCity AI.	Executed notebook, Test cases, Flowchart
3	Apply Data Science and Analytics Foundations to SafeCity AI.	Raw and cleaned data, Data dictionary, Notebook
4	Apply Statistical Intuition for Engineers to SafeCity AI.	Statistical question, Method justification, Notebook
5	Apply Machine Learning Pipelines to SafeCity AI.	Problem framing, Data audit, Pipeline
6	Apply Neural Networks and Deep Learning Foundations to SafeCity AI.	Architecture diagram, Training notebook, Learning curves
7	Apply Computer Vision and Image Intelligence to SafeCity AI.	Dataset card, Training notebook, Error gallery
8	Apply Natural Language Processing and Generative AI to SafeCity AI.	Corpus card, Baseline, Retrieval workflow
9	Apply Responsible AI and Reproducible Engineering to SafeCity AI.	Stakeholder map, Bias analysis, Privacy plan
10	Apply MLOps, APIs, Docker and Cloud Foundations to SafeCity AI.	Model artifact, API, Dockerfile
11	Apply Specialization and Capstone Sprint to SafeCity AI.	Sprint board, Specialization evidence, Integrated system
12	Apply Capstone Completion and Professional Defense to SafeCity AI.	Release archive, Final report, Evaluation pack

Evaluation criteria

- Problem and user clarity
- Data provenance and quality
- Baseline and experiment discipline

- Error analysis and responsible AI
- Integration and reproducibility
- Documentation, demonstration and individual contribution

Suggested literature-search strings

- "Incident pattern analysis" dataset model evaluation review
- "Resource prioritization" dataset model evaluation review
- "Bias-aware public-safety reporting" dataset model evaluation review

> **Project-specific risks**
>
> - Discriminatory profiling
> - Location stigma
> - Privacy
> - Misinterpreting correlation
>
> Do not promise real-time, medical, financial, safety-critical or autonomous use unless the system and evidence genuinely support it. Classroom prototypes must be labelled as prototypes.

Abstract worksheet

Problem and context	
Proposed method	
Data	
Expected contribution	
Limitations	

Final defense prompts

- What exact problem was solved?
- Why was this method chosen over the baseline?
- Which failure mode matters most?
- What did each member contribute?

- What evidence supports the result?
- What would be required before real deployment?

ML-WATER-11

AquaMind AI

Water quality, distribution and resource monitoring using ML

Project intent

This project must produce a demonstrable engineering workflow, not only a model notebook. The team will define the user and decision, establish a data and evaluation protocol, build a reproducible baseline, add the approved intelligence layer, review risk, integrate a usable interface or service and defend the final evidence.

Minimum viable product

Monitor water-quality or distribution readings, detect unusual patterns and present traceable alerts with measurement units.

Recommended research questions

- How can quality anomaly detection be designed, measured and validated within the available data and 12-week scope?
- How can demand forecasting be designed, measured and validated within the available data and 12-week scope?
- How can leak and distribution analytics be designed, measured and validated within the available data and 12-week scope?

Candidate data

- Public water-quality datasets
- Synthetic flow and pressure series
- Sensor calibration records

Candidate features

- pH
- Turbidity
- Conductivity
- Flow
- Pressure
- Rolling change

Baselines to compare

- Regulatory or engineering thresholds
- Rolling anomaly score
- Tree-based classifier

Primary evaluation

- Anomaly recall
- False alerts
- Detection delay
- Missing-data resilience

Recommended system architecture

Data layer

Source register, permission record, schema, validation, raw and processed versions.

Intelligence layer

Baseline, approved model or rules, evaluation, error analysis and model card.

Application layer

User workflow, API or notebook interface, logging, evidence and deployment plan.

Minimum data plan

Unit of analysis	Define one row, image, document, event or sensor window.
Target / output	Define the exact prediction, classification, retrieval or recommendation.
Data quality	Record missingness, duplicates, imbalance, annotation quality and drift risk.
Privacy	Exclude unnecessary personal or sensitive data and document access.

Twelve-week execution path

Week	Project action	Required evidence
1	Apply Computational Thinking and Engineering Discipline to AquaMind AI.	Architecture diagram, Problem statement, State or flow model
2	Apply Python Workspaces and Programming Foundations to AquaMind AI.	Executed notebook, Test cases, Flowchart
3	Apply Data Science and Analytics Foundations to AquaMind AI.	Raw and cleaned data, Data dictionary, Notebook
4	Apply Statistical Intuition for Engineers to AquaMind AI.	Statistical question, Method justification, Notebook
5	Apply Machine Learning Pipelines to AquaMind AI.	Problem framing, Data audit, Pipeline
6	Apply Neural Networks and Deep Learning Foundations to AquaMind AI.	Architecture diagram, Training notebook, Learning curves
7	Apply Computer Vision and Image Intelligence to AquaMind AI.	Dataset card, Training notebook, Error gallery
8	Apply Natural Language Processing and Generative AI to AquaMind AI.	Corpus card, Baseline, Retrieval workflow
9	Apply Responsible AI and Reproducible Engineering to AquaMind AI.	Stakeholder map, Bias analysis, Privacy plan
10	Apply MLOps, APIs, Docker and Cloud Foundations to AquaMind AI.	Model artifact, API, Dockerfile
11	Apply Specialization and Capstone Sprint to AquaMind AI.	Sprint board, Specialization evidence, Integrated system
12	Apply Capstone Completion and Professional Defense to AquaMind AI.	Release archive, Final report, Evaluation pack

Evaluation criteria

- Problem and user clarity
- Data provenance and quality
- Baseline and experiment discipline

- Error analysis and responsible AI
- Integration and reproducibility
- Documentation, demonstration and individual contribution

Suggested literature-search strings

- "Quality anomaly detection" dataset model evaluation review
- "Demand forecasting" dataset model evaluation review
- "Leak and distribution analytics" dataset model evaluation review

> **Project-specific risks**
>
> - Uncalibrated sensors
> - Unsafe quality claims
> - Unit mismatch
> - Missing context
>
> Do not promise real-time, medical, financial, safety-critical or autonomous use unless the system and evidence genuinely support it. Classroom prototypes must be labelled as prototypes.

Abstract worksheet

Problem and context	
Proposed method	
Data	
Expected contribution	
Limitations	

Final defense prompts

- What exact problem was solved?
- Why was this method chosen over the baseline?
- Which failure mode matters most?
- What did each member contribute?

- What evidence supports the result?
- What would be required before real deployment?

ML-SAAS-12

NexoraOps AI

AI-enabled SaaS dashboard for operations, analytics and workflow automation

Project intent

This project must produce a demonstrable engineering workflow, not only a model notebook. The team will define the user and decision, establish a data and evaluation protocol, build a reproducible baseline, add the approved intelligence layer, review risk, integrate a usable interface or service and defend the final evidence.

Minimum viable product

Deliver a role-aware operational dashboard with workflow records, analytics, AI-assisted prioritization and a documented API.

Recommended research questions

- How can operational dashboard be designed, measured and validated within the available data and 12-week scope?
- How can workflow prioritization be designed, measured and validated within the available data and 12-week scope?
- How can api and role-based saas architecture be designed, measured and validated within the available data and 12-week scope?

Candidate data

- Synthetic organizational workflow records
- Portal-style task metadata
- User and role definitions

Candidate features

- Task age
- Priority
- Owner workload
- Status transitions
- Completion history

Baselines to compare

- Rule-based priority
- Sortable dashboard
- Simple ranking model

Primary evaluation

- Task retrieval latency
- Ranking relevance
- Workflow completion
- Authorization test coverage

Recommended system architecture

Data layer

Source register, permission record, schema, validation, raw and processed versions.

Intelligence layer

Baseline, approved model or rules, evaluation, error analysis and model card.

Application layer

User workflow, API or notebook interface, logging, evidence and deployment plan.

Minimum data plan

Unit of analysis	Define one row, image, document, event or sensor window.
Target / output	Define the exact prediction, classification, retrieval or recommendation.
Data quality	Record missingness, duplicates, imbalance, annotation quality and drift risk.
Privacy	Exclude unnecessary personal or sensitive data and document access.

Twelve-week execution path

Week	Project action	Required evidence
1	Apply Computational Thinking and Engineering Discipline to NexoraOps AI.	Architecture diagram, Problem statement, State or flow model
2	Apply Python Workspaces and Programming Foundations to NexoraOps AI.	Executed notebook, Test cases, Flowchart
3	Apply Data Science and Analytics Foundations to NexoraOps AI.	Raw and cleaned data, Data dictionary, Notebook
4	Apply Statistical Intuition for Engineers to NexoraOps AI.	Statistical question, Method justification, Notebook
5	Apply Machine Learning Pipelines to NexoraOps AI.	Problem framing, Data audit, Pipeline
6	Apply Neural Networks and Deep Learning Foundations to NexoraOps AI.	Architecture diagram, Training notebook, Learning curves
7	Apply Computer Vision and Image Intelligence to NexoraOps AI.	Dataset card, Training notebook, Error gallery
8	Apply Natural Language Processing and Generative AI to NexoraOps AI.	Corpus card, Baseline, Retrieval workflow
9	Apply Responsible AI and Reproducible Engineering to NexoraOps AI.	Stakeholder map, Bias analysis, Privacy plan
10	Apply MLOps, APIs, Docker and Cloud Foundations to NexoraOps AI.	Model artifact, API, Dockerfile
11	Apply Specialization and Capstone Sprint to NexoraOps AI.	Sprint board, Specialization evidence, Integrated system
12	Apply Capstone Completion and Professional Defense to NexoraOps AI.	Release archive, Final report, Evaluation pack

Evaluation criteria

- Problem and user clarity
- Data provenance and quality
- Baseline and experiment discipline

- Error analysis and responsible AI
- Integration and reproducibility
- Documentation, demonstration and individual contribution

Suggested literature-search strings

- "Operational dashboard" dataset model evaluation review
- "Workflow prioritization" dataset model evaluation review
- "API and role-based SaaS architecture" dataset model evaluation review

> **Project-specific risks**
>
> - Broken access control
> - Private records
> - Automation bias
> - Always-on cloud cost
>
> Do not promise real-time, medical, financial, safety-critical or autonomous use unless the system and evidence genuinely support it. Classroom prototypes must be labelled as prototypes.

Abstract worksheet

Problem and context	
Proposed method	
Data	
Expected contribution	
Limitations	

Final defense prompts

- What exact problem was solved?
- Why was this method chosen over the baseline?
- Which failure mode matters most?
- What did each member contribute?

- What evidence supports the result?
- What would be required before real deployment?

ML-IOT-13

EdgePulse AI

IoT, sensor data and edge intelligence for smart monitoring systems

Project intent

This project must produce a demonstrable engineering workflow, not only a model notebook. The team will define the user and decision, establish a data and evaluation protocol, build a reproducible baseline, add the approved intelligence layer, review risk, integrate a usable interface or service and defend the final evidence.

Minimum viable product

Ingest simulated sensor telemetry, run an edge-friendly anomaly rule or compact model and synchronize events when connectivity returns.

Recommended research questions

- How can telemetry pipeline be designed, measured and validated within the available data and 12-week scope?
- How can edge anomaly detection be designed, measured and validated within the available data and 12-week scope?
- How can offline and intermittent-connectivity design be designed, measured and validated within the available data and 12-week scope?

Candidate data

- Generated telemetry
- Public IoT sensor datasets
- Device and connectivity logs

Candidate features

- Rolling statistics
- Battery level
- Signal quality
- Device ID
- Anomaly score

Baselines to compare

- Threshold rule
- Rolling z-score
- Compact tree model

Primary evaluation

- Detection latency
- Memory use
- False alarms
- Offline event retention

Recommended system architecture

Data layer

Source register, permission record, schema, validation, raw and processed versions.

Intelligence layer

Baseline, approved model or rules, evaluation, error analysis and model card.

Application layer

User workflow, API or notebook interface, logging, evidence and deployment plan.

Minimum data plan

Unit of analysis	Define one row, image, document, event or sensor window.
Target / output	Define the exact prediction, classification, retrieval or recommendation.
Data quality	Record missingness, duplicates, imbalance, annotation quality and drift risk.
Privacy	Exclude unnecessary personal or sensitive data and document access.

Twelve-week execution path

Week	Project action	Required evidence
1	Apply Computational Thinking and Engineering Discipline to EdgePulse AI.	Architecture diagram, Problem statement, State or flow model
2	Apply Python Workspaces and Programming Foundations to EdgePulse AI.	Executed notebook, Test cases, Flowchart
3	Apply Data Science and Analytics Foundations to EdgePulse AI.	Raw and cleaned data, Data dictionary, Notebook
4	Apply Statistical Intuition for Engineers to EdgePulse AI.	Statistical question, Method justification, Notebook
5	Apply Machine Learning Pipelines to EdgePulse AI.	Problem framing, Data audit, Pipeline
6	Apply Neural Networks and Deep Learning Foundations to EdgePulse AI.	Architecture diagram, Training notebook, Learning curves
7	Apply Computer Vision and Image Intelligence to EdgePulse AI.	Dataset card, Training notebook, Error gallery
8	Apply Natural Language Processing and Generative AI to EdgePulse AI.	Corpus card, Baseline, Retrieval workflow
9	Apply Responsible AI and Reproducible Engineering to EdgePulse AI.	Stakeholder map, Bias analysis, Privacy plan
10	Apply MLOps, APIs, Docker and Cloud Foundations to EdgePulse AI.	Model artifact, API, Dockerfile
11	Apply Specialization and Capstone Sprint to EdgePulse AI.	Sprint board, Specialization evidence, Integrated system
12	Apply Capstone Completion and Professional Defense to EdgePulse AI.	Release archive, Final report, Evaluation pack

Evaluation criteria

- Problem and user clarity
- Data provenance and quality
- Baseline and experiment discipline

- Error analysis and responsible AI
- Integration and reproducibility
- Documentation, demonstration and individual contribution

Suggested literature-search strings

- "Telemetry pipeline" dataset model evaluation review
- "Edge anomaly detection" dataset model evaluation review
- "Offline and intermittent-connectivity design" dataset model evaluation review

> **Project-specific risks**
>
> - Intermittent connectivity
> - Clock drift
> - Device spoofing
> - Resource constraints
>
> Do not promise real-time, medical, financial, safety-critical or autonomous use unless the system and evidence genuinely support it. Classroom prototypes must be labelled as prototypes.

Abstract worksheet

Problem and context	
Proposed method	
Data	
Expected contribution	
Limitations	

Final defense prompts

- What exact problem was solved?
- Why was this method chosen over the baseline?
- Which failure mode matters most?
- What did each member contribute?

- What evidence supports the result?
- What would be required before real deployment?

INNOWORKS PRESS

Applied AI Project Engineering Handbook

Team structures, project gates, research directions, data plans, architecture, evaluation, deployment, reporting and publication pathways.

https://internship.chatakeinnoworks.com