DRAFT — UNOFFICIAL — NOT FOR OPERATIONAL USE

MISSION TRAINING PLAN

MTP-MSS

USAREUR-AF

MISSION TRAINING PLAN (MTP)

MAVEN SMART SYSTEM (MSS) — USAREUR-AF

HEADQUARTERS
UNITED STATES ARMY EUROPE AND AFRICA
(USAREUR-AF)
Wiesbaden, Germany

DRAFT — NOT FOR OFFICIAL USE. FOR TRAINING PLANNING PURPOSES ONLY.

26 MARCH 2026

DRAFT — UNOFFICIAL — NOT FOR OPERATIONAL USE

MISSION TRAINING PLAN (MTP)

MAVEN SMART SYSTEM (MSS) — USAREUR-AF

HEADQUARTERS, UNITED STATES ARMY EUROPE AND AFRICA Wiesbaden, Germany | 2026

AUTHORITY: Issued under authority of USAREUR-AF C2DAO; implements USAREUR-AF Command Training Guidance. Provides planning and evaluation guidance for commanders and training managers responsible for MSS training across the formation.

DISTRIBUTION: Distribution authorized to U.S. Government agencies and their contractors only. Other requests must be referred to Headquarters, USAREUR-AF, C2DAO, Wiesbaden, Germany.

1. PURPOSE AND SCOPE

1-1. Purpose

This MTP provides the USAREUR-AF unit commander and training manager a structured framework to plan, resource, execute, and assess Maven Smart System (MSS) training across all personnel levels. It operationalizes the MSS training curriculum (SL 1 through SL 4 series) by defining tasks, conditions, standards, time requirements, and evaluation criteria.

1-2. Training Philosophy

MSS training is progressive and role-based. Bypassing foundational training (SL 1) to accelerate to SL 3 produces personnel who can operate tools without understanding what they are doing — a data governance and analytical risk.

Principle	Implication
Prerequisites are not suggestions	Commanders must verify completion before assigning personnel to advanced levels
Role determines training level	Assign based on actual job function, not desire or seniority

Principle	Implication
Practical exercise is the standard	Training is complete only when the task is performed to standard
Sustainment is required	MSS proficiency degrades within 6 months without use or refresher training

1-3. Training Levels Summary

Level	Publication	Audience	Days	Prerequisite
SL 1	Maven User	All personnel	1	None
SL 2	Builder	All staff	5	SL 1
SL 3	Advanced Builder	Data-adjacent specialists	5	SL 2
SL 4A	Intelligence WFF	G2/S2 Intel staff	3	SL 3 (Required)
SL 4B	Fires WFF	Fires/FSCOORD staff	3	SL 3 (Required)
SL 4C	Movement & Maneuver WFF	G3/S3 M2 staff	3	SL 3 (Required)
SL 4D	Sustainment WFF	G4/S4 SUST staff	3	SL 3 (Required)
SL 4E	Protection WFF	Protection staff	3	SL 3 (Required)
SL 4F	Mission Command WFF	MC/G6/S6 staff	3	SL 3 (Required)
SL 4G	ORSA	ORSA analysts	5	SL 3 (Required) + quant background
SL 4H	AI Engineer	AI/ML specialists	5	SL 3 (Required) + coding background
SL 4M	ML Engineer	MLEs	5	SL 3 (Required) + coding background
SL 4J	Program Manager	PMs, G8/S8	4	SL 3 (Required)
SL 4K	Knowledge Manager	KMOs, 37F	4	SL 3 (Required)
SL 4L	Software Engineer	SWEs	5	SL 3 (Required) + coding background
SL 4N	UI/UX Designer	UI/UX designers	5	SL 3 (Required) + design tool familiarity

Level	Publication	Audience	Days	Prerequisite
SL 4O	Platform Engineer	Platform engineers	5	SL 3 (Required) + Linux sysadmin + containers/Git
SL 5G	Advanced ORSA	Senior ORSA analysts	5	SL 4G (Required)
SL 5H	Advanced AI Engineer	Senior AI engineers	5	SL 4H (Required)
SL 5M	Advanced ML Engineer	Senior MLEs	5	SL 4M (Required)
SL 5J	Advanced Program Manager	Senior PMs	3	SL 4J (Required)
SL 5K	Advanced Knowledge Manager	Senior KMOs	3	SL 4K (Required)
SL 5L	Advanced Software Engineer	Senior SWEs	5	SL 4L (Required)
SL 5N	Advanced UI/UX Designer	Senior UI/UX designers	3	SL 4N (Required) + 2+ MSS design portfolio
SL 5O	Advanced Platform Engineer	Senior platform engineers	3	SL 4O (Required) + operational experience
T3-I	Instructor Certification	C2DAO-selected instructor candidates	5 + practicum	SL 3 (Required) + C2DAO selection
T3-F	MSC Force Multiplier	MSC-level data NCOs, S6, UDT nominees	0.5	SL 2 (Required) + CDR nomination

NOTE

SL 4A–F (WFF functional tracks) require SL 1, SL 2, and SL 3 as hard prerequisites. SL 4G–O (specialist tracks) also require SL 3 as a hard prerequisite. SL 5G–O (advanced specialist tracks) require the corresponding SL 4G–O. There are NO SL 5A–F tracks.

NOTE

"Training days" reflects initial qualification training with instructor support. Self-paced completion may require more time. Refresher/sustainment training is shorter — see Section 11.

FOUNDRY BOOTCAMP (FBC): A quarterly 5-day applied build event that operates **outside** the SL 1 through SL 5 training chain. Prereq: SL 2 Go on file + command-validated project. Does not grant SL 3 credit. Does not unlock SL 4 enrollment. Governed by FBC-SOP-001 (training_management/FOUNDRY_BOOTCAMP_SOP.md). Appears in training records as a separate event type, not as a TM course completion.

TRAIN-THE-TRAINER (T3): T3-I (Instructor Certification) and T3-F (MSC Force Multiplier) operate **outside** the SL 1 through SL 5 training chain. T3-I certifies C2DAO instructors; T3-F certifies Unit Data Trainers at the MSC level. T3-I is offered semi-annually; T3-F is offered quarterly aligned with MTT rotations. See Instructor Tier Definitions, Unit Data Trainer SOP, and MTT Operations SOP for employment and sustainment guidance.

AUTHORITATIVE REFERENCES

Publication	Title	Relevance
AR 350-1	Army Training and Leader Development	Master regulation for Army training policy; governs all institutional training programs
TR 350-70	Army Learning Policy and Systems	TRADOC master regulation governing training program design, course administration, and learning products
ADP 7-0	Training	Army training doctrine; establishes principles for training management across the force
FM 7-0	Training	Unit training management procedures; provides commander guidance for planning, executing, and assessing training

NOTE

TR 350-70 is published by TRADOC at adminpubs.tradoc.army.mil, not DA APD.

2. TRAINING OBJECTIVES BY LEVEL

2-1. SL 1 — Maven User

TLO: Access, navigate, and consume data products on MSS in support of operational requirements.

ELO	Description	Evaluation Method
10-01	Log in via CAC authentication and navigate to unit projects	Practical: observer validates login and navigation
10-02	Locate and view a dataset in a Workshop application	Practical: open designated application, filter to unit data
10-03	Execute an authorized Action in a Workshop application	Practical: submit status update via designated Action
10-04	Use Contour to build a basic chart and filter	Practical: create filtered bar chart from provided dataset
10-05	Use Quiver to explore an Object Type	Practical: navigate to assigned Object Type, apply filter, export view
10-06	Access and interact with an AIP Logic workflow or Agent interface	Practical: submit query to designated AIP interface
10-07	Identify classification markings and state authorized export procedures	Written/verbal: answer 3 of 4 classification scenario questions correctly
10-08	Troubleshoot at least 2 common access issues using SL 1 procedures	Practical: diagnose seeded access problem from troubleshooting annex

2-2. SL 2 — Builder

TLO: Build, configure, and publish Foundry data products on MSS using visual (no-code) platform tools.

ELO	Description	Evaluation Method
20-01	Create a Foundry project with correct naming, markings, and folder structure	Practical: inspector validates against naming convention checklist
20-02	Build a Pipeline Builder pipeline that ingests and transforms a provided dataset	Practical: pipeline runs; output schema validated

ELO	Description	Evaluation Method
20-03	Create an Object Type and populate it via pipeline output	Practical: Object Type in Ontology Manager with correct properties
20-04	Create a Link Type connecting two Object Types	Practical: link traversable in Quiver
20-05	Configure a basic Action that writes to an Object Type property	Practical: execute Action; verify property updated
20-06	Build and publish a Workshop application with table, filter, and Action button	Practical: application accessible to test users; table and filter functional
20-07	Manage project access (grant Viewer role to test user)	Practical: test user confirms Viewer access; cannot edit
20-08	Create a Foundry branch, make a change, and submit for promotion	Practical: branch created; change visible; promotion request submitted

2-3. SL 3 — Advanced Builder

TLO: Design and build complex, multi-source data products on MSS using advanced platform capabilities in accordance with C2DAO governance standards.

ELO	Description	Evaluation Method
30-01	Build a multi-page Workshop application with conditional logic and variable passing	Practical: application behaves as designed across 3+ pages; evaluator tests 5 conditional scenarios
30-02	Build a Pipeline Builder pipeline with multi-source join and computed aggregation	Practical: output validated against known-answer dataset
30-03	Design an Ontology schema (Object Types, Link Types, Actions) — documented design	Design review: evaluated against 6-item rubric (see Appendix C)
30-04	Conduct advanced Contour analysis: pivot, calculated columns, and saved view	Practical: Contour workbook reproduces provided reference output
30-05	Build a multi-object Quiver dashboard with linked views	Practical: filters propagate across linked views; evaluated by observer
30-06	Configure an AIP Logic workflow (configure inputs/outputs/triggers, not author)	Practical: workflow triggers correctly; routes output to designated dataset
30-07	Interpret a data lineage graph for a provided dataset	Verbal/written: correctly identify upstream sources, transforms, and consumers
30-08	Complete the full C2DAO promotion workflow: branch, test, submit, receive approval	Practical: evaluator plays data steward; trainee executes end-to-end

2-4. SL 4A through SL 4F — WFF Functional Tracks

TLO (common): Apply MSS data products to support warfighting function mission requirements; build and operate WFF-specific dashboards, pipelines, and ontology-backed tracking systems.

Each WFF track shares a common task structure adapted to WFF-specific data and mission context. Prerequisite: SL 1, SL 2, and SL 3 (all required).

Track	WFF	Audience	Key Focus Areas
SL 4A	Intelligence	G2/S2 Intel staff	PMESII analysis, ISR tracking, threat object ontology
SL 4B	Fires	Fires/FSCOORD staff	Target tracking, fire mission pipelines, effects dashboards
SL 4C	Movement & Maneuver	G3/S3 M2 staff	FRAGORD tracking, battle rhythm apps, unit disposition
SL 4D	Sustainment	G4/S4 SUST staff	Class IX tracking, maintenance dashboards, logistics pipelines
SL 4E	Protection	Protection staff	Risk assessment tracking, force protection dashboards
SL 4F	Mission Command	MC/G6/S6 staff	Battle rhythm management, comms status, commander's dashboard

Common ELOs for all WFF tracks:

ELO	Description	Evaluation Method
40WFF-01	Build a WFF-specific Pipeline Builder pipeline from provided exercise data	Practical: pipeline runs; output schema correct for WFF use case
40WFF-02	Create WFF Object Types and populate via pipeline	Practical: Object Types visible in Ontology Manager; properties correctly typed
40WFF-03	Configure a Workshop application with WFF-specific filters and status indicators	Practical: application displays correct data; filters functional
40WFF-04	Configure an Action to support a WFF workflow (e.g., status update, submission)	Practical: Action executes; correct property updated
40WFF-05	Build a multi-page WFF dashboard (summary → unit detail)	Practical: variable passing works; Page 2 filters by Page 1 selection
40WFF-06	Apply data governance: classification, naming, branching, and promotion workflow	Practical: product meets C2DAO naming and marking standards; promotion submitted

2-5. SL 4J — Program Manager

TLO: Design and operate program management data products on MSS that support commander decision-making for schedule, resource, and portfolio health.

ELO	Description	Evaluation Method
40J-01	Design a program data architecture (4 Object Types) for a provided scenario	Design review: evaluated against PM data model rubric
40J-02	Build a milestone tracking pipeline from a provided IMS spreadsheet	Practical: pipeline output validates against known-answer milestone set
40J-03	Build a Workshop milestone dashboard with RAG status and data-as-of timestamp	Practical: dashboard displays correct status; timestamp present
40J-04	Build a budget execution Quiver visualization showing obligation rate vs. target	Practical: chart displays correctly with reference line at quarterly target
40J-05	Configure a snapshot pipeline in Append mode for historical trend analysis	Practical: run pipeline 2× and verify cumulative records
40J-06	Produce an IPR product from MSS meeting the PM Dashboard Standards Checklist	Product review: evaluated against TM-40J Appendix A

2-6. SL 4K — Knowledge Manager

TLO: Design and operate knowledge management systems on MSS that capture, organize, and deliver institutional knowledge in support of unit continuity and mission effectiveness.

ELO	Description	Evaluation Method
40K-01	Design a knowledge ontology (5+ Object Types) for a provided unit KM scenario	Design review: evaluated against knowledge architecture checklist
40K-02	Configure a Workshop AAR submission form that writes to the AAR Object Type	Practical: submitter test confirms form writes correctly; required fields enforced
40K-03	Configure a lessons-learned intake pipeline with tagging and distribution logic	Practical: pipeline run with provided test data; output reviewed for accuracy
40K-04	Configure an AIP Logic summarization workflow for document intake	Practical: workflow processes a provided document; output reviewed by SME
40K-05	Build a knowledge browser application with search, filter, and drill-down	Practical: evaluator submits 5 test queries; application returns correct results
40K-06	Build and demonstrate a PCS knowledge transfer package for a specific role	Product review: package contains complete documentation per TM-40K Ch 9

2-7. SL 4G — ORSA

TLO: Conduct operations research and systems analysis on MSS, producing quantitatively rigorous decision support products meeting commander and ORSA product standards.

ELO	Description	Evaluation Method
40G-01	Configure a Code Workspace with required packages and verify Foundry dataset connectivity	Practical: workspace runs provided test script without errors
40G-02	Build and validate a regression model for a provided readiness dataset	Practical: model results reviewed against reference solution; validation statistics correct
40G-03	Build a time series forecast with documented assumptions and confidence bounds	Practical: forecast output reviewed; uncertainty quantification present
40G-04	Run a Monte Carlo simulation for a provided COA comparison scenario	Practical: output distribution and confidence intervals reviewed by ORSA evaluator
40G-05	Formulate and solve a linear programming resource allocation problem	Practical: LP solution reviewed; constraint formulation documented
40G-06	Produce a commander brief including point estimates with uncertainty bounds	Product review: brief reviewed against ORSA product standards checklist

2-8. SL 4H — AI Engineer

TLO: Author, test, and deploy AI-enabled capabilities on MSS using AIP Logic, Agent Studio, and LLM integration patterns in compliance with USAREUR-AF AI safety requirements.

ELO	Description	Evaluation Method
40H-01	Author an AIP Logic workflow with prompt engineering, conditional chain logic, and structured JSON output	Practical: workflow runs against provided dataset; output schema validates
40H-02	Configure and test an AIP Agent Studio agent with at least two tools and defined memory scope	Practical: agent responds correctly to 5 evaluator queries; tool calls logged
40H-03	Build an LLM integration pipeline with ontology grounding and RAG	Practical: pipeline retrieves correct ontology context; evaluator validates output grounding
40H-04	Implement human-in-the-loop checkpoints for all write-capable Actions in a workflow	Practical: evaluator confirms no Action executes without a visible review/confirm step
40H-05	Write Python transforms that extract and format Ontology data for AIP Logic context	Practical: transform output matches expected schema; runs without error

ELO	Description	Evaluation Method
40H-06	Complete the AIP Authorization Checklist (TM-40H Appendix A) for a proposed workflow; identify prohibited use cases	Written review: evaluator scores checklist completion and prohibited-use identification

2-9. SL 4M — Machine Learning Engineer

TLO: Build, evaluate, deploy, and govern machine learning models on MSS using Foundry Code Workspaces and MLOps tooling in compliance with USAREUR-AF model governance standards.

ELO	Description	Evaluation Method
40M-01	Configure Code Workspace with required packages, GPU allocation, and Foundry dataset connectivity	Practical: provided test script runs without errors; dataset connection confirmed
40M-02	Build a feature engineering pipeline for a provided Foundry dataset meeting documented feature standards	Practical: output dataset reviewed against feature specification; no null leakage
40M-03	Train and evaluate a supervised model meeting defined accuracy, precision/recall, and calibration thresholds	Practical: evaluation metrics reviewed by technical evaluator against acceptance criteria
40M-04	Deploy a trained model to a Foundry model serving endpoint and verify live inference	Practical: inference endpoint returns predictions for 10 test records; latency within spec
40M-05	Implement a model monitoring pipeline with drift detection and alert configuration	Practical: pipeline detects seeded drift event; alert routes correctly
40M-06	Complete a model governance document meeting USAREUR-AF model documentation standards	Product review: document reviewed against SL 4M governance checklist

2-10. SL 4L — Software Engineer

TLO: Build, test, and deploy production-quality software applications and integrations on MSS using OSDK, Platform SDK, TypeScript Functions on Objects, and Slate, following USAREUR-AF code review and deployment standards.

ELO	Description	Evaluation Method
40L-01	Authenticate to Foundry Ontology via OSDK; execute a paginated, filtered object query	Practical: query returns correct records; pagination handles >1 page
40L-02	Execute an Action via OSDK with full validation logic and structured error handling	Practical: valid Action executes; invalid input triggers correct error response

ELO	Description	Evaluation Method
40L-03	Build a TypeScript Function on Objects implementing a computed property	Practical: computed property returns correct values for 10 test objects; edge cases handled
40L-04	Write and test a TypeScript Action validator with at least 3 distinct validation conditions	Practical: evaluator tests 5 scenarios; validator passes/blocks correctly in all cases
40L-05	Build a Slate application integrated with the Foundry API displaying live ontology data	Practical: application renders correctly; data refreshes on state change
40L-06	Complete a C2DAO code review and deployment workflow for a provided OSDK application	Practical: PR created, review comments addressed, deployment checklist completed end-to-end

NOTE

Developer track evaluations (SL 4H, 40M, 40L, 50H, 50M, 50L) require a qualified technical reviewer: SL 4 certified instructor, C2DAO data engineer, or equivalent. Do not evaluate developer track ELOs using a non-technical observer.

2-11. SL 4N — UI/UX Designer

TLO: Design, prototype, test, and deliver user-centered operational data product interfaces on MSS using Workshop design patterns, accessibility standards, and C2DAO design review processes.

ELO	Description	Evaluation Method
40N-01	Conduct user research (stakeholder interviews and task analysis) for a provided operational scenario	Written review: research findings document evaluated against user research checklist
40N-02	Create wireframes and information architecture for a multi-page Workshop application	Design review: wireframes evaluated against information architecture rubric
40N-03	Build a high-fidelity interactive prototype in Workshop with conditional visibility and state transitions	Practical: prototype demonstrates required interactions; evaluator tests 5 user flows
40N-04	Conduct a user test with structured task scenarios and document findings with prioritized recommendations	Product review: test plan, observation notes, and findings document evaluated
40N-05	Create a design system artifact (component library with documented patterns and naming conventions)	Product review: design system evaluated against completeness and developer handoff standards

ELO	Description	Evaluation Method
40N-06	Submit design deliverables through C2DAO design review process with accessibility compliance documentation	Practical: deliverables meet C2DAO standards; accessibility checklist completed

2-12. SL 4O — Platform Engineer

TLO: Deploy, monitor, secure, and maintain infrastructure supporting MSS operational data products using Kubernetes, CI/CD pipelines, and C2DAO infrastructure standards.

ELO	Description	Evaluation Method
40O-01	Configure a Kubernetes deployment with correct namespace, resource quotas, and pod lifecycle management	Practical: deployment runs; resource limits enforced; pod health checks passing
40O-02	Execute a deployment strategy (rolling update) with zero-downtime verification and rollback capability	Practical: evaluator triggers rollback; service maintains availability throughout
40O-03	Build a monitoring and alerting configuration with metrics collection, log aggregation, and threshold alerts	Practical: monitoring dashboard displays correct metrics; alert fires on seeded anomaly
40O-04	Implement security hardening: network policies, RBAC, secrets management, and audit logging	Practical: evaluator attempts unauthorized access; policies block correctly
40O-05	Build a CI/CD pipeline with automated testing gates, vulnerability scanning, and promotion environments	Practical: pipeline executes end-to-end; gate blocks deployment on test failure
40O-06	Complete C2DAO infrastructure deployment checklist with change management documentation	Written review: checklist evaluated for completeness; change documentation meets standards

3. INDIVIDUAL TASK LIST

3-1. SL 1 Tasks (All Personnel)

Task ID	Task	Go Standard
MSS-10-T01	Access MSS via CAC	Successfully authenticates; navigates to unit project

Task ID	Task	Go Standard
MSS-10-T02	Navigate to a named dataset in Workshop	Locates dataset within 3 minutes without assistance
MSS-10-T03	Execute an Action	Action executes without error; property updated
MSS-10-T04	Filter a Workshop table and export the result	Filter applied; export completed; data correct
MSS-10-T05	Identify the classification marking on a dataset	States marking correctly; states authorized export procedure
MSS-10-T06	Submit a basic Contour chart	Chart displays correct data per provided parameters
MSS-10-T07	Request access to a missing resource	Submits correct access request to unit MSS administrator

3-2. SL 2 Tasks (Builder Track)

Task ID	Task	Go Standard
MSS-20-T01	Create a Foundry project per naming convention	Project created; naming matches convention; markings applied
MSS-20-T02	Build a Pipeline Builder pipeline (single source → output)	Pipeline runs; output schema and row count validated
MSS-20-T03	Create an Object Type with 5+ properties	Object Type visible in Ontology Manager; properties correctly typed
MSS-20-T04	Build and test an Action	Action executes; writes correct value; audit log confirmed
MSS-20-T05	Build a Workshop application (table + filter + Action)	Application published; filter works; Action accessible to authorized role
MSS-20-T06	Manage project access roles	Viewer access confirmed; Editor cannot elevate to Owner
MSS-20-T07	Execute a branching workflow	Branch created; change made; promotion requested

3-3. SL 3 Tasks (Advanced Builder)

Task ID	Task	Go Standard
MSS-30-T01	Build a multi-page Workshop application	3+ pages; conditional logic demonstrated; variable passing validated
MSS-30-T02	Build a multi-source Pipeline Builder join	Join produces correct output; row counts validated
MSS-30-T03	Document an Ontology design for a provided scenario	Design document covers all required elements; reviewed by instructor
MSS-30-T04	Build a Contour workbook with pivot and calculated columns	Output matches reference; pivot correct
MSS-30-T05	Build a multi-object Quiver dashboard	Linked views filter correctly; evaluated by observer
MSS-30-T06	Manage the C2DAO promotion workflow	Workflow completed end-to-end with evaluator as data steward

3-4. SL 4A–F Tasks (WFF Functional Tracks — common task set, WFF-adapted scenarios)

Task ID	Task	Go Standard
MSS-40WFF-T01	Build a WFF-specific pipeline from provided exercise data	Pipeline runs without error; output schema correct
MSS-40WFF-T02	Create WFF Object Types and populate via pipeline write step	Object Types populated; properties correctly typed; row count matches
MSS-40WFF-T03	Configure a Workshop application with WFF filters and status indicators	Application displays live data; filters functional; classification marking present
MSS-40WFF-T04	Configure an Action to support a WFF workflow	Action executes; correct Object property updated; access restriction enforced
MSS-40WFF-T05	Build a multi-page WFF dashboard	Page 1 → Page 2 variable passing works; Page 2 data filtered by Page 1 selection

Task ID	Task	Go Standard
MSS-40WFF-T06	Apply C2DAO governance: naming, marking, branching, and promotion	Product meets naming standards; promotion request submitted with complete description

3-5. SL 4N Tasks (UI/UX Designer)

Task ID	Task	Go Standard
MSS-40N-T01	Conduct user research for a provided operational scenario	Research document complete; findings actionable; stakeholder needs identified
MSS-40N-T02	Create wireframes and information architecture for a Workshop application	Wireframes cover all required pages; navigation hierarchy documented
MSS-40N-T03	Build a high-fidelity interactive prototype with conditional visibility	Prototype renders correctly; 5 user flows pass evaluator testing
MSS-40N-T04	Conduct user test and document findings with prioritized recommendations	Test plan followed; findings documented with severity ratings; top 3 fixes identified
MSS-40N-T05	Create a design system component library with documentation	Component library contains all required elements; naming conventions followed
MSS-40N-T06	Submit deliverables through C2DAO design review with accessibility compliance	All deliverables submitted; accessibility checklist complete; no critical failures

3-6. SL 4O Tasks (Platform Engineer)

Task ID	Task	Go Standard
MSS-40O-T01	Configure a Kubernetes deployment with namespace, quotas, and health checks	Deployment running; resource limits enforced; health checks passing
MSS-40O-T02	Execute a rolling update with zero-downtime and rollback	Update completes; no service interruption; rollback successful on command
MSS-40O-T03	Build monitoring and alerting with metrics dashboard and threshold alerts	Dashboard displays correct metrics; alert fires on seeded event
MSS-40O-T04	Implement security hardening (network policies, RBAC, secrets management)	Unauthorized access attempts blocked; audit log captures events
MSS-40O-T05	Build a CI/CD pipeline with testing gates and promotion environments	Pipeline runs end-to-end; gate blocks on test failure

Task ID	Task	Go Standard
MSS-40O-T06	Complete deployment checklist with change management documentation	Checklist complete; change documentation meets C2DAO standards

4. TRAINING STANDARDS AND GO/NO-GO CRITERIA

4-1. Go/No-Go Standard

Training Level	Go Standard
SL 1	Pass 6 of 7 tasks (T01–T07); pass classification questions (3 of 4)
SL 2	Pass 6 of 7 tasks (T01–T07); Pipeline Builder pipeline runs without error
SL 3	Pass 5 of 6 tasks (T01–T06); Ontology design document reviewed and approved
SL 4A–F (WFF)	Pass all 6 WFF tasks (T01–T06); WFF dashboard meets standard; data governance requirements met
SL 4G	Pass 5 of 6 ELOs; ORSA product brief reviewed by qualified ORSA evaluator
SL 4H/M/L	Pass developer track practical exercise; evaluated by SL 4 certified technical reviewer
SL 4J	Pass 5 of 6 ELOs; IPR product meets all PM Dashboard Standards Checklist items
SL 4K	Pass 5 of 6 ELOs; PCS transfer package reviewed and approved by instructor
SL 4N	Pass 5 of 6 ELOs; design portfolio reviewed by qualified UX evaluator; accessibility checklist complete
SL 4O	Pass 5 of 6 ELOs; deployment verified by qualified infrastructure evaluator; monitoring operational

4-2. No-Go Actions

A trainee who does not meet the Go standard: 1. Receives specific written feedback on which task(s) failed and what was deficient. 2. Receives a remediation assignment (specific TM section, specific procedure to practice). 3. Completes a re-evaluation on the failed task(s) within 5 duty days. 4. May not advance to the next TM level until the current level Go standard is met.

NOTE

No-Go at SL 1 often indicates an access provisioning issue (CAC enrollment, account not created) — not a training failure. Distinguish between access failures and competency failures. Fix access issues before assessing competency.

4-3. Evaluation Documentation

Training evaluations are documented in the Unit Training Status Matrix (Appendix A). Record at minimum:

Field	Description
Trainee name and unit	—
TM level evaluated	—
Date of evaluation	—
Go/No-Go result	—
Evaluator name and TM certification level	—
Re-evaluation date (if applicable)	—

5. BLOCKS OF INSTRUCTION — TIME REQUIREMENTS

5-1. SL 1 — 1 Training Day (8 hours)

Block	Time	Content	Method
1	0800–0900	MSS overview; data literacy fundamentals (Data Literacy Technical Reference Ch 1 summary)	Lecture/discussion
2	0900–1000	MSS login, navigation, project access	Instructor-led lab
3	1000–1100	Workshop applications: consuming data, tables, filters	Instructor-led lab
4	1100–1200	Actions: executing status updates and form submissions	Instructor-led lab
—	1200–1300	Lunch	—

Block	Time	Content	Method
5	1300–1400	Contour: basic chart and filter	Guided practice
6	1400–1500	Quiver: exploring Object Types	Guided practice
7	1500–1530	AIP interface overview and demonstration	Demonstration
8	1530–1600	Classification markings and export procedures	Lecture/scenario
9	1600–1700	Practical exercise (all tasks)	Evaluation

5-2. SL 2 — 5 Training Days (40 hours)

Day 1 — Project Fundamentals and File Ingestion

Block	Time	Content	Method
—	0800–0830	SL 2 overview; what trainees will build by Day 5; course standards and Go criteria	Brief
1	0830–1000	Project creation: naming conventions, classification markings, folder structure	Instructor-led lab
—	1000–1015	Break	—
2	1015–1100	File ingestion: upload a CSV; inspect schema, types, and row count	Instructor-led lab
3	1100–1200	Dataset explorer: column profiling, null detection, type mismatches	Instructor-led lab
—	1200–1300	Lunch	—
4	1300–1500	Pipeline Builder orientation: canvas layout, step library, input/output dataset config	Instructor-led lab
5	1500–1530	C2DAO naming conventions: datasets, pipelines, Object Types	Lecture/discussion
6	1530–1700	Individual practice: create a second project, ingest a provided dataset, confirm naming compliance	Guided practice

Day 2 — Pipeline Builder: Clean and Transform

Block	Time	Content	Method
—	0800–0830	Review: Day 1 questions; confirm all trainees have projects and ingested data	Review
7	0830–1030	Pipeline: filter step, rename step, CAST for type correction	Instructor-led lab
—	1030–1045	Break	—
8	1045–1200	Pipeline: calculated columns — string functions, conditional logic (IF/CASE), COALESCE for nulls	Instructor-led lab
—	1200–1300	Lunch	—
9	1300–1500	Pipeline: date and time functions — DATEDIFF, DATE_TRUNC, CURRENT_DATE; test with known-answer records	Instructor-led lab
—	1500–1515	Break	—
10	1515–1700	End-to-end pipeline practice: build a complete clean-and-transform pipeline from raw input to typed filtered output; run and verify	Guided practice

Day 3 — Pipeline Builder: Joins and Ontology Manager

Block	Time	Content	Method
—	0800–0830	Review: Day 2 questions; pipeline troubleshooting — error messages, schema mismatches	Review
11	0830–1030	Pipeline: join step — inner/left join, key selection, handling duplicates post-join, output column selection	Instructor-led lab
—	1030–1045	Break	—
12	1045–1200	Pipeline: group-by aggregation; union step basics; output dataset configuration (overwrite vs. append)	Instructor-led lab
—	1200–1300	Lunch	—
13	1300–1500	Ontology Manager: create an Object Type — properties, types, Primary Key, display name expression	Instructor-led lab

Block	Time	Content	Method
—	1500–1515	Break	—
14	1515–1630	Ontology Manager: create a Link Type — connecting two Object Types, cardinality, directionality	Instructor-led lab
15	1630–1700	Ontology practice: design a second Object Type from a provided scenario; verify naming compliance	Guided practice

Day 4 — Ontology Write Step, Actions, and Workshop Applications

Block	Time	Content	Method
—	0800–0845	Review: Day 3 questions; access control model: Viewer vs. Editor roles	Review/discussion
16	0845–0945	Pipeline: Ontology write step — connect pipeline output to Object Type; configure property mapping; run and verify	Instructor-led lab
—	0945–1000	Break	—
17	1000–1130	Actions: create a basic Action — parameter, write rule, access restriction; test from Ontology Manager	Instructor-led lab
—	1130–1145	Break	—
18	1145–1300	Workshop orientation: canvas, widget library, Object Type binding — table widget with live data	Instructor-led lab
—	1300–1400	Lunch	—
19	1400–1530	Workshop: filter widget, metric widget, bar chart widget — layout and data source configuration	Instructor-led lab
—	1530–1545	Break	—
20	1545–1700	Workshop: connecting an Action button — trigger, confirmation prompt, post-action refresh	Instructor-led lab

Day 5 — Publishing, Governance, and Practical Exercise

Block	Time	Content	Method
—	0800–0845	Review: Day 4 questions; Workshop publishing and access configuration	Review

Block	Time	Content	Method
21	0845–1000	Workshop: publishing, access control — grant Viewer and Editor roles; confirm role behavior	Instructor-led lab
—	1000–1015	Break	—
22	1015–1130	Foundry branching: create, make change, submit for data steward promotion with complete description	Instructor-led lab
—	1130–1200	Review: C2DAO governance standards and Go/No-Go criteria for practical exercise	Review
—	1200–1300	Lunch	—
23	1300–1700	Practical exercise (all SL 2 tasks)	Evaluation

5-3. SL 3 — 5 Training Days (40 hours)

Day 1 — Advanced Workshop

Block	Time	Content	Method
—	0800–0830	SL 3 overview; what trainees will build by Day 5; design-first methodology	Brief
1	0830–1030	Multi-page Workshop: page navigation, parameter configuration, URL deep links	Instructor-led lab
—	1030–1045	Break	—
2	1045–1200	Conditional logic: show/hide panels, conditional formatting on tables, dynamic widget visibility rules	Instructor-led lab
—	1200–1300	Lunch	—
3	1300–1500	Variable passing: passing object selections between pages; filtered detail views driven by page-1 selection	Instructor-led lab
—	1500–1515	Break	—
4	1515–1700	Design exercise: build a 3-page operations dashboard (portfolio → unit detail → historical trend); instructor critique	Guided practice

Day 2 — Advanced Pipeline Builder

Block	Time	Content	Method
—	0800–0830	Review: Day 1 questions; Workshop design critique debrief	Review
5	0830–1030	Multi-source joins: inner/left/outer, handling fan-out after join, post-join deduplication patterns	Instructor-led lab
—	1030–1045	Break	—
6	1045–1200	Union transforms: combining datasets with compatible schemas, handling schema mismatches	Instructor-led lab
—	1200–1300	Lunch	—
7	1300–1500	Group-by aggregations: count, sum, min/max, computed aggregation columns; aggregate-then-join patterns	Instructor-led lab
—	1500–1515	Break	—
8	1515–1630	Output configuration: overwrite vs. append mode; append mode for snapshot pipelines and historical records	Instructor-led lab
9	1630–1700	Scheduled pipeline configuration: schedule expression, build failure alert setup	Instructor-led lab

Day 3 — Ontology Design

Block	Time	Content	Method
—	0800–0900	Review: Day 2 questions; common pipeline errors from the lab	Review
10	0900–1000	Ontology design methodology: domain analysis, entity identification, relationship mapping, Action design	Lecture
—	1000–1015	Break	—
11	1015–1200	Individual design exercise: translate a provided mission requirement to a documented Ontology schema (Object Types, Link Types, cardinality, Actions)	Guided practice
—	1200–1300	Lunch	—

Block	Time	Content	Method
12	1300–1500	Design critique: each trainee presents schema; class critiques against the 6-item design rubric; instructor facilitates	Workshop
—	1500–1515	Break	—
13	1515–1700	Build the approved design: create the Ontology from the Day 3 design exercise; connect pipeline output via Ontology write step	Instructor-led lab

Day 4 — Analytics Tools and AIP Logic

Block	Time	Content	Method
—	0800–0830	Review: Day 3 questions; common Ontology build errors	Review
14	0830–1030	Contour: pivot tables, calculated columns, parameter controls, workbook structure, saving and sharing analysis views	Instructor-led lab
—	1030–1045	Break	—
15	1045–1200	Quiver: multi-object analysis, linked views, cross-filter propagation, drilling between Object Types	Instructor-led lab
—	1200–1300	Lunch	—
16	1300–1430	AIP Logic configuration: connecting triggers, inputs, outputs; human review queue design	Instructor-led lab
—	1430–1445	Break	—
17	1445–1600	Data lineage: reading a lineage graph; identifying upstream sources, transforms, and downstream consumers; using lineage to diagnose pipeline issues	Instructor-led lab
18	1600–1700	C2DAO production standards: what constitutes a production-ready data product; quality gates	Lecture/discussion

Day 5 — Governance and Practical Exercise

Block	Time	Content	Method
19	0800–0900	Full C2DAO promotion workflow: branch → change → description → submit → respond to feedback → approval	Instructor-led lab

Block	Time	Content	Method
20	0900–1000	Full-stack review: trace raw source → pipeline → Ontology → Workshop → governance; identify production-readiness gaps	Review
—	1000–1015	Break	—
—	1015–1100	Practical exercise scenario brief; design planning time — document Ontology schema on paper before touching the platform	Brief
—	1100–1130	Design review: evaluator reviews Ontology design against 6-item rubric; trainees with fatal design flaws correct before proceeding	Evaluation
—	1130–1200	Buffer: open lab — resolve any tool access or environment issues	Buffer
—	1200–1300	Lunch	—
21	1300–1700	Practical exercise (all SL 3 tasks)	Evaluation

5-4. SL 4A–F — WFF Functional Tracks (3 Days each; prereq: SL 3)

Each WFF track runs 3 days (24 hours). The schedule below applies to all 6 tracks (A–F). Scenario content and Object Type design vary by WFF; the instructional sequence and evaluation structure are identical.

Day 1 — Data Architecture and Pipeline Fundamentals

Block	Time	Content	Method
1	0800–0900	WFF role on MSS; WFF-specific data products and use cases; course overview	Lecture/discussion
2	0900–1100	WFF Object Type design: key entities, properties, primary keys; document before building	Instructor-led lab
3	1100–1200	Pipeline Builder: ingest provided WFF dataset; clean and transform	Instructor-led lab
—	1200–1300	Lunch	—
4	1300–1500	Pipeline: join step, computed status columns, RAG logic, Ontology write step	Instructor-led lab
5	1500–1700	Individual practice: second pipeline build using alternate provided dataset	Guided practice

Day 2 — Workshop Applications and Actions

Block	Time	Content	Method
6	0800–1000	Workshop: WFF summary dashboard — table, filter widgets, status indicators	Instructor-led lab
7	1000–1200	Workshop: multi-page application — Page 1 summary → Page 2 unit detail (variable passing)	Instructor-led lab
—	1200–1300	Lunch	—
8	1300–1500	Actions: configure status update and submission Actions for WFF workflow	Instructor-led lab
9	1500–1700	Access control, classification markings, WFF-specific export procedures	Lecture/discussion

Day 3 — Governance and Practical Exercise

Block	Time	Content	Method
10	0800–1000	C2DAO naming conventions, branching, promotion workflow — WFF application context	Instructor-led lab
11	1000–1200	Supervised practice run: rebuild the prior day's dashboard with a new dataset	Guided practice
—	1200–1300	Lunch	—
12	1300–1400	Practical exercise scenario brief	BRF
13	1400–1700	Practical exercise (all 6 tasks)	Evaluation

5-5. SL 4J — Program Manager (4 Days)

Day 1 — Agile Fundamentals and User Story Craft

Block	Time	Content	Method
1	0800–0830	The Technical PM role on MSS; connecting operational requirements to technical execution	Brief
2	0830–1000	Agile for data projects: Scrum framework, sprint cadence, backlog management; applied exercise: story sizing	Instructor-led lab

Block	Time	Content	Method
—	1000–1015	Break	—
3	1015–1200	User stories and acceptance criteria: format, quality standards, Definition of Ready; applied exercise: rewrite requirements	Instructor-led lab
—	1200–1300	Lunch	—
4	1300–1500	Kanban for operational support; sprint ceremonies — military data team execution standards; sprint planning exercise	Instructor-led lab
—	1500–1515	Break	—
5	1515–1700	User story quality workshop: peer review against Definition of Ready; common failure patterns	Workshop

Day 2 — ML/AI Lifecycle and Requirements Elicitation

Block	Time	Content	Method
—	0800–0900	Day 1 review; gate criteria quiz — six ML lifecycle phases	Review
6	0900–1030	ML/AI project lifecycle: six phases, gate criteria, cross-track handoffs, common gate-skip failure modes	Lecture
—	1030–1045	Break	—
7	1045–1200	Requirements elicitation from commanders and staff: structured intake, translation problem; interview simulation	Instructor-led lab
—	1200–1300	Lunch	—
8	1300–1430	Requirements document drill: problem statement, success criteria, constraints, definition of done; peer review	Instructor-led lab
—	1430–1445	Break	—
9	1445–1700	Stakeholder expectations: delivery timelines, scope tradeoffs, cross-track coordination; PM vs. technical authority	Lecture/discussion

Day 3 — Project Tracking Systems and Risk Management

Block	Time	Content	Method
—	0800–0830	Day 2 review; requirements document quality debrief	Review
10	0830–1030	Project tracking system architecture: Ontology design, Sprint Board specification, PM requirements spec	Instructor-led lab
—	1030–1045	Break	—
11	1045–1200	Commander-facing project status dashboard: health roll-up, milestone status, blocking issues; automated alerts	Instructor-led lab
—	1200–1300	Lunch	—
12	1300–1500	Risk management for data projects: risk register, dependency management, cross-track blockers	Instructor-led lab
—	1500–1515	Break	—
13	1515–1700	Dependency mapping exercise: identify cross-track dependencies for practical exercise scenario; instructor review	Instructor-led lab

Day 4 — Delivery, Change Management, and Practical Exercise

Block	Time	Content	Method
14	0800–0900	Delivery planning: scope/timeline/quality tradeoffs, release planning, Definition of Done, production readiness review	Lecture
15	0900–0945	Change management: user adoption, resistance management, rollout sequencing, platform governance	Lecture
—	0945–1000	Practical exercise scenario brief; evaluator distributes materials	Brief
—	1000–1015	Final prep; environment check	Buffer
16	1015–1200	Practical exercise — Phase 1 (Tasks 1–4): requirements document, user stories, sprint board spec, dashboard spec	Evaluation
—	1200–1300	Lunch	—
17	1300–1700	Practical exercise — Phase 2 (Tasks 5–7): dependency/risk register, production readiness brief, change management summary	Evaluation

5-6. SL 4K — Knowledge Manager (4 Days)

Day 1 — Knowledge Architecture, AAR Systems, and Lessons Learned

Block	Time	Content	Method
1	0800–0900	KM role on MSS; knowledge architecture methodology; why KM systems fail and what makes them survive personnel turbulence	Brief
2	0900–1100	Ontology: Knowledge Object Types — Document, Lesson, AAR, SOP, ExpertiseProfile; Link Types; design on paper before building	Instructor-led lab
—	1100–1115	Break	—
3	1115–1200	Workshop: AAR submission form — required-field validation, submission confirmation, routing to AAR Object Type	Instructor-led lab
—	1200–1300	Lunch	—
4	1300–1500	Lessons learned pipeline: intake, deduplication, tagging taxonomy design, distribution routing by unit/classification/echelon	Instructor-led lab
—	1500–1515	Break	—
5	1515–1700	AIP Logic: document summarization; automatic theme extraction; human review queue — all AIP outputs begin as Draft	Guided lab

Day 2 — Search, Retrieval Applications, and AIP Prompt Engineering

Block	Time	Content	Method
—	0800–0830	Day 1 review; AIP review gate requirements — the non-negotiable standard	Review
6	0830–1030	Workshop: knowledge browser — search by keyword, filter by tag/unit/date, drill-down to lesson detail	Instructor-led lab
—	1030–1045	Break	—
7	1045–1200	SOP/doctrine version control: lifecycle management, version tagging, SOP review notification workflow	Instructor-led lab
—	1200–1300	Lunch	—
8	1315–1530	AIP prompt iteration lab (extended): test against 5 documents, score extraction quality, revise, retest minimum 3 cycles	Instructor-led lab

Block	Time	Content	Method
—	1530–1545	Break	—
9	1545–1700	Prompt comparison debrief: before/after sharing, structural changes, common prompt failure patterns	Workshop

Day 3 — Expertise Mapping, PCS Continuity, and Package Draft

Block	Time	Content	Method
—	0800–0830	Day 2 review; privacy act and classification of knowledge objects	Review
10	0830–1030	Personnel expertise mapping: ExpertiseProfile Object Type, skills taxonomy, SME directory; Privacy Act authorities	Instructor-led lab
—	1030–1045	Break	—
11	1045–1145	PCS knowledge transfer methodology: key person dependency analysis, transfer package design, Foundry project handoff	Instructor-led lab
—	1145–1200	PCS package requirements brief: Chapter 9 completeness criteria	Brief
—	1200–1300	Lunch	—
12	1300–1700	PCS package draft lab (full afternoon): each trainee produces complete draft; submitted by 1700 for instructor review	Instructor-led lab

Day 4 — Governance, Instructor Review, and Practical Exercise

Block	Time	Content	Method
—	0800–0830	Day 3 review; access management, C2DAO governance for knowledge objects	Review
13	0830–1045	PCS package instructor review: written feedback from overnight review; individual conferral; revision against Chapter 9 criteria	Workshop
—	1045–1100	Break	—
14	1100–1145	Practical exercise scenario brief; review Go criteria for AIP gate and PCS package completeness	Brief
—	1145–1200	Buffer: questions / final environment check	Buffer

Block	Time	Content	Method
—	1200–1300	Lunch	—
15	1300–1700	Practical exercise (evaluated)	Evaluation

5-7. SL 4G — ORSA (5 Days)

Day 1 — Environment Setup and Statistical Modeling

Block	Time	Content	Method
1	0800–0900	ORSA role; analytical product standards; Code Workspace orientation	Lecture/discussion
2	0900–1100	Code Workspace setup: Python/R environment, packages, Foundry dataset connectivity	Instructor-led lab
3	1100–1200	Regression: linear and logistic for readiness forecasting	Instructor-led lab
—	1200–1300	Lunch	—
4	1300–1500	Classification models: feature selection, training, validation statistics	Instructor-led lab
5	1500–1700	Model validation standards; documenting assumptions; writing outputs to Foundry	Guided practice

Day 2 — Time Series and Monte Carlo

Block	Time	Content	Method
6	0800–1000	Time series: stationarity, ACF/PACF, ARIMA for readiness trend forecasting	Instructor-led lab
7	1000–1200	ARIMA/SARIMA forecast build: readiness trends and logistics demand signals	Guided practice
—	1200–1300	Lunch	—
8	1300–1500	Monte Carlo simulation: COA comparison under uncertainty; distribution selection	Instructor-led lab

Block	Time	Content	Method
9	1500–1700	Sensitivity analysis; logistics planning risk (Class IX stockage level modeling)	Guided practice

Day 3 — Optimization and Wargame Analysis

Block	Time	Content	Method
10	0800–1100	Linear programming: resource allocation formulation, constraint setup, scipy/lpSolve	Instructor-led lab
11	1100–1200	Scheduling optimization: maintenance window scheduling against operational constraints	Guided practice
—	1200–1300	Lunch	—
12	1300–1500	Wargame/exercise data architecture: collection templates, aggregation pipelines	Instructor-led lab
13	1500–1700	Post-exercise analysis: AAR aggregation pipeline, outcome measurement	Guided practice

Day 4 — Decision Support and Commander Brief

Block	Time	Content	Method
14	0800–1000	Quiver/Contour for ORSA: readiness forecast dashboard, COA comparison visualization	Instructor-led lab
15	1000–1200	Communicating uncertainty: confidence intervals, prediction intervals, briefing posture	Lecture/scenario
—	1200–1300	Lunch	—
16	1300–1700	Practical exercise: regression + forecast + commander brief	Evaluation

Day 5 — Communicating Results and Practical Exercise

Block	Time	Content	Method
15	0800–0900	Communicating uncertainty: confidence intervals, prediction intervals, briefing posture	Lecture
16	0900–1000	Common ORSA brief failures: point estimates without bounds, methods-paper language	Discussion

Block	Time	Content	Method
—	1000–1015	Break	—
17	1015–1200	Practical exercise scenario brief and ORSA product standards review	BRF
—	1200–1300	Lunch	—
18	1300–1700	Practical exercise: regression + time series + commander brief	Evaluation

5-8. SL 4H — AI Engineer (5 Days)

Day 1 — Safety, Architecture, and AIP Logic Fundamentals *(Block 1 mandatory — no exceptions)*

Block	Time	Content	Method
1	0800–1000	AI safety: human-in-the-loop requirements, OPSEC, prohibited use cases (Appendix B)	Lecture — mandatory
2	1000–1200	AIP platform architecture: AIP Logic, Agent Studio, Code Workspaces, LLM endpoints	Lecture/discussion
—	1200–1300	Lunch	—
3	1300–1500	AIP Logic: first workflow, prompt templates, input/output configuration	Instructor-led lab
4	1500–1700	AIP Logic: conditional chains, error handling, structured JSON output	Guided practice

Day 2 — Advanced AIP Logic and Python Transforms

Block	Time	Content	Method
5	0800–1000	AIP Logic: multi-step chains, looping, Action integration	Instructor-led lab
6	1000–1200	Python transforms for AIP: extracting and formatting Ontology data for LLM context	Instructor-led lab
—	1200–1300	Lunch	—
7	1300–1500	LLM integration patterns: context construction, grounding, RAG architecture	Lecture/lab

Block	Time	Content	Method
8	1500–1700	Ontology integration: connecting AIP Logic outputs to Object properties via Actions	Guided practice

Day 3 — Agent Studio and Evaluation

Block	Time	Content	Method
9	0800–1100	Agent Studio: architecture, tool configuration, memory, orchestration	Instructor-led lab
10	1100–1200	Testing AI outputs: red-teaming, adversarial prompts, evaluation frameworks	Lecture/exercise
—	1200–1300	Lunch	—
11	1300–1500	AI Output Validation Framework (Appendix C): applying to a workflow	Guided practice
12	1500–1700	AIP Authorization Checklist (Appendix A): full authorization walkthrough	Workshop

Day 4 — Production Deployment and Practical Exercise

Block	Time	Content	Method
13	0800–1000	Production deployment: pipeline scheduling, monitoring, rollback procedures	Lecture/demo
14	1000–1200	Practical exercise preparation: review scenario, plan design	Individual prep
—	1200–1300	Lunch	—
15	1300–1700	Practical exercise: author, test, and document AIP Logic workflow end-to-end	Evaluation

Day 5 — Production Deployment and Practical Exercise

Block	Time	Content	Method
15	0800–0900	Production deployment: pipeline scheduling, monitoring, failure alerting, rollback	Lecture
16	0900–1000	Practical exercise scenario brief and workflow design time	BRF

Block	Time	Content	Method
—	1000–1015	Break	—
17	1015–1200	Authorization checklist guidance; evaluation criteria review	BRF
—	1200–1300	Lunch	—
18	1300–1700	Practical exercise: author → test → authorize → deploy AIP workflow	Evaluation

5-9. SL 4M — Machine Learning Engineer (5 Days)

Day 1 — Workspace and Feature Engineering

Block	Time	Content	Method
1	0800–0900	MLE role on MSS; model governance overview; Code Workspace orientation	Lecture
2	0900–1100	Code Workspace setup: GPU allocation, package management, Foundry dataset connectivity	Instructor-led lab
3	1100–1200	Feature engineering: null handling, encoding, scaling, leakage prevention	Lecture/lab
—	1200–1300	Lunch	—
4	1300–1700	Feature engineering pipeline build: Foundry dataset → feature matrix	Instructor-led lab

Day 2 — Model Training and Evaluation

Block	Time	Content	Method
5	0800–1100	Model training: scikit-learn/PyTorch patterns, cross-validation, hyperparameter tuning	Instructor-led lab
6	1100–1200	Evaluation: accuracy, precision/recall, calibration, ROC-AUC; acceptance thresholds	Lecture/lab
—	1200–1300	Lunch	—

Block	Time	Content	Method
7	1300–1700	Model evaluation exercise: evaluate 2 competing models; select and document	Guided practice

Day 3 — Deployment and MLOps

Block	Time	Content	Method
8	0800–1000	Model deployment: serving endpoints, inference API, integration with Ontology Actions	Instructor-led lab
9	1000–1200	MLOps on Foundry: experiment tracking, model registry, versioning	Instructor-led lab
—	1200–1300	Lunch	—
10	1300–1500	Monitoring: drift detection, data quality alerts, retraining triggers	Instructor-led lab
11	1500–1700	Operational use cases: readiness prediction, logistics demand, anomaly detection	Lecture/case study

Day 4 — Governance and Practical Exercise

Block	Time	Content	Method
12	0800–1000	Model governance: documentation standards, model cards, responsible AI requirements	Lecture/workshop
13	1000–1200	Practical exercise preparation: review scenario, plan pipeline	Individual prep
—	1200–1300	Lunch	—
14	1300–1700	Practical exercise: feature pipeline → train → evaluate → deploy → governance doc	Evaluation

Day 5 — Governance and Practical Exercise

Block	Time	Content	Method
15	0800–0900	Model governance: model card completion, assumptions, limitations, responsible AI declaration	Lecture
16	0900–1000	Deployment approval and C2DAO governance for deployed models	BRF

Block	Time	Content	Method
—	1000–1015	Break	—
17	1015–1200	Practical exercise scenario brief; planning time	BRF
—	1200–1300	Lunch	—
18	1300–1700	Practical exercise: feature pipeline → train → evaluate → deploy → monitor → governance	Evaluation

5-10. SL 4L — Software Engineer (5 Days)

Day 1 — OSDK Fundamentals

Block	Time	Content	Method
1	0800–0900	SWE role on MSS; USAREUR-AF 5-layer data stack; code review standards	Lecture
2	0900–1100	OSDK setup: authentication, client initialization, first object query	Instructor-led lab
3	1100–1200	OSDK: filtering, sorting, pagination, and ResourceIterator patterns	Instructor-led lab
—	1200–1300	Lunch	—
4	1300–1500	OSDK: Link traversal; querying related objects	Instructor-led lab
5	1500–1700	OSDK: Action execution; error handling and retry patterns	Guided practice

Day 2 — OSDK Advanced and Platform SDK

Block	Time	Content	Method
6	0800–1000	OSDK: Object subscriptions (real-time change notifications); bulk operations	Instructor-led lab
7	1000–1200	Platform SDK: dataset read/write transactions, file resources, branch management	Instructor-led lab
—	1200–1300	Lunch	—

Block	Time	Content	Method
8	1300–1700	Platform SDK exercise: build a dataset pipeline integration	Guided practice

Day 3 — TypeScript Functions, Action Validators, and Slate

Block	Time	Content	Method
9	0800–1000	TypeScript Functions on Objects: structure, computed properties, bulk query patterns	Instructor-led lab
10	1000–1200	FOO testing patterns; performance considerations	Guided practice
—	1200–1300	Lunch	—
11	1300–1500	TypeScript Action validators: multi-condition logic, error messages, testing	Instructor-led lab
12	1500–1700	Slate applications: structure, Foundry API integration, widget binding	Instructor-led lab

Day 4 — CI/CD, Security, and Practical Exercise

Block	Time	Content	Method
13	0800–1000	CI/CD: code repository discipline, PR workflow, C2DAO deployment checklist	Lecture/demo
14	1000–1200	Security and compliance: OSDK token handling, input sanitization, OPSEC for app code	Lecture/lab
—	1200–1300	Lunch	—
15	1300–1700	Practical exercise: OSDK query → Action validator → Slate UI end-to-end	Evaluation

Day 5 — CI/CD, Security, and Practical Exercise

Block	Time	Content	Method
15	0800–0900	CI/CD: repository discipline, PR workflow, automated testing, C2DAO deployment checklist	Lecture
16	0900–1000	Security and compliance: token handling, input sanitization, OPSEC for app code	Lecture

Block	Time	Content	Method
—	1000–1015	Break	—
17	1015–1200	Practical exercise scenario brief; planning time	BRF
—	1200–1300	Lunch	—
18	1300–1700	Practical exercise: OSDK → validator → Slate UI → deployment checklist	Evaluation

5-11. SL 4N — UI/UX Designer (5 Days)

Day 1 — User Research and Information Architecture

Block	Time	Content	Method
1	0800–0900	UI/UX role on MSS; design principles for operational data products	Lecture
2	0900–1100	User research methods: stakeholder interviews, task analysis, persona development	Instructor-led lab
3	1100–1200	Information architecture: organizing complex data for command audiences	Instructor-led lab
—	1200–1300	Lunch	—
4	1300–1500	Wireframing fundamentals: low-fidelity prototyping for Workshop applications	Instructor-led lab
5	1500–1700	Design critique: peer review of wireframes against operational requirements	Workshop

Day 2 — Accessibility and Visual Design

Block	Time	Content	Method
6	0800–1000	Accessibility standards: WCAG compliance for government applications	Lecture
7	1000–1200	Responsive design: multi-device layouts for field and garrison use	Instructor-led lab
—	1200–1300	Lunch	—

Block	Time	Content	Method
8	1300–1500	Color theory and typography for data visualization; RAG standards	Instructor-led lab
9	1500–1700	Dashboard layout patterns: summary-detail, drill-down, comparison; anti-patterns	Instructor-led lab

Day 3 — Prototyping and User Testing

Block	Time	Content	Method
10	0800–1000	High-fidelity prototyping: Workshop design with real data binding	Instructor-led lab
11	1000–1200	Interactive prototyping: conditional visibility, state transitions, user flow design	Instructor-led lab
—	1200–1300	Lunch	—
12	1300–1500	User testing methodology: test plan design, task scenarios, observation protocols	Lecture
13	1500–1700	Usability testing lab: conduct test with peer; document findings; prioritize fixes	Guided practice

Day 4 — Design Systems and Developer Collaboration

Block	Time	Content	Method
14	0800–1000	Design system creation: component library, style guide, reusable widget templates	Instructor-led lab
15	1000–1200	Design system documentation for developer handoff; naming conventions	Instructor-led lab
—	1200–1300	Lunch	—
16	1300–1500	Design-development collaboration: specification documents, design tokens, handoff workflow	Instructor-led lab
17	1500–1700	Design review process: C2DAO standards review; iteration workflow	Discussion

Day 5 — Portfolio Standards and Practical Exercise

Block	Time	Content	Method
18	0800–0900	Design portfolio standards: complete operational UI/UX deliverable requirements	Lecture
—	0900–0915	Break	—
19	0915–1100	Practical exercise scenario brief and design planning time	BRF
—	1100–1200	Buffer: open lab	Buffer
—	1200–1300	Lunch	—
20	1300–1700	Practical exercise: user research → wireframe → prototype → test → design system artifact	Evaluation

5-12. SL 4O — Platform Engineer (5 Days)

Day 1 — Kubernetes and Container Fundamentals

Block	Time	Content	Method
1	0800–0900	Platform engineering role on MSS; infrastructure architecture overview	Lecture
2	0900–1100	Kubernetes fundamentals: cluster architecture, namespaces, resource quotas, pod lifecycle	Instructor-led lab
3	1100–1200	Container management: image registry, build pipeline, vulnerability scanning	Instructor-led lab
—	1200–1300	Lunch	—
4	1300–1500	Foundry infrastructure: platform components, service dependencies, health checks	Instructor-led lab
5	1500–1700	Infrastructure-as-code: configuration management for repeatable deployments	Instructor-led lab

Day 2 — Deployment and Networking

Block	Time	Content	Method
6	0800–1000	Deployment strategies: rolling updates, blue-green, canary; rollback procedures	Instructor-led lab

Block	Time	Content	Method
7	1000–1200	Deployment lab: execute rolling update; verify zero-downtime; trigger rollback	Instructor-led lab
—	1200–1300	Lunch	—
8	1300–1500	Service mesh and networking: ingress, load balancing, service discovery, TLS	Instructor-led lab
9	1500–1700	Storage management: persistent volumes, storage classes, backup and recovery	Instructor-led lab

Day 3 — Monitoring and Observability

Block	Time	Content	Method
10	0800–1000	Monitoring architecture: metrics collection, log aggregation, distributed tracing	Instructor-led lab
11	1000–1200	Alerting configuration: threshold alerts, anomaly detection, escalation routes	Instructor-led lab
—	1200–1300	Lunch	—
12	1300–1500	Observability dashboard build: health status, resource utilization, SLI/SLO tracking	Instructor-led lab
13	1500–1700	Incident response: runbook development, escalation procedures, post-incident review	Instructor-led lab

Day 4 — Security and CI/CD

Block	Time	Content	Method
14	0800–1000	Security hardening: network policies, RBAC, secrets management, audit logging	Instructor-led lab
15	1000–1200	Access control: service accounts, pod security standards, least-privilege	Instructor-led lab
—	1200–1300	Lunch	—
16	1300–1500	CI/CD pipeline design: GitOps workflow, testing gates, promotion environments	Instructor-led lab
17	1500–1700	CI/CD lab: build pipeline with test, scan, and deploy stages	Instructor-led lab

Day 5 — Standards and Practical Exercise

Block	Time	Content	Method
18	0800–0900	C2DAO infrastructure standards: change management, deployment approval, documentation	Lecture
—	0900–0915	Break	—
19	0915–1100	Practical exercise scenario brief and infrastructure planning time	BRF
—	1100–1200	Buffer: open lab	Buffer
—	1200–1300	Lunch	—
20	1300–1700	Practical exercise: deploy → monitor → alert → secure → CI/CD → deployment checklist	Evaluation

6. TRAINING SCHEDULE TEMPLATES

6-1. Individual Integration Training Schedule (New Arrival)

Week 1 — Foundation

Day	Training	Location	Access Requirement
Mon	SL 1 (full day)	MSS training environment	Basic MSS account (provisioned before arrival)
Tue	Supervised practice: unit operational MSS environment	G6/data team desk	Read-only access to unit projects
Wed	SL 2 Day 1 (if builder role) OR continuation practice	Training environment	Builder access provisioned
Thu	SL 2 Day 2 + evaluation	Training environment	Builder access
Fri	Review evaluation feedback; remediation if needed	—	—

Weeks 2–3 — Specialized Track (if applicable)

Continue with SL 3 or SL 4 track based on role. No more than one TM level per week for initial qualification.

6-2. Unit-Wide Training Event (S6 Shop or Data Team)

Week 1

Day	Training	Audience
Mon	Commander brief: Data Literacy for Senior Leaders summary (2 hrs); unit data strategy overview	All O-5+, S6 leadership
Tue	SL 1 (all staff)	All personnel
Wed	SL 1 Evaluation (morning); SL 2 Day 1 (designated builders, afternoon)	Split: all / builders
Thu	SL 2 Day 2	Designated builders
Fri	SL 2 Evaluation; SL 3 orientation brief	Builders

Week 2

Day	Training	Audience
Mon	SL 3 Day 1	Data-adjacent specialists
Tue	SL 3 Day 2	Data-adjacent specialists
Wed	SL 3 Day 3 + Evaluation	Data-adjacent specialists
Thu	Confirm all SL 4 enrollments; brief functional staff on WFF track options; brief specialists on G–O track options	Training NCO/Officer
Fri	SL 4 scheduling coordination complete; access provisioning confirmed for all Week 3+ participants	Training NCO/Officer

Week 3 (SL 4 WFF Tracks)

Day	Training	Audience
Mon–Wed	SL 4A (Intel WFF) — 3 days	G2/S2 Intel staff
Mon–Wed	SL 4D (Sustainment WFF) — 3 days (parallel session if capacity allows)	G4/S4 staff

Day	Training	Audience
Thu–Fri or following week	Remaining WFF tracks (B, C, E, F) as scheduled	Applicable WFF functional staff

NOTE

All SL 4 tracks (A–F WFF and G–O specialist) require SL 3 completion. Ensure all SL 4 participants have SL 3 on file before enrolling. WFF and specialist (G–O) training can run concurrently in separate sessions once SL 3 is complete.

Week 4 (SL 4 Specialist Tracks — 5-day courses)

Day	Training	Audience
Mon–Fri	SL 4G (ORSA) — 5 days	ORSA personnel
Mon–Fri	SL 4H (AI Engineer) — 5 days (parallel session if capacity allows)	AI Engineer personnel
Following week	SL 4M (ML Engineer), SL 4L (Software Engineer) — 5 days each, as scheduled	ML Eng, SWE personnel

Week 5 (SL 4 Specialist Tracks — 4-day courses)

Day	Training	Audience
Mon–Thu	SL 4J (Program Manager) — 4 days	G8/S8, Program Managers
Mon–Thu	SL 4K (Knowledge Manager) — 4 days (parallel session if capacity allows)	Knowledge Managers, S6

NOTE

SL 4J and SL 4K are 4-day courses (32 hours each). Access provisioning for SL 4H, SL 4M, and SL 4L requires 10–35 days (GPU workspaces, developer tokens) — initiate immediately after SL 3 enrollment is confirmed.

7. RESOURCE REQUIREMENTS

7-1. Access Requirements by Training Level

CAUTION

Access provisioning is on the critical path for all MSS training. Request access a minimum of 5 duty days before the training start date. Training events that begin without provisioned access will fail. The training officer owns the access request timeline.

Training Level	Access Required	Who Provisions	Lead Time
SL 1	Standard MSS user account (Viewer in training project)	Unit MSS Administrator	3–5 duty days
SL 2	Builder access in training project	Unit MSS Administrator	3–5 duty days
SL 3	Editor access; AIP Logic configuration access	Unit MSS Admin + C2DAO	5–7 duty days
SL 4A–F (WFF)	Builder access	Unit MSS Administrator	3–5 duty days
SL 4G	Code Workspace (Python/R)	C2DAO	7–10 duty days
SL 4H	AIP Logic authoring + Agent Studio access	C2DAO	7–10 duty days
SL 4M	Code Workspace with GPU; model registry access	C2DAO	10+ duty days
SL 4J	Builder access + GFEBS training data export	Unit MSS Admin + G8	5–7 duty days
SL 4K	Builder access + AIP Logic configuration	Unit MSS Admin + C2DAO	5–7 duty days
SL 4L	OSDK developer access; TypeScript environment	C2DAO	10+ duty days
SL 4N	Workshop design access (Editor + AIP Logic configuration)	C2DAO	7–10 duty days

Training Level	Access Required	Who Provisions	Lead Time
SL 4O	Kubernetes cluster access + CI/CD pipeline access	C2DAO	10+ duty days

7-2. Training Environment Requirements

All MSS training is conducted in the **USAREUR-AF MSS Training Environment** — not the production environment. The training environment contains: - Synthetic datasets representative of unit operational data (non-sensitive) - Pre-configured training projects at each TM level - Seeded Object Types and pipelines for exercise use - Test user accounts for access control exercises

If the training environment is unavailable, coordinate with C2DAO for an approved production substitute with restricted permissions.

7-3. Instructor Hardware Requirements

Requirement	Standard
Instructor workstation	Laptop with CAC reader, MSS access, projector output
Projector/display	Minimum 80" display or projector visible to all trainees
Network	Unclassified network with MSS access (confirm connectivity before training day)
Trainee workstations	One per trainee; CAC-capable; network access to MSS
Backup	Instructor maintains a pre-recorded demo video for each major task in case of MSS outage

7-4. SL 1 Throughput Risk — G-3 Action Required

RISK FLAG — FOR G-3 ACTION

SL 1 (Maven User, 1 day) is mandatory for **all** USAREUR-AF personnel within 30 days of assignment. At current instructor capacity and training iteration frequency, the program cannot process the full formation at the rate required to meet this standard.

The gap: - SL 1 required population: all USAREUR-AF military and Civilian personnel (theater-wide) - Current certified SL 1 instructor pool: insufficient to run concurrent iterations at scale - Current training frequency: insufficient to process new arrivals within the 30-day window during high-accession periods

Impact: Portions of the formation are operating MSS without SL 1 qualification, creating data governance risk (unqualified users making classification and access decisions) and operational risk (untrained users misreading or mishandling MSS data products).

Options for G-3 consideration: 1. **Instructor billet increase** — authorize additional C2DAO training billets or detail SL 2/30-certified personnel from units as SL 1 instructors on a rotational basis (qualification standard: Section 8-1) 2. **eLearning conversion of SL 1** — convert SL 1 to a self-paced computer-based training (CBT) that can be completed asynchronously; reserve the practical evaluation (Tasks 4-1, 4-2, and Ch 6 security markings) as the only instructor-administered element; this is the highest-leverage option for throughput 3. **Unit-embedded delivery** — certify one SL 2+ instructor per major subordinate command (MSC) to deliver SL 1 within their formation; C2DAO provides materials and certifies instructors; practical evaluation remains C2DAO-administered

Recommended G-3 action: Direct C2DAO to present options 2 and 3 with cost/risk analysis within 30 days. Until resolved, direct unit commanders to report SL 1 completion rates in the monthly Unit MSS Training Status Report (Section 10-2) and flag units below 70% completion for command attention.

8. INSTRUCTOR REQUIREMENTS AND CERTIFICATION

8-1. Instructor Qualification Standards

TM Level	Instructor Requirement
SL 1	SL 2 certified or higher; completed SL 1 evaluation as a trainee
SL 2	SL 3 certified or higher
SL 3	SL 4 certified in any track; or C2DAO-certified data engineer
SL 4A (Intel WFF)	SL 4A certified; G2/S2 background or Intel functional familiarity; SL 3 proficiency
SL 4B (Fires WFF)	SL 4B certified; Fires/FSCOORD background; SL 3 proficiency
SL 4C (M2 WFF)	SL 4C certified; G3/S3 movement and maneuver background; SL 3 proficiency
SL 4D (SUST WFF)	SL 4D certified; G4/S4 sustainment background; SL 3 proficiency
SL 4E (PROT WFF)	SL 4E certified; Protection background; SL 3 proficiency

TM Level	Instructor Requirement
SL 4F (MC WFF)	SL 4F certified; Mission Command/G6 background; SL 3 proficiency
SL 4G (ORSA)	Active-duty ORSA (FA49) with Code Workspace proficiency OR C2DAO-approved contractor
SL 4H (AI Eng)	C2DAO AI engineer or SL 4H certified individual
SL 4M (MLE)	C2DAO ML engineer or SL 4M certified individual
SL 4J (PM)	SL 4J certified; PM/resource management background
SL 4K (KM)	SL 4K certified; KM background (37F or equivalent)
SL 4L (SWE)	C2DAO software engineer or SL 4L certified; TypeScript/Python proficiency
SL 4N (UI/UX)	SL 4N certified; UI/UX design background; Workshop design proficiency; C2DAO UX SME designation
SL 4O (Platform)	SL 4O certified; Platform engineering background; Kubernetes/container proficiency; C2DAO infrastructure SME

8-2. Instructor Certification Process

1. Complete the TM for the level you will instruct, plus one level above (or a SL 4 track for SL 3 instructors).
2. Pass the practical evaluation for the target level.
3. Co-instruct one full training event at the target level, observed by a C2DAO data engineer or existing certified instructor.
4. Receive written certification from C2DAO.
5. Renew certification annually by completing a TM-level practical exercise review.

8-3. Instructor-to-Trainee Ratios

TM Level	Maximum T:I Ratio
SL 1	20:1 (lecture); 10:1 (lab)
SL 2	10:1 (all portions)
SL 3	8:1 (all portions)
SL 4A–F (WFF)	10:1 (all portions — high-volume functional staff courses)
SL 4G/H/M/L/N/O	5:1 (intensive coding/modeling content)

TM Level	Maximum T:I Ratio
SL 4J/K	10:1 (all portions)

9. PRACTICAL EXERCISE DESIGNS

9-1. SL 1 — "Unit SITREP Access"

Scenario: S3 NCO assigned to unit. Unit uses MSS to track SITREP submissions from subordinate elements. Access the SITREP dashboard, check for missing submissions, and submit a correction.

Tasks: 1. Log in to MSS using CAC. 2. Navigate to the `[UNIT]-S3-SITREP-Tracker` application. 3. Filter the table to show only submissions from the last 7 days. 4. Identify which element has not submitted for today. 5. Execute the "Submit SITREP Update" Action to correct a seeded status error. 6. Export the filtered table to CSV. 7. State the classification marking and authorized distribution list for exports.

Go standard: All 7 tasks completed; classification question answered correctly.

9-2. SL 2 — "Build a Unit Readiness Tracker"

Scenario: S4 officer needs a simple equipment readiness tracker. Provided: Excel spreadsheet with equipment identifiers, assigned units, and current C-ratings.

Tasks: 1. Create a Foundry project named per unit convention. 2. Upload the provided Excel file; build a Pipeline Builder pipeline that cleans and outputs a validated dataset. 3. Create an `Equipment` Object Type with: `equipment_id`, `unit`, `c_rating`, `last_updated` properties. 4. Configure the pipeline to write Equipment objects. 5. Create an Action that allows an authorized user to update `c_rating`. 6. Build a Workshop application with: table, filter by `unit` and `c_rating`, and the Update Action button. 7. Grant a test user Viewer access; verify they cannot edit. 8. Create a branch, change the application title, and submit for promotion.

Go standard: All 8 tasks completed without instructor assistance; pipeline runs without error.

9-3. SL 4A–F — WFF Functional Track Practical Exercises

Instructor selects from the applicable WFF track scenario bank. Scenarios below are representative for each track. All require SL 1, SL 2, and SL 3 completion.

SL 4A (Intel WFF) — "ISR Status Tracker"

Scenario: G2 analyst requires a dashboard showing ISR asset status across the AOR. Provided: synthetic ISR asset tracking spreadsheet.

Tasks: 1. Build a pipeline ingesting the ISR asset dataset; compute availability status per asset. 2. Create ISR Asset Object Type with required properties. 3. Build a Workshop application: table filtered by asset type and status. 4. Configure an "Update Asset Status" Action with appropriate access restriction. 5. Build a multi-page dashboard: Page 1 = AOR summary, Page 2 = unit-level detail. 6. Branch, submit product for promotion with complete description.

Go standard: All 6 tasks completed; pipeline runs without error; governance requirements met.

SL 4B (Fires WFF) — "Target Tracking Dashboard"

Scenario: FSCOORD requires a target status and engagement tracking application. Provided: synthetic target tracking dataset.

Tasks (same structure as 40A — Fires-specific data and Object Types). Go standard: all 6 tasks.

SL 4C (Movement & Maneuver WFF) — "Unit Disposition Tracker"

Scenario: S3 requires a unit disposition and FRAGORD compliance tracker. Provided: synthetic unit status dataset.

Tasks (same structure — M2-specific Object Types). Go standard: all 6 tasks.

SL 4D (Sustainment WFF) — "Class IX Readiness Dashboard"

Scenario: S4 requires a Class IX on-hand vs. requirement dashboard with shortfall flagging.

Tasks (same structure — SUST-specific data model). Go standard: all 6 tasks.

SL 4E (Protection WFF) — "Force Protection Status Tracker"

Scenario: Protection officer requires a threat indicator and force protection measure tracking dashboard.

Tasks (same structure — PROT-specific Object Types). Go standard: all 6 tasks.

SL 4F (Mission Command WFF) — "Battle Rhythm and Comms Status Dashboard"

Scenario: G6 requires a battle rhythm event tracker and comms system status dashboard for the commander.

Tasks (same structure — MC/G6-specific data model). Go standard: all 6 tasks.

9-4. SL 3 — "Multi-Source Operations Dashboard"

Scenario: S3 requires a dashboard combining personnel readiness (one dataset) and equipment readiness (second dataset) at the unit level, with battalion filter and drill-down to individual unit status.

Tasks: 1. Build a Pipeline Builder pipeline joining the two provided datasets on `unit_id`; compute overall readiness score per unit. 2. Design (document) the Ontology schema for a `Unit` Object Type and `ReadinessReport` Object Type with appropriate links. 3. Build a multi-page Workshop application: Page 1 = portfolio view (all units), Page 2 = unit detail (linked from Page 1), Page 3 = historical trend. 4. Build a Contour workbook showing readiness by battalion with a calculated deviation-from-standard column. 5. Submit the pipeline and application through the C2DAO promotion workflow.

Evaluator rubric: See Appendix C.

---

10. TRAINING STATUS TRACKING

10-1. Unit Training Status Matrix

The Unit Training Status Matrix (Appendix A) is the primary tracking tool. Maintained by the unit MSS Administrator or designated training NCO; reviewed monthly.

Required fields per Soldier/Civilian:

Field	Description
Name	Last, First MI
Rank/Grade	E/O/W/GS grade
MOS/Position	Assigned role
SL 1 Date	Date of Go evaluation (or N/A if not yet completed)
SL 2 Date	Date of Go evaluation
SL 3 Date	Date of Go evaluation
SL 4 Track	Applicable track(s)
SL 4 Date	Date of Go evaluation
Refresher Due	Date of next sustainment evaluation
Notes	Remediation status, access issues, special certifications

10-2. Unit Training Status Report (UTSR)

Monthly one-pager submitted to C2DAO summarizing unit MSS training status.

```
UNIT MSS TRAINING STATUS REPORT
Unit: [Unit Designation]          Date: [YYYYMMDD]
Reporting Officer: [Name, Rank]   Contact: [Email]

PERSONNEL SUMMARY:
  Total personnel requiring SL 1: [#]     SL 1 Complete: [#] ([%])
  Total personnel requiring SL 2: [#]     SL 2 Complete: [#] ([%])
  Total personnel requiring SL 3: [#]     SL 3 Complete: [#] ([%])
  Total WFF track assignments (SL 4A–F):  [#]  WFF Complete: [#] ([%])
    Breakdown: 40A[#] 40B[#] 40C[#] 40D[#] 40E[#] 40F[#]
  Total specialist track assignments (SL 4G–O): [#]  Specialist Complete: [#] ([%])
    Breakdown: 40G[#] 40H[#] 40J[#] 40K[#] 40L[#] 40M[#] 40N[#] 40O[#]

ISSUES / SHORTFALLS:
  Access provisioning delays: [Y/N — if yes, describe]
  Personnel awaiting remediation: [#]
  Instructor availability: [adequate / insufficient]
  WFF track scheduling conflicts (functional staff unavailability): [describe if any]

NEXT TRAINING EVENT: [Date, Level, # of trainees]
```

11. SUSTAINMENT TRAINING REQUIREMENTS

11-1. Sustainment Standards

Condition	Sustainment Requirement
Active daily MSS user	No formal sustainment; supervisory spot-check quarterly
Occasional user (weekly to monthly)	Practical task evaluation every 12 months
Infrequent user (<monthly)	Practical task evaluation every 6 months
MSS access not used in 6+ months	Re-evaluate at current TM level before resuming independent work
PCS arrival with prior MSS training	Re-evaluate SL 1 (access/navigation may differ by theater); waive with supervisor verification

11-2. Sustainment Event Design

A sustainment event does not require a full-day class — design as a practical task check:

Level	Duration	Method
SL 1	1 hour	Complete one task scenario from the SL 1 practical exercise bank (Appendix C). Evaluated by immediate supervisor or unit MSS Administrator.
SL 2/30	2–3 hours	Complete one of: build a new pipeline in a test project, modify an existing Workshop application, or complete a branch/promote workflow. Evaluated by SL 3+ certified individual.
SL 4	4 hours	Complete a representative task from the relevant SL 4 track. SL 4J/K sustainment may include producing a real unit product (PM dashboard update, AAR pipeline run) observed by a supervisor.

11-3. Access Deactivation Policy

Personnel who have not accessed MSS in 90 days will have their access reviewed by the unit MSS Administrator. Access will be deactivated at 180 days of inactivity per USAREUR-AF access management policy. Reactivation requires: 1. Supervisor endorsement of continued need 2. Completion of SL 1 practical re-evaluation 3. MSS Administrator re-provisioning

APPENDIX A — UNIT TRAINING STATUS MATRIX (TEMPLATE)

```
UNIT MSS TRAINING STATUS MATRIX
Unit: _____________________     As of: ___________
Maintained by: _________________     Reviewed: ________

NAME          | RANK | MOS  | SL 1  | SL 2  | SL 3  | SL 4  | REFRESH
              |      |      | DATE  | DATE  | DATE  | TRK/DT | DUE
--------------+------+------+-------+-------+-------+--------+--------
              |      |      |       |       |       |        |
              |      |      |       |       |       |        |
              |      |      |       |       |       |        |

STATUS CODES: C = Complete (Go)  I = In Progress  N = Not Started
              R = Remediation    W = Waived (supervisor verification)
```

APPENDIX B — INDIVIDUAL TRAINING RECORD (TEMPLATE)

```
MSS INDIVIDUAL TRAINING RECORD

Name: _____  Rank/Grade: _______  MOS/Position: __________
Unit: _____  Email: _____________________________

SL 1 EVALUATION
  Date: __________  Evaluator: _________________  Result: GO / NO-GO
  Tasks Failed (if any): _____________________________________
  Re-evaluation Date (if applicable): _____________  Result: _______

SL 2 EVALUATION
  Date: __________  Evaluator: _________________  Result: GO / NO-GO
  Tasks Failed: ______________________________________________
  Re-evaluation: _____________________________________________

SL 3 EVALUATION
  Date: __________  Evaluator: _________________  Result: GO / NO-GO
  Tasks Failed: ______________________________________________
  Re-evaluation: _____________________________________________

SL 4 TRACK: _______
  Date: __________  Evaluator: _________________  Result: GO / NO-GO
  Tasks Failed: ______________________________________________
  Re-evaluation: _____________________________________________

SUSTAINMENT EVALUATIONS
  Date: ________  Level: _____  Result: GO / NO-GO  Evaluator: _________
  Date: ________  Level: _____  Result: GO / NO-GO  Evaluator: _________
  Date: ________  Level: _____  Result: GO / NO-GO  Evaluator: _________

INSTRUCTOR CERTIFICATIONS
  TM Level: _____  Certified: ________  Certifying Authority: ___________
  TM Level: _____  Certified: ________  Certifying Authority: ___________
```

APPENDIX C — PRACTICAL EXERCISE SCENARIO BANK

SL 1 Scenarios (Instructor selects one per evaluation)

Scenario 10-A: Access the `[Unit]-Readiness-Dashboard` application. Identify which subordinate element has the lowest C-rating. Export the table filtered to that element. State the classification of the dataset.

Scenario 10-B: Navigate to the `[Unit]-SITREP-Tracker`. Submit a SITREP for today using the designated Action. Verify submission appears in the tracker table.

Scenario 10-C: Use Contour to build a bar chart showing count of personnel by rank from the provided personnel dataset. Apply a filter for a specific unit. Save the chart.

Scenario 10-D: You receive an error message when trying to access a dataset. Using SL 1 troubleshooting procedures, identify the likely cause (one of three seeded scenarios: wrong marking, no project access, dataset moved). State the correct resolution action.

SL 2 Scenarios (Instructor selects one per evaluation)

Scenario 20-A: Readiness Tracker — see Section 9-2.

Scenario 20-B: Build a training schedule tracker. Given an Excel file with trainee name, TM level, date, and result, build a Pipeline Builder pipeline, create a TrainingRecord Object Type, configure an Update Status Action, and build a Workshop application showing training completion percentages by unit.

Scenario 20-C: Given a logistics dataset (equipment, location, status), build a Pipeline Builder pipeline that filters to RED status items and a Workshop application displaying them with a mark-resolved Action.

SL 4J Scenarios (PM — Instructor selects one per evaluation)

Scenario 40J-A: G8 data builder for a theater sustainment command program. Given a provided IMS spreadsheet (15 milestones) and a GFEBS obligation extract, build the full PM stack: Object Types, ingestion pipelines, milestone RAG dashboard, and obligation rate Quiver chart with a Q2 reference line. Application must display a data-as-of timestamp.

Scenario 40J-B: Command has 6 programs across 3 battalions. Using provided synthetic program data, build a Contour portfolio health matrix showing schedule and resource status per program, sorted by overall health descending. Export as PDF formatted for a commander IPR.

Scenario 40J-C: An existing PM dashboard has no data-as-of date and uses hardcoded status values in the pipeline. Identify and fix both issues. Document the changes in pipeline description fields and re-run to verify.

SL 4K Scenarios (KM — Instructor selects one per evaluation)

Scenario 40K-A: Brigade KMO needs an AAR capture system. Build the Object Type, Workshop submission form with required-field validation, and a knowledge browser showing submitted AARs filterable by unit and event type.

Scenario 40K-B: Configure an AIP Logic summarization workflow that extracts structured lessons from provided documents, creates Draft Lesson objects, and routes them to a KM Review Queue Workshop application. Produce the AIP Authorization checklist for the workflow.

Scenario 40K-C: Design a PCS knowledge transfer package for an outgoing S3 operations sergeant who has been the unit MSS builder for 18 months. Using TM-40K Ch 9 procedures, produce the complete transfer package: key person dependency analysis, knowledge transfer artifacts list, and Foundry project handoff documentation.

SL 4G Scenarios (ORSA — Evaluator selects one per evaluation)

Scenario 40G-A: Using a provided readiness dataset (12 months, C-ratings by battalion), build: (1) a linear regression to forecast C2-level probability at 6 months, (2) a time series forecast with 90% confidence intervals, (3) a commander brief showing the forecast with explicit uncertainty bounds. No point estimate without its confidence range.

Scenario 40G-B: G3 needs to compare two COAs for a logistics operation. Using provided demand and resource data, run a Monte Carlo simulation for each COA (1,000 trials minimum), produce a probability distribution for on-hand stock at D+30, and brief the comparison with risk characterization.

Scenario 40G-C: Given a maintenance schedule constraint problem (20 vehicles, 5 bay slots, maintenance windows, operational commitments), formulate and solve a linear program that maximizes vehicles available during a defined readiness window. Document all constraints; present the solution in language the S4 can brief.

SL 4H Scenarios (AI Engineer — Evaluator selects one per evaluation)

Scenario 40H-A: Build an AIP Logic workflow that accepts unstructured SITREP text, extracts structured fields (unit, date, activity, status, issues), maps them to a SITREP Object Type via an Action, and routes uncertain extractions to a human review queue. Complete the AIP Authorization Checklist.

Scenario 40H-B: Configure an Agent Studio agent that answers natural language questions about unit readiness by querying the Readiness Object Type. The agent must refuse questions outside its defined scope and must not take any write Actions without an explicit confirmation step.

SL 4M Scenarios (ML Engineer — Evaluator selects one per evaluation)

Scenario 40M-A: Given a provided equipment failure dataset, build: feature engineering pipeline, binary classifier for failure prediction within 30 days, evaluation report against acceptance thresholds, and a deployed inference endpoint. Produce the model governance document.

Scenario 40M-B: An existing model is deployed and showing data drift (evaluator seeds drift into the monitoring dataset). Diagnose the drift, characterize its type, and produce a written recommendation: retrain, adjust thresholds, or investigate data quality issue. Document the decision.

SL 4L Scenarios (SWE — Evaluator selects one per evaluation)

Scenario 40L-A: Build a TypeScript application using OSDK that: (1) queries Unit objects filtered by readiness status, (2) paginates results into a Slate table view, (3) triggers a status update Action on user click with confirmation modal, and (4) handles error states with user-visible error messages. Complete the C2DAO deployment checklist.

Scenario 40L-B: Given a provided Action that writes to a Personnel Object Type, write a TypeScript validator enforcing: required field presence, valid rank enumeration, date range validation (no future dates), and cross-field logic (if status = DEPLOYED, location must be non-null). Test with 8 provided test cases (4 valid, 4 invalid).

SL 4N Scenarios (UI/UX Designer — Evaluator selects one per evaluation)

Scenario 40N-A: A brigade S3 needs a redesigned operations dashboard. Conduct user research (interview the evaluator acting as S3), create wireframes for a 3-page Workshop application, build a high-fidelity prototype, conduct a user test with a peer, and produce a design system artifact documenting the component library.

Scenario 40N-B: An existing readiness dashboard has been flagged for poor usability. Conduct a heuristic evaluation, document findings, redesign the information architecture, build an improved prototype with accessibility compliance, and submit through C2DAO design review.

SL 4O Scenarios (Platform Engineer — Evaluator selects one per evaluation)

Scenario 40O-A: Deploy a provided application to a Kubernetes cluster. Configure namespace, resource quotas, rolling update strategy, monitoring dashboard with alerting, and network policies. Build a CI/CD pipeline that tests, scans, and deploys. Complete the deployment checklist.

Scenario 40O-B: An existing deployment is experiencing intermittent failures (evaluator seeds issues). Diagnose using monitoring tools, implement alerting for the failure condition, apply security hardening, and produce a runbook documenting the incident response procedure.

SL 3 Scenario Design Rubric (Evaluator Reference)

Criterion	Description	Weight
Completeness	Design covers all required Object Types and Link Types	20%
Correctness	Properties correctly typed; links directionally correct	20%
Normalization	No property redundancy; proper use of links vs. embedded data	15%
Naming convention	Follows USAREUR-AF naming convention throughout	15%

Criterion	Description	Weight
Governance	Access model defined; data steward identified	15%
Feasibility	Design is buildable within SL 3 toolset	15%

Go: Score ≥75% with no zero-score on any criterion.

USAREUR-AF Operational Data Team MTP-MSS | Version 1.0 | March 2026