Semester 1 – Final Report

Project Name	Unified Deep Learning Benchmark for Satellite Image Restoration and Generation
Faculty Supervisor Name	Ismayil Shahaliyev
Team Lead Name & ID	Royana Huseynova 11270 Executive Summary and Milestone Development
Team member #1 Name & ID • What sections did this person contribute to?	Pasha Zulfugarli 13731 Executive Summary, Milestone Development, and Role Allocation
Team member #2 Name & ID • What sections did this person contribute to?	Huseyn Sadatkhanov 13770 Milestone Development, Risk Assessment and Mitigation Planning, and Appendices
Team member #3 Name & ID • What sections did this person contribute to?	Nijat Alisoy 17366 Work Breakdown, Evaluation, and Appendices

1. Executive Summary

The project **Unified Deep Learning Benchmark for Satellite Image Restoration and Generation** aims to create a standardized evaluation pipeline for three key satellite image processing tasks: **Cloud Removal (CR), Super-Resolution (SR), and High-Definition (HD) image generation**. Current benchmarks such as AllClear for cloud removal or PROBA-V for super-resolution operate independently and use different preprocessing steps, metrics, and evaluation protocols, which makes comparing models difficult and inconsistent. Our Fall Semester work identified this gap and demonstrated the need for a unified system capable of supporting multiple datasets, such as AllClear, SEN12MS-CR-TS, WorldStrat, PROBA-V SR, and SAT25K, while applying consistent normalization, band alignment, and metric evaluation.

3Activities included studying dataset structures and spectral properties, implementing **PyTorch-based scripts** to load, visualize, and normalize multi-band images, designing the system's **class structure**, organizing the project folder hierarchy, and defining ten use cases describing how researchers will inspect outputs, compare models, validate datasets, and analyze preprocessing sensitivity. these efforts led to a preliminary framework, including scripts for loading and visualizing datasets and a class diagram

showing how datasets, models, and evaluation workflows connect, providing a clear foundation to build on in Semester 2, when we will focus on implementing the full pipeline, integration of baseline models for CR, SR, and HD tasks, consistent evaluation of models on different datasets and enhancing the visualization tools to make it easier to compare and understand the results.

2. Team Structure and Role Allocation

During Semester 1, the focus was on understanding satellite datasets, developing core data loading and visualization utilities, designing the system structure, and organizing the project repository. Individual contributions are outlined below.

Team Lead – Royana Huseynova (11270)
Royana contributed to developing the PyTorch scripts for loading and normalizing satellite images, ensuring consistent data processing. She also designed the class diagram to clearly represent the relationships among datasets, preprocessing steps, models, and evaluation metrics, providing a structured overview of the system.

Team Member – Nijat Alisoy (17366)
Nijat contributed to creating PyTorch scripts for loading and normalizing satellite images and developed visual previews to help inspect and better understand the datasets. He also supported defining how datasets, models, and evaluation metrics interact within the system, strengthening the overall system design.

Team Member – Pasha Zulfugarli (13731)
Pasha focused on analyzing the structure and spectral characteristics of the datasets, refining scripts to correctly handle multi-band images. He reviewed relevant PyTorch and deep learning documentation to ensure accurate data processing and organized the project folder hierarchy in the Git repository for clarity and accessibility.

Team Member – Huseyn Sadatkhanov (13770)
Huseyn implemented PyTorch scripts for loading and normalizing satellite images and developed visualization routines to explore the datasets. He contributed to organizing the project folder hierarchy and helped design the system workflow to ensure consistent and structured data processing.

Proposed Role Allocation for Semester 2

Semester 2 work focuses on implementing the complete benchmarking pipeline, integrating multiple datasets through a unified preprocessing interface, executing baseline models for CR, SR, and HD generation, and ensuring robust evaluation through metrics, visualization, and testing. To achieve this efficiently, responsibilities will be allocated according to each member's demonstrated Semester 1 contributions and the technical requirements of the Semester 2 milestones.

Team Lead – Royana Huseynova (11270)
Royana will lead the overall implementation and integration of the benchmarking pipeline, ensuring that all modules work together under a consistent architecture. Her responsibilities will include managing the Evaluation Engine development, enforcing reproducibility mechanisms (configuration logging, fixed seeds, and standardized experiment execution), coordinating milestone progress, and validating that metric computation is correct and stable across repeated runs. Royana will also oversee the final integration of visual outputs and ensure that the repository remains organized and aligned with the project's documentation and reporting requirements.

Team Member – Nijat Alisoy (17366)
Nijat will focus on dataset integration and preprocessing implementation. His responsibilities will include building and maintaining dataset loaders under the unified data interface, implementing normalization and spectral band alignment rules, validating dataset integrity, and generating preview samples to verify preprocessing correctness. He will also support performance optimization of dataset loading to ensure the pipeline remains efficient when handling large-scale imagery.

Team Member – Pasha Zulfugarli (13731)
Pasha will focus on research validation and benchmarking consistency, ensuring that datasets and preprocessing steps align with remote sensing standards and that evaluation protocols remain fair across tasks. His responsibilities will include maintaining and documenting band mapping rules, verifying dataset-specific properties, supporting metric applicability definitions, and contributing to structured reporting outputs (tables and comparisons across datasets and models). He will also contribute to maintaining clear documentation for dataset usage and preprocessing assumptions to support reproducibility and transparency.

Team Member – Huseyn Sadatkhanov (13770)
Huseyn will focus on visualization, qualitative analysis, and testing support. His responsibilities will include implementing qualitative comparison outputs (input/output/ground truth panels), developing difference maps and zoom-based visual inspection tools, and ensuring that visual outputs remain aligned and comparable across datasets. In addition, he will contribute strongly to testing efforts, including unit testing of preprocessing and metric modules, integration testing across dataset-model-metric workflows, and system-level testing of full end-to-end execution to confirm pipeline reliability.

3. Work Breakdown

The work breakdown for this project was defined based on the core use cases described in the User Design Document. These use cases reflect how researchers evaluate satellite image models, compare outputs, inspect visual quality, and ensure reproducible results, and they directly guide task identification. Task dependencies are defined to ensure a logical and sequential execution of the project and to support the construction of the Gantt chart. Dataset handling and preprocessing represent the initial phase of the project and must be completed before any model evaluation can begin, as all models rely on consistently prepared inputs. Model evaluation and metric computation are therefore dependent on the completion of preprocessing tasks. Visualization, qualitative analysis, and reproducibility mechanisms depend on the successful execution of model evaluation, since they require generated outputs and computed metrics. This finish-to-start dependency structure ensures that tasks are executed in the correct order and allows the work breakdown structure to be directly mapped to the project Gantt chart.

The first major task is dataset handling and preprocessing, which includes loading multiple public datasets such as AllClear, SEN12MS-CR-TS, WorldStrat, PROBA-V SR, and SAT25K. Sub-tasks include pixel normalization, spectral band alignment, dataset validation, and generation of preview samples to ensure consistent inputs across models.

Sub-tasks include:

- **Dataset integration and loader implementation:** building dataset-specific loaders (GeoTIFF readers via Rasterio) under a common interface so that all datasets can be accessed through the same pipeline entry point.
- **Data validation and integrity checks:** detecting missing tiles, corrupted files, empty/no data regions, and inconsistent shapes or resolutions; logging dataset statistics to confirm correctness.
- **Normalization and scaling:** applying consistent pixel normalization rules across datasets (e.g., handling different intensity ranges, no data values, and saturation).
- **Spectral band alignment and mapping:** defining configurable band selection/mapping rules so that models receive consistent band ordering and comparable inputs (e.g., Sentinel-2 RGB composites, multi-band inputs for CR tasks, etc.).
- **Preview sample generation:** producing visual previews (true-color RGB, band-wise visual checks, and sample triplets where applicable) to confirm that preprocessing produces valid and comparable inputs across datasets and tasks.

The second major task is model evaluation and metric computation. This task implements executing baseline or candidate models for each task and evaluating outputs in a consistent manner. This involves running cloud removal, super-resolution, and high-definition image generation models through the evaluation pipeline.

Sub-tasks include:

- **Model integration layer:** defining a unified model interface (input/output contracts) so that different architectures can be executed in the same pipeline without rewriting evaluation code.
- **Task-specific execution workflows:** supporting CR (cloudy input → restored output), SR (low-res input → super-res output), and HD generation (conditional or generative output → high-resolution result), while maintaining consistent handling of batch processing.
- **Metric library integration:** implementing standardized evaluation metrics, including visual fidelity metrics (**PSNR, SSIM**), spectral metrics (**SAM, ERGAS**), and no-reference/perceptual metrics (**QNR, FID**) where applicable. Metric computation will follow a consistent protocol across datasets and tasks, including correct handling of masks, no data regions, and multi-band evaluation.

- **Structured result reporting:** saving metric outputs in standardized formats (CSV/JSON), enabling cross-model comparisons, aggregation by dataset/task, and reproducible reporting of results.
- **Baseline experiments:** running initial benchmark experiments to validate that model execution and metric computation behave correctly and yield stable results across repeated runs.

The third major task focuses on visual inspection and reproducibility support. Sub-tasks include generating comparison triplets, difference maps, and zoom-in views, as well as logging configurations, preprocessing settings, and seeds. Task dependencies follow a logical order: preprocessing must be completed before model evaluation; evaluation must precede visualization, and reproducibility mechanisms rely on consistent execution of all previous tasks.

Sub-tasks include:

- **Qualitative comparison outputs:** generating side-by-side visual comparisons (input / output / ground truth where available), including consistent image scaling and alignment so that comparisons are meaningful.
- **Difference maps and error visualization:** producing absolute difference images, zoom-in crops, and region-of-interest views to highlight changes and artifacts introduced by models.
- **Experiment configuration and logging:** recording preprocessing settings, dataset versions/splits, model checkpoints, metric settings, random seeds, and runtime environment details so experiments can be replicated and audited.
- **Reproducibility enforcement:** ensuring deterministic execution where feasible (seed control, consistent preprocessing, fixed evaluation scripts) and producing structured run folders with all artifacts (configs, metrics, images, logs).
- **Comparison across runs:** enabling researchers to compare two models (or two preprocessing settings) on the same dataset subset and generate consistent summary outputs to support analysis and reporting.

4. Milestone Development and Timeline

All remaining project work will be completed by April 30, 2026, as scheduled for the project timeline. During semester 2, the focus will be on turning the system design and use cases

developed in semester 1 into a fully developed and tested benchmarking pipeline that can be reliably used to evaluate satellite.

The project milestones are defined based on functional capability and experimental reliability, ensuring that each development phase contributes directly to a usable and well-founded benchmarking system. To reflect these priorities, the semester 2 milestones define a clear development path from implementation to evaluation.

Milestone 1 – Model Execution and Metric Evaluation

Timeline: January 2026

This milestone implements the core benchmarking functionality of the system. Evaluation of metrics including PSNR, SSIM, SAM, ERGAS, QNR, and FID will be computed in a unified manner, and results will be recorded in structured formats suitable for comparison and analysis. Use https://github.com/chaofengc/IQA-PyTorch

Success Criteria:

- Metrics are computed correctly for each task
- Results remain consistent across repeated runs
- We tested it for Satellite imagery
- We pushed the code to the GitHub repository.

Milestone 2 – Dataset Integration and Preprocessing

Timeline: February 2026

This milestone focuses on implementing a unified data handling layer that allows multiple satellite datasets to be processed in a consistent manner. Key activities include integration of public datasets used in the project (AllClear, SEN12MS-CR-TS), applying uniform preprocessing steps such as normalization and band alignment, and validating dataset integrity. Sample visual previews will be generated to verify correct alignment and preprocessing before model evaluation.

Success Criteria:

- All datasets load through a common interface
- Preprocessing is applied consistently across datasets
- Input data is visually and numerically verified

Milestone 3 – Visualization and Analysis Support

Timeline: March 2026

This milestone focuses on making the results easy to understand and compare. The system will produce visual outputs that show the original input image, the model output, and the reference image side by side, along with clear difference views and close-up regions to highlight changes.

All experiment settings and parameters will be recorded so that results can be reviewed, compared, and repeated in a controlled and transparent way.

Success Criteria:

- Visual results are displayed clearly and match the same locations in the images, making differences easy to see
- All experiment settings are saved so results can be checked, reviewed, and repeated later
- The system makes it easy to compare results from different models and datasets

Milestone 4 – Testing, Optimization, and Final Evaluation

Timeline: March 2026

This milestone focuses on validating the correctness and reliability of the benchmarking pipeline and preparing final project deliverables. Testing will concentrate on ensuring that dataset handling, metric computation, and evaluation workflows operate as intended.

The following testing approaches will be used:

- Unit testing to verify individual components such as preprocessing functions and metric calculations
- Integration testing to ensure correct interaction between datasets, models, and evaluation modules
- System-level testing to validate end-to-end execution of the benchmarking pipeline

Final benchmark experiments will be executed, results will be analyzed, and documentation will be completed.

Success Criteria:

- Core pipeline components execute correctly from input to output
- Evaluation results are consistent across repeated runs
- Final benchmark outputs are complete and clearly interpretable

Milestone 5 – Preparation for Presentation, Final Report, and Repository Launch

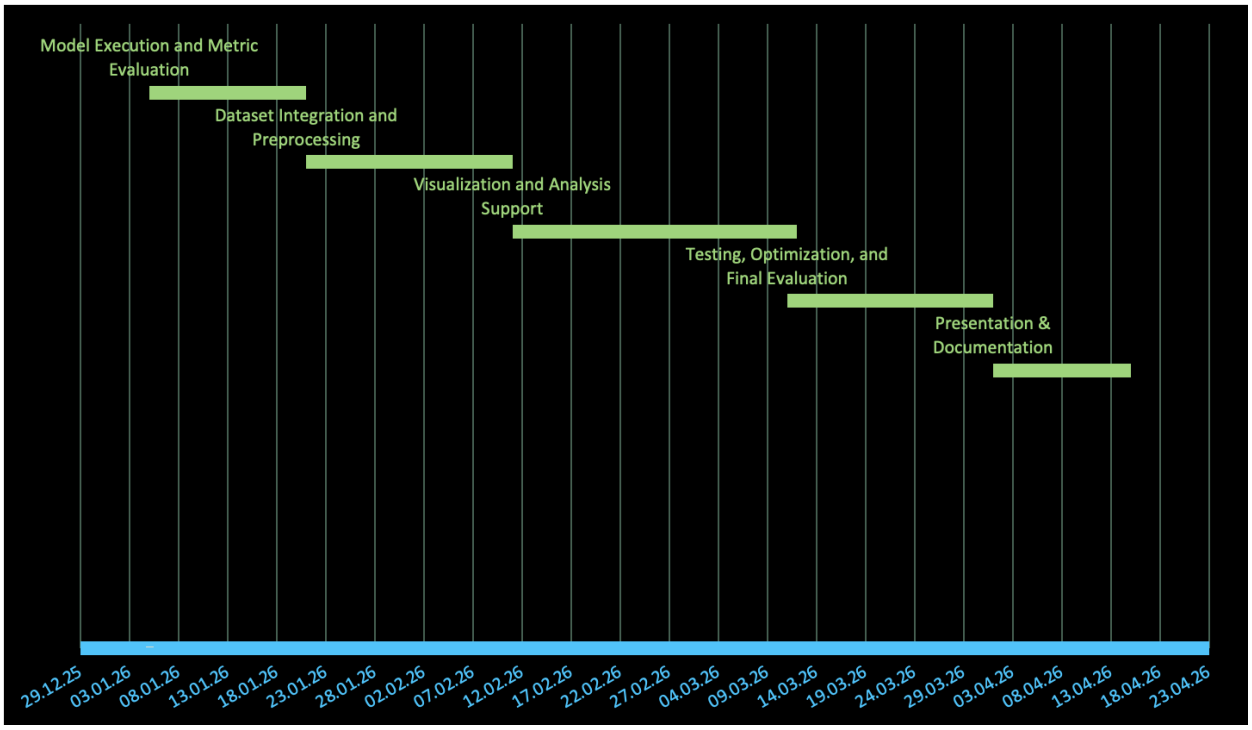
Timeline: April 2026

This milestone focuses on finalizing all project deliverables and preparing for submission. Activities include compiling results and visualizations, preparing the final report, documenting the benchmarking workflow, and launching the project repository to share code, datasets, and evaluation tools.

Success Criteria:

- Final report is complete, clear, and summarizes all project outcomes.
- Project repository is live, organized, and accessible, containing all code, datasets, and experiment documentation.
- Supporting materials, including dataset previews, model evaluation results, and experiment configurations, are uploaded and properly formatted.

Gantt Chart for project timeline:



5. Risk Assessment and Mitigation Planning

The following section outlines key technical and project risks for semester 2, together with planned mitigation strategies:

1.Potential risk is dataset inconsistency and incompatibility across the supported satellite datasets. Differences in spatial resolution, spectral band availability, and metadata quality may lead to preprocessing errors or unfair model comparisons. This risk will be mitigated by implementing dataset validation checks, configurable band-mapping rules, and consistent normalization procedures across all datasets.

2.Risk involves computational resource limitations, particularly when processing high-resolution satellite imagery or running multiple evaluation experiments. To address this, the system will support batch-based evaluation, selective dataset subsets, and the use of lightweight baseline models to ensure experiments remain feasible within available resources.

3. Another risk is metric misinterpretation or misuse, as not all evaluation metrics are equally applicable to all tasks or datasets. This risk will be mitigated by clearly defining

metric applicability for each task and documenting assumptions and limitations associated with each metric.

4. Risk is inconsistent with experimental results caused by changes in configuration settings or execution conditions across runs. This risk will be mitigated by using fixed experiment configurations, logging all relevant parameters, and following consistent execution procedures for every evaluation.

Contingency planning:

If full-scale benchmark execution is not feasible due to time or computational constraints, the project will prioritize core benchmarking functionality and experimental reliability. In such cases, evaluations will be conducted on representative subsets of datasets while maintaining consistent preprocessing and evaluation settings.

If certain datasets or models cannot be fully integrated, the system will still demonstrate its generality by supporting multiple tasks and at least one complete dataset–model–metric evaluation workflow per task. This ensures that the primary project objectives remain achievable even under constrained conditions.

6. Final / Confirmed Selection of Tools, Technologies, and Resource Planning

The project is implemented using Python, selected for its strong support for scientific computing, image processing, and deep learning workflows. The benchmarking pipeline and supporting utilities rely on widely adopted open-source libraries that are commonly used in satellite image analysis.

Core dependencies include PyTorch for numerical operations, Matplotlib and Pillow for visualization and image export, and Rasterio for reading and processing GeoTIFF (.tif) satellite imagery. These tools enable consistent handling of multi-band satellite data, including Sentinel-2–style imagery, and support tasks such as normalization, band selection, and visual inspection.

Development and experimentation are conducted in a cross-platform environment compatible with Windows, macOS, and Linux, with Visual Studio Code used as the primary development environment. Local workstations with optional GPU support are used for experimentation, while cloud-based resources may be leveraged if additional computational capacity is required for large-scale evaluations.

All software libraries, tools, and datasets used in the project are open-source or publicly available for research purposes, including the AllClear dataset and related satellite imagery datasets. As a result, the project does not require licensing costs or software procurement.

No specialized hardware purchases are planned. Existing personal or university-provided computing resources are sufficient to support development, visualization, and benchmarking tasks. If necessary, cloud-based computer resources may be used within free or academic usage limits, without affecting the overall project budget.

The project infrastructure was established and stabilized during semester 1, prior to the main development phase of semester 2. This included configuring the Python development environment, creating isolated virtual environments, installing required open-source dependencies, and verifying access to large-scale satellite image datasets such as AllClear.

Core utility scripts for dataset inspection and visualization were implemented and executed to validate correct handling of multi-band GeoTIFF satellite imagery across different platforms. These scripts confirmed proper reading of spectral bands, normalization behavior, and visual output generation, ensuring that the data pipeline functions as expected.

All dependency versions and environment configurations have been documented to maintain consistency across development machines. With the infrastructure fully in place, semester 2 work can focus exclusively on pipeline implementation, evaluation, and analysis without additional setup.

7. Evaluation and Validation Strategy

The evaluation and validation strategy focuses on ensuring the correctness, reliability, and usability of the benchmarking pipeline. Testing approaches are selected based on the nature of the system as a research-oriented evaluation framework.

Unit Testing will be used to validate individual components of the pipeline. This includes testing preprocessing functions, dataset loaders, and metric computation modules to ensure they behave as expected under controlled inputs.

Integration Testing will verify correct interaction between major system components, including datasets, models, preprocessing modules, and evaluation logic. This ensures that components function correctly when combined into a unified pipeline.

System Testing will be conducted to validate full end-to-end execution of the benchmarking workflow, from dataset input through model execution to metric output and visualization. This confirms that the system operates correctly under realistic usage scenarios.

Project success will be evaluated using both functional and experimental criteria. Key indicators include:

- Correct and consistent computation of evaluation metrics across datasets and tasks
- Successful execution of end-to-end benchmarking experiments
- Clear and interpretable quantitative and visual outputs
- Consistent results across repeated experiment runs

These criteria ensure that the system is usable as a reliable benchmarking tool rather than a simple demonstration.

Formal user acceptance testing is not required for this project, as the system is intended for use by researchers, familiar with satellite image evaluation workflows. Informal internal testing will be conducted by team members to simulate typical research usage scenarios and verify usability and clarity of outputs.

8. Ethical, Legal, and Professional Considerations

The project exclusively uses publicly available satellite imagery datasets intended for research purposes, such as AllClear and related datasets. These datasets do not contain personally identifiable information or sensitive personal data. As a result, the project does not pose direct privacy risks related to individuals or private entities.

All data is processed in its original research-provided form without attempting to infer or extract sensitive information beyond what is explicitly contained in the datasets. Dataset usage follows the terms and conditions specified by the original data providers.

From an ethical perspective, the primary concern in satellite image processing is the risk of misinterpretation or misuse of generated or enhanced imagery. Image restoration or

generation techniques may visually improve data while potentially introducing artifacts or misleading details.

To address this, the project emphasizes quantitative evaluation and transparent reporting rather than relying solely on visual quality. Multiple evaluation metrics are used to assess not only visual similarity but also spectral and structural consistency. This reduces the risk of presenting visually appealing but scientifically misleading results.

Additionally, the benchmarking system is designed to promote fair and consistent model comparison, avoiding biased evaluation practices that could favor specific models, datasets, or preprocessing choices.

The project follows standard academic research practices. This includes clearly documenting experiments, reporting results transparently, and properly citing all datasets, software libraries, and external tools used throughout the project.

All software used in the implementation is open-source and is used in accordance with its licensing terms. No proprietary, restricted, or paid tools are required. The project complies with university academic integrity guidelines and follows commonly accepted practices in machine learning and remote sensing research.

9. Conclusion and Forward Strategy

During semester 1, the project focused on establishing the technical foundation and initial tooling required for a unified satellite image benchmarking system. The team explored the structure and characteristics of large-scale satellite datasets, with particular emphasis on the AllClear dataset and similar GeoTIFF-based imagery.

As a concrete outcome, a cross-platform utility script was developed to support dataset inspection and visualization. This tool enables reliable reading of multi-band satellite images, construction of true-color RGB previews, handling of no data and empty tiles, and visual validation of normalization and scaling behavior. The successful implementation and execution of this tool confirmed correct access to datasets and validated the basic data handling assumptions needed for later benchmarking work.

By the end of semester 1, the project infrastructure was fully set up, dependencies were stabilized, and preliminary dataset understanding was achieved, providing a solid starting point for full pipeline development in semester 2.

In semester 2, the project will build directly on the established infrastructure and dataset tooling. The focus will shift to implementing a complete benchmarking pipeline that integrates multiple satellite datasets through a unified preprocessing interface.

Baseline models for cloud removal, super-resolution, and high-definition image generation will be executed within this pipeline, and evaluation of metrics will be computed consistently across tasks. Visualization tools developed in semester 1 will be extended to support qualitative comparison alongside quantitative evaluation. Testing and validation will ensure correct end-to-end execution, followed by final benchmark experiments and documentation.

The benchmarking framework can be extended to support additional datasets, models, and satellite image processing tasks. The modular design allows future contributors to reuse and expand the system without modifying core components.In the long term, the project can serve as a practical reference implementation for standardized evaluation practices in satellite image restoration and generation, supporting more reliable and transparent model comparison in remote sensing research.

Appendices

A. Research Report:
https://docs.google.com/document/d/1QmQ_U5gRzTdU83ulaYwgJEiQnqcgDfBYls9Z_TkG8es/edit?tab=t.0

B. User Design:
https://adauniversity-my.sharepoint.com/:w:/r/personal/pzulfugarli13731_ada_edu_az/_layouts/15/Doc.aspx?sourcedoc=%7BF0670D0E-DCEE-44B0-BC97-BBE357BC5BCA%7D&file=User%20Design.docx.docx&fromShare=true&action=default&mobileredirect=true

C. Supplementary diagrams:

UML Class Diagram:

https://drive.google.com/file/d/1SyosNaPZx-fHfOX2rzm44OSESEXnp9V-/view?usp=sharing

https://www.figma.com/design/rEF6XX0suEK3RX6ZDPE7dN/SDP-2026?node-id=262-2&t=z6hBXMo5nYjN9T3F-1

D. Preliminary code or prototypes:
https://github.com/Royana-Huseynova/SDP-2026