**8-Week Thesis Plan**

AI-Powered Wildfire Detection and Monitoring Using Visual Imagery with Explainable Deep Learning Models

**Week 1: Data Acquisition & Engineering**

* Collect and preprocess multi-year Canadian wildfire imagery (GeoTIFF/Landsat/Sentinel, NASA FIRMS).
* Tile images into standardized 256×256 patches, generate pixel-level fire/no-fire masks, confirm label validity by random visual inspection.
* Organize splits for train/validation/test; save processed tile directories and metadata for reproducibility.
* Write code scripts for cleaning, labeling, and automate pipeline steps.

**Week 2: Exploratory Analysis & Monitoring Setup**

* Visualize fire/no-fire samples for spatial pattern and ground truth check.
* Analyze class balance, noise, and outliers in labeled data (spectral histograms, fire frequency maps).
* Configure real-time monitoring demo using Streamlit; validate ingestion and visualization with random tiles.
* Document observations and impact for monitoring robustness.

**Week 3: Baseline Model Training**

* Implement and train a lightweight baseline CNN classifier and fire pixel segmentation model.
* Record performance: accuracy, F1, recall, IoU; plot learning and confusion matrices for both classes.
* Save all code/notebooks for baseline model reproducibility and downstream benchmarking.

**Week 4: Transfer Learning & Advanced Models**

* Apply ResNet50 and DenseNet121 architectures and transfer learning on fire/no-fire tiles using Keras/TensorFlow.
* Conduct data augmentation (flips, rotations, zooms) and tune hyperparameters for detection precision.
* Compare training logs and validation metrics across architectures; discuss suitability for operational monitoring.

**Week 5: Explainable Deep Learning Implementation**

* Integrate Grad-CAM and SHAP for CNN, ResNet, and DenseNet models to localize feature importance.
* Generate visual overlays for fire/no-fire decisions on test tiles to aid human interpretability.
* Summarize explainability outcomes for critical and ambiguous cases; enhance dashboard for explanation display.

**Week 6: Real-time Monitoring Pipeline**

* Deploy best model into a scalable pipeline, incorporating continuous satellite feed simulation.
* Automate prediction and visualization of fire progression on new imagery.
* Develop or expand Streamlit dashboard for live reporting to agencies and stakeholders; validate with synthetic events.

**Week 7: Final Model Evaluation and Error Analysis**

* Run held-out test set and operational scenarios, document model strengths, failure modes, and edge cases.
* Analyze explainability visualizations (Grad-CAM, SHAP overlays) for trustworthiness and actionable insights.
* Refine monitoring pipeline, add statistical reporting (precision/recall per month/region).

**Week 8: Reporting, Demo, & Deployment**

* Prepare all thesis deliverables: code repos, annotated notebooks, full pipeline documentation.
* Write detailed thesis chapters—methods, monitoring workflow, experiments, explainability results, limitations.
* Finalize live Streamlit dashboard for demonstration, include video or scenarios for presentation.
* Conduct mock stakeholder demo and integrate feedback for deployment readiness.

My teacher expectation:

Week 1 results:

* A dataset usable to complete the task of interest

Week 2 results:

* data analysis with basic statistics, insights, and things to tell me more about the data
* issues with the dataset that require attention

Week 3 results:

* Baseline model with the metrics (accuracy, F1 score etc.) + confusion matrix

Week 4 results:

* Metrics (accuracy, F1 and others) for the more advanced models than the baseline

Week 5 results:

* Report (with images, tables) based on GRAD-CAM, SHAP etc.
* Prepare a report on Bayesian Interpretation with the confusion matrix (<https://betterexplained.com/articles/an-intuitive-and-short-explanation-of-bayes-theorem/>)
* Other artifacts that help explain the model (provide later)

Week 6:

* I am not sure about the data pipeline.
* In my opinion, on Week 6 building some cool Streamlit app with dashboards would be more than sufficient.
* Hence, model in Streamlit with some user interface for inference and dashboards should be enough.

Week 7 results:

* Model error analysis and improvement

Week 8 results:

* Finalize model, streamlit, inference
* Finalize the report

you can check this working solution based on satellite imagery - <https://github.com/pereira-gha/activefire>

dataset and instruction how to download it is included

I like this one - <https://data.mendeley.com/datasets/fgvscdjsmt/4>

also you can check this one <https://www.kaggle.com/datasets/brsdincer/wildfire-detection-image-data>

Wildfire Detection Image Data

For Machine Learning Process

consolidated list of datasets here - <https://github.com/AlimTleuliyev/wildfire-detection>