Medical Decision Support Project: Next 2 Weeks Guide (3-Person Team)

WHAT WE ARE BUILDING

We are building a system that helps doctors diagnose pneumonia and decide on treatment. The system takes patient information (symptoms, lab results, vital signs) and outputs two things: (1) How likely the patient has pneumonia, and (2) What treatment to recommend.

We are building this system two different ways and comparing which one works better.

WHY PNEUMONIA?

Pneumonia is an infection in the lungs that makes it hard to breathe. It is common, serious, and kills people if not treated. Doctors need to diagnose it quickly.

We chose pneumonia because:

- **Clear diagnostic signals.** Patients show specific symptoms (fever, cough, shortness of breath) and lab results (low oxygen, high white blood cells). We can measure these things.
- **Established treatment guidelines.** If a patient has pneumonia, doctors know what medications work. There are maybe 3-4 main treatment options depending on severity.
- **Good datasets exist.** Hospitals have recorded thousands of pneumonia cases with symptoms, lab values, and outcomes. These datasets are publicly available for research.

Compared to other diseases, pneumonia is straightforward. Cancer or rare genetic disorders would be much harder because the data is messier and treatment options are less clear. Pneumonia is a good teaching example.

WHAT OUR SYSTEM DOES

Step 1: We get patient data (fever: yes, cough: yes, oxygen level: 88, white blood cell count: 11000)

Step 2: Our system runs analysis and outputs:

> "This patient has 82% probability of pneumonia. Recommend hospitalization with oxygen therapy."

That is our goal. Two things: probability and treatment recommendation.

TWO DIFFERENT APPROACHES WE WILL BUILD

We are building the system two ways. This comparison is the whole point.

APPROACH A: BAYESIAN NETWORK (Complex but Interpretable)

- This is a diagram showing cause-and-effect. Fever and cough point to high probability of pneumonia.

Pneumonia points to oxygen being low.

- We can trace the logic: "Why did the system say pneumonia? Because the patient had fever and low oxygen, and we know 75% of patients with those symptoms have pneumonia."
- **Advantages:** We can explain the decision. We can tell doctors why the system thinks there is pneumonia.
- **Disadvantages:** Takes longer to build. Might not be as accurate as other methods.

APPROACH B: RANDOM FOREST (Simple but Black Box)

- This is a machine learning model that learns patterns from thousands of patient records. It is very good at finding patterns but does not explain why.
- A doctor sees: "This patient has pneumonia" but cannot ask why. The model learned it from data, not from logical rules.
- **Advantages:** Often more accurate. Faster to code.
- **Disadvantages:** Doctors do not know why the system made the decision. Harder to trust.

We will build both, measure accuracy for each, and then write about which is better for real hospitals.

WHY THIS MATTERS

In real hospitals, doctors need to trust the AI system. If the AI says "admit this patient to the hospital" but the doctor does not understand why, the doctor will ignore it. So even if Approach B is slightly more accurate, Approach A might be better because doctors understand it.

That is what we will discuss in our final report.

COMPLETE 12-WEEK PROJECT PLAN

OVERVIEW: WHAT WE WILL DELIVER

By week 12, we will have completed:

- A working Bayesian Network system for pneumonia diagnosis and treatment recommendation
- A working Random Forest model built on the same data for comparison
- Accuracy measurements for both systems (precision, recall, F1 score, calibration)
- Treatment recommendation rules validated on test data
- A comprehensive final report comparing both approaches
- A presentation with results and insights
- Code repositories with working implementations

PHASE BREAKDOWN: WEEKS 1-12

Week(s)	Phase Name	Main Activities	Owner(s)	Deliverable
1-2	Data Setup & Team Initialization	Download Kaggle dataset, apply for MIMIC, assign roles, explore data, understand patterns	All (A leads)	Dataset + role assignments + initial summary
3-4	Decision Tree Development	Build and optimize decision tree, test on validation set, extract tree structure, document rules	Person A	Working decision tree model + accuracy metrics
5-6	Bayesian Network Construction	Convert tree structure to BN, calculate probability tables, implement Laplace smoothing, validate on test set	Person B	Working BN model + probability tables + test results
7-8	Treatment Integration & System Testing	Write treatment recommendation rules, code treatment function, integrate with BN, comprehensive testing	Person C (with A & B support)	Integrated system with treatment recommendations + test results
9-10	Random Forest Comparison Model	Build Random Forest classifier, train on same dataset, test on test set, collect all comparison metrics	Person A	Working Random Forest model + full comparison metrics vs BN
11-12	Final Report & Presentation	Write comprehensive report, create visualizations, prepare presentation slides, plan demo	Person C (input from A & B)	Final written report + presentation slides + demo code

---

DETAILED WEEKLY BREAKDOWN

WEEKS 1-2: DATA SETUP & TEAM INITIALIZATION

Goals

- Have working dataset ready for modeling
- Understand data quality and patterns

- Have clear role assignments
- Have first working model (decision tree)

Person A Activities

- Download Kaggle Pneumonia Dataset
- Apply for MIMIC-IV credentials
- Explore dataset: check data types, missing values, distributions
- Split data: 700 training, 150 validation, 150 test
- Begin decision tree coding with scikit-learn

Person B Activities

- Review research papers (P1, P2, P3 from Manus document)
- Learn pgmpy library basics (watch tutorials, practice with toy examples)
- Understand Bayesian Network structure and probability concepts

Person C Activities

- Review research papers (P4, P5, P8 on treatment and decision trees)
- Document treatment guidelines from literature
- Begin creating outline for final report

By End of Week 2

> We have: Downloaded dataset, applied for MIMIC, assigned roles, built first decision tree with 80+ percent accuracy, extracted treatment rules, written week 1-2 summary

WEEKS 3-4: DECISION TREE DEVELOPMENT

Goals

- Optimize decision tree for accuracy
- Document tree structure for BN conversion
- Extract clinical insights from tree

Person A Activities

- Tune decision tree hyperparameters (max depth, min samples split)
- Test on validation set, adjust parameters
- Calculate precision, recall, F1 score
- Document tree structure (text file showing decision rules)
- Save tree visualization as image

Person B Activities

- Study Person A's tree structure
- Begin designing BN structure based on tree
- Set up pgmpy project and test basic BN creation

Person C Activities

- Receive tree structure from Person A
- Refine treatment rules based on actual tree output
- Start writing about decision tree approach

Deliverable by End of Week 4

> Decision tree finalized with documented accuracy metrics, tree structure visualization, and decision rules extracted

WEEKS 5-6: BAYESIAN NETWORK CONSTRUCTION

Goals

- Build working Bayesian Network from decision tree structure
- Calculate all probability tables
- Test BN on test dataset

Person A Activities

- Provide training data to Person B in clean format
- Keep test data secure (Person B does not see test data yet)

Person B Activities

- Create BN structure in pgmpy based on decision tree
- Calculate Conditional Probability Tables (CPTs) from training data
- Implement Laplace smoothing for rare combinations
- Test BN inference on sample patients
- Debug and fix any errors

Person C Activities

- Prepare final treatment rule code
- Start writing section about treatment recommendations

Deliverable by End of Week 6

> Working Bayesian Network with probability tables, test on training data showing 75+ percent accuracy, documented code with comments

WEEKS 7-8: TREATMENT INTEGRATION & SYSTEM TESTING

Goals

- Integrate treatment recommendation system with BN
- Test complete end-to-end system
- Measure all required metrics

Person A Activities

- Provide test dataset to Person B
- Run decision tree on test set, collect final accuracy metrics

Person B Activities

- Run BN on test set (first time seeing test data)
- Calculate accuracy metrics for BN
- Generate calibration curves
- Measure inference time

Person C Activities

- Code treatment recommendation function
- Integrate treatment function with BN output
- Test complete system on 20 sample patients
- Document any issues and fixes

All Team Activities

- Test end-to-end: patient data → BN → diagnosis probability → treatment recommendation
- Check for bugs and edge cases

Deliverable by End of Week 8

> Fully integrated Bayesian Network system with treatment recommendations, tested on test dataset, all accuracy metrics collected

WEEKS 9-10: RANDOM FOREST COMPARISON MODEL

Goals

- Build Random Forest on same training data
- Test on same test dataset
- Collect all comparison metrics

Person A Activities

- Build Random Forest classifier with scikit-learn
- Train on same training data used for BN and decision tree
- Tune hyperparameters (number of trees, max depth, etc.)
- Test on exact same test dataset
- Calculate precision, recall, F1 score, calibration curves, inference time

Person B & C Activities

- Help debug if issues arise
- Start preparing comparison charts and visualizations

Deliverable by End of Week 10

> Working Random Forest model with complete test results showing direct comparison to Bayesian Network and decision tree approaches

WEEKS 11-12: FINAL REPORT & PRESENTATION

Goals

- Write comprehensive final report
- Create visualizations comparing approaches
- Prepare presentation
- Prepare for questions

Person C Activities (Lead)

- Collect all results and metrics from A and B
- Write Introduction (project overview)
- Write Methodology section (explain all three approaches)
- Write Results section (present findings)
- Write Discussion section (trade-offs, insights)
- Write Conclusion section (which approach is better and why)

Person A Activities

- Provide decision tree and Random Forest results
- Provide accuracy comparison data
- Help create visualizations (accuracy charts, confusion matrices)

Person B Activities

- Provide Bayesian Network results
- Help create BN structure visualization (draw actual network diagram)

- Explain probability concepts in report

All Team Activities

- Create presentation slides (5-8 slides maximum)
- Prepare demo: run live example of system on new patient data
- Practice presentation delivery
- Prepare answers to likely questions

Report Structure

- **Title Page:** Project name, team members, date
- **Executive Summary:** One paragraph overview of what we built and findings
- **Introduction:** Why pneumonia matters, why we chose it, project goals
- **Related Work:** Summary of 3-4 research papers we used
- **Methodology:** Decision tree approach, Bayesian Network approach, Random Forest approach, treatment recommendations
- **Data:** Dataset description, size, features, preprocessing
- **Results:** Accuracy results for all three approaches, visualizations, calibration curves
- **Discussion:** Trade-offs between approaches, why each approach matters, insights
- **Conclusion:** Which approach is better for real hospitals and why
- **References:** All papers and sources we used

Deliverable by End of Week 12

> Comprehensive written report (8-10 pages), presentation slides, working code, and preparation for oral presentation

---

KEY MILESTONES & DECISION POINTS

Week(s)	Milestone	Decision/Gate
End of Week 2	First working model	If accuracy < 70%, adjust tree parameters or reconsider features
End of Week 4	Optimized decision tree	If accuracy < 75%, do additional feature engineering
End of Week 6	Bayesian Network working	If BN has issues, may extend week 7 for debugging
End of Week 8	Complete integrated system	If treatment rules need adjustment, update before week 9
End of Week 10	All comparison metrics	All data collected and ready for final report
End of Week 12	Final report + presentation	Project complete

OUR PLAN FOR THE NEXT 2 WEEKS (DETAILED)

WEEK 1: GET OUR DATA AND SET UP

Task 1: Download one public dataset

> We will go to Kaggle and download the "Pneumonia Dataset with Clinical Data" (it is free, no verification needed). This is a spreadsheet with 1000+ patient records. Each row has: fever (yes/no), cough (yes/no), oxygen level, white blood cell count, diagnosis (pneumonia or not).

- **Why:** We need data to build with. We cannot build a system without examples.
- **Time:** 15 minutes download and unzip.

Task 2: Apply for MIMIC-IV access

> We will go to PhysioNet.org and apply for MIMIC-IV access. This is real hospital data with thousands of pneumonia cases. It is bigger and more realistic than Kaggle data, but requires credentials.

- **Why:** Kaggle data is practice. MIMIC-IV is the real thing. If we get approved, we use MIMIC. If not, we finish with Kaggle.
- **Time:** 10 minutes to apply. Then we wait. Takes 3-7 days to get approved.

Task 3: Understand our data

> We will open the Kaggle dataset in Excel or Python. We will look at 10 rows and understand what each column means. Example:
>
> Patient 1: fever=yes, cough=yes, oxygen=92, WBC=10000, diagnosis=pneumonia
> Patient 2: fever=no, cough=no, oxygen=98, WBC=7000, diagnosis=no pneumonia
>
> This helps us see the pattern: patients with pneumonia have fever, cough, and low oxygen.

- **Time:** 30 minutes.

Task 4: Decide our roles for the next 12 weeks

> **Person A: Data & Modeling Lead**
>
> - Download and prepare datasets (weeks 1-2)
> - Build decision tree (weeks 3-4)
> - Build Random Forest (weeks 9-10)
> - Handle all data cleaning and preprocessing

> **Person B: Bayesian Network Specialist**
>
> - Learn pgmpy library (weeks 1-4)
> - Build Bayesian Network using pgmpy library (weeks 5-6)
> - Calculate probability tables from data
> - Test BN on test dataset
> - Deliver results to report writer

> **Person C: Treatment & Report Lead**
>
> - Review treatment guidelines from papers (weeks 1-2)
> - Write treatment recommendation rules (weeks 7-8)
> - Code the treatment recommendation function

- Collect results from all team members (weeks 9-10)
- Write final report and prepare presentation (weeks 11-12)

- **Why:** So we each know what we are responsible for and we do not duplicate work.
- **Time:** 30 minutes team meeting.

WEEK 2: BUILD OUR FIRST SIMPLE MODEL (PRACTICE)

Task 1: Build a decision tree

> Person A will open Python and use scikit-learn library to build a decision tree. This is a diagram that asks questions: "Does patient have fever? If yes, go left. If no, go right."
>
> The output will look like:
>
> - If fever=yes AND oxygen<90: Likely pneumonia
> - If fever=yes AND oxygen>90: Maybe pneumonia
> - If fever=no: Likely no pneumonia
>
> This is our first working model. It takes 2-3 hours of coding and learning.

- **Why:** This is practice. We will use this structure to build our Bayesian Network in weeks 5-6.
- **Time:** 4-6 hours of work for Person A.

Task 2: Test the decision tree

> Person A will take 100 patient records that the decision tree never saw before. Person A will run the tree on them and count how many we got right.
>
> Example: "Out of 100 patients, the tree correctly diagnosed 87 of them. That is 87% accuracy."

- **Why:** So we know if our model works or if it is garbage.
- **Time:** 1 hour.

Task 3: Document what we learned

> Person C will look at the decision tree output and write down the rules in plain language:
>
> "Rule 1: If fever AND low oxygen, recommend hospitalization.
> Rule 2: If fever AND normal oxygen, recommend outpatient antibiotics.
> Rule 3: If no fever, recommend observation."
>
> These rules will become our treatment recommendation system later.

- **Why:** So treatment recommendation is not a mystery later. We are basing it on our data.
- **Time:** 1 hour.

Task 4: Write up week 1-2 summary

> Person C will write 1 page:
>
> - What dataset we used (Kaggle Pneumonia Dataset)
> - How many patients (1000)

> - Accuracy of our first decision tree (87%)
> - What we learned (fever and low oxygen are strong indicators of pneumonia)
>
> This becomes part of our final report.

- **Time:** 1-2 hours.

WHAT WE WILL HAVE BY END OF WEEK 2

- A downloaded dataset ready to use
- An application for MIMIC-IV submitted (just waiting for approval)
- Our team roles clearly assigned
- A working decision tree model (even if not perfect)
- Test results showing 80+ percent accuracy
- Treatment rules written down
- A summary document for our report

At this point we are no longer worried about whether this is possible. We have a working model. Weeks 3-12 are just making it better and comparing our two approaches.

OUR TIMELINE: ALL 12 WEEKS

Weeks	Phase	What We Do	Owner	Key Deliverable
1-2	Data & Setup	Download data, build first model, assign roles	All (A leads)	Working decision tree
3-4	Decision Tree	Optimize tree, extract structure, document rules	A	Optimized tree + metrics
5-6	Bayesian Network	Build BN with probability calculations	B	Working BN system
7-8	Treatment & Testing	Add treatment recommendations, end-to-end testing	C (with A & B)	Complete BN + treatment system
9-10	Random Forest	Build comparison model, collect metrics	A	RF model + comparison data
11-12	Report & Presentation	Write report, create slides, prepare demo	C (with A & B input)	Final report + presentation

Each phase builds on the last. By week 8 we have a working system. Weeks 9-12 are comparison and documentation.

WHAT COULD GO WRONG (AND HOW WE WILL HANDLE IT)

Problem 1: MIMIC-IV access is denied or takes too long

Our solution: We already have Kaggle data. We finish with Kaggle. Our report says "We used Kaggle dataset due to MIMIC access delays. Real hospital data would improve generalization."

Problem 2: Decision tree gets only 60% accuracy

Our solution: That is OK. We use it anyway. We explain in our report: "Initial decision tree achieved 60% accuracy. Our Bayesian Network improved this to 75%."

Problem 3: Bayesian Network code is confusing

Our solution: That is normal. pgmpy library has good documentation. Person B will watch a 20-minute tutorial on YouTube. After learning the basics, the code becomes clear. We might need to extend weeks 5-6 to weeks 5-7.

Problem 4: One of us gets sick or busy

Our solution: That is why we have assigned roles. If Person A is stuck, Person B or C can help. We are a team, not individuals.

Problem 5: We run out of time

Our solution: The minimum viable project is: (1) Decision tree, (2) Test results, (3) Treatment rules, (4) Final report. We skip the Random Forest comparison if we have to. We still get a passing grade.

KEY PRINCIPLE

This project is not about getting everything perfect. It is about building a working system, comparing two approaches, and writing about the trade-offs. Even 75% accuracy is fine. Even using Kaggle instead of MIMIC is fine.

Our professor cares about: Did we understand the methodology? Did we build it? Did we test it? Did we compare approaches? Did we think critically about results?

We do not need 99% accuracy or perfect code. We need a working system and honest analysis.

NEXT STEPS

This week:

- All three of us meet and confirm our roles
- Person A downloads Kaggle dataset immediately
- Person A applies for MIMIC-IV access
- All three of us examine the first 10 rows of the dataset
- Person B begins learning pgmpy basics
- Person C reviews treatment guidelines from research papers

Next week:

- Person A builds decision tree with scikit-learn
- Person A tests it on 100 new patient records
- Person C documents the treatment rules we discovered
- Person C writes our week 1-2 summary

We have everything we need. Let's start now.