Emergency Room (ER) Triage System – Complete Project Overview

1. Project Goal

This project builds an Emergency Room (ER) Triage System that helps nurses and doctors identify which patients should receive treatment first. The system records patient vital signs and symptoms, calculates priority using medical triage rules, and displays a real-time waiting list sorted by urgency. It automatically updates whenever a patient's vitals change.

2. Team Roles and Responsibilities

Person	Role	Responsibilities
A	Backend Developer	Develops the triage algorithm (priority logic). Reads vitals and symptoms, calculates score, level, and reason.
B	Database Manager	Creates and manages the MySQL database. Builds the patients table, inserts test data, and ensures consistent structure.
C	Frontend Developer (JSP/HTML)	Designs and implements the web pages for nurses: patient intake form, waiting queue table, and update vitals page. Ensures a clean, easy-to-navigate interface.
D	Backend Assistant & QA Tester	Assists the backend developer with algorithm and servlet integration. Runs tests, fixes errors, and prepares the demo presentation.

3. Tools Used

• Git + GitHub – Team collaboration and version control
• MySQL – Database for patient records
• Java (Servlets) – Backend and triage logic
• JSP/HTML – Frontend pages
• Apache Tomcat – Runs the web app
• Eclipse – Development environment

4. Database (patients table)

The MySQL table stores each patient's information: id, name, age, gender, bp_sys, bp_dia, hr, rr, spo2, temp, symptom, created_at, and status. This table provides the data used by the triage algorithm to determine urgency.

5. Triage Algorithm Overview

The triage algorithm decides who should receive medical attention first. It uses two key processes: detecting life-threatening 'Red Flags' and calculating a 'Score' from patient vitals and symptoms. The algorithm then maps each patient to a Triage Level (1–3) and sorts the waiting list accordingly.

5.1 Red Flags – Immediate Life Threats

Red Flags are critical conditions that indicate immediate danger to life. If any are detected, the patient is instantly assigned Level 1 (Immediate) and marked as 'In Treatment'. These include severe failures in airway, breathing, circulation, or neurological function (ABCD model).

• Airway/Breathing: $SpO_2 \leq 88\%$, RR ≥ 35 with distress, 'severe dyspnea', 'cyanosis', 'asthma severe', 'pneumothorax', or 'chest trauma'
• Circulation: SBP < 90 mmHg, 'severe bleeding', 'massive trauma', 'GI bleed', or 'chest pain' with jaw/arm/diaphoresis (heart attack)
• Neurological: 'unresponsive', 'head injury' with 'unequal pupils' or 'seizure', 'stroke', 'slurred speech', or 'one-sided weakness'
• Combined Trauma: gunshot, stab wounds, high falls, or multiple failing vitals (e.g., SpO_2 84%, BP 78/40, RR 38)

In real ERs, such patients go directly to treatment. The system still records them automatically as 'In Treatment' so staff can manage room occupancy and documentation later.

5.2 Scoring System (for Non-Red-Flag Patients)

If no red flags are detected, the algorithm calculates a score based on abnormal vital signs and critical symptom keywords. Each deviation from normal adds weighted points to reflect how unstable the patient may become.

Vital Sign Scoring

SpO_2: ≤90 (+4), 91–93 (+2), 94–95 (+1)
SBP: <90 (+4), 90–99 (+3), 100–109 (+1)
HR: ≥140 or <50 (+3), 120–139 (+2), 110–119 (+1)
RR: ≥30 (+2), 22–29 (+1), <8 (+3)
Temp: ≥39.5 or ≤35 (+1)
Age: ≥65 (+1)

Symptom Scoring

- Cardiac: 'chest pain', 'jaw pain', 'sweating', 'nausea' (+2–4)
- Respiratory: 'difficulty breathing', 'asthma', 'cyanosis' (+1–3)
- Neurological: 'confusion', 'seizure', 'slurred speech', 'stroke' (+2–4)
- Trauma/Bleeding: 'bleeding', 'fracture', 'stab', 'gunshot' (+2–4)
- Infection: 'fever', 'infection', 'sepsis', 'vomiting' (+1–2)

The total score is the sum of all applicable points. Multiple mild abnormalities can combine into a moderate or high score.

5.3 Mapping Total Score to Triage Level

After computing the total score, the system maps it to a triage level:

- Score ≥ 9 → Level 1 (Immediate)
- Score 6–8 → Level 2 (Very Urgent)
- Score 3–5 → Level 3 (Urgent)
- Score 0–2 → Level 4/5 (Low Acuity – optional)

This mapping ensures that even patients without explicit red flags but with multiple abnormal vitals are prioritized correctly.

5.4 Realistic Handling of Red Flag Patients

In real emergency departments, Level 1 (red flag) patients are taken directly to treatment rooms. However, their data is still entered into the system by another nurse or later by the same nurse for record-keeping and triage management. This ensures accurate tracking of room usage and critical case statistics.

In this system, when a red flag is detected, the patient is assigned 'In Treatment' automatically. The main waiting queue only lists Levels 2 and 3. This mirrors real hospital behavior: immediate patients are treated first, while others are ordered by urgency.

6. Queue Sorting Logic

1. Red flag patients (Level 1) → Status = In Treatment (not in queue)
2. Remaining patients are sorted by:
   - Triage Level (2 before 3)
   - Higher score within same level

- Earlier arrival time
- Older age if tied

7. Demo Flow and Explanation

1. Add a Level 2 or Level 3 patient → appears in waiting queue sorted by score.
2. Add a critical (red flag) patient → automatically set as 'In Treatment' and excluded from the queue.
3. Update a patient's vitals (e.g., SpO_2 drops) → re-triaged automatically; if new red flag → moves to In Treatment.
4. Demonstrate that queue always shows only waiting patients (Levels 2 and 3).

8. Summary

The ER triage algorithm combines medical reasoning and data logic to prioritize patients fairly and safely. Red Flag patients receive immediate care but remain documented for hospital tracking. Others are scored based on vitals and symptoms, mapped to urgency levels, and displayed in an automatically updating queue.

9. Detailed Queueing Logic

The queueing logic determines the order in which patients are treated. It ensures that the most urgent cases are handled first while maintaining fairness and medical realism. The queue constantly updates based on each patient's triage level, score, and arrival time.

9.1 Purpose of the Queue

The queue is not a simple first-come, first-served list. Instead, it is a dynamic priority queue that ranks patients based on urgency. It automatically reorganizes itself whenever new patients arrive, vitals change, or someone is moved into treatment.

9.2 Who Appears in the Queue

Only patients whose status is marked as 'Waiting' are displayed in the queue. Those who are currently in treatment or already discharged are excluded. This ensures the queue always shows only patients who are waiting to be seen.

- Waiting → displayed in queue
- In Treatment → excluded (already being treated)
- Treated → excluded (completed cases)

9.3 Sorting Priorities

Each patient is evaluated using four sorting priorities in this exact order:

1. Triage Level (ascending): Level 1 is most urgent, followed by Level 2, then Level 3.
2. Score (descending): Within the same level, patients with higher scores appear first.
3. Arrival Time (ascending): If patients share the same level and score, earlier arrivals appear first.
4. Age (descending): If all else is tied, older patients are seen first due to frailty considerations.

9.4 Example of Queue Sorting

Example:
- Faisal – Level 2, Score 8, Arrived 12:10, Age 50 → 1st
- Noura – Level 2, Score 8, Arrived 12:12, Age 70 → 2nd
- Ahmed – Level 2, Score 6, Arrived 12:05, Age 80 → 3rd
- Sara – Level 3, Score 9, Arrived 12:00, Age 30 → 4th

Result: Faisal appears first because he has the same level and score as Noura but arrived earlier.

9.5 Real-Time Updates

The queue reacts immediately to any change:
- If a patient's vitals worsen and trigger a red flag, their status becomes 'In Treatment' and they are removed from the queue.
- If a new patient arrives with a high score, they are instantly inserted into the correct

position.
• If a patient improves and their score decreases, their level is downgraded and they move lower in the queue.

9.6 Database Query Logic

The system retrieves the waiting list using a structured SQL query that automatically applies the medical priority rules:

```
SELECT * FROM patients
WHERE status = 'Waiting'
ORDER BY triage_level ASC, triage_score DESC, created_at ASC, age DESC;
```

This ensures that urgency, score, time, and age are all factored into the display order in a single step.

9.7 User Interface Behavior

The queue page displays patients with clear color-coded badges:
• Red (Level 1): Critical – In Treatment, not shown in queue
• Orange (Level 2): Very Urgent – top of queue
• Yellow (Level 3): Urgent – lower priority

Each entry also shows the reason for the assigned level (e.g., 'SpO_2 89% + chest pain → Level 2').

9.8 Medical Realism in Presentation

During demonstrations, it is explained that Level 1 patients bypass the queue entirely to reflect real-life ER behavior. The waiting list represents Levels 2 and 3, and it dynamically reorders itself as conditions change. This approach models both fairness and accuracy—critical patients receive immediate care, while stable patients wait safely.

9.9 Visual Priority Bands

To improve clarity, each triage level is represented by color and expected wait time:
• Level 1 (Red) – Immediate care (0 minutes)
• Level 2 (Orange) – Very urgent (within 10–15 minutes)
• Level 3 (Yellow) – Urgent but stable (within 30–60 minutes)

9.10 Summary

The queueing logic integrates clinical urgency with data-driven fairness. It ensures that critical cases receive immediate attention and that other patients are managed efficiently. The combination of triage level, score, time, and age creates a queue that mirrors real hospital decision-making. The system's dynamic updates and clear presentation make it both educational and operationally realistic.