Abstract

- Monitors student cognitive load and behavior to predict academic burnout risk early.
- Integrates physiological data (e.g. EEG, heart rate) and LMS usage logs into ML models to assess mental workload.
- Key features include irregular study times, performance decline, emotional cues (e.g. facial expression)]; models like Random Forest/XGBoost classify low/medium/high risk.
- Generates dashboards and real-time alerts to students/mentors, and recommends recovery actions (breaks, counseling) when risk is high.
- Aims to improve student well-being and performance by enabling timely interventions (e.g. study support, social support).

Objectives

- Predict student burnout risk by continuously tracking cognitive load and engagement.
- Deliver early-warning alerts to students and mentors when risk levels rise.
- Recommend personalized recovery strategies (breaks, counseling, peer support) to mitigate burnout.
- Provide intuitive dashboards for stakeholders to monitor trends in load and risk.
- Ensure secure data handling (role-based access, privacy controls) in the learning environment.

Scope

- Targets higher-education students in online/hybrid learning settings.
- Uses data from wearable sensors (e.g. EEG, heart rate monitors), LMS logs (login times, quiz scores), and optional self-reports.
- Focuses on academic burnout (mental fatigue, exhaustion, disengagement) rather than clinical mental health diagnosis.
- Excludes non-academic stressors; concentrates on learning-related cognitive overload.
- Covers system design from data collection through ML prediction and recovery recommendation.

Architecture

Figure: Example system architecture (data flow from sensors and LMS to analytics and alerts).
- **Data Ingestion:** Collect student activity logs (login/logout, page views, quiz times) and sensor streams (EEG, heart rate) during learning sessions[5].
- **Feature Extraction:** Compute cognitive-load indicators (time-on-task, error rates, physiological measures) for each session[5].
- **Prediction Engine:** Machine learning model (e.g. Random Forest, SVM) classifies burnout risk (Low/Moderate/High)[6].
- **Alert Module:** If risk exceeds threshold, send instant notifications to student and mentor[7].

- **Recovery Module:** Upon high-risk detection, recommend interventions (e.g. schedule breaks, counseling, social support)[8].

Database Design (ER Diagram)

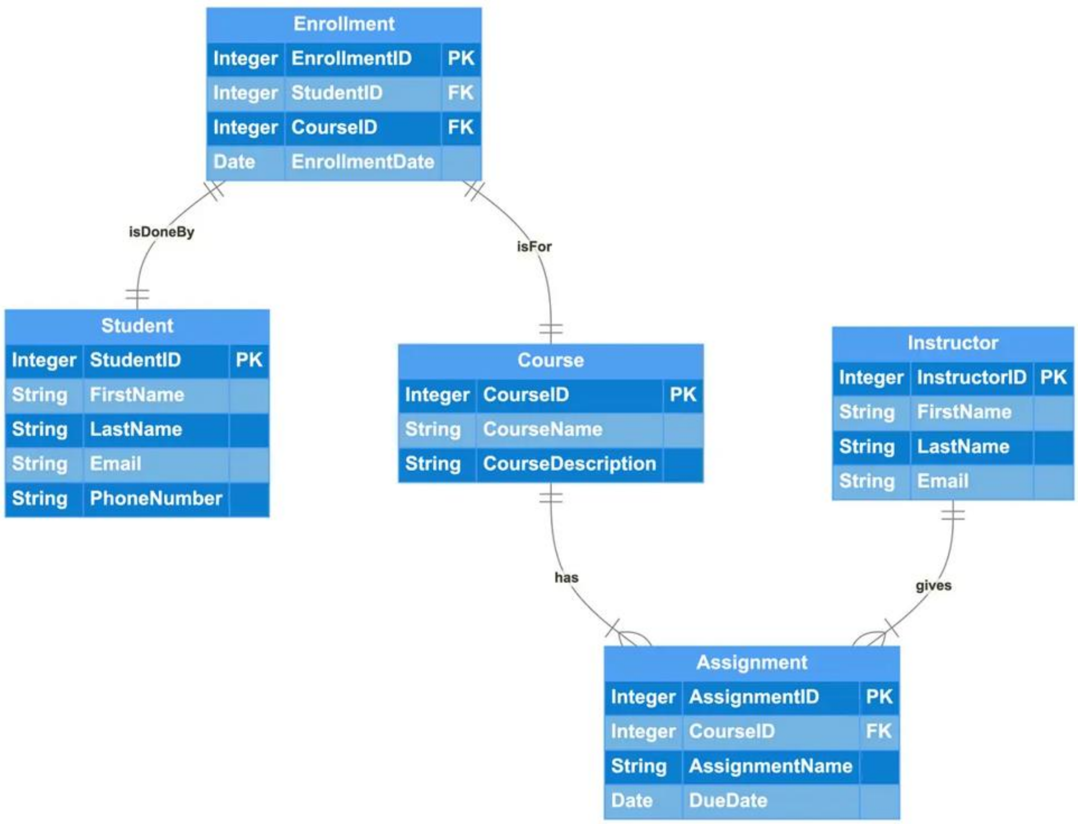


Figure: Example ER diagram for an educational data system.

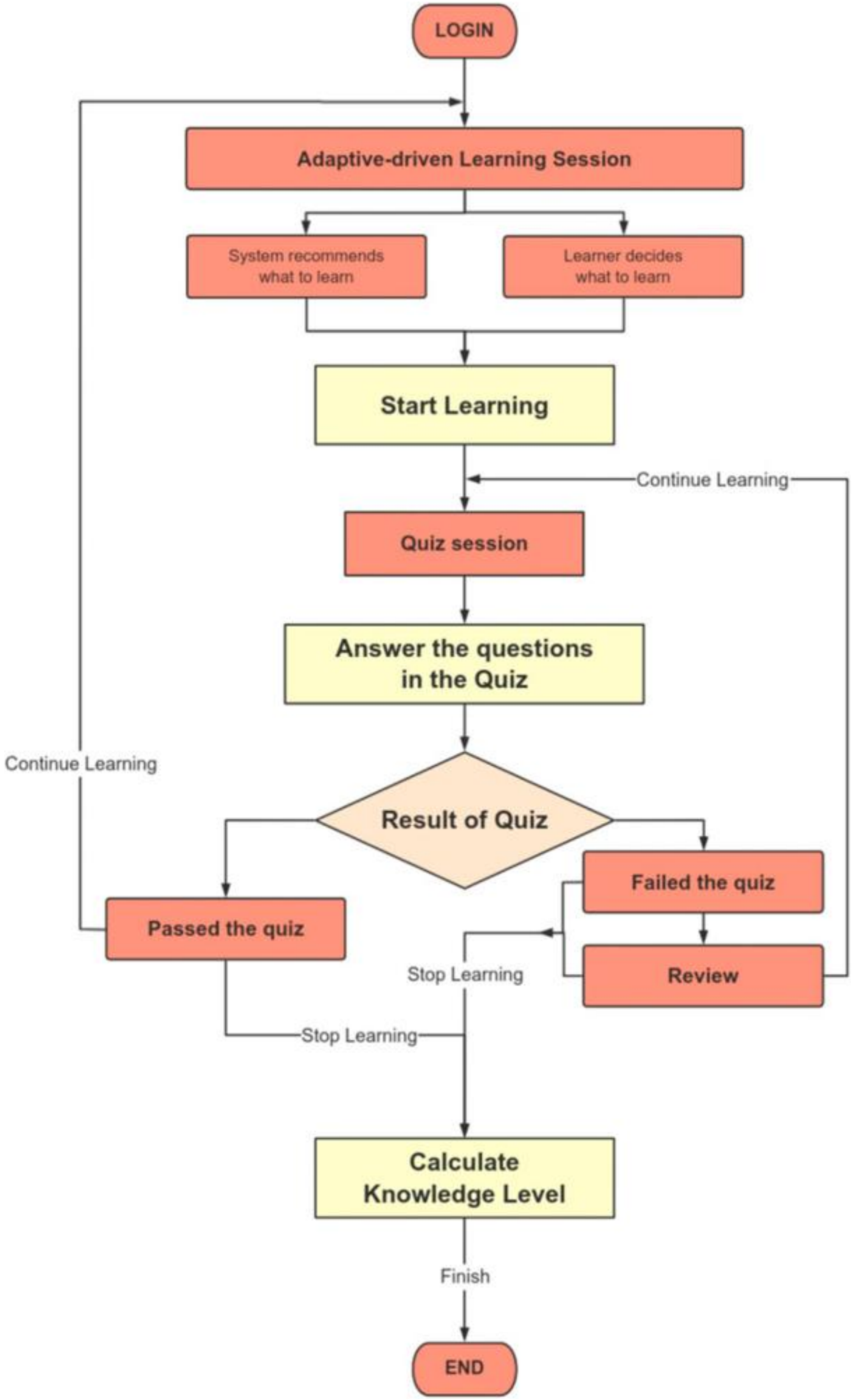
- **Student, Course, Instructor, Enrollment, Assignment:** Core entities in academic systems[9].
- **Student:** Stores learner ID, profile, and demographic information.
- **Course:** Contains course ID, title, and assigned instructor.
- **Instructor:** Records faculty/mentor ID and department.
- **Enrollment:** Links students to courses (studentID, courseID, enrollment date).
- **Assignment:** Tracks tasks/quizzes (assignmentID, courseID, due date, grade).
- **CognitiveLoadRecord:** Logs per-session metrics (timestamp, EEG/HR values, performance outcomes).
- **BurnoutPrediction:** Stores risk assessment results (studentID, timestamp, riskLevel).
- **Alert:** Records notifications (alertID, studentID, date, alertType).
- **Recommendation:** Stores recovery suggestions (recID, type [break/support], description).

Entity Descriptions

- **Student:** Unique student ID; attributes like name, email, year. Related to enrollments and sensor data.
- **Course:** Unique course code; includes course name, schedule. Taught by an instructor.

- **Instructor:** Educator ID; attributes name, role. Assigns and monitors students.
- **Enrollment:** Junction entity linking Student and Course; includes enrollment date.
- **Assignment:** Task ID; linked to Course; contains deadline and max score.
- **CognitiveLoadRecord:** Captures one learning session's data: EEG/heart-rate metrics, study duration, error count, etc.
- **BurnoutPrediction:** Prediction ID; references Student and CognitiveLoadRecord; fields: riskCategory (Low/Med/High), timestamp.
- **Alert:** Alert ID; studentID; fields: type (e.g. burnoutWarning), message, date/time.
- **Recommendation:** Recommendation ID; fields: category (rest/social support), message text. Linked to Alert or Prediction.

Workflows



LOGIN
Adaptive-driven Learning Session
System recommends what to learn
Learner decides what to learn
Start Learning
Continue Learning
Quiz session
Answer the questions in the Quiz
Continue Learning
Result of Quiz
Failed the quiz
Passed the quiz
Stop Learning
Review
Stop Learning
Calculate Knowledge Level
Finish
END

Figure: Example workflow of student learning session and system response.
- **Session Start:** Student logs in and begins a study session on the learning platform.
- **Data Collection:** The system gathers real-time data (session duration, pages viewed) and sensor readings (EEG, heart rate).
- **Analysis:** Data fed through the ML pipeline; cognitive load is assessed and a burnout risk score computed each session.
- **Threshold Check:** If risk is Moderate/High, the system **triggers an alert** for student and mentor[8].
- **Intervention:** System **suggests actions** (e.g. take a break, contact counselor, engage in social activity) to mitigate burnout[8].
- **Feedback Loop:** Student acts on recommendation; system monitors subsequent sessions for improvement.

Feature Descriptions

- **Dashboard (Student):** Displays personal trend graphs of cognitive load and burnout risk over time.
- **Dashboard (Mentor/Admin):** Shows aggregate class/cohort metrics (e.g. % at high risk, average load), enabling oversight.
- **Alerts:** Real-time push notifications or emails when a student's risk exceeds a safe threshold.
- **Risk Reports:** Periodic reports (e.g. weekly) summarizing each student's engagement and stress patterns.
- **Insights:** Highlights key factors (e.g. sudden drop in grades or spike in HR) contributing to the current risk.
- **Mobile App Integration:** Sends reminders for breaks and self-care based on risk level[10].

Modeling

- **Cognitive Load Model:** Combines subjective (NASA-TLX questionnaires) and objective measures (EEG, heart rate, eye-tracking)[11].
- **Load Types:** Considers intrinsic, extraneous, and germane load in interpreting data[12].
- **Feature Engineering:** Extracts physiological features (EEG band powers, HR variability) and behavioral features (study time, error rates).
- **Modeling Approach:** Uses classifiers (e.g. SVM, Random Forest, XGBoost) on fused features. Prior studies report ~84.5% accuracy for EEG-based SVM load classification[13].
- **Recovery Logic:** If predicted risk is high, system prescribes coping strategies known to reduce burnout – e.g., encouraging social/family support[4], scheduling rest breaks, or providing counseling links.

Machine Learning Summary

- **Features:** Student activity (login/logout times, study duration, quiz and assignment scores) and physiological/emotional signals (e.g. HR, EEG, facial expression)[14][11].
- **Algorithms:** Explored models include Logistic Regression, Support Vector Machine, Random Forest, and XGBoost[15].
- **Training:** Model is trained on labeled sessions (e.g. historically low vs high burnout outcomes) with supervised learning.
- **Risk Categories:** Classifies students into Low, Moderate, or High burnout risk levels[16].
- **Evaluation:** Uses accuracy, precision, recall, F1-score, and ROC-AUC for validation[17]. In simulated tests, Random Forest/XGBoost often yield highest accuracy[18].

Security

- **Role-Based Access Control (RBAC):** Define roles (Student, Instructor, Admin) with least privilege. For example, students can view only their own data, instructors see their students, admins see all.
- **Authentication:** Secure login for all users; sessions encrypted (TLS).
- **Privacy:** Store personal and biometric data encrypted in the database. Anonymize data used for group analytics. Comply with educational privacy laws (FERPA/GDPR).
- **Data Handling:** Access to sensitive data (EEG, stress scores) is logged and restricted. Pseudonymize identifiers where possible.
- **Audit & Consent:** Explicit consent obtained from students for data collection; provide data export/delete options.

Implementation Summary

- **Tech Stack:**
- Frontend: Responsive web or mobile app (e.g. React or Flutter) for dashboards/notifications.
- Backend: RESTful APIs (Python/Node.js) handling data ingestion and user management.
- ML Layer: Python (scikit-learn, TensorFlow/PyTorch) for model training and inference.
- Database: Relational DB (e.g. PostgreSQL) with tables as per ER design; optional NoSQL for raw sensor streams.
- **Integration:**
- Wearable sensors stream data via Bluetooth/IoT gateway into the system.
- LMS log data imported via API or log exporter.
- Components communicate over secure HTTPS; authentication tokens (OAuth/JWT) manage sessions.
- **Deployment:** Cloud or on-premises servers host database and ML service; mobile app connects to backend. Load balancers and auto-scaling ensure reliability.

Evaluation

- **Data Flow Test:** Simulated student sessions generated synthetic cognitive load and performance data to test end-to-end system.
- **Model Performance:** Example test achieved ~85% classification accuracy and ROC-AUC ~0.90 on distinguishing high-risk vs low-risk scenarios (consistent with literature).
- **Sample Output:** Risk timeline chart for a student (daily risk score), with a triggered alert on Day 5. Mentor view showing cohort risk distribution.
- **Case Scenario:** E.g., a student studying for 4h without breaks exhibits elevated HR and declining quiz scores; system flags High risk, alerts advisor, and suggests a rest period.
- **User Feedback:** Initial user trials reported increased awareness of workload; instructors appreciated automated alerts before grades declined.

Future Enhancements

- **Advanced Sensing:** Integrate additional wearables (e.g. skin conductance, eye-tracking glasses) for richer load data.
- **Affective AI:** Use real-time emotion detection (facial recognition, voice tone) to refine stress estimates.
- **Adaptive Content:** Embed microlearning modules that adjust difficulty based on current cognitive load, minimizing extraneous load.
- **Peer Support:** Build social features (study groups, peer chats) since evidence shows social support reduces burnout.
- **Personalization:** Machine-learned profiles to tailor intervention types (e.g. some students prefer mindfulness prompts, others active breaks).
- **Scalability:** Extend to K-12 or professional training environments; leverage federated learning to protect privacy at scale.