

**CAPSTONE PROJECT REGISTER**

**Class**: **Duration time**: from ..…….../20…. To ..….…./20…..

**(\*) Profession:** <Software Engineer> **Specialty**: <ES> Hộp Văn bản <IS> Hộp Văn bản <JS> Hộp Văn bản

**(\*) Kinds of person make registers:**  Lecturer Hộp Văn bản Students

**1. Register information for supervisor (if have)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Fullname | Phone | **E-Mail** | **Title** |
| Supervisor | Lại Đức Hùng | 0976710580 | Hungld5@fe.edu.vn | Mr. |
| Supervisor |  |  |  | Mr. |

**2. Register information for students (if have)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Full name** | **Student code** | **Phone** | **E-mail** | **Role in Group** |
|  |  |  |  |  | Leader |
|  |  |  |  |  | Member |
|  |  |  |  |  | Member |
|  |  |  |  |  | Member |

**3. Register content of Capstone Project**

**(\*) 3.1. Capstone Project name:**

* English: DevEval AI - Machine Learning-Powered Developer Skills Assessment Platform
* Vietnamese: Nền tảng Đánh giá Kỹ năng Lập trình viên áp dụng trí tuệ nhân tạo
* Abbreviation: DEAI

1. **Context:**

The tech industry and educational institutions face significant challenges in accurately assessing developer skills during hiring processes and academic evaluations:

1. Traditional assessment methods are often subjective and inconsistent
2. Manual evaluation of programming skills is time-consuming and resource-intensive
3. Current platforms provide limited insights into skill gaps and learning needs
4. Lack of personalized learning paths based on assessment results

Therefore, there is a demand for an intelligent platform that can automatically assess developer skills through multiple-choice questions, provide detailed insights, and offer personalized improvement recommendations using machine learning.

1. **Proposed Solutions**

* Develop an AI-powered assessment platform using adaptive testing algorithms
* Implement ML models for question difficulty calibration and skill analysis
* Create comprehensive question banks with intelligent categorization
* Provide automated skill gap analysis and recommendations
* Enable tracking of learning progress over time
* **Functional requirement**
* **Web Application**
* Users can register and login using email or institutional accounts
* Users can take various technical assessments with multiple-choice questions
* Users can receive immediate feedback on their performance
* Users can view detailed analysis of their skill profile
* Users can access personalized learning recommendations
* Users can track their progress through a dashboard
* Users can generate skill certificates
* **Assessment Engine**
* Support multiple technical domains (Programming, Database, DevOps, etc.)
* ML-based adaptive question selection
* Real-time performance analysis
* Pattern recognition in user responses
* Automated difficulty calibration
* Learning path generation
* **Admin Web System**
* Admin can login to the system
* Admin can manage user accounts and roles
* Admin can manage question banks and categories
* Admin can view system analytics and usage statistics
* Admin can monitor ML model performance
* Admin can customize assessment parameters
* Admin can generate performance reports
* **Institution Portal:**
* Institutions can create custom assessments
* Institutions can view candidate performance analytics
* Institutions can manage question banks
* Institutions can generate comprehensive reports
* Institutions can track learning outcomes
* **Non-functional requirement:**
* Response time for question loading should be under 2 seconds
* System should handle concurrent assessments efficiently
* High availability and scalable architecture
* Secure assessment environment
* Data privacy compliance

(\*) 3.2. Main proposal content (including result and product)

1. **Theory and practice (document):**

* Students should apply the software development process and UML 2.0 in the modelling system.
* The documents include User Requirement, Software Requirement Specification, Architecture Design, Detail Design, System Implementation, and Testing Document, Installation Guide, sources code, and deployable software packages.
* Server-side technologies:
* Server: Node.js with Express.js framework
* Database: MongoDB for flexible data storage
  + - * Store question banks with rich metadata
      * User profiles and progress tracking
      * Assessment results and analytics
* Authentication: JWT (JSON Web Tokens)
* API: RESTful API architecture
* ML Integration:
  + - TensorFlow.js for client-side predictions
    - Python microservice for complex ML tasks
* Model serving via REST API
* Client-side technologies:
* Web Client: HTML5, CSS3, Javascript, ReactJS
* State Management: Redux
* UI Components: Material-UI/Tailwind
* Chart Libraries: Chart.js/D3.js

1. **Products:**

* Interactive Web Application for skill assessment
* Admin Dashboard for system management
* REST API system for data management
* ML models for adaptive testing and analysis
* Complete documentation package.

1. **Proposed Tasks:**

* Task package 1: Core Platform Development (Web UI, Assessment Engine)
* Task package 2: ML Model Development (Question Analysis, Adaptive Testing)
* Task package 3: Analytics & Recommendations Engine
* Task package 4: Integration & Testing
* Task package 5: Documentation (System Analysis, Design, User Manuals).

4. Other comments (propose all relative things if have).

|  |  |
| --- | --- |
| **Supervisor (If have)**  *(Sign and full name)* | HCM, date 07/11 /2024  **On behalf of Registers**  *(Sign and full name)* |