

**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 | Phạm Thanh Trí | 0937359396 | Tript9@fe.edu.vn | Mr. |
|  |  |  |  |  |

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

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Full name** | **Student code** | **Phone** | **E-mail** | **Role in Group** |
| 1 | Lưu Ka Ka | SE171652 | 0938659975 | kalkse171652@fpt.edu.vn | Leader |
| 2 | Dương Viết Hoàng | SE173046 | 0919893174 | hoangdvse173046@fpt.edu.vn | Member |
| 3 | Phan Kang Min | DE170353 | 0342555702 | minpkde170353@fpt.edu.vn | Member |
| 4 | Nguyễn Lê Tiến Phát | SE183630 | 0388918553 | phatnltse183630@fpt.edu.vn | Member |

**3. Register content of Capstone Project**

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

* English: AI Enhance – Online Data Gathering And Assignment Report Generator for Business Students
* Vietnamese: AI Enhance – Hệ thống thu thập và hỗ trợ phân tích dữ liệu trực tuyến cho sinh viên ngành digital marketing
* Abbreviation: AE

1. **Context:**

The of market‑related data—stock prices, cryptocurrencies, commodities, exchange rates, business news, and social‑media signals—has grown rapidly and is scattered across WebSocket feeds, public APIs, RSS channels, and websites.

Marketing students must manually collect and clean this information, a labor‑intensive process that often yields unstable datasets and unreliable analytical results.

AE proposes an open, student‑friendly platform that:

1. Automatically crawls multiple online sources and synchronizes data in real time.
2. Generates AI‑driven reports (PDF / Markdown) ready for assignments and theses.
3. Lets users extend capabilities through paid packages (higher link quotas, data access APIs).
4. **Proposed Solutions**

* Web App Crawler + AI Report Generator
  + Students receive 4 free crawl attempts to validate the service.
  + Paid packages expand link quotas and unlock premium APIs.
* Lecturer Management: upload Excel student rosters; the system auto‑creates accounts and emails credentials.
* Admin Dashboard: monitors usage, revenue, and “unsupported link” errors.
* Integrated Wallet & Payments with detailed transaction history.
* Reports exportable as PDF or .md files with auto‑generated charts.
* **Functional requirement**
* **Web App – Student**
* Register (activation link sent by Staff).
* Join a class (via class code or Lecturer invitation).
* Enter up to 4 URLs, crawl them, and download AI‑generated reports.
* Join a group within the class (or be assigned by the Lecturer).
* Submit reports directly to the class / group workspace for Lecturer review.
* View grades awarded by the Lecturer.
* **Web App – Lecturer**
* Register → await Staff approval.
* Create Classes and Groups.
* Upload Excel roster; system issues student accounts automatically.
* Review and grade student reports; export grade sheets.
* **Staff System**
* Approve Student and Lecturer registrations; validate Excel files.
* Reset passwords and provide technical support.
* **Web App – User (Paid Package)**
  + - Register account.
    - Browse and purchase a package; payment is processed instantly.
    - Enter URLs according to the purchased quota, crawl data, and generate AI reports.
    - Access premium data APIs tied to the active package.
    - Review your receipt after each purchase.
* **Admin Web**
  + - Manage all accounts, classes, and service packages.
    - Real‑time dashboard: request counts, crawler errors, revenue metrics.
    - Configure the list of “unsupported” domains.
    - **Crawler Agent (system):** Manage and monitor AI/automated agents that crawl data from provided links based on requests.
    - **Web Data Extract Agent (system):** Manage and configure AI agents that process crawled data, apply filtering rules, and generate analytical reports.
* **Non-functional requirement:**
* Security: JWT, RBAC, SHA‑256 encryption for sensitive data.
* Scalability: micro‑service architecture with background queues.
* Crawl + report generation ≤ 10 s per link (< 1 MB).

(\*) 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: .NET, Python, Microservice, Apache Kafka.
* Database Design: SQL Server, Redis, MongoDB.
* Client-side technologies:
* Web Client: TypeScript, NextJS.

1. **Products:**

* Web API for System.
* Admin / Staff Web
* Web App for Students/Users/Lecturer (crawler & AI reports)

1. **Proposed Tasks:**

* Task package 1: Develop the Web application of the System for Admin.
* Task package 2: Develop the Online Data Gathering & Automated Report Generator system.
* Task package 3: Build – Deploy and Test the system.
* Task package 4: Prepare all the required documents: System analysis and Design, Test plan, Installation manual, User manual.

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

|  |  |
| --- | --- |
| **Supervisor (If have)**  *(Sign and full name)* | HCM, date …… ………. /20 …  **On behalf of Registers**  *(Sign and full name)* |