1. Problem Definition

**1.1. Name of this Capstone Project**

• **Official name**: Examination Tools applying Block Chain Technology.

•**Vietnamese name**: Xây dựng hệ thống quản lý đề thi và việc thi cử sử dụng công nghệ Block-Chain.

• **Abbreviation**: ExamTool.

**1.2. Problem Abstract**

This project is created to assist the school to fight against cheating in the examination as well as develop an exam administration to be more optimal. In the process of research and analysis, we decided to implement ExamTool by using Extension. But this way have a lot of problem because Extension can not intervene the system deeply. So we have to change the idea to using an application to kill processes of the system.

The Desktop application, which is built on Window, can deep embedding the system and kill unauthorized processes. Firstly, application will send a request to server to verify if this application is valid to access the website. After verified, application will be allowed to access resources for exam questions. Students will do the test during the given time.

Block-chain technology will be applied to the results, which make sure them will not to be lost or changed. For each bad behaviors once being discovered during test time, we are going to deduct time or stop the test immediately.

Regarding to the question management function, we find it difficult to input the question to the question bank as well as organize the examination. Therefore, a variety of template formats are going to be provided in the word file, which facilitates the lecturers to import the file. Parsing process will be resolved by the clients. In addition, we also look up the potential duplicate questions before importing them into the question bank, then notify to the lecturer for resolve.

The tests will be generated based on the Learning Outcome of a subject or manually selected by the staff. Thus, the exam will be guaranteed to contain all knowledge of a subject, then notify to the leaders for the approval. If any errors occur during the review process, the reviewer can directly edit the question right on the test. The question which has been corrected will be synchronized versus the question in the question bank.

**1.3. Project Overview**

**1.3.1 Current Situation**

*Below are the problems encountered in this project:*

* **New techniques:** The team is new to block-chain techniques.
* **Lack of knowledge** about the window system.
* **Lack of knowledge** about examination system.

**1.3.2 The Proposed System**

**1.3.3. Boundaries of the system**

*Our system supports:*

* **Import question** file with many question types.
* **Matching questions** and **show all questions** can be duplicated.
* **Synchronized question** in question bank after edited in exam.
* **Random questions** in exam **base on L.O**.
* **Killing process** real-time.
* ExamTool only support for Window 10.

*Our system hasn’t supported:*

* ExamTool provide for others O.S like Linux or Mac
* Exam Management System do not support Internet Explorer and Microsoft Edge.

**1.3.4. Future Plans**

In future, we will support:

- Create an Exam by import file “List of students can take exam”.

- Support import question with different types as Matching, Fill Blank, Indicate Mistake, Reading from GIFT, XML files.

**1.3.5. Development Environment**

**1.3.5.1. Hardware requirements**

|  |  |  |
| --- | --- | --- |
| **Hardware** | **Minimum Requirements** | **Recommended** |
| Internet Connection | Cable | Cable |
| Operating System | Window Server 2008 | Window Server 2016 |
| Computer Processor | Intel® Xeon ® 3.0GHz | Intel® Xeon ® Processors |
| Computer Memory | 4GB RAM | 8GB RAM or more |

**1.3.5.2. Software requirements**

|  |  |  |
| --- | --- | --- |
| **Software** | **Name / Version** | **Description** |
| Environment |  |  |
| Modeling tool | Star UML |  |
| IDE | Visual Studio Enterprise 2017  Visual Studio Code 1.23.1  Webstorm 2018.2 | Programming tools |
| DBMS | SQL Server 2017 | Used to create & manage the database for system |
| Source control | SVN 1.10.1 | Used for source control |

1. Coding Convention

**JavaScript:**

*Naming Conventions:*

* Variable and function names written as **camelCase**.
* Global variables written in **UPPERCASE**.
* Constants written in **UPPERCASE**.

*Function Conventions:*

* Put the opening bracket at the end of the first line.
* Use one space before the opening bracket.
* Put the closing bracket on a new line, without leading spaces.
* Do not end a complex statement with a semicolon.

*Others***:**

* Always put spaces around operators ( = + - \* / ), and after commas.
* Always use 4 spaces for indentation of code blocks.
* Always end a simple statement with a semicolon..
* Always end an object definition with a semicolon.

**C#:**

*Naming Conventions:*

* Use **PascalCase** for public property, method and type name.
* For parameters and local variables, use **camelCase**.
* For private fields, use prefix \_**camelCase** with an **\_**.
* Named constants with **ALLCAPS**.
* Vertically align curly brackets.
* Prefix interfaces with the letter **I**. Interface names are noun (phrases) or adjectives.

**Reference:**

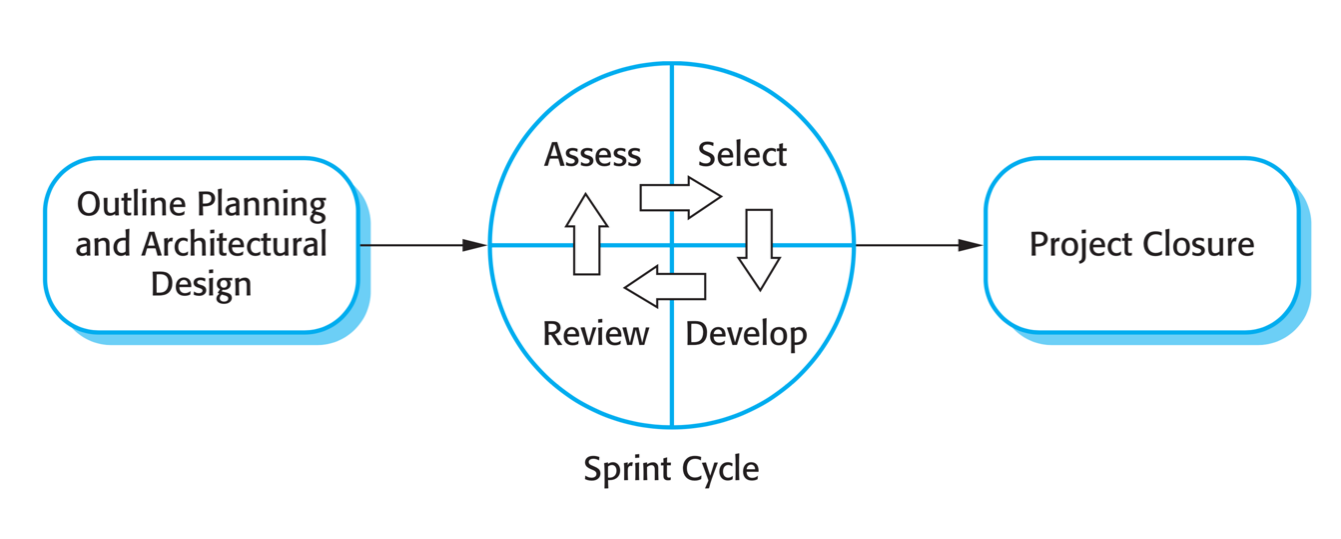
* [C# Style Guide and Coding Convention](https://docs.microsoft.com/en-us/dotnet/csharp/programming-guide/inside-a-program/coding-conventions)
* [JavaScripts Style Guide and Coding Convention](https://www.w3schools.com/js/js_conventions.asp)

# 2. Project Organization

2.1. Software Process Model

This project is developed using the Scrum model – part of an agile framework for Software development project. Our team chooses the Scrum model because of the following reasons:

* Our team only has 3 members, and tasks are assigned vertically, do all steps from design, coding, testing, and implementation. Scrum is the most suitable model for the small and medium project.
* In the project, there are many new technologies that need to be learned. With the Scrum model, the team can learn and develop in parallel to meet the deadline.
* The product owner can change the requirement or extend scope. The team will adapt to change better.

Figure 1: The Scrum Process

*Reference: Software Engineering 9th by Somerville, page 73*

**2.2. Roles and Responsibilities**

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Full name** | **Role in group** | **Responsibilities** |
| 1 | Kiều Trọng Khánh | Project Owner | * Specify scope and user requirement * Give out technique and business analysis support * Control the development process |
| 2 | Cao Trung Hiếu | Scrum master | * Create Sprint Backlog and Product Backlog * Make sure the Scrum teams understand and follow the process. * Help the team master scrum artifacts such as Sprint Backlog, Product Backlog, ... * Writing report * Always be present to answer questions and give advice when product owner or scrum member needs. |
| 3 | Cao Trung Hiếu  Nguyễn Minh Hưng  Trương Tấn Sang | Scrum team members | * Clarifying requirements * Prepare documents * Designing database * GUI Design * Coding * Testing |

Table 6: Roles and Responsibilities Details

**2.3. Tools and Techniques**

|  |  |
| --- | --- |
| Tool/Technique | Name and version |
| Front-end | Angular 2+, Jquery, boostrap |
| Back-end | .Net framework 4.7.2 |
| IDE | Visual Studio Enterprise 2017, Visual Studio Code 1.23.1, Webstorm 2018.2 |
| Database | SQL Server 2017 |
| Modeling Tool | Star UML |
| Version Control | Tortoise SVN 1.10.1 |
| Task Management | Trello |

Table 7: Tools and Techniques

**3. Project Management Plan**

**3.1. Product Backlog**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sprint** | **Story ID** | **Story** | **Task ID** | **Task** |
| 1 | 1 | Introduction document | 1.1 | Project Information |
|  |  |  | 1.2 | Introduction |
|  |  |  | 1.3 | Current Situation |
|  |  |  | 1.4 | Problem Definition |
|  |  |  | 1.5 | Proposed Solution |
|  |  |  | 1.6 | Role and Responsibility |
|  |  |  | 1.7 | Functional Requirements |
|  | 2 | Product Backlog | 2.1 | Create Product Backlog |
|  | 3 | Project management plan | 3.1 | Problem Definition |
|  |  |  | 3.2 | Project Organization |
|  |  |  | 3.3 | Project management plan |
|  |  |  | 3.4 | Coding Convention |

**3.2. Sprint Backlog**

**3.2.1. Sprint 1 (10.09.2018 – 23.09.2018): Project initiation**

**3.2.1.1. Goal**

1.1 Project Information

1.2 Introduction

1.3 Current Situation

1.4 Problem Definition

1.5 Proposed Solution

1.6 Role and Responsibility

1.7 Functional Requirements

2.1 Create Product Backlog

3.1 Problem Definition

3.2 Project Organization

3.3 Project management plan

3.4 Coding Convention

**3.2.1.2. Development**

|  |  |  |
| --- | --- | --- |
| **Task ID** | **Task** | **Responsible** |
| 1.1 | Project Information | HungNM |
| 1.2 | Introduction | HungNM |
| 1.3 | Current Situation | HungNM |
| 1.4 | Problem Definition | HieuCT |
| 1.5 | Proposed Solution | HieuCT |
| 1.6 | Role and Responsibility | SangTT |
| 1.7 | Functional Requirements | SangTT |
| 2.1 | Create Product Backlog | HieuCT |
| 3.1 | Problem Definition | HungNM |
| 3.2 | Project Organization | HieuCT |
| 3.3 | Project management plan | HieuCT, SangTT |
| 3.4 | Coding Convention | SangTT |

**3.2.2. Sprint 2 (24.09.2018 – 8.10.2018): Software Document**

**3.2.2.1. Goal**

1.1 User Requirement Specification

1.2 Software Requirement Specification

1.3 Software System attribute

1.4 Conceptual diagram

1.5 Design Overview

1.6 System Architectural Design

1.7 Component Diagram

2.1 Detailed Description

3.1 Interface

3.2 Interface Design

3.3 Algorithms

**3.2.2.2. Development**

|  |  |  |
| --- | --- | --- |
| **Task ID** | **Task** | **Responsible** |
| 1.1 | User Requirement Specification | HungNM |
| 1.2 | Software Requirement Specification | HungNM |
| 1.3 | Software System attribute | HungNM |
| 1.4 | Conceptual diagram | HieuCT |
| 1.5 | Design Overview | HieuCT |
| 1.6 | System Architectural Design | SangTT |
| 1.7 | Component Diagram | SangTT |
| 2.1 | Detailed Description | HieuCT |
| 3.1 | Interface | HungNM |
| 3.2 | Interface Design | HieuCT, HungNM |
| 3.3 | Algorithms | HieuCT, SangTT |

**3.2.3. Sprint 3 (9.10.2018 – 23.10.2018): Basic feature and core**

**3.2.3.1. Goal**

1.1 System Problem definition

1.2 Implement Algorithms

1.3 Import question from Moodle

1.4 Manage Exam

1.5 Manage Course

1.6 Manage Question

**3.2.3.2. Development**

|  |  |  |
| --- | --- | --- |
| **Task ID** | **Task** | **Responsible** |
| 1.1 | System Problem definition | HieuCT |
| 1.2 | Implement Algorithms | HieuCT |
| 1.3 | Import question from Moodle | HungNM |
| 1.4 | Manage Exam | HungNM, SangTT |
| 1.5 | Manage Question | HungNM, SangTT |

**3.2.4 Sprint 4 (24.10.2018 – 8.11.2018): Software Document**

**3.2.4.1. Goal**

1.1 Teacher Import and Manage Question

1.2 Manage LO

1.3 Manage Chapter

1.4 Staff generate test exam

1.5 Student App Implement

1.6 Realtime kill Process Implement

1.7 Leader Approve Test Exam

2.1 Student Take Exam

3.1 Review Student Test Exam

3.2 Blockchain Implement

**3.2.4.2. Development**

|  |  |  |
| --- | --- | --- |
| **Task ID** | **Task** | **Responsible** |
| 1.1 | Teacher Import and Manage Question | HungNM |
| 1.2 | Manage LO | SangTT, HungNM |
| 1.3 | Manage Chapter | SangTT, HungNM |
| 1.4 | Staff generate test exam | SangTT, HungNM |
| 1.5 | Student App Implement | HieuCT |
| 1.6 | Realtime kill Process Implement | HieuCT |
| 1.7 | Leader Approve Test Exam | SangTT, HungNM |
| 2.1 | Student Take Exam | HieuCT |
| 3.1 | Review Student Test Exam | HieuCT |
| 3.2 | Blockchain Implement | SangTT |

**3.2.5 Sprint 5 (9.11.2018 – 23.11.2018): Software Document**

**3.2.5.1. Goal**

1.1 IT Get Student Screen

1.2 Login with FPT API

1.3 Manager Approve Exam

1.4 Run test With FPT Student

1.5 Report bug

1.6 Fix bug

1.7 Test document

**3.2.5.2. Development**

|  |  |  |
| --- | --- | --- |
| **Task ID** | **Task** | **Responsible** |
| 1.1 | IT Get Student Screen | HieuCT |
| 1.2 | Login with FPT API | SangTT, HungNM |
| 1.3 | Manager Approve Exam | SangTT, HungNM |
| 1.4 | Run test With FPT Student | SangTT, HungNM |
| 1.5 | Report bug | HieuCT |
| 1.6 | Fix bug | HieuCT |
| 1.7 | Test document | SangTT, HungNM |

**3.2.6 Sprint 6 (24.11.2018 – 13.12.2018): Complete Document**

**3.2.6.1. Goal**

1.1 Installation Guide

1.2 User Manual

1.3 Paper Document

**3.2.6.2. Development**

|  |  |  |
| --- | --- | --- |
| **Task ID** | **Task** | **Responsible** |
| 1.1 | Installation Guide | HieuCT |
| 1.2 | User Manual | SangTT, HungNM |
| 1.3 | Paper Document | SangTT, HungNM |