***The World Islamic Sciences and Education University***

جامعة العلوم الاسلامية العالمية

Faculty of Information Technology

كلية تكنولوجيا المعلومات

****

GRADUATION PROJECT

**Title**

*ThinkTankIT*

**Students**

Bara Akram Ibrahim Jaber (3220601009) CS

Mohammad Ismael Yousef Hussein (3210601036) CS

Esraa Jamel Qaddoum (3200606016) SCIN

**Supervisor**

Dr. Firas Al-Zoubi

SEMESTER I

2024/2025

# Acknowledgements

To the esteemed The World Islamic Sciences and Education University,

The profound sense of accomplishment achieved on this project is truly a testament to the steadfast collaboration between a remarkable group of individuals. Our journey thus far has been marked by invaluable guidance and encouragement from those who have tirelessly dedicated themselves to the success of the project.

In expressing our deep gratitude, we thank God Almighty, who inspires us and guides us to the right path crowned with success. We also express our appreciation and special thanks to our supervisor, Dr. Firas Al-Zoubi, whose wisdom, inspiration, and constructive comments were pivotal in shaping this project and whose guidance is a beacon that illuminates our path toward excellence.

Furthermore, we extend our sincere thanks to our parents and colleagues, whose unwavering support has been a constant source of motivation. Their contributions were effective in achieving the achievement we have reached today.

May God bless you all and thank you.

# Abstract

The development of an ThinkTankIT at WISE University serves as an essential component of the student information system, designed to improve the efficiency, transparency, and accessibility of the team formation process. The platform simplifies the traditional, manual process of forming graduation project teams by automating key tasks such as profile creation, team management, and supervision approvals. This ensures accurate and timely processing of team-related requests while reducing administrative workload for both students and supervisors.

Students are required entry to personalized profiles so that showcasing their skills, links to \*\*GitHub\*\* and \*\*LinkedIn\*\*, as well as a brief bio describing their experience and achievements. team leaders can invite suitable students to join their teams. Team membership changes, such as leaving or removing members, are only permitted with supervisor approval, ensuring academic alignment and integrity.

Key features of the platform include secure login for authentication, a dashboard for supervisors to monitor and manage teams, real-time notifications for team-related updates, and tools for defining project focus areas and technologies.

The system is implemented using Node.js and Express.js for the back-end and PostgreSQL for data storage, while the front-end is developed using HTML, JavaScript, CSS. By transitioning to this automated system, students can easily connect, collaborate, and manage their projects, while supervisors can ensure quality and consistency in team formation.

we employed the Scrum methodology, a flexible and efficient framework designed to deliver value throughout the project's development phase. It provides an intuitive interface for students to log in, and interact with the system, making it a practical solution to a real-world problem faced by the university's student community.

In summary, the ThinkTankIT aims to enhance student satisfaction, streamline team management, and foster collaboration, offering an efficient and accessible solution to meet the needs of graduating students and supervisors alike.

**الملخص**

يعمل تطوير ThinkTankIT في جامعة WISE كعنصر أساسي في نظام معلومات الطلاب، المصمم لتحسين الكفاءة والشفافية وإمكانية الوصول إلى عملية تشكيل الفريق. تبسط المنصة العملية اليدوية التقليدية لتشكيل فرق مشروع التخرج من خلال أتمتة المهام الرئيسية مثل إنشاء الملف الشخصي وإدارة الفريق وموافقات الإشراف. يضمن ذلك معالجة دقيقة وفي الوقت المناسب للطلبات المتعلقة بالفريق مع تقليل عبء العمل الإداري لكل من الطلاب والمشرفين .

يطلب من الطلاب الدخول إلى ملفات تعريف مخصصة لعرض مهاراتهم وروابطهم إلى GitHub وLinkedIn، بالإضافة إلى سيرة ذاتية موجزة تصف خبراتهم وإنجازاتهم. يمكن لقادة الفريق دعوة الطلاب المناسبين للانضمام إلى فرقهم. لا يسمح بتغييرات عضوية الفريق، مثل ترك الأعضاء أو إزالتهم، إلا بموافقة المشرف، مما يضمن المواءمة الأكاديمية والنزاهة.

تشمل الميزات الرئيسية للمنصة تسجيل الدخول الآمن للمصادقة، ولوحة معلومات للمشرفين لمراقبة الفرق وإدارتها، وإشعارات في الوقت الفعلي للتحديثات المتعلقة بالفريق، وأدوات لتحديد مجالات وتقنيات تركيز المشروع .

يتم تنفيذ النظام باستخدام Node.js و Express.js للواجهة الخلفية وPostgreSQLلتخزين البيانات، في حين تم تطوير الواجهة الأمامية باستخدام HTML وJavaScript وCSS. من خلال الانتقال إلى هذا النظام الآلي، يمكن للطلاب بسهولة الاتصال والتعاون وإدارة مشاريعهم، في حين يمكن للمشرفين ضمان الجودة والاتساق في تشكيل الفريق.

استخدمنا منهجية Scrum ، وهو إطار مرن وفعال مصمم لتقديم قيمة طوال مرحلة تطوير المشروع. يوفر واجهة بديهية للطلاب لتسجيل الدخول والتفاعل مع النظام، مما يجعله حلا عمليا لمشكلة في العالم الحقيقي التي يواجهها مجتمع طلاب الجامعة.

باختصار، يهدف ThinkTankIT إلى تعزيز رضا الطلاب، وتبسيط إدارة الفريق، وتعزيز التعاون، وتقديم حل فعال ويمكن الوصول إليه لتلبية احتياجات الطلاب المتخرجين والمشرفين على حد سواء .

List of abbreviations

|  |  |
| --- | --- |
| TTIT | ThinkTankIT |
| WISE | The world Islamic Science & Education University |
| IT | Information Technology |
| HTML | Hypertext Markup Language |
| CSS | Cascading Style Sheets |
| JS | Java Script |
| SQL | Structured Query Language |
| UI | User Interface |
| ER | Entity Relationship |
| CS | Computer Science |
| SCIN | Security and confidentiality of information and networks |

Contents

[Acknowledgements 1](#_Toc188204362)

[Abstract 2](#_Toc188204363)

[CHAPTER 1 8](#_Toc188204364)

[INTRODUCTION 8](#_Toc188204365)

[1.1Overview 9](#_Toc188204366)

[1.2 Problem Statement 9](#_Toc188204367)

[1.3 Project objectives 10](#_Toc188204368)

[1.4 Research strategy (Framework) 11](#_Toc188204369)

[1.5 Scope 12](#_Toc188204370)

[1.6 Gantt chart 13](#_Toc188204371)

[7.1 Project outline 13](#_Toc188204372)

[CHAPTER 2 14](#_Toc188204373)

[LITERATURE REVIEW 14](#_Toc188204374)

[2.1 Overview 15](#_Toc188204375)

[2.2 Comparative Study 15](#_Toc188204376)

[2.2.1 Related Work 15](#_Toc188204377)

[2.3 summary 17](#_Toc188204378)

[CHAPTER 3 19](#_Toc188204379)

[METHODOLOGY 19](#_Toc188204380)

[3.2 Feasibility study 20](#_Toc188204381)

[3.2.1 Technical feasibility 20](#_Toc188204382)

[3.2.2 Operational feasibility 21](#_Toc188204383)

[3.3 Requirements 21](#_Toc188204384)

[3.3.1 Types of requirements: 22](#_Toc188204385)

[3.3.1.2 Non-functional requirement 23](#_Toc188204386)

[3.4 Tools 24](#_Toc188204387)

[3.5 Methodology process 25](#_Toc188204388)

[CHAPTER 4 29](#_Toc188204389)

[DESIGN MODELS 29](#_Toc188204390)

[4.2 Context diagram-0 30](#_Toc188204391)

[30](#_Toc188204392)

[4.3 Data flow Diagram-1 31](#_Toc188204393)

[4.4 Use Case Diagram 32](#_Toc188204394)

[4.5 ER Diagram 33](#_Toc188204395)

[CHAPTER 5 34](#_Toc188204396)

[EXPERIMENTS AND RESULTS 34](#_Toc188204397)

[5.1 Overview 35](#_Toc188204398)

[5.2 Test Design 35](#_Toc188204399)

[5.4 Summery 55](#_Toc188204400)

[CHAPTER 6 56](#_Toc188204401)

[CONCLUSION AND FUTURE WORKS 56](#_Toc188204402)

[6.1 Overview 57](#_Toc188204403)

[6.2 Summary of the project 57](#_Toc188204404)

[6.3 Achieved objectives 58](#_Toc188204405)

[Objective 01: 58](#_Toc188204406)

[Objective 02: 58](#_Toc188204407)

[6.4 Limitation 59](#_Toc188204408)

[6.5 Future Work 59](#_Toc188204409)

[REFERENCES 60](#_Toc188204410)

[Appendices 61](#_Toc188204411)

[61](#_Toc188204412)

List of Tables

[Table 1 comparison of features for related works 18](#_Toc188227924)

[Table 2 Operational Feasibility 21](#_Toc188227925)

[Table 3 Non-functional Requirements 24](#_Toc188227926)

[Table 4 Scrum Sprints 26](#_Toc188227927)

List of Figures

[Figure 1 Project Sprints 12](#_Toc188227933)

[Figure 2 Gantt chart 13](#_Toc188227934)

[Figure 3 Sprints Process 25](#_Toc188227935)

[Figure 4 Context diagram-0 30](#_Toc188227936)

[Figure 5 Data flow Diagram -1 31](#_Toc188227937)

[Figure 6 Use Case Diagram 32](#_Toc188227938)

[Figure 7 ER Diagram 33](#_Toc188227939)

[Figure 8 Login Screen 1 35](#_Toc188227940)

[Figure 9 Login Screen 2 36](#_Toc188227941)

[Figure 10 Login Screen 3 36](#_Toc188227942)

[Figure 11 Login Screen 4 37](#_Toc188227943)

[Figure 12 Login Screen 4 37](#_Toc188227944)

[Figure 13 Home Screen 1 38](#_Toc188227945)

[Figure 14 Home Screen 2 38](#_Toc188227946)

[Figure 15 Home Screen 3 39](#_Toc188227947)

[Figure 16 Home Screen 5 39](#_Toc188227948)

[Figure 17 Team Screen 1 40](#_Toc188227949)

[Figure 18 Team Screen 2 40](#_Toc188227950)

[Figure 19 Home Screen 6 41](#_Toc188227951)

[Figure 20 User profile Screen 41](#_Toc188227952)

[Figure 21 Home Screen 7 42](#_Toc188227953)

[Figure 22 Create Team Screen 1 42](#_Toc188227954)

[Figure 23 Create Team Screen 2 43](#_Toc188227955)

[Figure 24 Create Team Screen 4 44](#_Toc188227956)

[Figure 25 profile Screen 1 44](#_Toc188227957)

[Figure 26 profile Screen 2 45](#_Toc188227958)

[Figure 27 profile Screen 3 45](#_Toc188227959)

[Figure 28 Team Screen 3 46](#_Toc188227960)

[Figure 29 Team Screen 4 46](#_Toc188227961)

[Figure 30 Team Screen 5 47](#_Toc188227962)

[Figure 31 Team Screen 6 47](#_Toc188227963)

[Figure 32 Team Screen 7 48](#_Toc188227964)

[Figure 33 Team Screen 8 48](#_Toc188227965)

[Figure 34 Team Screen 9 49](#_Toc188227966)

[Figure 35 Admin Screen 1 49](#_Toc188227967)

[Figure 36 Admin Screen 2 50](#_Toc188227968)

[Figure 37 Admin Screen 3 50](#_Toc188227969)

[Figure 38 Home Screen 8 51](#_Toc188227970)

[Figure 39 Home Screen 9 51](#_Toc188227971)

[Figure 40 Team Screen 10 52](#_Toc188227972)

[Figure 41 Question 1 53](#_Toc188227973)

[Figure 42 Question 2 53](#_Toc188227974)

[Figure 43 Question 3 54](#_Toc188227975)

[Figure 44 Question 4 54](#_Toc188227976)

[Figure 45 Question 5 55](#_Toc188227977)

# 

# CHAPTER 1

# INTRODUCTION

## 1.1Overview

The primary objective of this graduation project is to develop a web-based platform that addresses the challenges faced by students at **The World Islamic Sciences and Education University** in forming or joining teams for their graduation projects. Many students struggle to find suitable teammates with compatible skills, interests, and goals, which can negatively impact their ability to successfully complete their projects. This platform aims to solve this issue by creating an efficient and user-friendly system for team formation.

The website allows students to log in, create teams, or join existing ones based on their areas of expertise, project focus (e.g., web application or mobile application), and preferred programming languages. By incorporating these filters, the platform ensures that students can collaborate with peers who align with their technical skills and project requirements.

The project was developed by a team of three members using modern web technologies, including **HTML**, **CSS**, **JavaScript**, **Node.js**, **Express.js**, and **PostgreSQL**. A local host server was used for development and testing purposes. The system includes key features such as user authentication, team creation and browsing, and matching students based on their preferences and skills.

This platform not only addresses a specific need within the university but also demonstrates the practical application of web development technologies to solve real-world problems. By facilitating efficient team formation, the system aims to enhance collaboration, productivity, and overall project success for students.

## 1.2 Problem Statement

Students at **The World Islamic Sciences and Education University** face significant challenges in forming or joining teams for their graduation projects. The absence of an efficient system for team matching makes it difficult for students to find suitable teammates

who align with their skills, project goals, and technical preferences. This often results in delays, mismatched teams, and suboptimal project outcomes.

Additionally, the process of finding teammates is largely unstructured, relying on personal connections or word-of-mouth communication. This not only limits opportunities for collaboration but also leaves many students without the necessary resources to complete their projects effectively. As a result, students are unable to fully leverage their potential or work on projects that align with their expertise and interests.

To address this problem, there is a need for a structured, technology-driven platform that enables students to easily connect with peers, form teams, and collaborate based on shared goals, areas of expertise, and preferred technologies.

## 1.3 Project objectives

The primary objectives of this graduation project are as follows:

1. Develop a web-based platform that enables students at The World Islamic Sciences and Education University to easily create or join teams for their graduation projects.
2. Design an intuitive and user-friendly interface that simplifies the process of registering, logging in, browsing teams, and creating new teams
3. Enable students to have profiles that showcase their skills and some links to their accounts on GitHub and LinkedIn.
4. Enable students to find other students and form graduation project teams based on their skills.
5. Solve the specific issue of team formation within the university by Enable graduation project supervisors to track these teams and help connect students who are unable to find a team

## 1.4 Research strategy (Framework)

The strategic choice of a research methodology profoundly influences the process of data collection and analysis, thereby serving as a pivotal determinant in research endeavors. This section aims to meticulously examine diverse research strategies available, emphasizing their nuances and implications to aid in informed selection.

In the domain of software development, the Software Development Life Cycle (SDLC) stands as a foundational process governing product or service design and development. Established in the 1960s, SDLC consists of systematic stages, offering comprehensibility and structured implementation. However, its rigidity concerning alterations after initial stages poses limitations.

Conversely, Agile methodology, originating in 2001 through collaborative efforts among 17 software developers, adopts an iterative approach in project management. It boasts faster

execution, particularly beneficial for small-scale projects, and its adaptability to dynamic changes in requirements stands as a distinguishing feature.

Within the Agile framework, Scrum serves as a refined subset that fosters effective team collaboration, primarily in managing complex product development. Employing meetings, roles, and tools, Scrum optimizes team structure and workload management, beneficial not only for software development but also for diverse teams united toward a shared objective.

The selection of Scrum methodology for our project is substantiated by several compelling reasons, encompassing its multifaceted benefits:

* Enhanced Adaptability and Flexibility.
* Fostering Innovation and Creativity.
* Cost Efficiency.
* Continuous Quality Improvement.
* Organizational Alignment and Synergy.
* Heightened Employee and Customer Satisfaction.
* Risk Mitigation and Transparency.

Additionally, Scrum promotes a culture of accountability, empowers cross-functional collaboration, and facilitates an environment where team members take ownership of their tasks, contributing to heightened productivity and efficiency.

A diagram of a process

Description automatically generated with medium confidence

Figure Project Sprints

## 1.5 Scope

The TTIT website is designed for the IT department at the W.I.S.E University. The system focuses solely on team formation and does not include project management or progress tracking features. Besides, it allows students to entry their profiles where they can showcase their abilities while finding teams or projects.

Moreover, it also assists the supervisors in monitoring graduation projects’ teams conducted by these students to enhance their academic experience and ensure quality and alignment with academic standards.

## 1.6 Gantt chart



Figure Gantt chart

## 7.1 Project outline

* **Chapter 2:** compares the system with other existing systems, highlighting the unique functionalities and important features in The ***TTIT*** website.
* **Chapter 3:** discusses the project's approach, functional and non-functional needs, and the feasibility assessment.
* **Chapter 4**: explains the system's functionality using a series of graphs for clarity.
* **Chapter 5**: examines the implementation and evaluation of the outcomes.
* **Chapter 6**: summarizes the project idea and the future work that can be applied in the near future.

# CHAPTER 2

# LITERATURE REVIEW

## 2.1 Overview

This chapter provides an overview of existing platforms designed to assist IT students in forming teams for their graduation projects. These platforms aim to facilitate collaboration, enhance team management, and centralize project tracking. By offering tools that enable students to connect with peers possessing complementary skills, these systems streamline the team formation process, foster better communication, and improve task organization within academic projects.

The chapter also includes a comprehensive **Literature Review** that examines previous studies and research on the development of educational platforms tailored to university project teams. Relevant sources such as **Google Scholar**, **academic journals**, and other credible resources were consulted to explore best practices, challenges, and technological approaches used in similar systems. This review aims to establish a foundation for the creation of an effective and user-centric team-building platform, specifically designed to meet the needs of university students engaged in collaborative projects.

By analyzing related works and synthesizing findings from the literature, this chapter identifies the key functionalities and features that contribute to the success of team-building platforms. It also highlights gaps in existing solutions, providing a clear direction for the design and implementation of a tailored system that addresses the unique requirements of students at **The World Islamic Sciences and Education University**. This systematic approach ensures the proposed platform will not only meet academic objectives but also offer practical benefits to its users.

## 2.2 Comparative Study

In this segment, we present the relevant work, which is also referred to as academic services or student applications. The purpose of this section is to spotlight endeavors undertaken by others that bear a connection to our own work.

### 2.2.1 Related Work

* **Team-Maker[[1]](#footnote-1)**

Is an online tool designed to create balanced and effective teams, commonly used in academic or professional settings. It is part of the **CATME system** (Comprehensive Assessment of Team Member Effectiveness), which includes other tools for team management and performance evaluation.

**Purpose**: Helps instructors create student teams based on skills and preferences.

* **Handshake[[2]](#footnote-2)**

Is a **professional networking platform** designed to connect **students** with **job opportunities** and **employers**, helping students build a strong professional network while still in school. It acts as a **bridge between students and employers**, offering access to job and internship opportunities and allowing students to showcase their skills and achievements in a professional manner.

**Purpose:** connect students with job and internship opportunities, helping them build professional networks and apply for roles.

* **National University of Singapore (NUS)[[3]](#footnote-3)**

They have an internal system that supports the registration of graduation projects and allows the entry of basic information about the project, such as the title, objectives, and academic supervisor.

**Purpose:** This platform helps streamline the management of graduation projects, keeping everything organized for both students and supervisors.

* TTIT System

It helps students overcome one of the biggest challenges in graduation projects, which is finding suitable team members. This system is currently limited to IT students and will be the first experience for students. After that, the experience will be improved for all students from all colleges in the university later

Provides additional features that were not addressed in any of the previous platforms, although not all of these features are available at the same time or within the same platform.

**such as:**

* Forming teams: enabling students to find suitable colleagues for graduation projects based on shared skills and interests.
* Student Profiles: Enable students to have profiles that showcase their skills and some links to their accounts on GitHub and LinkedIn.
* Supervision and Management: Assist academic supervisors in monitoring and managing the team formation process to ensure quality and alignment with academic standards and also to track these teams and help connect students who are unable to find a team.

## 2.3 summary

The lack of a current means to facilitate student acquaintance and teamwork makes the process of forming graduation project teams a major challenge difficult for both students and professors. The proposed system streamlines this process by enabling students to entry to personalized profiles so that showcasing their skills, links to \*\*GitHub\*\* and \*\*LinkedIn\*\*, as well as a brief bio describing their experience and achievements. team leaders can invite suitable students to join their teams. Team membership changes, such as leaving or removing

members, are only permitted with supervisor approval, ensuring academic alignment and integrity. finality all of those features are not available in to another works at same.

Table comparison of features for related works

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Feature/Aspect | A logo with a spiral design  Description automatically generated | A green and white logo  Description automatically generated |  |  |
| Allows supervisor tracking and support | Yes | No | No | No |
| Facilitates skill-based team formation | Yes | Yes | No | No |
| Allows joining existing teams | Yes | No | No | No |
| Tailored for graduation projects | Yes | No | No | Yes |
| Supports user profiles with links | Yes | No | Yes | No |

**In addition to that, this chapter highlights some of the existing platforms that have been already implemented in other universities, it demonstrates how these systems provide enhanced solutions for the problems noted.**

# CHAPTER 3

# METHODOLOGY

**3.1 Overview**

This chapter provides a detailed exploration of the comprehensive feasibility study, the methodology employed in the project, as well as the functional and non-functional requirements.

## 3.2 Feasibility study

The primary objective of a feasibility study is to evaluate the economic viability of a proposed business or project. The outcome of the study determines whether it is advisable to proceed with the proposed venture. If the feasibility study yields positive results, the next step is to develop a comprehensive business plan. However, if the findings indicate that the project is not a viable business idea, it should not be pursued. While it may be challenging to accept unfavorable results, identifying potential issues early in the process is far more beneficial than discovering them at a later stage.

### 3.2.1 Technical feasibility

A technical feasibility study reviews the technical resources available for the project, which determines if the right equipment, enough equipment, and the right technical knowledge are provided to complete this project's objectives.

*Table 2: Real life cost for Uni*

|  |  |
| --- | --- |
| Item | Cost (JOD) |
| Dedicated server | 70–150 per month |
| Database Costs | 10 per month |
| Load Balancers | 14–35 |
| Total estimated cost | 94-195 |

### 3.2.2 Operational feasibility

An operational feasibility study evaluates whether the organization can complete this project or not. This includes staffing requirements, organizational structure, and any applicable legal requirements. At the end of the operational feasibility study, your team will have a sense of whether you have the resources, skills, and competencies to complete this work.

Table Operational Feasibility

|  |  |
| --- | --- |
| Process | Percentage |
| Readiness and Training | 85% |
| Maintenance Viability | 75% |
| Workflow Efficiency | 70% |
| Performance | 80% |
| Process Integration | 80% |

## 3.3 Requirements

**3.3.1 Type Collected**

**3.3.1.1 Interview:**

We conducted interviews with potential users as part of the requirements gathering process to develop the functionality and role of the Go Hotel app. During these interviews, I sought to understand users' needs and expectations from the app. guiding questions included focusing on features that were important to them and how the app could improve their time management and task completion. Through these interviews, I was able to obtain valuable details about how they use similar applications, and the challenges they face in managing daily tasks. They also gave their opinions on user interfaces and personal preferences regarding app design. These interviews played a crucial role in guiding development directions to better meet user expectations and ensure a satisfactory and effective user experience.

**3.3.1.2 Questionnaire**:

A questionnaire was designed as part of the requirements gathering efforts in a hotel reservation development project. This questionnaire was an effective way to obtain opinions and details from multiple points of view, and to cover them all from everywhere. In preparing the questionnaire, we have defined a variety of questions covering different aspects of users' needs and expectations. For example, the questions included inquiries about desired features, how they preferred to organize tasks, their challenges with time management, and what problems they encountered with other applications. The questionnaire was effective in collecting a large amount of information from a wide range of participants. It also helped better direct development efforts to comprehensively meet user needs and cover all user feedback.

### 3.3.1 Types of requirements:

#### 3.3.1.1 Functional

These are statements of services the system should provide, how the system should react to particular inputs. In some cases, the functional requirements may also explicitly state what the system should not do.

*Table 3: Functional*

|  |  |
| --- | --- |
| **Requirement** | **Description** |
| User Login | The system must allow users to log in using their credentials. |
| Profile Management | Users must be able to view, and edit their profiles, including skills and project preferences. |
| Team Creation | Users must be able to create teams, specifying project type, skills required, and programming languages. |
| Team Browsing | Users must be able to browse existing teams based on filters such as project type, and languages. |
| Join Requests | The system must allow students to send a request to join teams. |
| Add Requests | The system must allow team leader to send a request to add students to his team. |
| Admin Panel | Administrator must be able to monitor and manage teams, delete teams, accept/reject teams creation requests and members exit/expulsion requests. |
| Notifications | The system must notify users of important actions, such as join requests and requests to refuse to establish a team. |
| Search for a student | The system must allow students to search for a student using students name or ID. |
| Search for a team | The system must allow students to search for a team using team name. |
| Edit team info | The system must allow team leader to edit team info. |
| Edit deadline | The system must allow administrator to edit the deadline. |

### 3.3.1.2 Non-functional requirement

There are constraints on the services or functions offered by the system. They include timing constraints, constraints on the development process, and constraints imposed by standards. Non-functional requirements often apply to the system as a whole, rather than individual system features or services.

Table Non-functional Requirements

|  |  |
| --- | --- |
| **Requirement** | **Description** |
| Usability | The system must have a user-friendly interface, ensuring ease of navigation and interaction. |
| Performance | The platform must handle up to a specified number of concurrent users without performance degradation. |
| Scalability | The system must be designed to accommodate additional users and features in the future. |
| Security | User data must be securely stored. |
| Availability | The platform must maintain a high level of uptime, ensuring reliability for users. |
| Compatibility | The application must be accessible across modern web browsers and devices. |
| Maintainability | The system must be developed using modular code to facilitate future updates and bug fixes. |

## 3.4 Tools

* Office365.
* Visual studio code.
* Draw io.
* **Language**: HTML, CSS, JS, Node.js, Express.js, PostgreSQL.
* **AI**: ChatGPT, GitHub Copilot.
* **Web-Browser**: Google chrome / Brave.
* **Hardware**: laptop

## 3.5 Methodology process

Scrum is one of the most famous development methodologies. It involves several environmental and technical variables (e.g., requirements, resources) that are likely to change during the process, which affects the development process in a way that makes it unpredictable and complex, if not handled probably. This requires flexibility in the system’s development process to be able to respond to the changes.

In addition to that, Scrum helps to improve the existing engineering practices in an organization, for it involves frequent management activities aiming at consistently identifying any deficiencies or impediments in the development process as well as the practices that are used. The website ***TTIT*** has been developed on Scrum as it is currently the most used and trusted framework of reference in the software industry and because of the many advantages, it is easily scalable, flexible to changes, produces higher software quality, and helps in the reduction of risks.

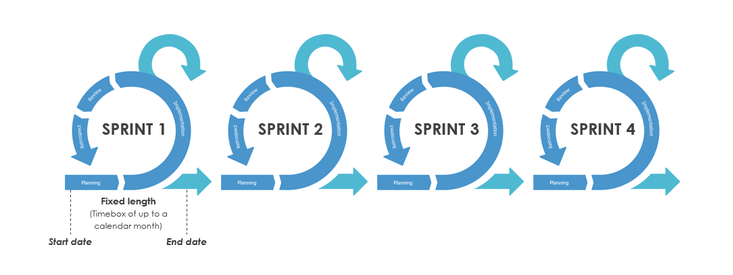


Figure Sprints Process

main roles in scrum: -

* Scrum master.
* Product owner.
* Scrum team.
* Customer
* Management

**Sprints** are iterative cycles where the functionality is developed or enhanced to produce new increments. Each Sprint includes the traditional phases of software development: requirements, analysis, design, evolution, and delivery phases. The architecture and the design of the system evolved during the Sprint development. One Sprint is planned to last from one week to one month. Also, there may be more than one team building the increment.

Table Scrum Sprints

|  |  |  |  |
| --- | --- | --- | --- |
| **Sprints** | **Objectives** | **Tasks** | **Deliverables** |
| Sprint 1  2 Weeks | 1-Identify and define the problem faced by students in forming and managing teams.  2-Understand user needs and expectations for the platform. | 1-Conduct surveys and interviews with students and faculty to identify key challenges and desired features.  2-Review existing platforms and best practices for team  formation and collaboration.  3-Define functional and non-functional requirements for the platform.  4-Draft initial syste diagrams, including use case and data flow diagrams (DFDs). | 1-Problem statement and list of platform requirements.  2-Initial system diagrams (use case, DFDs, and ERD). |
| Sprint 2  5 Weeks | 1-Design the overall architecture and structure of the platform.  2-Select the most appropriate technologies for the project. | 1-Create detailed Entity-Relationship Diagram (ERD) to define the database structure.  2-Design the user interface (UI) and user experience (UX) mockups for key features like registration, login, and team management.  3-Choose the technology stack:  **a-Frontend**: HTML, CSS, JavaScript.  **b-Backend**: Node.js and Express.js.  **c-Database**: PostgreSQL.  4-Set up the development environment, including a localhost server for testing. | 1-Finalized system design (ERD, UI/UX mockups).  2-Configured development environment. |
| Sprint 3  6 Weeks | 1-Build the core functionalities of the platform.  2-Ensure the platform works as intended through iterative testing and refinement. | 1- Implement user authentication (registration, login, and password management).  2-Develop features for team creation, browsing, and joining, including filters for skills, project type, and programming languages.  3-Integrate database functionality for storing user, team, and request data.  4-Perform unit testing for individual features and functionalities.  5-Conduct peer reviews and gather initial feedback for iterative improvements. | 1-Working prototype of the platform with core features implemented.  2-Iterative testing results and refined functionality. |
| Sprint 4  2 Weeks | 1-Validate the platform’s functionality and user experience.  2-Prepare for deployment and future scalability. | 1-Conduct extensive testing (functional, usability, and system testing) on the localhost server.  2-Perform user testing with a sample group of students to validate the system's effectiveness in solving the identified problem.  3-Gather and implement feedback from the supervisor, **Dr. Firas Al-Zoubi**, and users.  4-Document the system setup process for future deployment on a live server.  5-Finalize project documentation, including the Acknowledgement, Abstract, Problem Statement, Objectives, and Research Strategy. | 1-Fully functional platform tested and validated.  2-Deployment-ready system documentation.  3-Final project report. |

# CHAPTER 4

# DESIGN MODELS

## 4.2 Context diagram-0

A context diagram is high-level, an important tool for establishing the boundaries of the system being modeled. It doesn’t go into the detailed ins and outs of the system, they map out and entire system in a way that’s simple, clear, and easy to understand.

## 

Figure Context diagram-0

## 4.3 Data flow Diagram-1

A data flow diagram (DFD) is a graphical representation of the flow of information through a process or system. It uses standardized symbols and notations to show how data enters a system, is transformed by the system, and exits the system

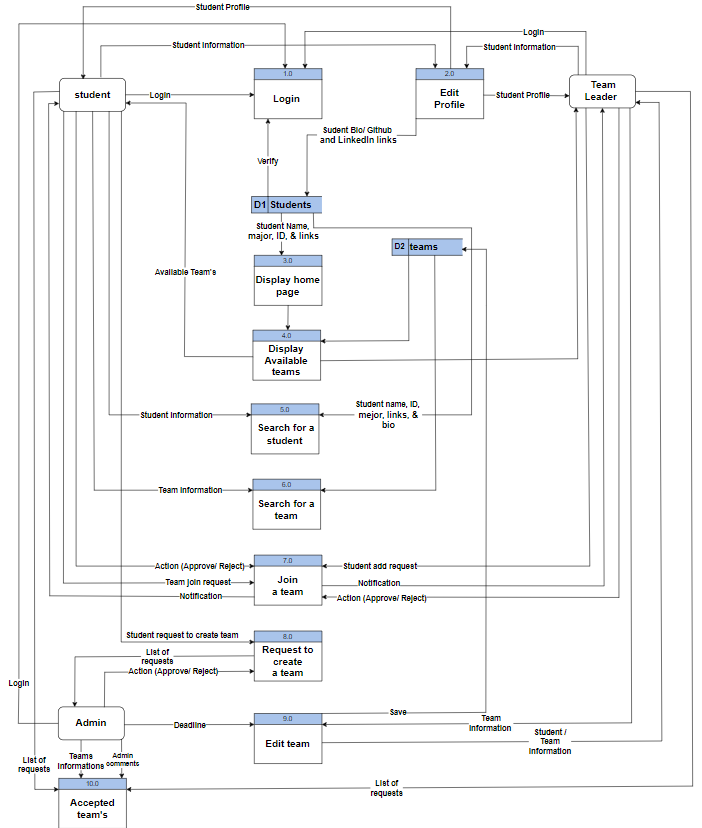


Figure Data flow Diagram -1

## 4.4 Use Case Diagram

A use-case diagram is one of the UML diagrams that models a system's behavior and helps capture its requirements. Its main purpose is to represent the interactions between the system and its actors.

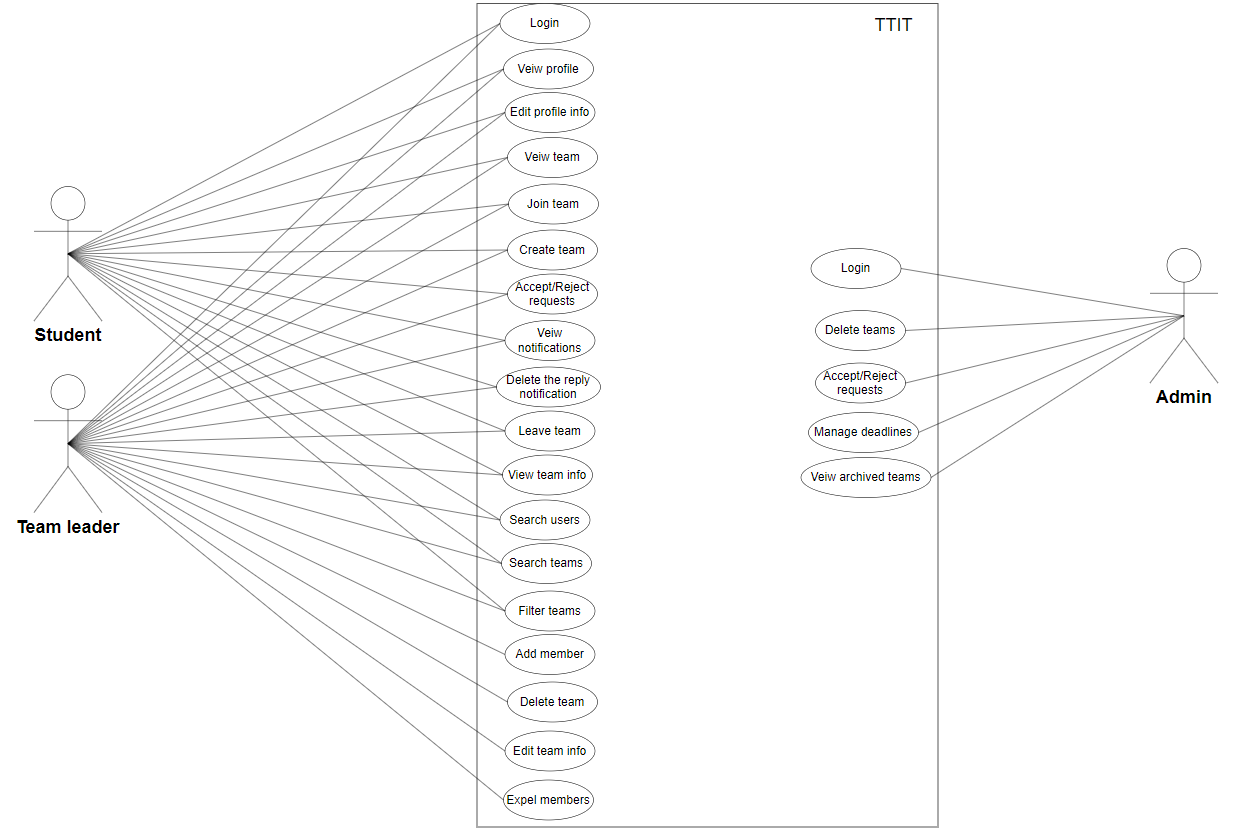


Figure Use Case Diagram

## 4.5 ER Diagram

An Entity-Relationship Diagram (ERD), also known as an ER diagram or ER model, is a graphical representation that depicts the relationships between entities (such as people, objects, places, concepts, or events) within an information system, for database design.

A diagram of a company

Description automatically generated

Figure ER Diagram

# CHAPTER 5

# EXPERIMENTS AND RESULTS

## 5.1 Overview

In this chapter, the system undergoes comprehensive testing to ensure that each unit performs its assigned tasks efficiently and aligns seamlessly with other units within the system. This process involves applying various software testing methodologies and evaluating the system’s performance in an environment that closely replicates the intended operational setting for which it was designed.

## 5.2 Test Design

Test design is a systematic process aimed at defining how testing should be conducted in an organized manner. This process involves selecting testing techniques, formulating test scenarios, creating test cases, choosing test data, and specifying expected outcomes. This comprehensive approach ensures the adoption of an effective and well-structured testing methodology.

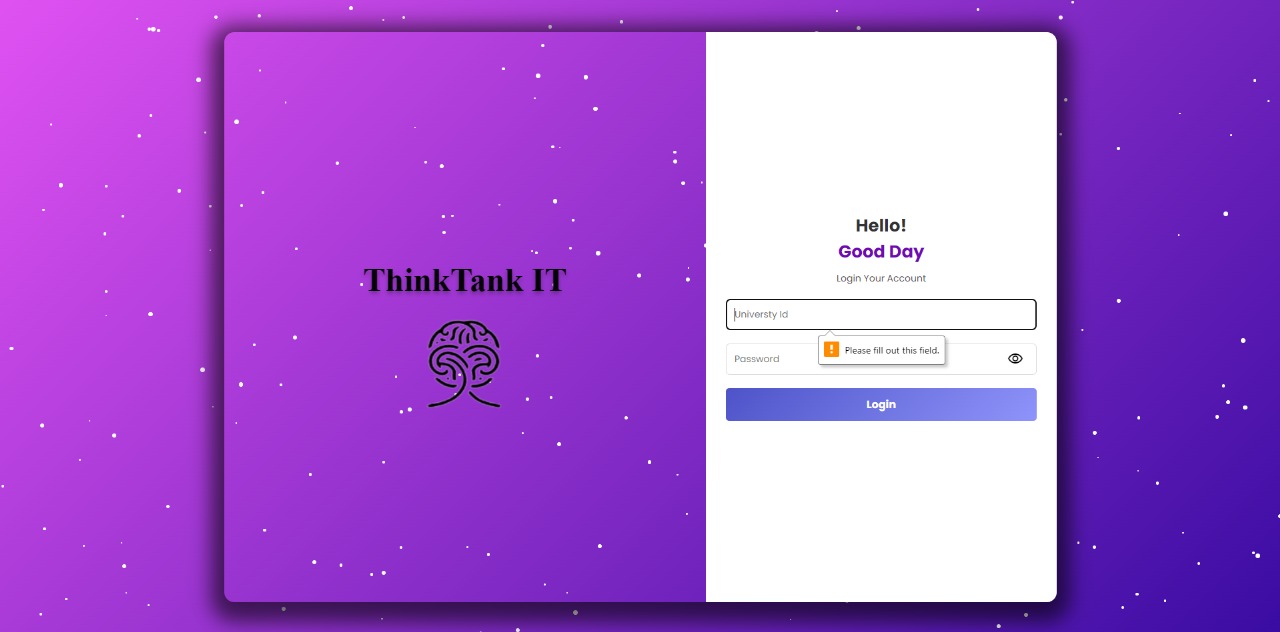
Login without ID and password

Figure Login Screen 1

A purple box with a purple background

Description automatically generatedLogin with invalid password

Figure Login Screen 2

A purple and white screen with a purple background

Description automatically generated with medium confidenceLogin where the ID is a string

Figure Login Screen 3

Login with invalid ID

A screenshot of a purple screen

Description automatically generated

Figure Login Screen 4

Login with valid ID and password

A purple screen with a purple background

Description automatically generated

Figure Login Screen 4

The home page after a successful login

A screenshot of a website

Description automatically generated

Figure Home Screen 1

Available teams

A screenshot of a computer

Description automatically generated

Figure Home Screen 2

Filter the available teams

A screenshot of a computer

Description automatically generated

Figure Home Screen 3

Filtered teams

A screenshot of a computer

Description automatically generated

Figure Home Screen 5

Team view after click team name

A screenshot of a computer

Description automatically generated

Figure Team Screen 1

Team view after click team name

A screenshot of a computer

Description automatically generated

Figure Team Screen 2

Search for a user

A screenshot of a computer

Description automatically generated

Figure Home Screen 6

User profile after click view button

A screenshot of a computer

Description automatically generated

Figure User profile Screen

Notifications after receiving a notification

A screenshot of a computer

Description automatically generated

Figure Home Screen 7

Creating a project with empty field

A screenshot of a computer

Description automatically generated

Figure Create Team Screen 1

Create a team

A screenshot of a computer

Description automatically generated

Figure Create Team Screen 2

Check the rule to create a team

A screenshot of a computer

Description automatically generated

Figure 23 Create Team Screen 3

Request sent successfully to admin

A screenshot of a computer

Description automatically generated

Figure Create Team Screen 4

Personal profile

A screenshot of a computer

Description automatically generated

Figure profile Screen 1

Profile edits: Add links and update bio

A screenshot of a computer

Description automatically generated

Figure profile Screen 2

Profile updated successfully

A screenshot of a computer

Description automatically generated

Figure profile Screen 3

Team overview displaying members and their details.

A screenshot of a computer

Description automatically generated

Figure Team Screen 3

Project details, programming languages, and fields utilized by the team.

A screenshot of a computer

Description automatically generated

Figure Team Screen 4

Team editing interface: Update name, project description, and organize members.

A screenshot of a computer

Description automatically generated

Figure Team Screen 5

The team’s name is being edited through the team management interface

A screenshot of a computer

Description automatically generated

Figure Team Screen 6

Team updated successfully

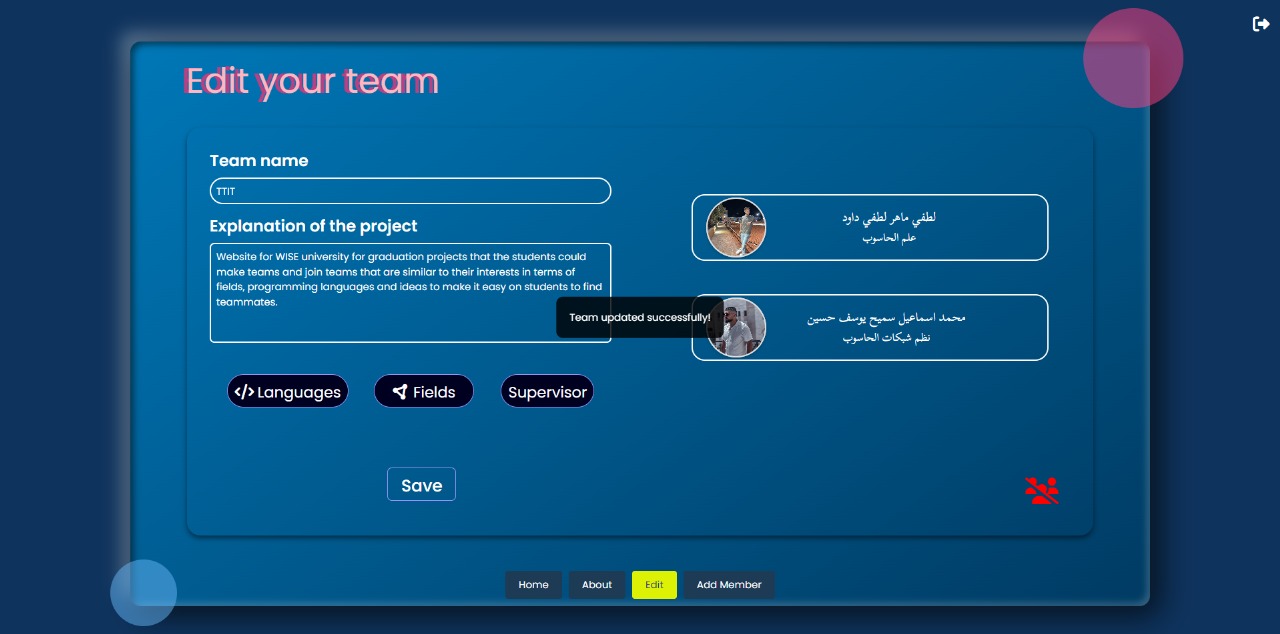


Figure Team Screen 7

A dedicated page for adding new team members

A screenshot of a computer

Description automatically generated

Figure Team Screen 8

Member added successfully.

A screenshot of a computer

Description automatically generated

Figure Team Screen 9

Admin panel interface for approving or rejecting team creation requests

A screenshot of a computer

Description automatically generated

Figure Admin Screen 1

Team archive: Displaying saved team details

A screenshot of a computer

Description automatically generated

Figure Admin Screen 2

Admin panel for managing member requests related to teams

A screenshot of a computer

Description automatically generated

Figure Admin Screen 3

The admin has approved the departure of a team member.

A screenshot of a computer

Description automatically generated

Figure Home Screen 8

The admin has approved removing a member from the team

A screenshot of a computer

Description automatically generated

Figure Home Screen 9

After a member leaves and another member is removed from the team

A screenshot of a computer

Description automatically generated

Figure Team Screen 10

5.3 User Acceptance Test

Based on user decisions, we discovered that 84.2% of people didn’t face an issue with our website and 15.8% faced it.

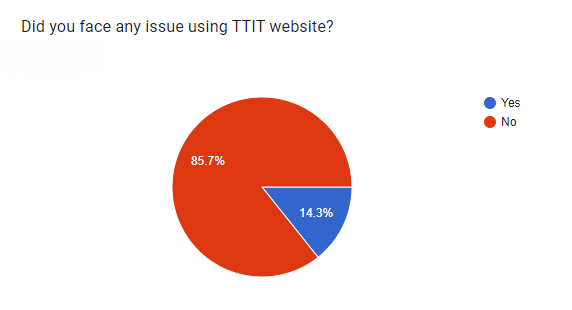


Figure Question 1

Based on user decisions, we discovered that 25.6% of people have the issue and 47.4% don’t.

A pie chart with numbers and a red circle

Description automatically generated

Figure Question 2

Based on user decisions, we discovered that 52.6% of people have the issue and 47.4% don’t.

A pie chart with a number of percentages

Description automatically generated

Figure Question 3

Based on user decisions, we discovered that 78.9% of people the website help them and 21.1% not.

A blue and red pie chart

Description automatically generated

Figure Question 4

Based on user decisions, we discovered that 89.5% of people find the website easy to use and 10.5% not.

A blue and red pie chart

Description automatically generated

Figure Question 5

## 5.4 Summery

After completing the system testing and making the necessary modifications, the operations were fully implemented without noticing any errors, as follows:

1. Access to the site via the student's university email
2. Ability to create a new team.
3. The supervisor can respond to requests and send notes about them, so that they appear to the student easily on the request page.

# CHAPTER 6

# CONCLUSION AND FUTURE WORKS

## 6.1 Overview

This chapter presents a holistic overview of the project’s key aspects, covering its objectives, achievements, main contributions, limitations, and future work.

## 6.2 Summary of the project

The ThinkTankIT System (TTIT) automates the process of forming and managing graduation project teams to streamline and expedite the process for users. This system enables students to entry profiles so that showcasing their skills and academic interests, find suitable team members, and form teams electronically at any time. Additionally, it allows students to request team changes, such as removing or adding members, with the supervisor's approval.

Supervisors receive notifications for every new team formation or change request to approve or reject. Supervisors can also monitor team progress, and ensure that the project aligns with academic standards.

Employing the **Scrum framework** allowed for an iterative and structured development process, ensuring that each sprint contributed to the completion of critical functionalities, from user authentication and team management to admin oversight and notification systems. This Agile approach also provided flexibility to adapt to changing requirements and facilitated continuous testing and improvement throughout the development lifecycle.

This system aims to enhance collaboration, improve team management, and reduce processing times for all stakeholders. By leveraging robust technology and secure protocols, the platform ensures the safe and reliable transfer of data while empowering students and supervisors with greater control over their academic interactions and workflows.

In conclusion, the **TTIT** website not only meets the immediate needs of its target audience but also demonstrates the effective application of modern software development practices in addressing real-world problems. It stands as a testament to the potential of technology to simplify and enhance collaborative efforts in academic environments.

## 6.3 Achieved objectives

### Objective 01:

Understand the current project scope and articulate the problem statement

comprehensively.

This involves gaining a thorough understanding of the existing situation and challenges faced by students within to creation the team process, as outlined in Chapter (2). This examination informed decision-making and enabled us to point out the essential functionalities of our system. Furthermore, we conducted a comparative analysis of our proposed features against those already implemented in existing systems with similar objectives.

### Objective 02:

Develop and implement the (TTIT) System, Select A Methodology:

We chose the scrum methodology due to its flexibility with requirements changing, responsiveness to feedback, and effectiveness with small teams.

* Determine System Functionalities:

Defining the system’s objectives and features it ought to offer, ensuring alignment with the user needs and expectations.

* Comparing Functionalities with other Existing Systems:

To Understand the similar functionalities of other existing systems and determine what needs to be improved in a system with a similar purpose.

* + Doing A Feasibility Study:

To evaluate Economic, Technical, And Operational aspects, providing essential information to verify the project’s sustainability and alignment.

* + Implement Scrum Methodology:

Breaking the project into multiple sprints to simplify implementation and fit within the time constraints.

* + Designing The System:

Formulate a comprehensive system design, incorporating various diagrams such as the Context diagram, Data Flow diagram, Use Case diagram, the ER (Entity-Relationship) diagram.

* + Coding:

Build the project by using Visual Studio Code and coding in (HTML, CSS, JS, Node.JS, Exspress.JS), PostgreSQL to build the database, and Local Host server to execute the code.

## 6.4 Limitation

* The platform does not include tools assess the skills listed by students
* All changes in team formation require supervisor approval, which could delay the process if the supervisor is unavailable or has a high workload.
* The system may not accommodate all students from the university or large numbers only including IT students
* There is no means of communication between students general and between the team member especially

## 6.5 Future Work

* Add tools to verify student skills, such as skill tests or coding challenges.
* Introduce automated approval processes or delegate approval tasks to assistants to reduce delays.
* Expand the system to include students from other departments, not just IT.
* Add messaging systems such as chatting to improve communication between students and team members.

# REFERENCES

1- <https://info.catme.org/features/team-maker/> (last accessed 14/10/2024)

2- <https://link.springer.com/content/pdf/10.1007/s12369-021-00763-z.pdf> (last accessed 14/10/2024)

3- <https://nus.edu.sg/> (last accessed 14/10/2024)

4- <https://engineering.futureuniversity.com/BOOKS%20FOR%20IT/Software-Engineering-9th-Edition-by-Ian-Sommerville.pdf> (last accessed 15/12/2024)

5- scrum.org <https://www.scrum.org/resources/blog/what-sprint-backlog-and-how-it-connects-other-elements-scrum> (last accessed 15/12/2024)

6- Miro,” what’s the context diagram and how do you use it?” <https://miro.com/blog/context-diagram/> (last accessed 15/12/2024 )

# Appendices

### A pie chart with a number of percentages Description automatically generated

1. <https://info.catme.org/features/overview/> (last accessed 14/10/2024) [↑](#footnote-ref-1)
2. <https://link.springer.com/content/pdf/10.1007/s12369-021-00763-z.pdf> (last accessed 14/10/2024) [↑](#footnote-ref-2)
3. . <https://scale.nus.edu.sg/programmes/undergraduate-programmes/bachelor-of-information-technology?gad_source=1&gclid=CjwKCAiAg8S7BhATEiwAO2-R6vjjO8h0pISWOEuWB618DOVIUKRTn12XGsgp26QHeRLiKrj96zILrhoC5jgQAvD_BwE> (last accessed 14/10/2024) [↑](#footnote-ref-3)