**Software Requirements Specification**

**for   
**

**Version 1.0 approved**

**Prepared by**

**Osama Zaid, Abdallah Wefky, Amr Khalid, Ahmed Mahmoud, Gehad Medhat, Shaimaa Yasser, Tasneem Emad**

**Faculty of Commerce, BIS Department**

**Assiut University**

**June 15, 2025**

**ACKNOWLEDGMENTS**

We sincerely thank our academic supervisors, Dr. Ramadan and Dr. Hesham, for their continuous guidance, insightful feedback, and unwavering support throughout this project. Their expertise and encouragement were crucial to the successful completion of our work.

We also extend our heartfelt thanks to Dr. Mostafa Qaid, Head of the Business Information Systems Department, for providing us with the necessary resources and academic environment to pursue this project.

We are grateful to the Faculty of Commerce, Assiut University, for providing the academic foundation and research facilities that made this project possible.

Finally, we acknowledge the support of our families and friends who encouraged us throughout this journey.

**Abstract**

**Makkeny** is a dynamic web-based learning platform designed to help recent graduates overcome career and skill development challenges.

It bridges the gap between academic knowledge and market needs by offering personalized learning paths, mentorship, and practical training in business, marketing, accounting, programming, and soft skills.

Built with React for the frontend, Node.js for application logic, and MongoDB for data management.

The platform features user-friendly interfaces, career guidance, tailored course recommendations, and Access to expert coaching and mentorship, reduces the time, effort, and cost associated with traditional training methods while providing more effective and targeted learning experiences.

Testing confirms Makkeny as an effective, affordable, and scalable solution for job market preparation.

**Table of Contents**

[1. Introduction 5](#_Toc200324109)

[1.1 Purpose 5](#_Toc200324110)

[1.2 Document Conventions 5](#_Toc200324111)

[1.3 Intended Audience 5](#_Toc200324112)

[1.4 Product Scope 5](#_Toc200324113)

[1.5 References 6](#_Toc200324114)

[2. Overall Description 6](#_Toc200324115)

[2.1 Product Perspective 6](#_Toc200324116)

[2.2 Product Functions 6](#_Toc200324117)

[2.3 User Classes and Characteristics 7](#_Toc200324118)

[2.4 Operating Environment 7](#_Toc200324119)

[2.5 Design and Implementation Constraints 7](#_Toc200324120)

[2.6 User Documentation 7](#_Toc200324121)

[2.7 Assumptions and Dependencies 7](#_Toc200324122)

[3. External Interface Requirements 7](#_Toc200324123)

[3.1 User Interfaces 7](#_Toc200324124)

[3.2 Hardware Interfaces 7](#_Toc200324125)

[3.3 Software Interfaces 7](#_Toc200324126)

[3.4 Communications Interfaces 7](#_Toc200324127)

[4. System Features 8](#_Toc200324128)

[4.1 User Register 8](#_Toc200324129)

[4.2 Course Filtering 8](#_Toc200324130)

[4.3 Course Enrollment Request 8](#_Toc200324131)

[4.4 Schedule Mentorship Session 8](#_Toc200324132)

[4.5 Cancel Enrollment 8](#_Toc200324133)

[4.6 Provide Feedback 8](#_Toc200324134)

[4.7 Add Course 8](#_Toc200324135)

[4.8 Edit Course 8](#_Toc200324136)

[4.9 Mentor Response 8](#_Toc200324137)

[5. Other Nonfunctional Requirements 8](#_Toc200324138)

[5.1 Performance Requirements 8](#_Toc200324139)

[5.2 Security Requirements 8](#_Toc200324140)

[5.3 Software Quality Attributes 8](#_Toc200324141)

[5.4 Business Rules 8](#_Toc200324142)

[5.5 Culture and political 8](#_Toc200324143)

[6. System Architecture and Design Models 9](#_Toc200324144)

[6.1 Architectural Design 9](#_Toc200324145)

[6.2 Diagrams Description: ERD, UML, etc…... 12](#_Toc200324146)

[7. Interface Design 13](#_Toc200324147)

[7.1 Overview of User Interface 13](#_Toc200324148)

[7.2 Screen Images 13](#_Toc200324149)

# **1. Introduction**

## **1.1 Purpose**

Makkeny is a web application designed to assist recent graduates, particularly from the Faculty of Commerce at Assiut University, in navigating the labor market. The application aims to provide a secure and user-friendly platform for career development and skill enhancement by offering targeted course recommendations, personalized career counseling, mentorship, and practical experience opportunities. The purpose of this document is to outline the software requirements for the Makkeny application to ensure it meets the needs of its users while maintaining security and usability.

## **1.2 Document Conventions**

1. Register a new account for users (Users, Instructors, Mentors, Admins).
2. Allow users to browse and enroll in courses.
3. Facilitate career counseling sessions and mentorship programs.
4. Enable users to apply for internships and workshops.
5. Provide admin functionalities to manage users, courses, and events

## **1.3 Intended Audience**

1. **Developers**: Responsible for implementing the application.
2. **Project Managers**: Oversee the project timeline and resources.
3. **Marketing Staff**: Promote the application to target users.
4. **Users**: Users, mentors, admins, and Instructors.
5. **Testers**: Ensure the application meets quality standards.
6. **Documentation Writers**: Create user manuals and help files.

## **1.4 Product Scope**

**Objective**:

Makkeny aims to help recent graduates find suitable career paths by providing access to industry relevant training and opportunities at affordable costs.

**Goals**:

* Facilitate skill enhancement through tailored courses and workshops.
* Provide personalized career counseling and mentorship.
* Offer practical experience via internships with partnered companies.

**Benefits**:

* Graduates gain access to affordable, targeted training.
* Companies can connect with potential talent.
* The platform earns a commission from course enrollments and partnerships

## **1.5 References**

Not Applicable.

# 

# **2. Overall Description**

## **2.1 Product Perspective**

**Makkeny** is a self-contained, user-friendly online platform that simplifies career development for graduates by connecting them with courses, mentors, and companies. It operates as a standalone web application accessible via modern browsers.

## **2.2 Product Functions**

## **2.3 User Classes and Characteristics**

## **2.4 Operating Environment**

## **2.5 Design and Implementation Constraints**

## **2.6 User Documentation**

## **2.7 Assumptions and Dependencies**

# **3. External Interface Requirements**

## **3.1 User Interfaces**

## **3.2 Hardware Interfaces**

## **3.3 Software Interfaces**

## **3.4 Communications Interfaces**

# **4. System Features**

## **4.1 User Register**

## **4.2 Course Filtering**

## **4.3 Course Enrollment Request**

## **4.4 Schedule Mentorship Session**

## **4.5 Cancel Enrollment**

## **4.6 Provide Feedback**

## **4.7 Add Course**

## **4.8 Edit Course**

## **4.9 Mentor Response**

# **5. Other Nonfunctional Requirements**

## **5.1 Performance Requirements**

## **5.2 Security Requirements**

## **5.3 Software Quality Attributes**

## **5.4 Business Rules**

## **Culture and political**

# **6. System Architecture and Design Models**

## **6.1 Architectural Design**

The system is composed of several core components, each representing a specific layer or function within the architecture:

* **User Interface Module**:

Responsible for handling user interactions. Includes components for user registration, request submission, and point tracking. Interfaces with the Business Logic module for processing user requests.

* **Business Logic Module**:

Manages the core functionalities of the application. Validates user requests and matches them with appropriate service providers. Calculates points and discounts for users. Interfaces with the Data Access module for accessing and updating data.

* **Service Provider Module**:

Handles activities related to service providers. Allows service providers to add, edit, and delete services offered. Interfaces with the Business Logic module for service request processing.

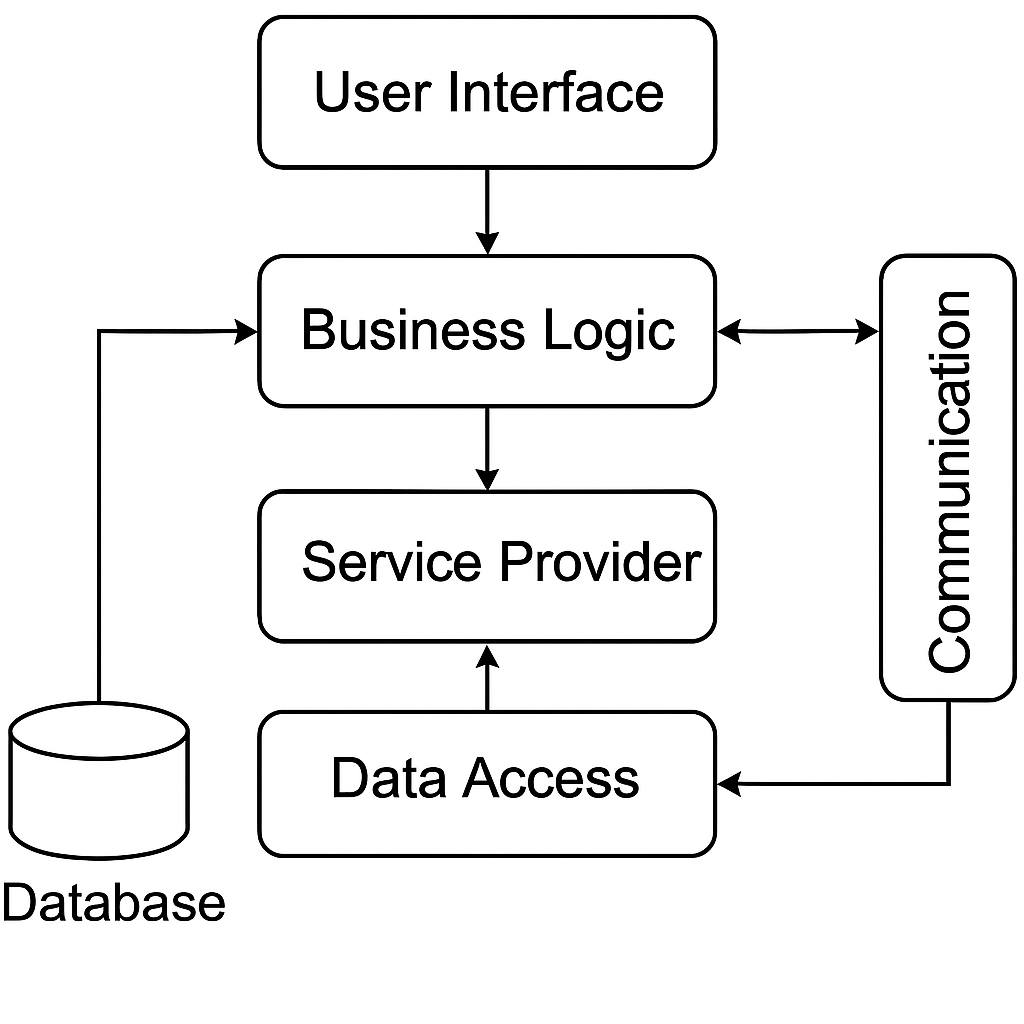
* **Data Access Module**:

Manages access to the data repositories. Handles CRUD operations for user accounts, service requests, and service provider information. Ensures data security and integrity.

* **Communication Module**:

Facilitates communication between users and service providers. Handles notifications, messages, and updates. Interfaces with both the User Interface and Business Logic modules.

Figure 1 System **Architectural Design**



**Description of the Diagram:**

The diagram illustrates the major subsystems: User Interface, Business Logic, Service Provider, Data Access, and Communication.

Arrows indicate the flow of information between subsystems.

User Interface interacts with Business Logic for request processing and Data Access for user information.

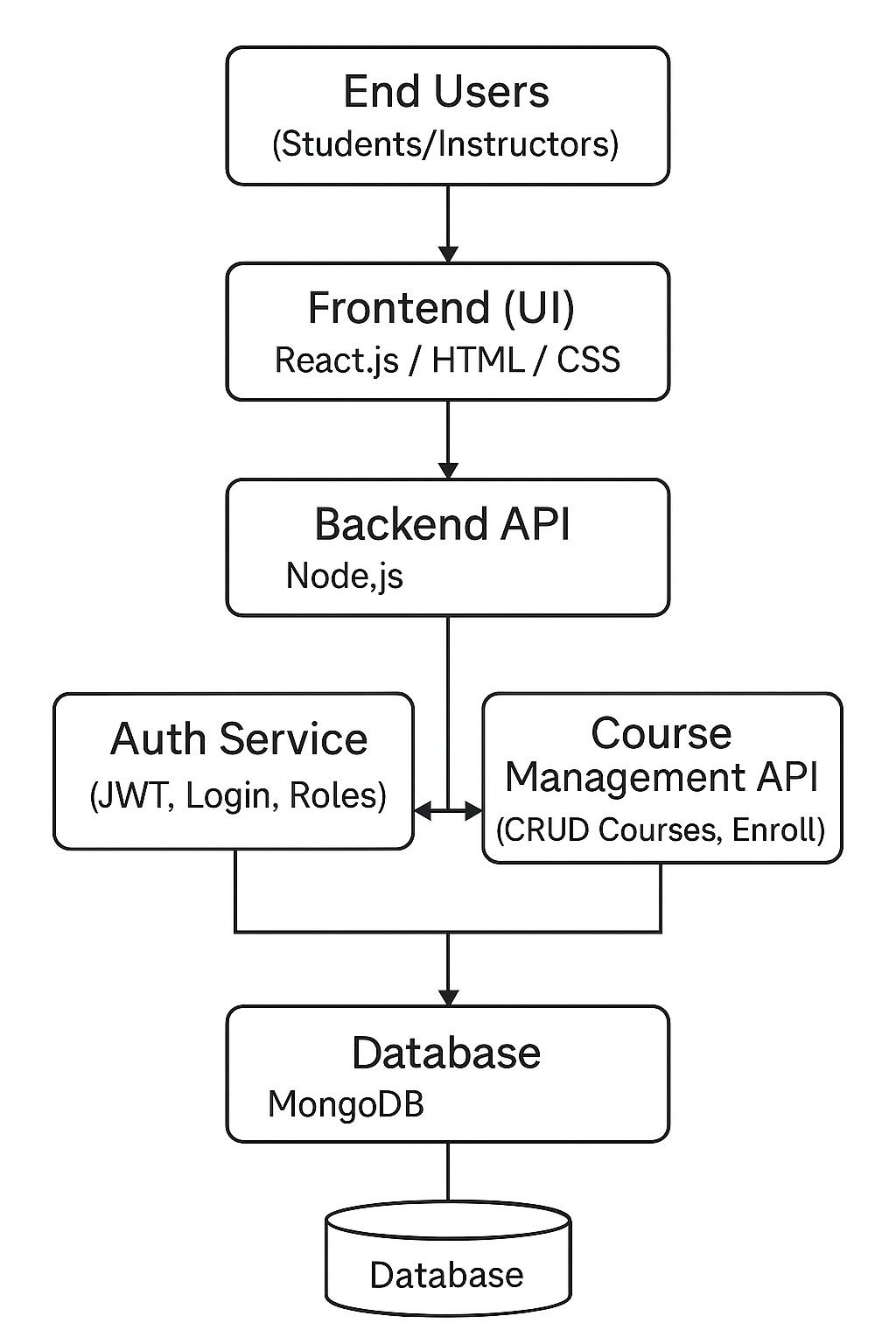
Service Provider communicates with Business Logic for service management.

Data Access manages access to the database, ensuring data integrity.

Communication facilitates interaction between users and service providers.

This architecture promotes modularity and separation of concerns, allowing for easier maintenance, scalability, and collaboration between subsystems to achieve the desired functionality of the system.

Figure 2 Example



**End Users:**

Represent the external actors of the system, including students, instructors, and administrators. Interact with the system via the User Interface Module to perform various actions such as registration, course enrollment, and system management.

**Frontend (UI):**

Corresponds to the User Interface Module. Responsible for handling user interactions such as registration, login, navigation, and request submission. Provides a user-friendly interface and connects directly to the Business Logic module for processing operations.

**Backend API:**

Represents the Business Logic Module. Handles the core functionalities of the application including request validation, course and user management, and coordination between services. Interfaces with the Data Access module for reading/writing data and with the Auth Service for user authentication and access control.

**Auth Service:**

A sub-module of the Business Logic or a dedicated microservice. Manages user authentication, authorization, and role-based access control. Ensures secure access to protected resources in the system.

**Course Management API:**

Represents the Service Provider Module. Allows instructors to create, update, and manage their courses. Enables linking of course content with enrolled users and integrates with the Business Logic for validation and tracking.

**Database:**

Corresponds to the Data Access Module. Manages CRUD operations for users, courses, sessions, and system logs. Ensures data consistency, security, and integrity across the platform.

## **6.2 Diagrams Description: ERD, UML, etc…...**

# **Interface Design**

## **7.1 Overview of User Interface**

## **7.2 Screen Images**