



**NATIONAL UNIVERSITY**  
of Computer & Emerging Sciences

## **“Personalized Study Planner on the Cloud”**

*Utilizing Microsoft Azure Services*

### **PROJECT PROPOSAL**

**COURSE CODE: CS4037**

**COURSE NAME: INTRODUCTION TO CLOUD COMPUTING**

**INSTRUCTOR: SIR MUHAMMAD SUDAIS**

# **Project Proposal**

**Title: Personalized Study Planner on the Cloud**

**Course: Introduction to Cloud Computing (CS 4037)**

**Instructor:** Muhammad Sudais

## **Group Members:**

- [SAAD AHMED – Project Lead] – 22K 4345
  - [MUHAMMAD HUZAIFA] – 22K4223
  - [MUHAMMAD YOUSUF] – 22K4457
  - [MURTAZA JOHAR] – 22K4508
  - [HASEEB MUJTABA] – 22K4307
  - [ALI RAZA SHAIKH] – 22K4313
  - [HUNAIN MEMON] – 22K5014
- 

## **1. Project Overview**

The **Personalized Study Planner on the Cloud** is a web-based application hosted on **Microsoft Azure** that enables students to plan, organize, and track their study schedules effectively. Students can register, log in, and manage their personalized schedules.

The system will integrate **Azure services** such as **App Service, SQL Database, Blob Storage, Azure Functions, and Azure Active Directory**, ensuring a **scalable, resilient, and secure cloud solution**.

## **2. Objectives**

1. To design and deploy a cloud-based study planner using Microsoft Azure.
2. To utilize core Azure services such as **App Service, SQL Database, Blob Storage, Azure Functions, and Active Directory**.
3. To implement a **scalable, secure, and resilient architecture** that can support multiple users simultaneously.

4. To gain practical experience in **CI/CD pipelines using Azure DevOps**.
5. To demonstrate **cloud monitoring, cost optimization, and security best practices**.

## 3. Project Requirements

### 3.1 Functional Requirements

- **Application:**
  - A web application built using ASP.NET/Python/Node.js hosted on **Azure App Service**.
  - Includes **user authentication system**.
- **User Features:**
  - User registration and login (via **Azure Active Directory**).
  - Students: create/update/delete study schedules (CRUD).
  - Upload and manage study resources (PDFs, images, notes) stored in **Azure Blob Storage**.
- **API Integration / Azure Function App:**
  - **Azure Function (time-triggered)**: sends daily reminders to students (e.g., via email).
  - **API**: Expose a secure API to fetch study schedules using **Azure API Management**.
- **Cloud Hosting:**
  - Hosted using **Azure App Service** for high availability.
- **Database:**
  - Store user data, tasks, and schedules in **Azure SQL Database**.
- **Storage:**
  - Store notes, PDFs, and other documents in **Azure Blob Storage**.
- **Monitoring & Logging:**

- Use **Azure Monitor** and **Application Insights** to track app performance, errors, and user activity.
- **Security:**
  - **Azure Active Directory (AAD):** Authentication and role-based access.
  - **Azure Key Vault:** Store sensitive data such as database connection strings and API keys.
- **Networking (Optional):**
  - Configure **Virtual Network (VNet)** and **Network Security Groups (NSGs)** for secure internal communication.

### 3.2 Non-Functional Requirements

- **Performance:** Application must handle **100+ concurrent users**.
- **Scalability:** Auto-scale with **Azure App Service scaling**.
- **Availability:** Maintain **99.95% uptime SLA**.
- **Security:** Ensure encryption, identity management, and secure access.
- **Cost Optimization:** Use resource scaling, monitoring, and cost management tools in Azure.

## 4. Azure Services to be Used

- **Compute:** Azure App Service, Azure Function Apps
- **Storage:** Azure Blob Storage
- **Database:** Azure SQL Database
- **Networking (Optional):** Azure VNet, Traffic Manager
- **Security:** Azure Active Directory, Azure Key Vault
- **Monitoring:** Azure Monitor, Application Insights
- **DevOps (Optional):** Azure DevOps for CI/CD pipelines

## **5. Deliverables**

- **Proposal Document** (this document).
- **Source Code** on GitHub/Azure DevOps with documentation.
- **Azure Resources** (App Service, SQL DB, Blob Storage, Functions, Monitor, AD, Key Vault).
- **Final Report** (architecture, services, challenges, optimizations).