# Riphah School of Computing and Innovation (RSCI), Lahore



Computer Organization and Assembly Language (Lab)

## Project Proposal

**Submitted to:** Ms. Amna Bibi

**Group Members:** 1

**Members Id and Names:**

**NAME:** Muhammad Abdullah; **SAP:** 53457

BSCS 3-B

28/10/2024

## Aim of the Project

The Basic System Kernel project aims to create a foundational system kernel that serves as a minimal operating environment, providing low-level access to hardware and fundamental system services. Developed in Assembly language, this project will enhance understanding of low-level programming concepts, memory management, interrupt handling, and system architecture by focusing on essential kernel operations.

## Brief Description

The Basic System Kernel is a console-based application developed in Assembly language that simulates core functionalities of a simple operating system kernel. This kernel handles essential processes like managing system memory, scheduling basic tasks, and handling input/output operations directly. The goal is to create a simplified kernel that interacts with the system hardware at a low level, allowing users to explore system internals and gain hands-on experience with memory management, process handling, and interrupt control in Assembly language.

## Functionality of Project

* **Memory Management:**
  1. Basic memory allocation and deallocation for efficient resource usage.
  2. Simple mechanisms to prevent memory leaks and ensure system stability.
* **Process Scheduling:**
  1. Implement a basic scheduler to manage process execution.
  2. Support for round-robin scheduling to ensure fairness in process handling.
* **Input/Output Handling:**
  1. Direct management of I/O operations for reading from and writing to hardware devices.
  2. Simplified console-based interface for displaying system messages and accepting user commands.
* **Interrupt Handling:**
  1. Implement interrupt service routines (ISRs) to handle hardware and software interrupts.
  2. Basic exception handling to ensure system stability and error recovery.
* **Security and Stability Measures:**
  1. Implement basic access control for system resources.
  2. Use kernel-level error handling for stability during abnormal process execution.
* **User-Friendly Console Interface:**
  1. Clear prompts for monitoring kernel activities.
  2. Simple menu for accessing features and observing kernel responses to various operations.