Skills Analysis

- **Familiarity with O-RAN Architecture / Specifications**
  - MAC protocol
  - PHY (physical layer) protocol – modulation schemes, coding schemes, and multiple access techniques such as OFDMA
  - RLC (Radio Link Control) protocol – provides reliability and error correction over the radio interface
  - Interface protocols including E1, F1, and S1, all of which perform different connectivity and communication functions between different network components
  - Near-RT and non-RT Radio Interface Controller functions

- **4G/5G/LTE Technologies and Standards**
  - Deployment scenarios such as network slicing and network function virtualisation (NFV)
  - Knowledge of core network protocols such as NGAP and S1AP for communication between the RAN and core network
  - Proficiency in configuring and managing RAN elements such as base stations, radio units, distributed units, and central units
  - 5G functional splits including 7.2, 7.3, and 7.2x splits, all of which define how base station functions are accomplished by different elements of the network
  - Familiarity with wireless propagation models and radio frequencies
  - PDCP (Packet Data Convergence Protocol)
  - RRC (Radio Resource Control)
  - SCTP (Stream Control Transmission Protocol)
  - GTP-U (GPRS Tunnelling Protocol – User Plane)

- **Network Protocols and Networking Concepts**
  - TCP
  - UDP
  - IP
  - IP routing
  - VLANs
  - Subnetting
  - QoS
  - SSL/TLS
  - Ethernet
  - DHCP
  - DNS
  - SNMP
  - SFTP / FTP
  - TCP/IP and OSI Protocol Suite
  - IPsec

- Advanced Encryption Protocols
- Software defined networking

- **Operating System Knowledge**
  - Ubuntu 22.04/Linux familiarity
  - General: Windows/Linux
  - Knowledge of hypervisors
    - Setting up virtualisation support in UEFI/BIOS settings
  - Creation and management of Virtual Machines
    - Disk partitioning
    - Installation of guest OSes
  - Containerisation technologies
    - Creating, pushing, pulling images to/from repositories
    - Familiarity with writing Docker files (i.e. Docker syntax) to define dependencies and the build steps of a container image
    - Container lifecycle and image management
  - Dual boot configuration of a guest and native OS
  - IDE familiarity (VS Code, VIM, PyCharm etc)
  - Shell scripting
  - Automation Tools (Ansible, Kubernetes)

- **Soft skills**
  - Comprehensive documentation abilities
  - Technical writing skills
  - Document design/outlining
  - Ability to create and modify test plans
  - Collaboration/teamwork
  - Knowledge of IT project management principles

As we completed our 5G Indoor Testbed R&D project, the skills list we developed earlier became even more crucial for efficient task delegation. It allowed us to leverage our team's strengths effectively. For example, Chris and Kat took charge of the hardware setup and establishing connectivity between the radio modules. Chris had gained valuable experience troubleshooting the hardware during Semester 1, and Kat supported the effort. Ed also contributed to hardware troubleshooting. Meanwhile, Ed and Sam focused on setting up the Docker package, drawing on their experience from personal projects, which helped streamline the containerization process. This approach ensured smoother coordination and faster progress throughout the project.

Training Plan

From the outset, our 5G Indoor Testbed project presented a significant learning curve, particularly in areas such as radio modules and O-RAN technology. As a group, we struggled to find solid, relatable examples or clear solutions for the challenges we faced. This was compounded by the fact that we had limited experience with radio modules, and there was a noticeable gap in available resources for addressing the specific issues we encountered. However, recognizing this from the start, we incorporated a flexible training plan to tackle these difficulties.

Our training plan, though not exhaustive, focused on acquiring the key skills necessary for the project's success. We emphasized gaining proficiency in 5G technologies, O-RAN architecture, and Software Defined Radios (SDRs)—all of which were central to the testbed's development. Additionally, we understood the importance of mastering containerization technologies like Docker and having a foundational understanding of virtualization and virtual machine (VM) management for efficient testbed setup.

By leveraging individual expertise within our team and committing to knowledge-sharing efforts, we bridged some of the gaps. For example, knowledge transfer through documentation, workshops, and resources such as related articles and crash courses helped ensure everyone was up to speed with the latest developments. This collaborative approach was crucial, especially given the technical complexity of the project.

Ultimately, while the project domain was predictably challenging, our flexible and proactive training plan—along with continuous internal collaboration—allowed us to navigate these obstacles more confidently. Although the radio module expertise initially seemed out of reach, our commitment to learning and teamwork helped us adapt and move forward effectively.