**Practical No.3**

**Introduction**

Use case diagrams provide a clear visualization of how users interact with your system. For IPCRA, they illustrate key functionalities like resume analysis, interview simulation, feedback generation, and real-time insights. This helps both developers and stakeholders understand the system flow from candidate input to interview refinement

**Objectives**

By completing this practical, you will be able to:

* Identify key actors such as Job Applicants, Admin, and HR Reviewers
* Associate use cases with include, extend, and generalization relationships
* Draw a complete use case diagram for IPCRA

**Importance of Scheduling**

Scheduling plays a central role in developing AI systems with modular integration. For IPCRA:

* Ensures NLP models are trained before feedback modules are built
* Coordinates resume parsing, simulation flow, and dashboard rendering
* Allocates resources efficiently across ML model training and UI/UX
* Syncs API calls and external datasets for accurate simulations
* Supports testing phases where accuracy and response relevance are critical

**Task Identification using WBS**

A Work Breakdown Structure (WBS) breaks down the entire FDMS project into smaller, manageable tasks. This helps in planning, scheduling, assigning responsibilities, and tracking progress.

WBS for FDMS

Level 1: Flight Delay Management System

**1. Requirements Gathering**

* Identify stakeholders (Job Seekers, HR Teams)
* Define system goals and constraints
* Gather functional (analysis, feedback) and non-functional (security, latency) requirements

**2. System Design**

* Use Case Diagram: Resume upload, mock interview initiation, feedback visualization
* UI/UX Wireframes for Chatbot and Dashboard
* Design flow for candidate scoring and analytic

**3. Development**

* Backend: Resume parsing (Python), Feedback engine (ML model)
* Frontend: Chat interface (React/HTML/CSS), Graph dashboard
* API Integrations: Job role matcher, LinkedIn scraping (if allowed)

**4. Feedback & Interview Module**

* Build NLP model for interview analysis
* Integrate semantic matching and keyword extraction
* Score responses based on tone, relevance, and clarity

**5. Notification System**

* Email alerts for scheduled mock interviews
* Post-interview report summaries
* Alerts for resubmission or re-interview

**6. Testing**

* Unit and Integration Testing
* Use test resumes to validate parser accuracy
* Evaluate model predictions with real response data

**7. Deployment**

* Host on cloud platform (Azure/GCP)
* Set up containerization and load balancing
* Secure data access with role-based authentication

**8. Maintenance & Monitoring**

* Monitor user queries and chatbot accuracy
* Refine ML models with new interview datasets
* Collect feedback for periodic updates

**Resource Type:**

| **Resource Type** | **Specific Resources** |
| --- | --- |
| **Human Resources** | Project Lead, ML Engineer, Frontend Developer, UX Designer, Tester |
| **Software Tools** | Python (spaCy, sklearn), React, Figma, Postman, Firebase, GitHub |
| **Hardware** | Developer laptops, GPU nodes for model training, cloud server |
| **Other Resources** | Resume datasets, interview Q&A sets, job role APIs, hosting platform |

**Monitoring – FDMS Project**

* **Project Monitoring**: Track task completion and sprint velocity
* **System Monitoring**: Analyze chatbot logs for accuracy and timeouts
* **Resource Monitoring**: Ensure model training doesn't bottleneck system performance
* **Quality Monitoring**: Use feedback loop to adjust scoring algorithms
* **Post-deployment Monitoring**: Track resume uploads, success rates, user engagement

**Conclusion**

IPCRA combines AI and resume analytics to elevate candidate readiness. Through scheduling, task decomposition via WBS, proper resource allocation, and real-time monitoring, development becomes seamless and performance stays reliable—just what recruiters and candidates both need.