**AI Project Proposal – Intelligent Event Reminder Bot**

# 1. Project Title

Intelligent Event Reminder Bot – Rules + Time Triggers

# 2. Team Members

Name: [Muhammad Aawad ali]  
ID.: [F2022266686]  
Section: [v8]  
Email: [f2022266686@gmail.com]

# 3. Problem Statement

People often forget their tasks or events due to busy schedules or poor time management. There is a need for an intelligent system that can assist in reminding users of their events based on time and rule-based conditions.

# 4. Objectives

- Identify the real-world problem of task and time mismanagement.  
- Design reminder scheduling using both time and rule-based triggers.  
- Handle natural language input for creating reminders.  
- Provide notifications through a graphical interface and voice alerts.  
- Test and optimize the bot for reliable performance.

# 5. Project Description

This project involves developing an intelligent event reminder bot capable of reminding users of their tasks or events. The bot will utilize both time-based triggers (e.g., "remind me at 4:00 PM") and rule-based logic (e.g., "remind me every Monday"). Users can input reminders in natural language, which the system will interpret using NLP techniques. Alerts will be delivered via GUI pop-ups and voice. The goal is to enhance productivity by helping users better manage their time and daily tasks.

# 6. Artificial Intelligence Technique(s) to be Used

- Rule-Based Systems  
- (Optional) Natural Language Processing (using spaCy or NLTK)

# 7. Proposed Environment

The bot will operate on a personal computer using a graphical user interface built with Tkinter. It will handle reminder scheduling and event management locally.

# 8. Tools / Languages / Libraries

- Programming Language: Python  
- Tools: Jupyter Notebook, VS Code  
- Libraries:  
 - schedule, datetime for time-based reminders  
 - pyttsx3 for voice alerts  
 - tkinter for GUI  
 - spaCy or NLTK (optional NLP)

# 9. Expected Challenges

- Managing delayed execution due to system resource limitations  
- Implementing accurate natural language parsing  
- Handling GUI compatibility and user input variations  
- Resolving conflicts when multiple reminders overlap

# 10. Evaluation Criteria

- Accuracy of delivering reminders  
- Timeliness of alerts  
- Ease of use and user interface quality  
- Success rate of NLP interpretation (if implemented)

# 11. Tentative Plan / Timeline

|  |  |
| --- | --- |
| Week | Task |
| 3–4 | Finalize requirements, collect inputs, design architecture |
| 5–6 | Develop reminder scheduler (time + rules), begin core logic |
| 7–8 | Build GUI, integrate NLP (optional), test on scenarios |
| 9 | Final testing, fix bugs, prepare final presentation |