CelerSMS AT Command Emulator Assignments

Instructions

- Use suitable virtual AT emualtors like https://www.celersms.com/at-emulator.htm OR any other suitable ones
- Log your tests and results on Jira
- Upload test code/scripts to github
- Share your work with your nominated seniors

1: Setup & Fundamentals

- Install https://www.celersms.com/at-emulator.htm
- Explore AT command layout in emulator UI
- Test: AT, ATI, ATZ

➢ **Assignment: Document responses and screenshots of each command**

2: Syntax & Status Commands

- Supported: AT+CSQ, AT+CPIN?, AT+COPS?
- Test signal quality, SIM status, and network operator info
- **Assignment: Extract numeric response (e.g., RSSI) manually or via script**

3: Command Variants & Help

- Check: AT+CMD=?, AT+CMD?, AT+CMD=<val> where applicable

➢ **Assignment: Group 5 supported commands by syntax types**

- Document help-style output manually (emulator does not auto-list all commands)

4: SMS Mode Simulation

- Supported: AT+CMGF=1, AT+CMGS, AT+CMGR
- Send and receive SMS from emulator UI
- **Assignment: Send test message, receive response, and record log**

5: PDP Context & Internet Setup

- Not supported in CelerSMS AT Emulator (skip or mark theoretical)

➢ **Assignment: Research APN configuration with AT+CGDCONT on real modem**

6: Socket Communication

- Not applicable in CelerSMS Emulator (limited to GSM SMS AT commands)

➢ **Assignment: Research AT+QIOPEN/AT+QISEND for any TCP modules of Cavli and create a presentation**

7: Error Simulation & Debugging

- Test AT+CMEE=2 and send unsupported commands to trigger error

➢ **Assignment: Log different error types and write manual response interpretation**

8: Custom AT Commands

- CelerSMS supports limited standard GSM commands only

➢ **Assignment: Research any suitable Cavli product and Propose 3 custom AT commands and mock expected responses manually**

9: Automation

- Automate AT test sequence via emulator’s scripting interface **(if any)**
- If not available, simulate sequence using keyboard macros or external automation

➢ **Assignment: Document flow manually or via batch tool like AutoHotkey**

10: Final Evaluation

- Create scenario with: AT → CSQ → CMGF → CMGS → CMGR

➢ **Assignment: Run full sequence in emulator, take screenshots, and present flow**

Expected Deliverables:

Item	Description
Command Log Sheet	For each step, provide:• Command• Expected and actual response• Notes on success/failure
Screenshots Folder	Screenshot each command execution & response in the emulator. Name them step1_AT.png, step2_CSQ.png, etc.
Flow Diagram	Create a flowchart or linear diagram showing command sequence and logic
Summary Slide/Page	Short summary of what was learned, any errors encountered, and workarounds if applicable.

Notes

- If a command is not supported, document that as part of your testing results.
- All these should be in your git remote repository and issues in Jira with proper reference to doc available in git