

“Why do astronauts prefer the Linux operating system?
Because you can't open Windows in space.”

- Somebody in the internet

Command Recognition

Command Structure

The system implements a **command parsing mechanism** that breaks down user input into structured commands

Command Structure

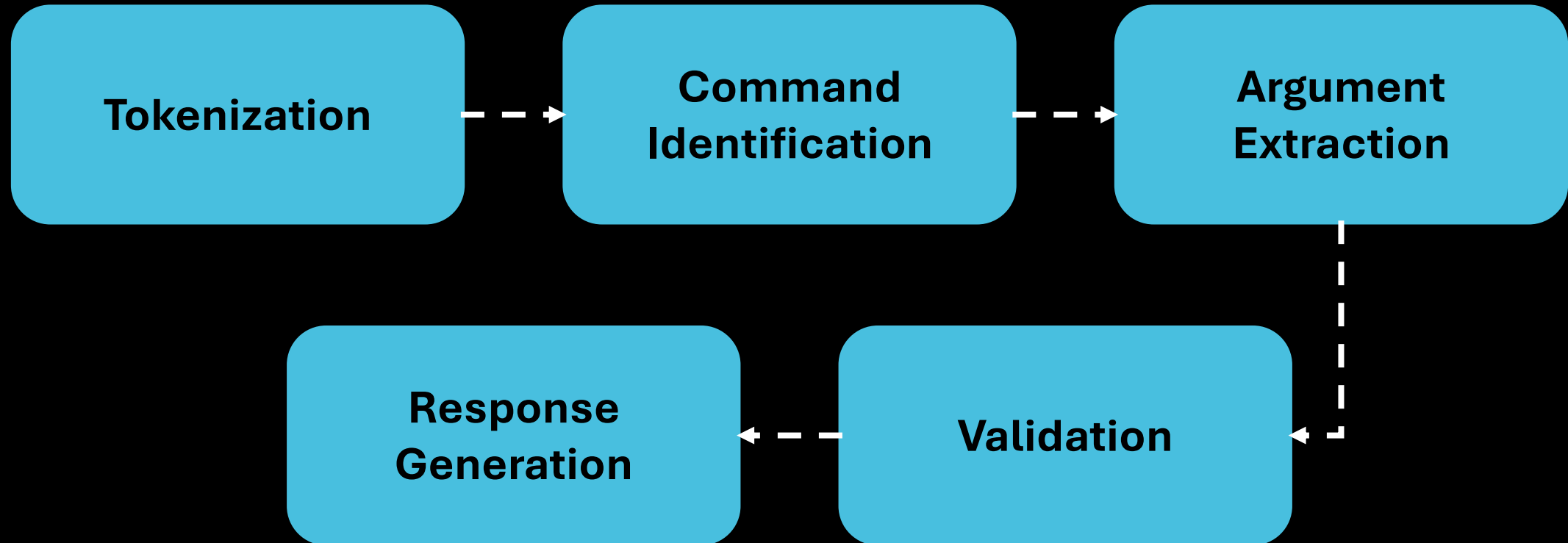
```
typedef struct CommandStruct {  
    std::string command;  
    std::vector<std::string> arguments;  
} Command;
```

Source: CommandHandler.cpp

Supported Commands

Command	Description
<code>help</code>	Display available commands and usage information
<code>start_marquee</code>	Begin marquee text animation
<code>stop_marquee</code>	Stop marquee text animation
<code>set_text <text></code>	Update the marquee display text
<code>set_speed <value></code>	Modify animation speed (milliseconds)
<code>clear/cls</code>	Clear the console screen
<code>status</code>	Show current system status (debug mode)
<code>exit</code>	Terminate the application

Parsing Algorithm



Command Interpreter

Producer

Implemented in
KeyboardHandler

Captures user keyboard input
Enqueues commands for processing
Maintains command history

Consumer

Implemented in
CommandHandler

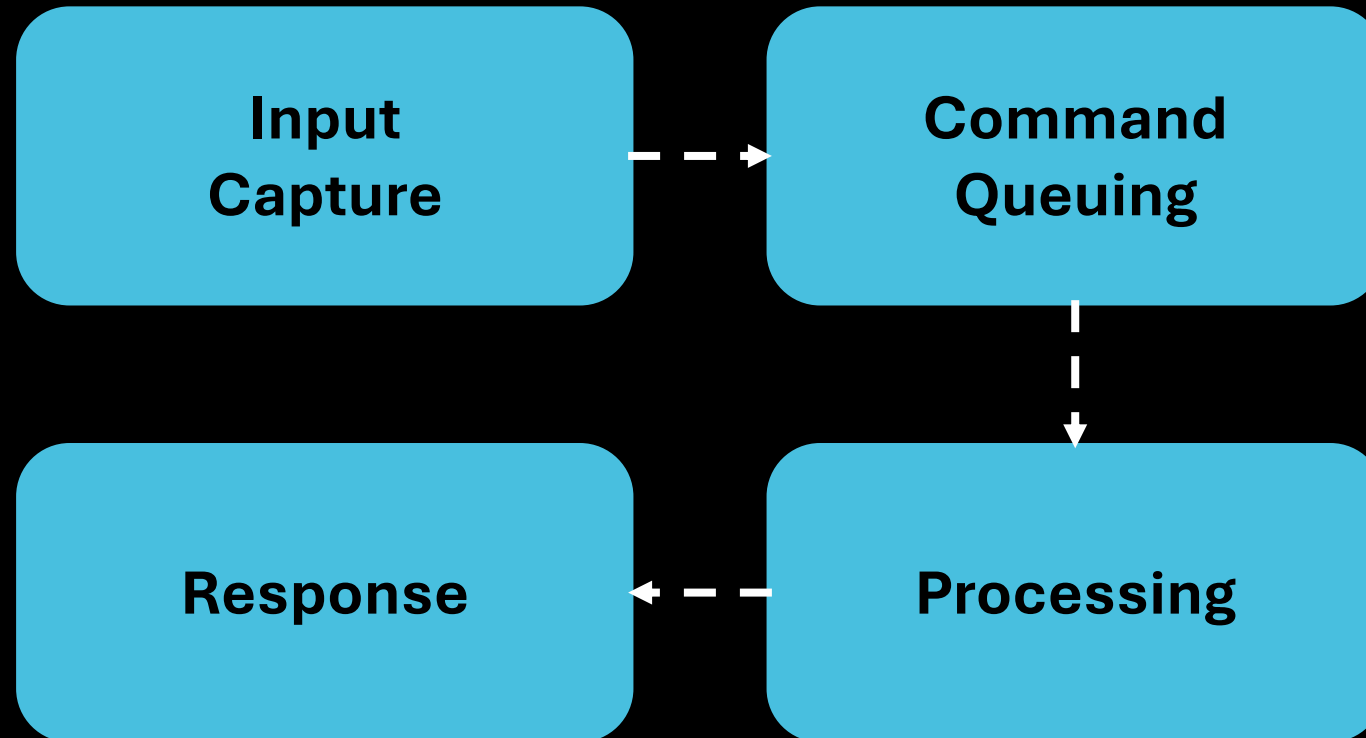
Processes queued commands
Executes command logic
Returns response messages

Thread Safety

```
class CommandHandler {  
private:  
    std::queue<std::string> commandQueue;  
    std::mutex queueMutex;  
  
public:  
    void enqueueCommand(const std::string& commandString) {  
        std::lock_guard<std::mutex> lock(queueMutex);  
        commandQueue.push(commandString);  
    }  
};
```

Source: CommandHandler.cpp

Command Execution Flow



Console UI

Layout



Fig 1. The finalized application UI.

Display Management

```
class DisplayHandler {  
private:  
    HANDLE hConsole;  
    int consoleWidth, consoleHeight;  
    int marqueeHeight, textConsoleHeight;  
    std::vector<std::string> textConsoleLines;  
    SimpleASCIIArt* asciiArt;  
  
public:  
    void updateDisplay();  
    void drawMarqueeSection();  
    void drawTextConsole();  
};
```

Source: DisplayHandler.cpp

System Architecture

Component Responsibilities

