|  |  |
| --- | --- |
| REPORT  OF CIA JAVA PROJECT | Abstract  This project utilizes deep file system manipulation techniques and integrates cutting-edge cryptographic algorithms to ensure data protection and to provide a robust and secure infrastructure.  DEVELOPER  SADAT ALI |

**GITHUB**

[**Sadatali9 (SADAT ALI LARIK) (github.com)**](https://github.com/Sadatali9)

**LINKEDIN**

[**linkedin.com/in/sadat-ali-7216b3275**](https://www.linkedin.com/in/sadat-ali-7216b3275)

**Table of Contents1. COMSATS INTELLIGENCE AGENCY** - Introduction - Responsibilities - Database Security**2. Main Menu Options** - Confidential Agent Profiles - Comprehensive Mission Briefing - Mission Initialization Protocols - Database Information - Continue Mission - Encryption - Decryption - Exit**3. Option 1: Confidential Agent Profiles** - Active-Duty Personnel - Reserve Operatives - Agent Performance Analytics - Agent Financial Ledger**4. Option 2: Comprehensive Mission Briefing** - Total Mission Log - Current Mission Dashboard - Mission Accomplishment Records - Mission Complexity Index**5. Option 3: Mission Initialization Protocols**

- New Mission Addition - Assigning Mission Difficulty - Agent Deployment - Projected Financial Estimate**6. Post Mission Initialization Options** - Encrypt Messages or Documents - Decrypt Messages or Files - Highly Strong Password Generator - (Alert) Delete Any File from Database - Make New Confidential File - Extract Data from Any File - Delete Data from Any File in Database - Search Data from File - Information of All Files in Database**7. Encryption Algorithms**

- ROT13 - ONESHIFT - FIVESHIFT - REVERSESHIFT - DOUBLESHIFT - CUSTOMIZE CIPHER - DIGITAL SIGNATURE GENERATOR (MAC) - ENCRYPT ADVANCE CRYPTO (HMAC)**8. Decryption Algorithms** - ROT13 - ONESHIFT - FIVESHIFT - REVERSESHIFT - DOUBLESHIFT - CUSTOMIZE CIPHER - DIGITAL SIGNATURE GENERATOR (MAC) - DECRYPT ADVANCE CRYPTO (HMAC)**9. Option 4: Database Information** - File Information - File Permissions**10. Option 5: Continue Mission** - Continue Existing Missions**11. Option 6: Encryption** - Encryption Algorithms**12. Option 7: Decryption** - Decryption Algorithms**13. Option 8: Exit**

- Terminate Program**14. Flowchart for Main Program** - Flowchart Illustration**15. Code**

**16.Output**

**COMSATS INTELLIGENCE AGENCY**

**CIA**

COMSATS Intelligence Agency is a secret agency, also known as CIA, working in a country. It is responsible to maintain peace and justice in the society secretly and runs secret missions in the country. It has several undercover agents that are actively participating in the missions throughout the country. It has a huge database of past missions, agents’ stats and agents’ finance etc. This database is secure as it saves the data in encrypted form and it is almost impossible to access this data without proper credentials.

When you open this system, you will get 8 options as follows:

* Confidential Agent Profiles
* Comprehensive Mission Briefing
* Mission Initialization Protocols
* Database information
* Continue mission
* Encryption
* Decryption
* Exit

After the user selects one option, he will get further options:

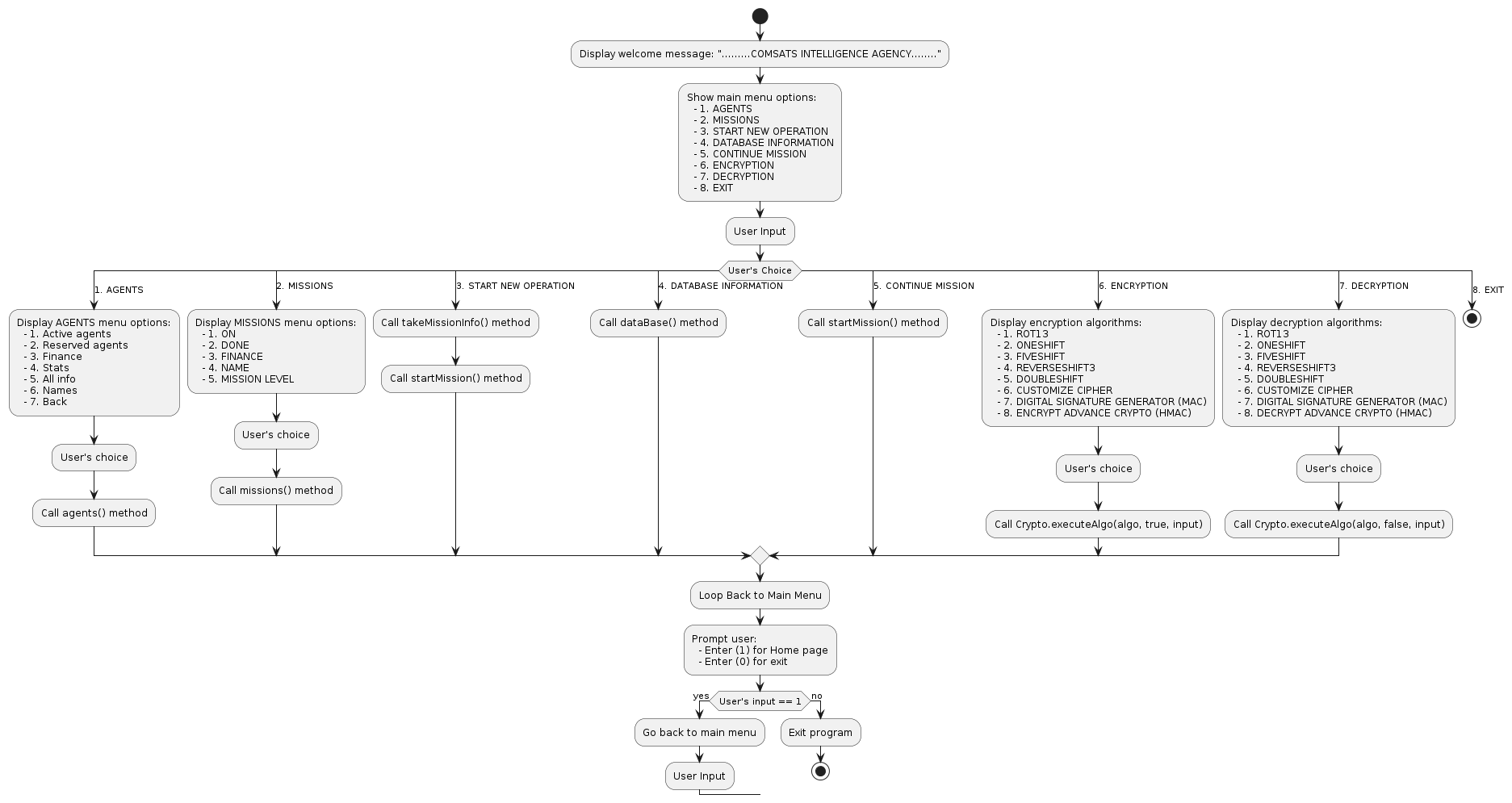
* If the user selects option 1, then he will get the following options:

**CONFIDENTIAL AGENT PROFILES**

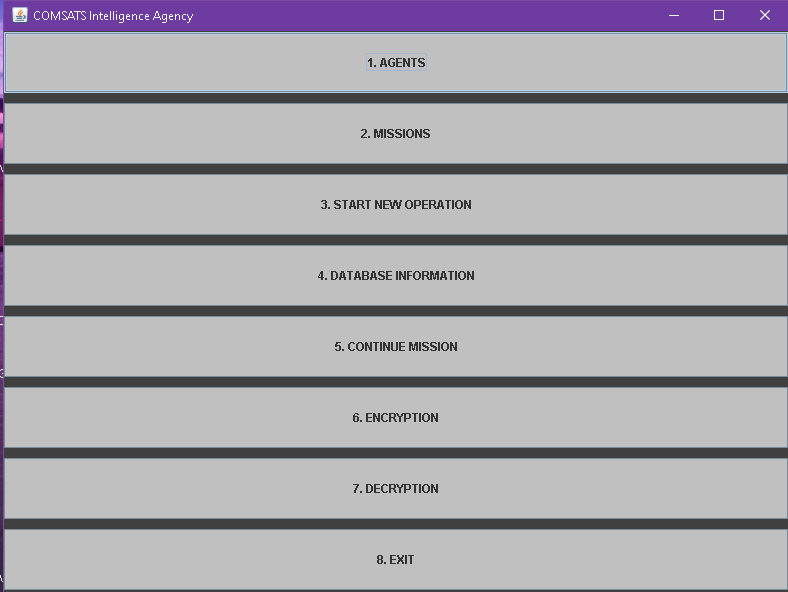
* **Active-Duty Personnel**This function will give the information about the active agents that are participating in missions.
* **Reserve Operatives**This function will give us the information about the agents that are reserved.
* **Agent Performance Analytics**This function will give us the information about the stats and history of agents.
* **Agent Financial Ledger**This function will give us the information about the finance of agents.
* If the user selects option 2, then he will get the following options:  
    
   **COMPREHENSIVE MISSION BREIEFING**
* **Total Mission Log**This function will tell us about the missions that are completed and have been running.
* **Current Mission Dashboard  
  T**his function will tell us about the missions that are in progress.
* **Mission Accomplishment Records**This function will tell us about the missions that have been completed.
* **Mission Complexity Index**This function will tell us the index of difficulty of a mission.
* If the user selects option 3, then he will get the following options:  
    
   **MISSION INITIALIZATION PROTOCOLS**
* **New Mission Addition**This function will help us to initiate a new mission.
* **Assigning Mission Difficulty**This function will help us to assign the index of difficulty to accomplish the mission.
* **Agent Deployment**This function will help us to deploy the agents for that particular mission.
* **Projected Financial Estimate**This function will tell us the cost to run that particular mission by taking information from the user and after this the mission will start.   
  After the mission starts, the following options will appear:
* **ENCRYPT MESSAGES OR DOCUMENTS**This function will encrypt the messages or documents and there are 9 algorithms available for this purpose that the user can select.
* **DECRYPT THE MESSAGE OR FILES**This function will decrypt the messages or files and there are also 9 algorithms available for this purpose.
* **HIGHLY STRONG PASSWORD GENERATOR.**This function will create a highly strong password for the user.
* **(ALERT)DELETE ANY FILE FROM DATABASE**This function will delete any file from the database and it is not possible to recover it.
* **MAKE NEW CONFIDENTIAL FILE.**This function will create a new confidential file as per the instructions of the user.
* **EXTRACT DATA FROM ANY FILE**This function will extract data from any file.
* **DELETE DATA ANY FILE FROM DATABASE**This function will delete the data of a file selected by the user.
* **SEARCH DATA FROM FILE**This function will search data from the file.
* **INFORMATION OF ALL FILES OF DATABASE**This function will give the user all files information in the database.The user will enter his/her choice and the function will start working.   
  There are 9 algorithms for encryption and decryption which are listed below:
* **ROT13**  
  ROT13 is a simple letter substitution cipher that replaces a letter with the 13th letter after it in the alphabet. For example, 'A' becomes 'N', 'B' becomes 'O', and so on. It was popularized as a simple encryption method for online communication or for hiding spoilers in plain sight. It's a symmetric cipher, meaning that the same algorithm is used for both encryption and decryption. Since ROT13 shifts each letter by the same amount, it's considered weak encryption and isn't suitable for securing sensitive information. It's more of a fun or casual way to obscure text rather than a serious encryption technique.
* **ONESHIFT**In this cipher, each letter of the alphabet is shifted by one position. For example, 'A' becomes 'B', 'B' becomes 'C', and so on. This is a type of Caesar cipher, which is one of the simplest and most widely known encryption techniques. Like ROT13, it's also a symmetric cipher, meaning that the same algorithm is used for both encryption and decryption. However, like ROT13, it's not considered secure for protecting sensitive information.
* **FIVESHIFT**  
  A "five-shift" cipher is a variation of the Caesar cipher, where each letter of the alphabet is shifted by five positions. For instance, 'A' becomes 'F', 'B' becomes 'G', and so forth. Like other Caesar ciphers, it's a symmetric encryption method, meaning that the same algorithm is used for both encryption and decryption. While it's slightly more complex than a one-shift cipher, it's still relatively simple and not considered secure for protecting sensitive information against modern cryptographic attacks.
* **REVERSESHIFT**  
  A "reverse shift" cipher, also known as a "reverse Caesar cipher" or "reverse ROT," is a type of encryption where each letter of the alphabet is shifted backwards by a certain number of positions. For example, in a reverse shift of one position, 'A' becomes 'Z', 'B' becomes 'A', 'C' becomes 'B', and so on.
* **DOUBLESHIFT**  
  A "double-shift" cipher, also known as a "double Caesar cipher," involves applying two shifts to each letter of the alphabet. For example, you might shift each letter forward by two positions, meaning 'A' becomes 'C', 'B' becomes 'D', and so on. Then, this shifted alphabet is shifted again by another fixed amount.
* **CUSTOMIZE CIPHER**  
  Customized ciphers are unique encryption methods crafted through tailored modifications to traditional cryptographic techniques. These alterations can include changes to the alphabet, shift values, or encryption procedures, resulting in personalized encryption schemes designed to suit specific needs or preferences.
* **DIGITAL SIGNATURE GENERATOR(MAC)**On a Mac, you can generate a digital signature using the built-in Keychain Access app or the Preview app. With Keychain Access, you can utilize certificates stored in your Keychain to sign documents, while Preview allows you to create a digital signature by capturing it with your trackpad or importing an image. These tools provide convenient ways to authenticate and sign documents digitally, offering flexibility and ease of use for various signing needs on macOS.
* **ENCRYPT ADVANCE CRYPTO(HMAC)**Advanced Encryption Standard (AES) is a symmetric encryption algorithm widely adopted for securing sensitive data due to its robustness and efficiency. HMAC (Hash-Based Message Authentication Code) is a cryptographic mechanism used for verifying the integrity and authenticity of data. By combining AES encryption with HMAC, a higher level of security is achieved, as AES ensures confidentiality while HMAC provides data integrity and authenticity verification. This combination is commonly employed in various applications, including secure communication protocols and data storage systems, to safeguard against unauthorized access and tampering.
* **DECRYPT ADVANCE CRYPTO (HMAC)**  
  Decrypting data encrypted with HMAC (Hash-Based Message Authentication Code) typically involves verifying the integrity and authenticity of the encrypted data using the HMAC key and algorithm, followed by decrypting the data using an appropriate decryption algorithm such as AES (Advanced Encryption Standard) if the data was symmetrically encrypted. HMAC ensures that the decrypted data has not been tampered with during transmission or storage by comparing the calculated HMAC value with the received HMAC value. If the HMAC verification is successful, the data is considered authentic and can be decrypted using the appropriate decryption key and algorithm.
* If the user selects option 4, then he will get the following options:  
   **DATABASE INFORMATION**

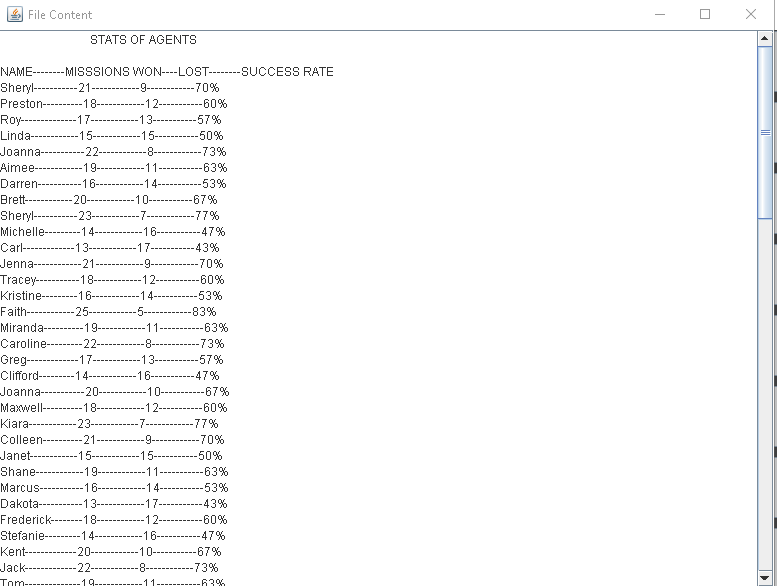
If the user selects this option, then he can get the information of all the files present in the database like the bytes of the files, permissions like whether it is readable or writable or not.

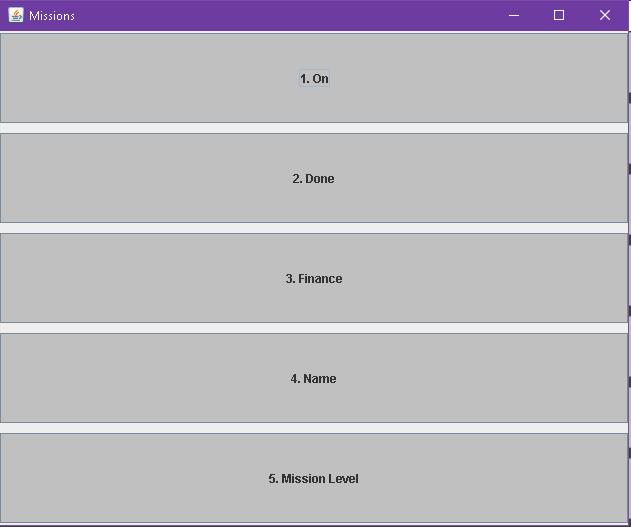
* If the user selects option 5, then he will get the following options:  
    
   **CONTINUE MISSION**By selecting this option, the user can continue the mission which is already in progress without initiating a new mission.
* If the user selects option 6, then he will get the following options:   
    
   **ENCRYPTION**The user can encrypt his messages or passwords by using this option for security reasons. There are further mechanisms to do this which we have already explained before. Here are the names:
* ROT13
* ONESHIFT
* FIVESHIFT
* REVERSESHIFT
* DOUBLESHIFT
* CUSTOMIZE CIPHER
* DIGITAL SIGNATURE GENERATOR(MAC)
* ENCYPT ADVANCE CRYPTO(HMAC)
* DECRYPT ADVANCE CRYPTO (HMAC)
* If the user selects option 7, then he will get the following options:  
    
   **DECRYPTION**  
  The user can select this option to decrypt a message or a file by using 9 algorithms which are explained before. Here are the names again:
* ROT13
* ONESHIFT
* FIVESHIFT
* REVERSESHIFT
* DOUBLESHIFT
* CUSTOMIZE CIPHER
* DIGITAL SIGNATURE GENERATOR(MAC)
* ENCYPT ADVANCE CRYPTO(HMAC)
* DECRYPT ADVANCE CRYPTO (HMAC)
* If the user selects option 8, then he will get the following options:  
   **EXIT**By selecting this option, the user can terminate the program and the app will be closed.

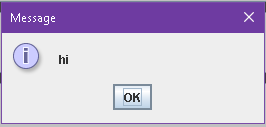
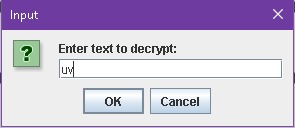
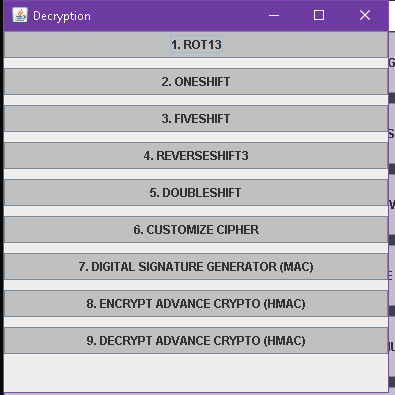
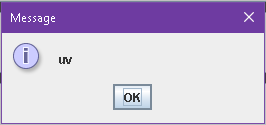
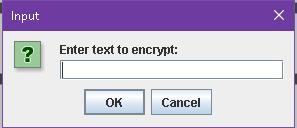
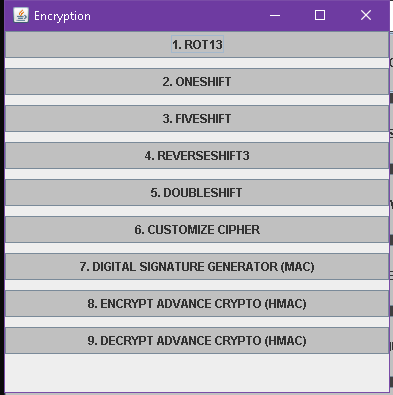
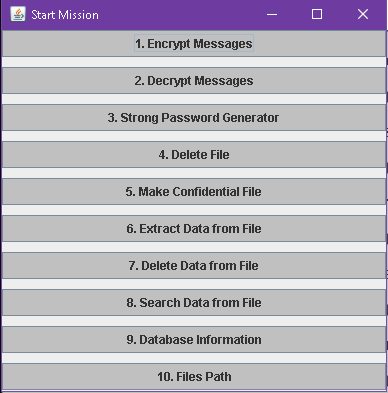
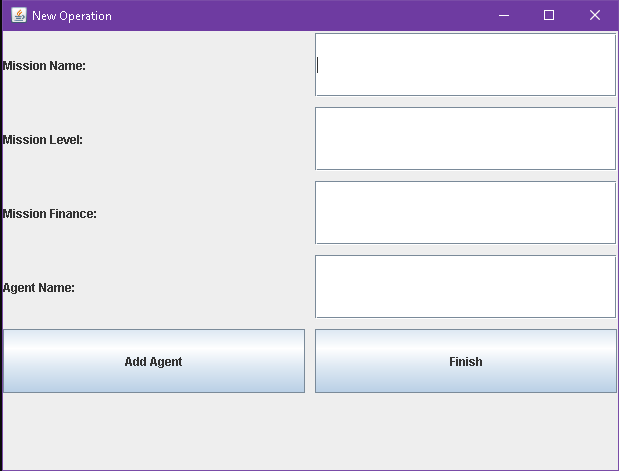
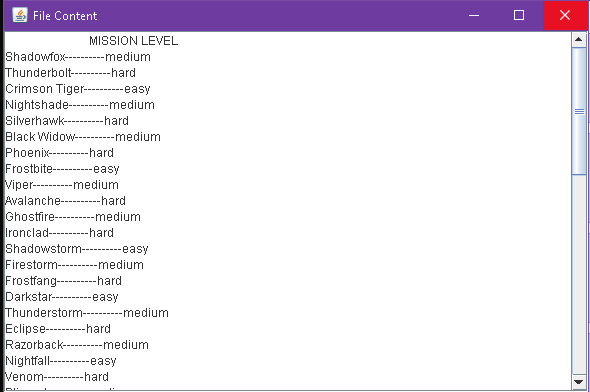
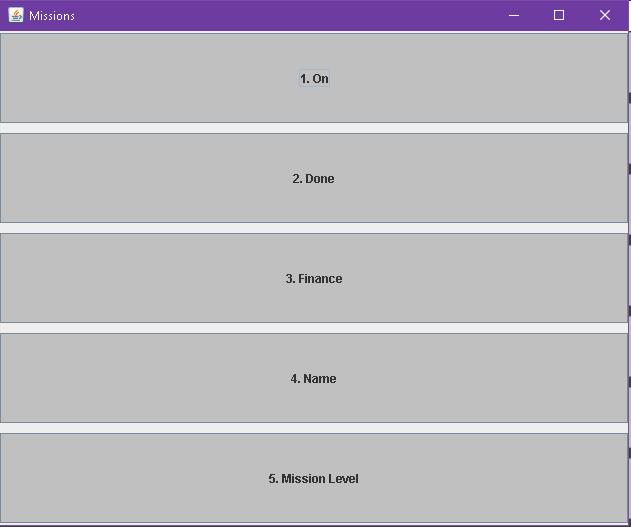
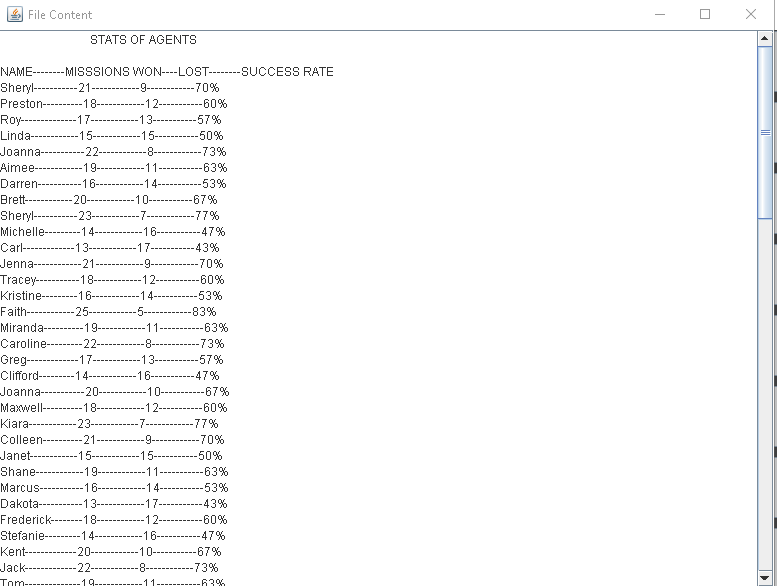
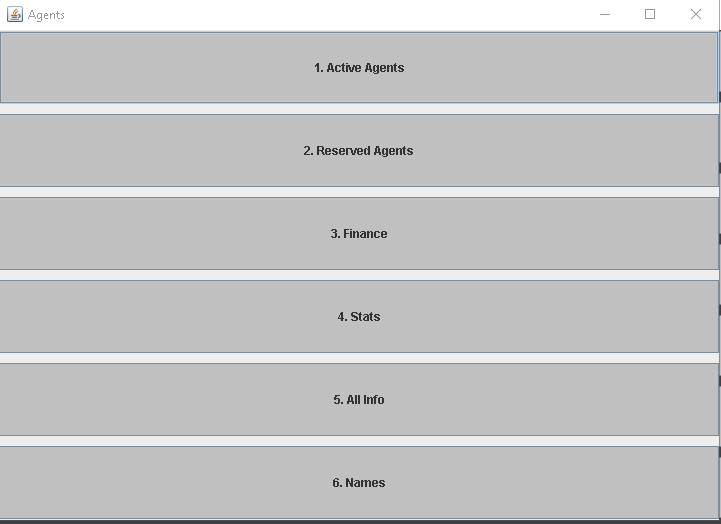
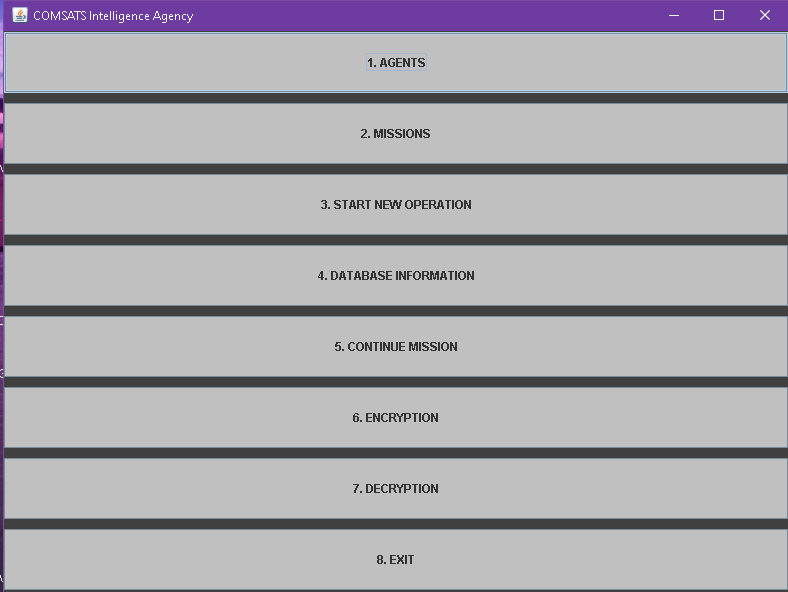
**FLOWCHART:**

**OUTPUT:**

****

****

****

****