# Textual Use Case

Project Aardvark

Note: For A -> B (Extensions), A extends B, and B is extension of A

For A -> B (Inclusions), B includes A, and A is inclusion of B

|  |  |
| --- | --- |
| Classical Cryptography Breaking Tools(1) | |
| Relations | 1. User (Direct) 2. User’s notes (1.1), Extends 3. Save progress (1.2), Extends 4. Read file (1.3), Extends 5. Undo Changes (1.4), Extends 6. Reset cipher text (1.5), Extends 7. Transpose (1.6), Extends 8. Shift Cipher (1.7), Extends 9. Block substitution (1.8), Extends 10. Substitute Character (1.9), Extends 11. Rearrange cText (1.10), Extends 12. Permutate (1.11), Extends 13. Calculate IC (1.12), Includes 14. Generate Letter Distribution Graph (1.13), Includes 15. Print Cipher Text (1.14), Includes 16. Suggestion System (1.15), Includes |
| Actor | User |
| Goal | Access Cryptography Breaking tools of a crypto project accordingly |
| Main Scenario | 1. User chooses an existing project from the front page 2. Application UI transitions to project view 3. User applies changes to the cipher text |
| Alternative Scenario | - |
| Assumptions | - |

|  |  |
| --- | --- |
| User’s notes(1.1) | |
| Relations | Classical Cryptography Breaking Tools, Extension |
| Actor | User |
| Goal | Lets user take notes on a crypto project |
| Main Scenario | 1. User taps “notes” icon 2. User types in notes 3. User clicks save to store the note |
| Alternative Scenario | Alt 3 The user clicks cancel  3.1 The program discards any changes to the note |
| Assumptions | - |

|  |  |
| --- | --- |
| Save progress(1.2) | |
| Relations | Classical Cryptography Breaking Tools, Extension |
| Actor | User |
| Goal | Saves current crypto project state into the device |
| Main Scenario | 1. User taps “save" icon 2. The system saves the current state of the project |
| Alternative Scenario | - |
| Assumptions | Assuming the device storage is not corrupted |

|  |  |
| --- | --- |
| Read file (1.3) | |
| Relations | Classical Cryptography Breaking Tools, Extension |
| Actor | User |
| Goal | Read a cipher text from the device |
| Main Scenario | 1. User taps “read cipher text file” icon 2. User selects a new file to read from through file browser 3. Redo the whole suggestion algorithm, frequency graph, and IC calculation on read new file |
| Alternative Scenario | Alt 3 The user clicks cancel  3.1 The system does not redo the whole algorithms mentioned in the main scenario |
| Assumptions | Assuming the project has already been created before |

|  |  |
| --- | --- |
| Undo changes(1.4) | |
| Relations | Classical Cryptography Breaking Tools, Extension |
| Actor | User |
| Goal | Reverts the state of the cipher text to before the latest change |
| Main Scenario | 1. User taps “undo" icon 2. Program reverts back the state of the ciphertext |
| Alternative Scenario | Alt 2 there is no more state to revert back to  2.1 Program displays error message |
| Assumptions | - |

|  |  |
| --- | --- |
| Reset cipher text(1.5) | |
| Relations | Classical Cryptography Breaking Tools, Extension |
| Actor | User |
| Goal | Reverts the cipher text to its original state |
| Main Scenario | 1. User taps “reset” icon 2. System reverts the cipher text to its original state |
| Alternative Scenario | - |
| Assumptions | - |

|  |  |
| --- | --- |
| Transpose(1.6) | |
| Relations | 1. Classical Cryptography Breaking Tools, Extension 2. Transpose By N (1.6.1), Extends |
| Actor | User |
| Goal | Apply specific transposition algorithm to the cipher text |
| Main Scenario | 1. User selects transposition button 2. User enters key 3. Apply transposition algorithm to the cipher text 4. Program saves the new state to “Change history” 5. The program re calculates the IC and frequency graph accordingly |
| Alternative Scenario | Alt 2 Invalid input or empty input  2.1 Displays error message  2.2 prompt user to reenter key.  Alt 4 User does not enter any key or taps “cancel"  4.1 Program does not save anything to “Change history" |
| Assumptions | - |

|  |  |
| --- | --- |
| Shift Cipher(1.7) | |
| Relations | Classical Cryptography Breaking Tools, Extension |
| Actor | User |
| Goal | Apply specific Shift cipher algorithm to the cipher text |
| Main Scenario | 1. User selects Shift cipher button 2. User chooses number of shift to apply 3. Apply Shift right or Shift left to the cipher text accordingly 4. Program saves the new state to “Change history” 5. The program re calculates the IC and frequency graph accordingly |
| Alternative Scenario | Alt 2 Invalid input or empty input  2.1 Displays error message  2.2 prompt user to reenter key.  Alt 4 User does not enter any key or taps “cancel"  4.1 Program does not save anything to “Change history" |
| Assumptions | - |

|  |  |
| --- | --- |
| Block substitution(1.8) ???? | |
| Relations | Classical Cryptography Breaking Tools, Extension |
| Actor | User |
| Goal |  |
| Main Scenario |  |
| Alternative Scenario |  |
| Assumptions |  |

|  |  |
| --- | --- |
| Substitute Character(1.9) | |
| Relations | 1. Classical Cryptography Breaking Tools, Extension 2. By Keyword (1.9.1), Extends 3. By Character (1.9.2), Extends |
| Actor | User |
| Goal | Apply specific Substitution algorithm to the cipher text |
| Main Scenario | 1.9.1 By Keyword   1. User clicks “substitution by keyword” button 2. User enters the keyword 3. Apply substitution by keyword to the cipher text 4. Program saves the new state to “Change history” 5. The program re calculates the IC and frequency graph accordingly   1.9.2 By Character   1. User picks a character, and a character to substitute with 2. User clicks “apply" button 3. Apply substitution by character to the cipher text 4. Program saves the new state to “Change history” 5. The program re calculates the IC and frequency graph accordingly |
| Alternative Scenario | (1.9.2) Alt 2 Invalid input or empty input  2.1 Displays error message  2.2 prompt user to reenter key.  (both) Alt 4 User does not enter any key or taps “cancel"  4.1 Program does not save anything to “Change history" |
| Assumptions | - |

|  |  |
| --- | --- |
| Rearrange cText(1.10) | |
| Relations | 1. Classical Cryptography Breaking Tools, Extension 2. Character per word (1.10.1), Extends 3. Word per line (1.10.2), Extends |
| Actor | User |
| Goal | Re arrange the cipher text view |
| Main Scenario | 1.10.1 Character per word   1. User picks number of character per word 2. The cipher text updates accordingly   1.10.2 Word per line   1. User picks word per line 2. The cipher text updates accordingly |
| Alternative Scenario | - |
| Assumptions | - |

|  |  |
| --- | --- |
| Permutate (1.11) | |
| Relations | Classical Cryptography Breaking Tools, Extension |
| Actor | User |
| Goal | Displays permutations of the cipher text |
| Main Scenario | 1. User presses “permutation” button 2. User enters key value for permutation 3. Program displays permutations for the text accordingly |
| Alternative Scenario | - |
| Assumptions | - |

|  |  |
| --- | --- |
| Calculate IC (1.12) | |
| Relations | 1. Classical Cryptography Breaking Tools, Inclusion 2. Find Period (1.12.1), Includes |
| Actor | - |
| Goal | Calculates IC of current cipher text state |
| Main Scenario | 1. The program calculates the IC of current cipher text 2. Then it displays the IC result accordingly   1.12.1 Find Period   1. The program calculates the IC of current cipher text with period 2 to N applied 2. Then it displays the IC results (with the period applied) accordingly |
| Alternative Scenario | - |
| Assumptions | * The project is newly created / already exist * The cipher text value is not null |

|  |  |
| --- | --- |
| Generate Letter Distribution Graph (1.13) | |
| Relations | 1. Classical Cryptography Breaking Tools, Inclusion 2. Letter Frequency period of N (1.13.1), Includes |
| Actor | User |
| Goal | Generate Letter distribution graph |
| Main Scenario | 1. The program generates Normal English letter distribution graph   1.13.1 Letter Frequency period of N  1. The program generates Letter frequency of current cipher text state according to the value of N (where N is user’s input) |
| Alternative Scenario | Alt 1 The user chooses another language letter distribution graph  1.1 The program generates letter distribution graph of the user’s choosing |
| Assumptions | * The project is newly created / already exist * The cipher text value is not null |

|  |  |
| --- | --- |
| Print Cipher Text (1.14) | |
| Relations | Classical Cryptography Breaking Tools, Inclusion |
| Actor | User |
| Goal | Print current state of cipher text and its original state |
| Main Scenario | 1. The program displays current state of the cipher text 2. Then, it displays the original state of the cipher text |
| Alternative Scenario | Alt 2 The original cipher text value is null  2.1. Prompt the user to enter a cipher text  2.2. Set original cipher text to user’s input, and set current cipher text to original state |
| Assumptions | * The project is newly created / already exist * The original state of the cipher text is displayed on demand |

|  |  |
| --- | --- |
| Suggestion System (1.15) | |
| Relations | Classical Cryptography Breaking Tools, Inclusion |
| Actor | - |
| Goal | Predict what algorithm the original cipher text used, and suggest steps |
| Main Scenario | 1. The program analyses the original cipher text and generates suggestions on how to break it |
| Alternative Scenario | - |
| Assumptions | * The project is newly created / already exist * The cipher text value is not null |

|  |  |
| --- | --- |
| Encryption / Decryption Tools (2) | |
| Relations | 1. User (Direct) 2. Paste to Clipboard (2.1), Extends 3. Copy from clipboard (2.2), Extends 4. Vigenere Cipher (2.3), Extends 5. Transposition Cipher (2.4), Extends 6. Shift Cipher (2.5), Extends 7. Substitution Cipher (2.6), Extends |
| Actor | User |
| Goal | Encrypt or Decrypt text inputted by the user |
| Main Scenario | 1. User goes to navigation menu 2. User clicks on Encrypt / Decrypt feature 3. The application brings the user to Encrypt / Decrypt activity 4. User paste text on clipboard to the input text view by pressing a button 5. User chooses one (or more) cryptographic algorithm to encrypt or decrypt the input text 6. User then copies the (changed) text to the clipboard by pressing a button |
| Alternative Scenario | - |
| Assumptions | - |

|  |  |
| --- | --- |
| Paste to Clipboard (2.1) | |
| Relations | Encryption / Decryption tools, Extension |
| Actor | User |
| Goal | Paste text from clipboard into the text view |
| Main Scenario | 1. User presses “Paste from clipboard” button 2. The program displays text from the clipboard |
| Alternative Scenario | Alt 2 The clipboard contains no text / is null  2.1: Displays error message |
| Assumptions | - |

|  |  |
| --- | --- |
| Copy from clipboard (2.2) | |
| Relations | Encryption / Decryption tools, Extension |
| Actor | User |
| Goal | Copy text from text view to the clipboard |
| Main Scenario | 1. User presses “Copy to clipboard” button 2. The program copies text inside text view |
| Alternative Scenario | Alt 2 The text view has not text / is null  2.1: Displays error message |
| Assumptions | - |

|  |  |
| --- | --- |
| Vigenere Cipher (2.3) | |
| Relations | Encryption / Decryption tools, Extension |
| Actor | User |
| Goal | Apply Vigenere cipher to the input text |
| Main Scenario | 1. User presses vigenere cipher button 2. User enters key 3. Program applies vigenere cipher accordingly |
| Alternative Scenario | Alt 3 Key is invalid or empty  3.1 Displays error message and prompt the user to reenter valid key |
| Assumptions | - |

|  |  |
| --- | --- |
| Tranposition Cipher(2.4) | |
| Relations | Encryption / Decryption tools, Extension |
| Actor | User |
| Goal | Apply Transposition cipher to the input text |
| Main Scenario | 1. User presses Transposition cipher button 2. User enters key 3. Program applies Transposition cipher accordingly |
| Alternative Scenario | Alt 3 Key is invalid or empty  3.1 Displays error message and prompt the user to reenter valid key |
| Assumptions | - |

|  |  |
| --- | --- |
| Shift Cipher(2.5) | |
| Relations | Encryption / Decryption tools, Extension |
| Actor | User |
| Goal | Apply Shift cipher to the input text |
| Main Scenario | 1. The user pick shift cipher key value (from 1 to 25) 2. The user presses the shift button and apply shift cipher to the input text according to the shift key value and the direction |
| Alternative Scenario | - |
| Assumptions | - |

|  |  |
| --- | --- |
| Substitution Cipher(2.6) | |
| Relations | Encryption / Decryption tools, Extension |
| Actor | User |
| Goal | Apply Substitution cipher to the input text |
| Main Scenario | 1. User presses Substitution cipher button 2. User enters key 3. Program applies Substitution cipher accordingly |
| Alternative Scenario | Alt 3 Key is invalid or empty  3.1 Displays error message and prompt the user to reenter valid key |
| Assumptions | - |

|  |  |
| --- | --- |
| Rename Project (3) | |
| Relations | User (Direct) |
| Actor | User |
| Goal | Rename an existing project |
| Main Scenario | 1. User long presses the project title 2. The user then enters a new title 3. User presses confirm button 4. program updates the project title |
| Alternative Scenario | Alt 3 Title already exist, or new title input field is empty  3.1 Displays error message and prompt the user to reenter valid title |
| Assumptions | One or more project already exist in the projects list |

|  |  |
| --- | --- |
| Create new project(4) | |
| Relations | User (Direct) |
| Actor | User |
| Goal | Create new project |
| Main Scenario | 1. User presses “+" (plus) button in the front page to create new project 2. User enters title of new project 3. User presses confirm button 4. Program then adds new project to the list |
| Alternative Scenario | Alt 3 Title already exist, or new title input field is empty  3.1 Displays error message and prompt the user to reenter valid title |
| Assumptions |  |

|  |  |
| --- | --- |
| Delete Project (5) | |
| Relations | User (Direct) |
| Actor | User |
| Goal | Creates new project |
| Main Scenario | 1. User presses “trash" icon at the side of each project title 2. The program displays confirmation pop up 3. The user presses confirm button 4. Program deletes the associated project |
| Alternative Scenario | Alt 3 The user presses cancel button  3.1 Program does not delete the associated project |
| Assumptions | - |