1 About project

This project implements several classical encryption algorithms (Playfair, Winston, Vigenère) using a modular and extensible architecture based on well-established software design patterns.

2 Cipher Logic Overview

2.1 Playfair Cipher

– The message is split into bigrams (letter pairs).
– If a pair contains identical letters, the second is replaced with `Z`.
– If the total length is odd, an `X` is added at the end.
– A 5×5 matrix is built using the first key (I = J).
– Each pair is processed as follows:
Same row → letters shift right
Same column → letters shift down
Rectangle → letters are swapped to opposite corners in the same row

2.2 Winston Cipher

– The message is split into bigrams (letter pairs).
– If a pair contains identical letters, the second is replaced with `Z`.
– If the total length is odd, an `X` is added at the end.
– A 5×5 matrix is built using the first key (I = J).
– A 5×5 matrix is built using the second key (I = J).
– Find the first bigram's letter in at the first table[row1][col1]
– Find the second bigram's letter in at the second table[row2][col2]
– Find result at the first table [row1][col2]
– Find result at the first table [row2][col1]

2.3 Vigenère Cipher

– If two identical letters appear in a row, the second is replaced with `Z`.
– If the message has an odd number of letters, add `X` at the end.
– The key is repeated to match the length of the message.
– Each letter is shifted forward in the alphabet by the alphabetical index of the corresponding key letter.

3 Project structure

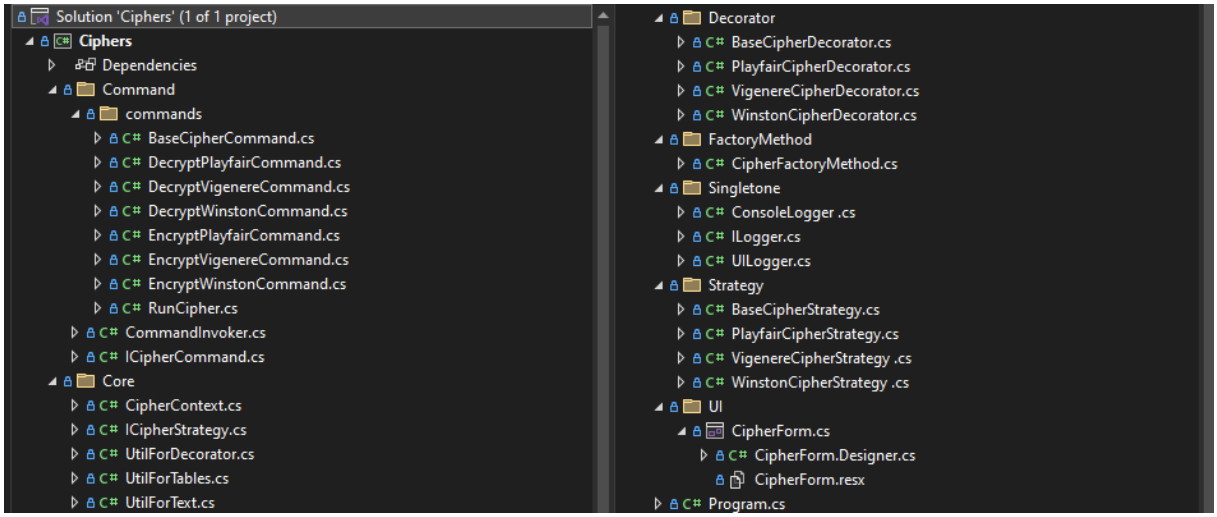

Figure 1: Project structure

3.1 Folder - Strategy

Used in: `ICipherStrategy`, concrete strategy classes.
Purpose: Make ciphers interchangeable.

3.2 Folder - Decorator

Used in: `BaseCipherDecorator`, adds logging.
Purpose: Extend cipher without modifying original.

3.3 Folder - Factory Method

Used in: `CipherFactoryMethod.Create()`.
Purpose: Centralized object creation based on user input.

3.4 Folder - Command

Used in: `ICipherCommand`, `RunCipher`, `CommandInvoker`.
Purpose: Encapsulate cipher action logic.

3.5 Folder - Singleton

Used in: `ConsoleLogger`, `UILogger`.
Purpose: Share a single logger across app.
Note: `UILogger` requires explicit initialization with a form.

3.6 Folder - Core

ICipherStrategy – interface shared by all cipher strategies.
CipherContext – structural holder for strategy and mode (not actively used but supports Strategy-compliant architecture).
UtilForText – provides text preprocessing:
– filters input
– generates bigrams
– repeats key for Vigenère
UtilForTables – table generation and lookup used in Playfair and Winston.
UtilForDecorator – printing tables and debug-formatting for decorators.

3.7 Folder - UI

CipherForm – main Windows Forms class that builds the user interface.
Purpose: collect user input and trigger cipher execution.
Includes:
– `ComboBox` for selecting cipher type and mode (Encrypt/Decrypt)
– `TextBox` inputs for first key, second key, and message
– `Button` to run the cipher process
– `textBoxLog` and `textBoxResults` to display logs and results
Works with: `CipherFactoryMethod`, `RunCipher`, and `UILogger`

4 Ciphers in action

4.1 Playfair

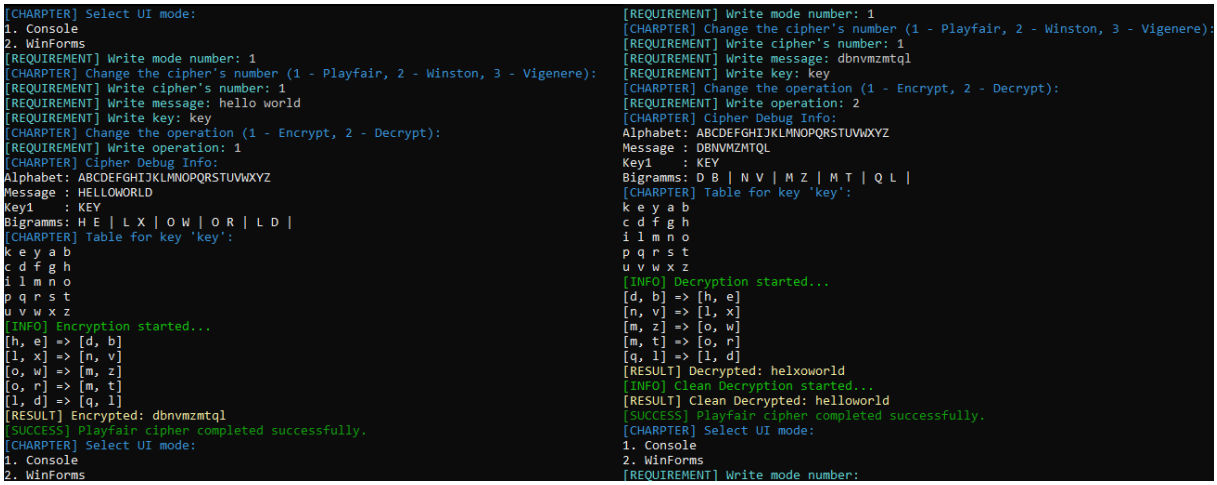

Figure 2: Playfair in console mode

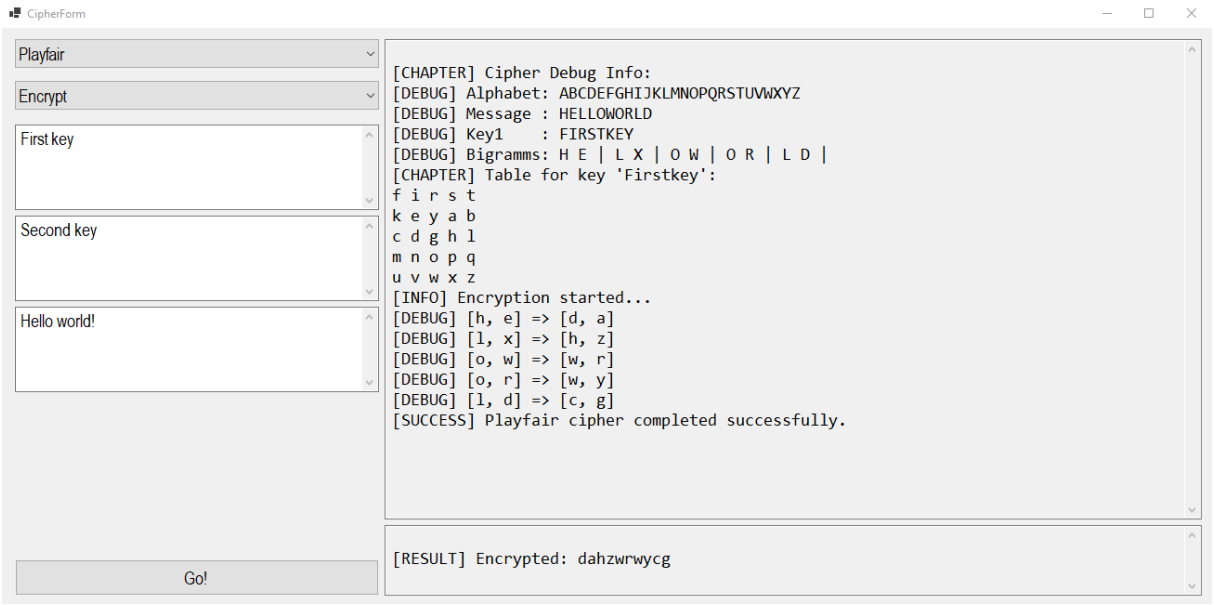

Figure 3: Playfair encrypt in WinForms mode

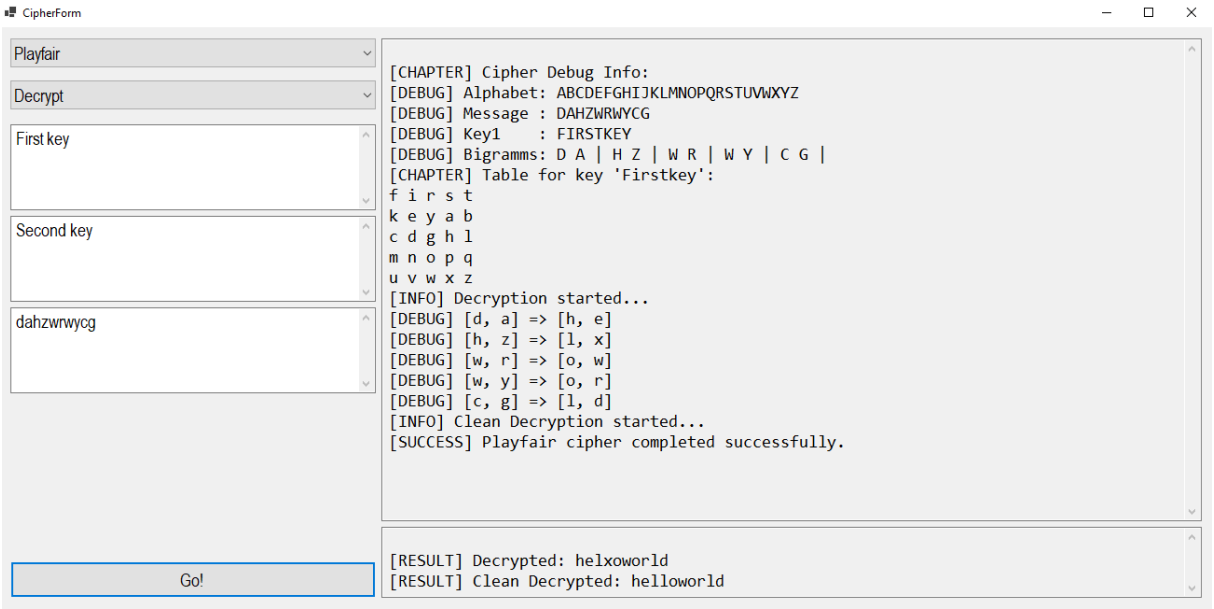

Figure 4: Playfair decrypt in WinForms mode

4.2 Winston

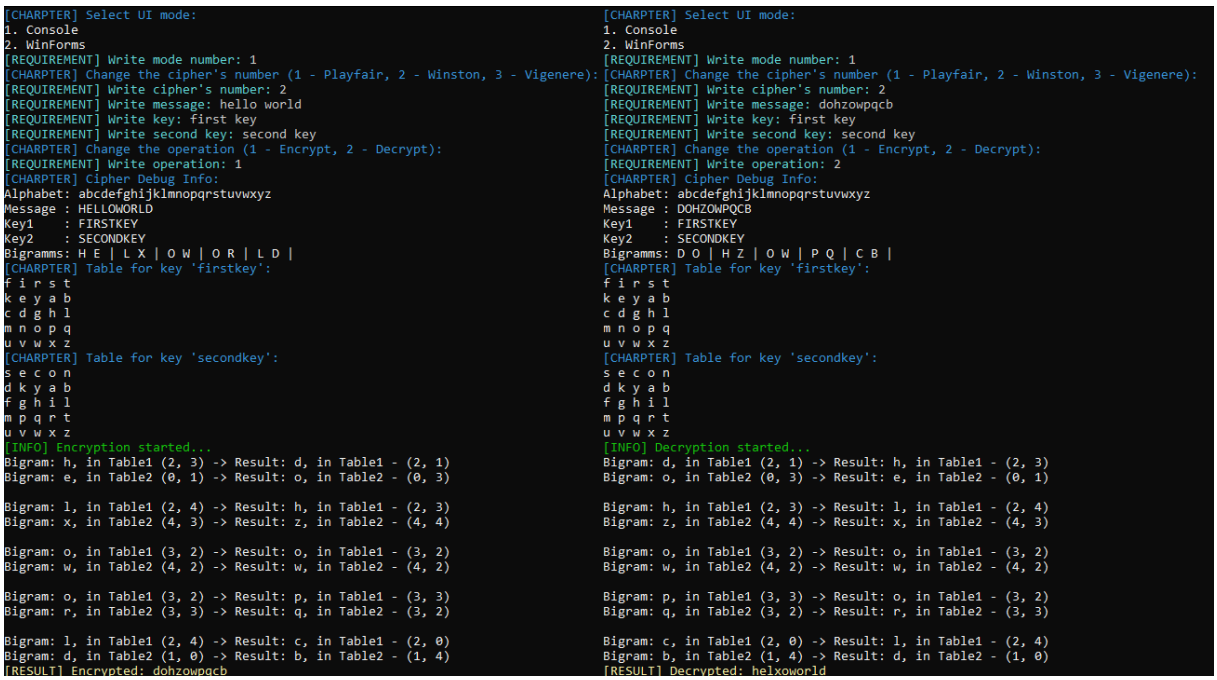

Figure 5: Winston in console mode

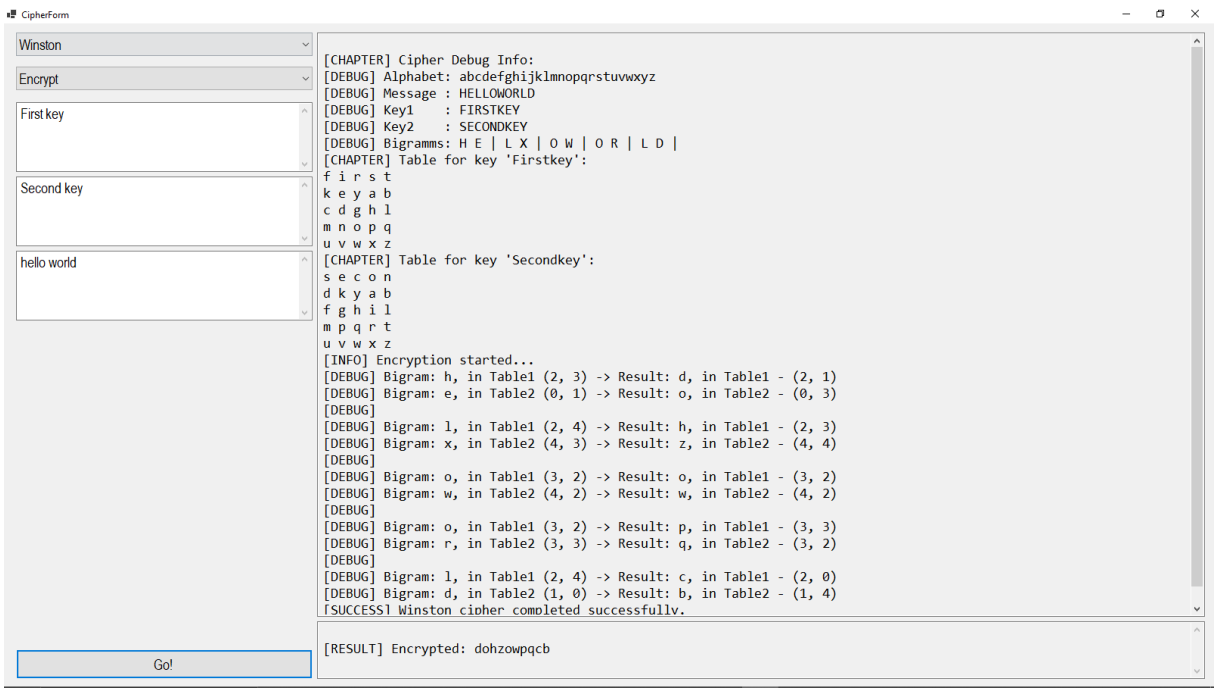

Figure 6: Winston encrypt in WinForms mode

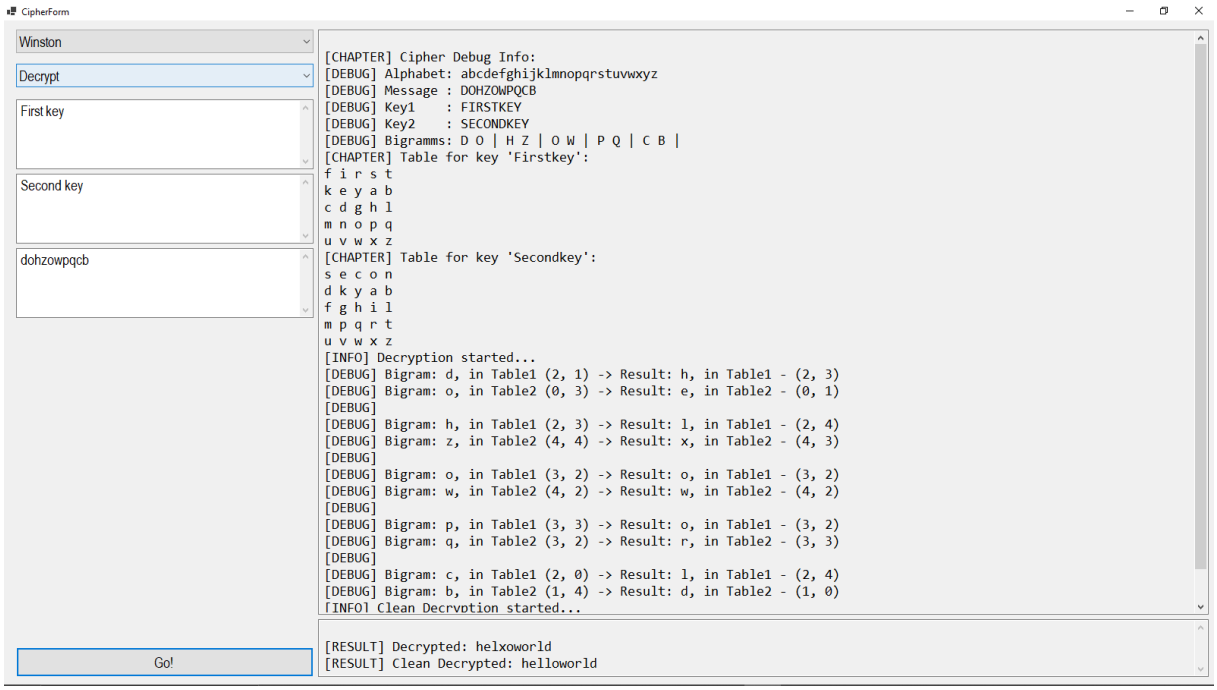

Figure 7: Winston decrypt in WinForms mode

4.3 Vigener

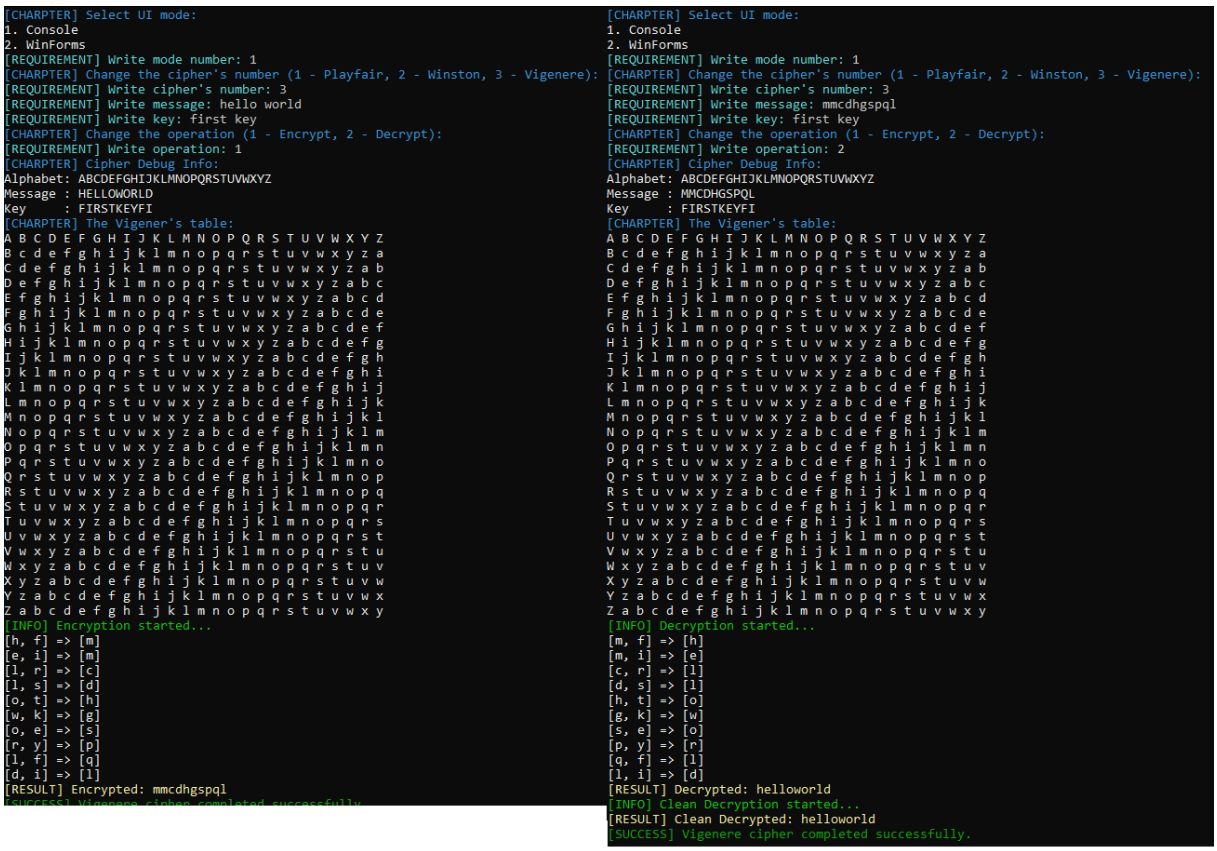

Figure 8: Vigenere in console mode

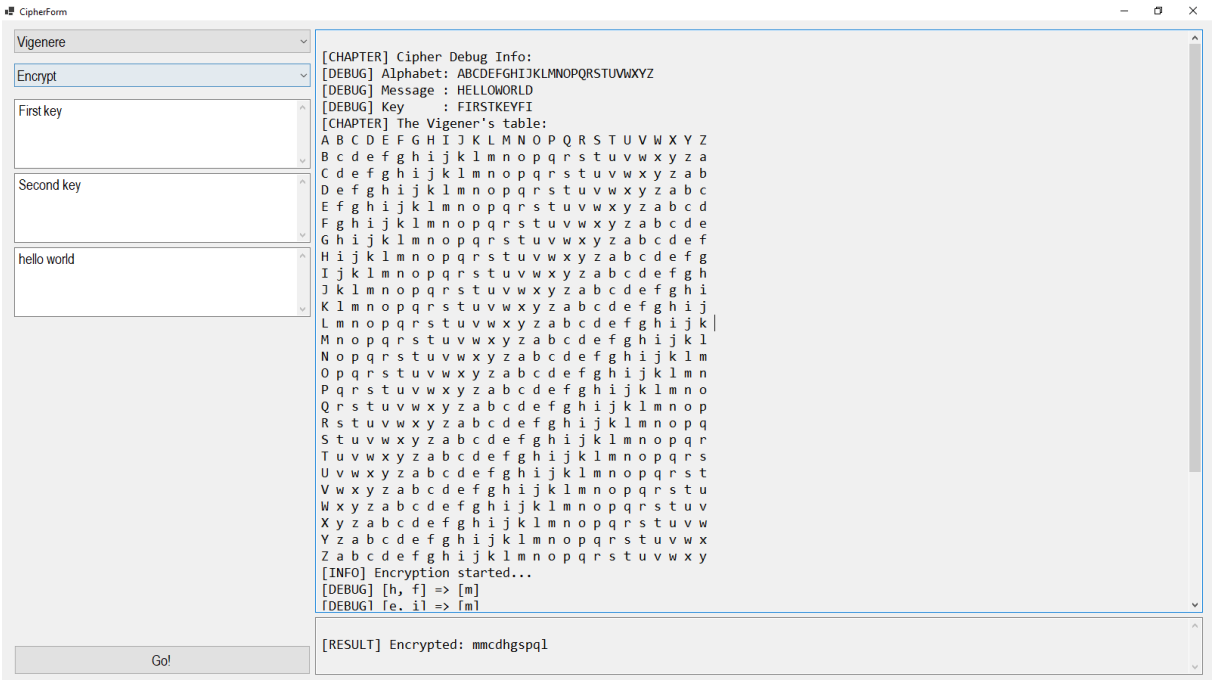

Figure 9: Vigener encrypt part 1 in WinForms mode

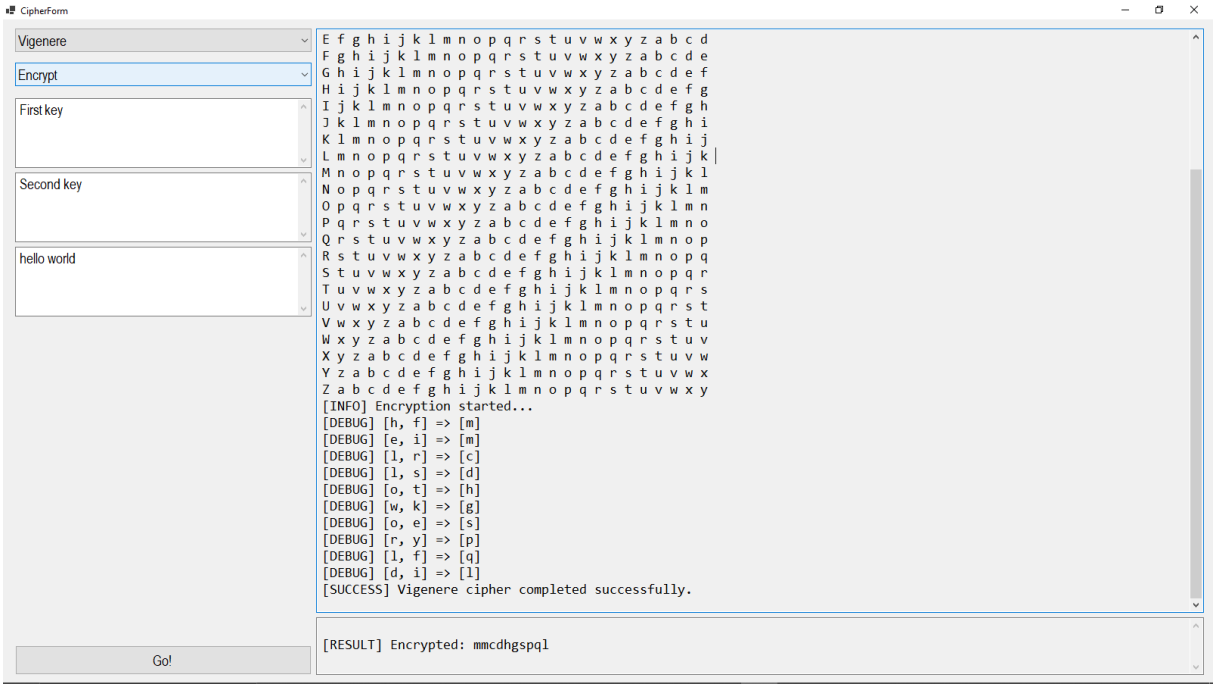

Figure 10: Vigener encrypt part 2 in WinForms mode

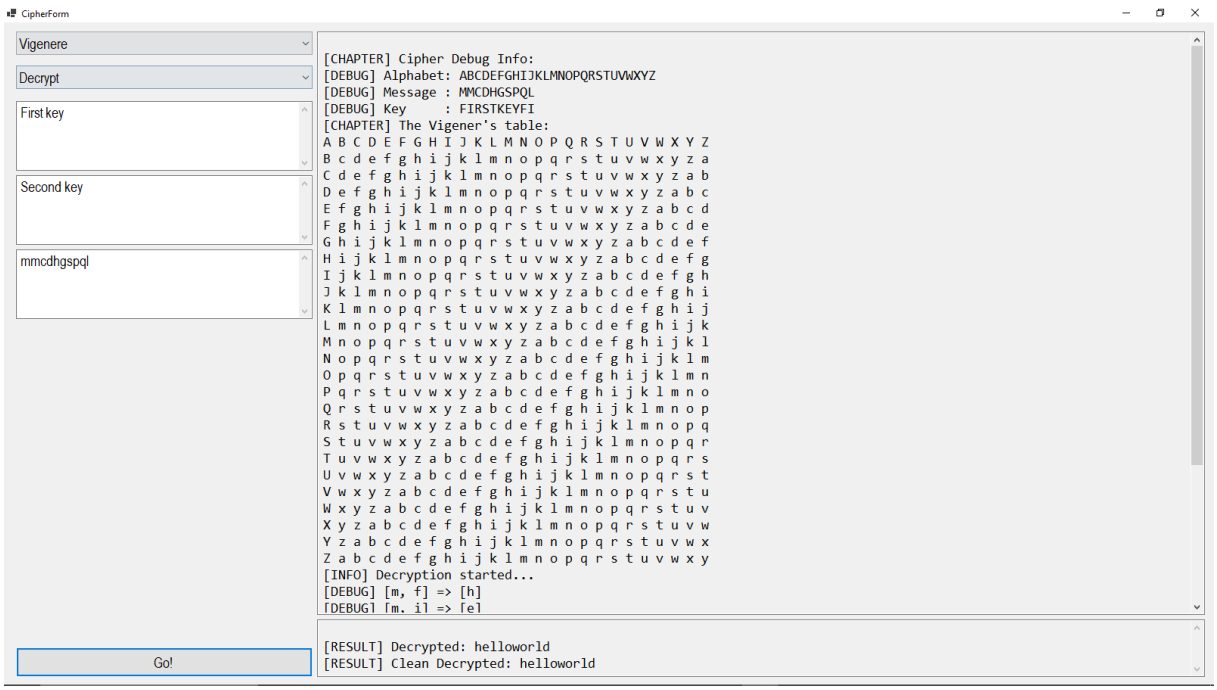

Figure 11: Vigener decrypt part 1 in WinForms mode

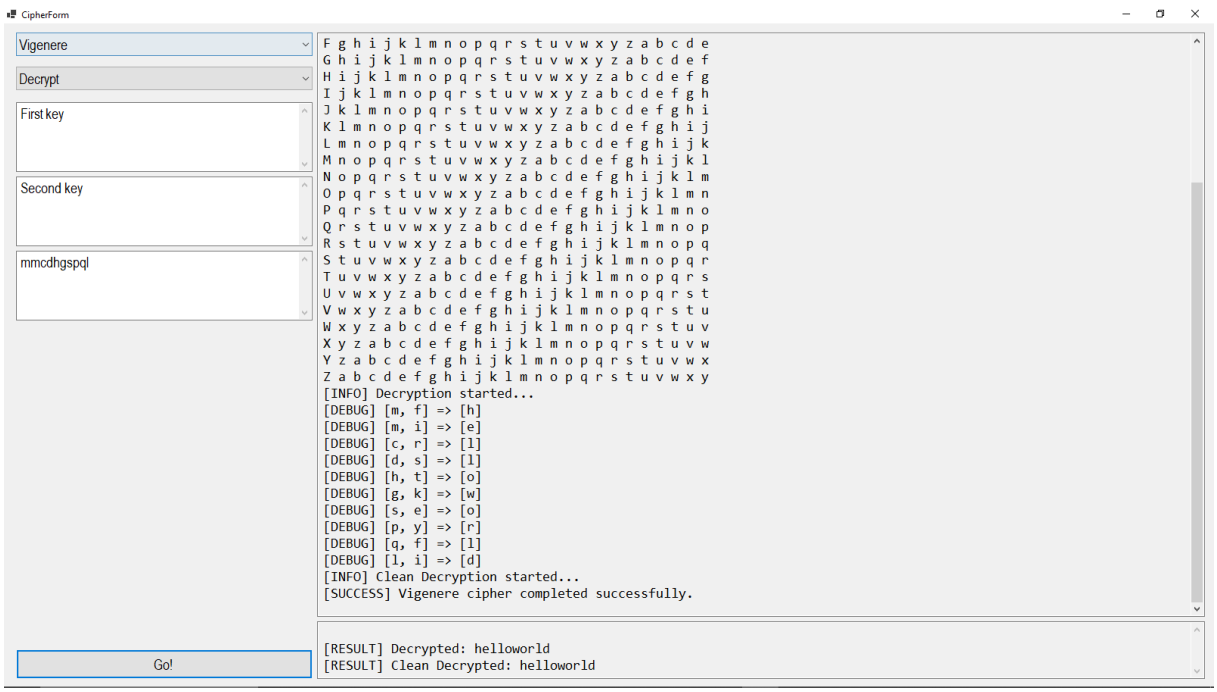

Figure 12: Vigener decrypt part 2 in WinForms mode