Step 4:  
What Does the program do?

Program will display the complete scripture text with the reference. When the user press Enter, it will hide random words in the scripture and if the User type quit, it will end the program.

What user inputs does it have?

User inputs: 1 Pressing Enter and Typing quit.

What output does it produce?

The program outputs the complete scripture, once when the User press Enter, it will display the updated scripture with some words hidden.

How does the program end?

The user type Quit

All words in the scripture are hidden.

Step 5, Determine the classes

* Scripture Class:

Keep track of the reference and the text of the scripture, hide words in the scripture and der the rendered display of the scripture.

* Reference Class:

Keep track of the book, chapter, and verse information.

* Word Class:

Keeps track of a single word and whether it is shown or hidden.

* Program: Main class
  + Handle user input and the overall flow of the program, It will display scriptures.

Step 6, Define Class Behaviors:

* Scripture:
  + HideRandomWord()
  + GetRenderedText()
  + IsCompletlyHidden()
* Reference:
  + Constructor for initializing book, chapter and verse information.
  + ToString() // Converts the reference information to a formatted string
* Word:
  + Hide()
  + Show()
  + IsHidden()
  + GetRenderedText()
* Program:
  + Main()
  + Display the complete scripture.

Step 7 Define class attributes:

* Scripture

Reference \_reference

List<Word>

* Reference:
  + String \_book
  + Int \_chapter

Int \_verseStart

Int \_VerseEnd

* Word:
  + String \_text
  + Bool \_Is hidden // A Boolean to store whether the word is hidden or not
* Scripture Class Constructor:
  + Public Scripture(string referenceText, string text)
* Reference Class Constructor:
  + // a constructor that takes the book, chapter and verse info.
* Word Class Constructor:
  + Public Word(String text): // Initializes the word’s text and sets IsHidden to ‘false’
* Scripture Class Constructors:
  + Public Scripture(string referenceText, string text) // This Constructor Initializes a Scripture object by taking a reference text. Internally store a list of word object for the scripture.
  + // The reference object is created based on the reference text.
* Reference Class Constructor:

Public Reference (String reference Text)

Public Reference (string referenceText, int verseEnd)

* Word Class Constructor:
  + Public Word (string text): // This constructor initialized a word object by taking the text of the word as a parameter. It sets the IsHidden attribute to false.

***DIAGRAMS***

**Reference**

\_book: string

\_chapter: int  
\_verseStart: int

\_verseEnd: int

Reference(referenceText:string)

Reference (referenceText: string, verseEnd:int

ToString(): string

**Scripture**

\_reference: Reference   
\_word: List<Word>  
--------------------------------------------------------

Scripture (reference text: string, txt: string)

HideRandomWord()

GetRenderedText(): string

IsCompletelyHideen(): bool

ToString(): string

\_text: string

\_isHidden: bool

-------------------------------------------------------  
Word(text: string)

Hide()

Show()

IsHidden(): bool

GetRenderedText():string

**Word**