**Graphical programming language application:**

Designing the visual composition and temporal behavior of a GUI is an important part of software application programming in the area of human–computer interaction. Its goal is to enhance the efficiency and ease of use for the underlying logical design of a stored program, a design discipline named usability. Methods of user-centered design are used to ensure that the visual language introduced in the design is well-tailored to the tasks.

The visible graphical interface features of an application are sometimes referred to as chrome or GUI (pronounced gooey). Typically, users interact with information by manipulating visual widgets that allow for interactions appropriate to the kind of data they hold. The widgets of a well-designed interface are selected to support the actions necessary to achieve the goals of users. A model–view–controller allows flexible structures in which the interface is independent from and indirectly linked to application functions, so the GUI can be customized easily. This allows users to select or design a different skin at will, and eases the designer's work to change the interface as user needs evolve. Good user interface design relates to users more, and to system architecture less.

**Agile Development:**

AGILE methodology is a practice that promotes continuous iteration of development and testing throughout the software development lifecycle of the project.

Importance of agile process.

* The **agile process** is broken into individual models that designers work on
* The customer has early and frequent opportunities to look at the product and make decision and changes to the project
* Small projects can be implemented very quickly. For large projects, it is difficult to estimate the development time.
* Error can be fixed in the middle of the project.
* Development process is iterative, and the project is executed in short (2-4) weeks iterations. Planning is very less.
* Documentation attends less priority than software development
* In agile testing when an iteration end, shippable features of the product is delivered to the customer. New features are usable right after shipment. It is useful when you have good contact with customers.
* It requires close communication with developers and together analyze requirements and planning

Methods used in project:

Planning:

* Service level agreements and its conditions.
* Information gathering.

Requirement analysis:

* Gather all the requirement required to implement project.

Design:

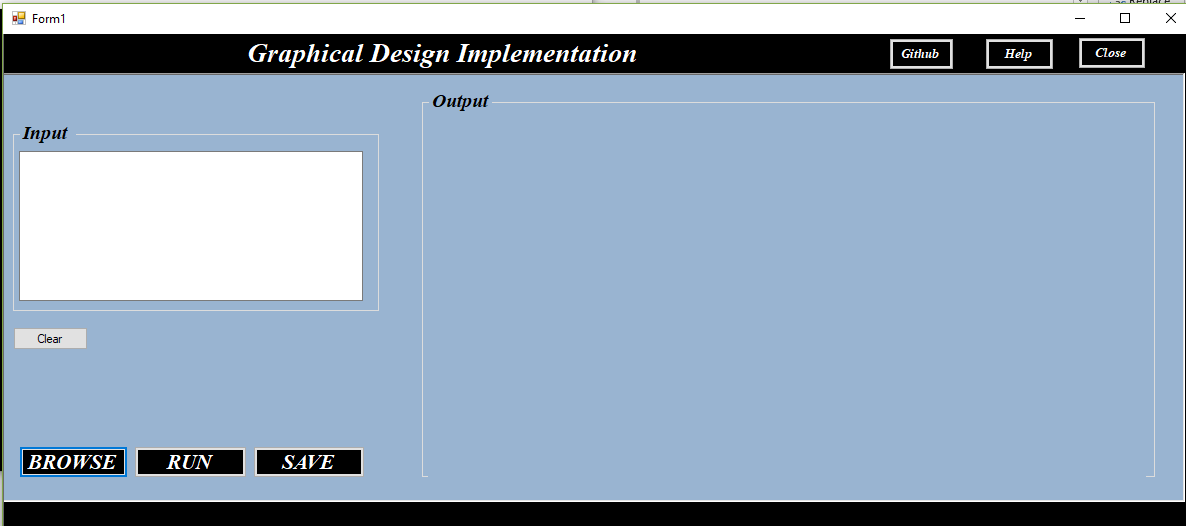
* Break down of tasks
* Test Scenario preparation for each task
* Regression Automation Framework

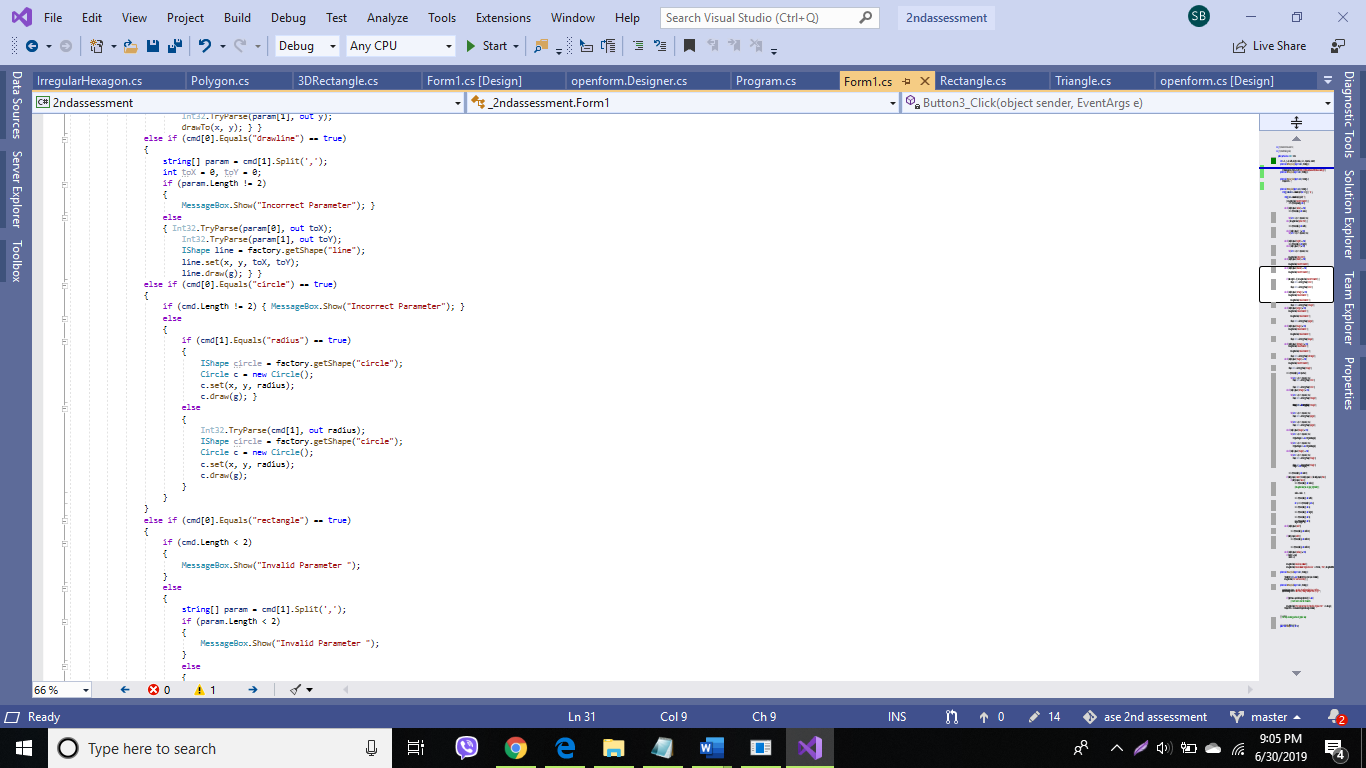
Execution:

* Coding
* Unit Testing
* Execution of Manual test scenarios
* Defect Report generation
* Conversion of Manual to Automation regression test cases
* Mid Iteration review
* End of Iteration review

Project:

Main form:





**Circle Class:**