**Project 4 – Team 8 - Product Backlog**

**Sprint table:**

* **P** stands for **priority**. User stories are ordered by priority.
* **E** stands for **estimation**. We have used the Fibonacci series as a measure.
* **No.of user stories =** 16
* **No. of tasks** = 30

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  | **Day** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** |
| **P** | **E** | **User stories** | **Tasks** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1 | 1 | As a user, I should be able to see a rectangle with an **open bracket symbo**l with a **box at the right end**, so that I can click on them. | Create a class **Open Bracket**  and extend the Shape class. | 1 |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Test the shape in workspace. |  |  |  |  | 1 |  |  |  |  |  |  |  |  |  |
| 1 | 1 | As a user, I should be able to see a rectangle with a **closed bracket** with a **circle at the left end**, so that I can click on them. | Create a class **Closed Bracket**  and extend the Shape class. | 1 |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Test the shape in workspace. |  |  |  |  |  | 1 |  |  |  |  |  |  |  |  |
| 1 | 1 | As a user, I should be able to see a rectangle with a **lesser than symbol** with **one circle at the left end and two circles at the right end**, so that I can click on them. | Create a class **LessThan**  and extend the Shape class. |  | 1 |  |  |  |  |  |  |  |  |  |  |  |  |
| Test the shape in workspace. |  |  |  |  |  | 1 |  |  |  |  |  |  |  |  |
| 1 | 1 | As a user, I should be able to see a rectangle with a **greater than symbol** with **one circle at the right end and two circles at the left end**, so that I can click on them. | Create a class **Great**  **Than**  and extend the Shape class. |  |  | 1 |  |  |  |  |  |  |  |  |  |  |  |
| Test the shape in workspace. |  |  |  |  |  |  | 1 |  |  |  |  |  |  |  |
| 1 | 1 | As a user, I should be able to see a rectangle with an **at (@) symbol** with **one circle at the left end and two circles at the right end**, so that I can click on them. | Create a class **AtThe**  **Rate**  and extend the Shape class. |  |  | 1 |  |  |  |  |  |  |  |  |  |  |  |
| Test the shape in workspace. |  |  |  |  |  |  | 1 |  |  |  |  |  |  |  |
| 1 | 1 | As a user, I should be able to see a rectangle with a **dash(-) symbol** with **one circle at each end**, so that I can click on them. | Create a class **Hyphen**  and extend the Shape class. |  |  | 1 |  |  |  |  |  |  |  |  |  |  |  |
| Test the shape in workspace. |  |  |  |  |  |  | 1 |  |  |  |  |  |  |  |
| 1 | 1 | As a user, I should be able to see a rectangle with **two horizontal lines** with **two bars at the right and left ends** so that I can click on them. | Create a class **Rectangle**  and extend the Shape class. |  |  |  |  | 1 |  |  |  |  |  |  |  |  |  |
| Test the shape in workspace. |  |  |  |  |  |  | 1 |  |  |  |  |  |  |  |
| 2 | 2 | As a user, I should be able to click on any box of any rectangle and connect with any other box. | Add mouse listeners for the boxes |  |  |  |  |  |  |  | 2 |  |  |  |  |  |  |
| Call line class to connect two shapes. |  |  |  |  |  |  |  | 1 |  |  |  |  |  |  |
| 3 | 1 | As a user, I should not be able to draw multiple lines from a single box. | Check if a box is already connected to any other box in a shape |  |  |  |  |  |  |  |  |  | 1 |  |  |  |  |
| 4 | 1 | As a user, I should be able to draw multiple lines from a horizontal bar. | Add mouse listener for the bar class |  |  |  |  |  |  |  |  | 2 |  |  |  |  |  |
| Allow multiple connection for the bar class |  |  |  |  |  |  |  |  | 1 |  |  |  |  |  |
| 5 | 1 | As a user, I should not be able to connect a box from one rectangle to a box in the same rectangle except the rectangle with @. | Write rules to check the type of class and allow /deny if it is in the same rectangle. |  |  |  |  |  |  |  |  |  | 1 |  |  |  |  |
| 6 | 3 | As a developer, I want to learn about parenthesis matching, so that I can perform compile function. | Learn about techniques to check if given expression is valid |  |  |  |  |  |  |  |  |  | 2 |  |  |  |  |
| Try with small examples to be valid. |  |  |  |  |  |  |  |  |  | 2 |  |  |  |  |
| 7 | 2 | As a developer, I want to learn and revisit the data structure used for storing shapes, so that I can perform compile function. | Analyze different data structures used to store shapes. |  |  |  |  |  |  |  |  |  |  | 2 |  |  |  |
| Compare efficiency and decide |  |  |  |  |  |  |  |  |  |  | 3 |  |  |  |
| 8 | 5 | As a user, I should be able to validate if the expressions in the workspace are correct so that I can compile them. | Create class compile |  |  |  |  |  |  |  |  |  |  |  | 4 |  |  |
| Write Rules for valid and invalid connection |  |  |  |  |  |  |  |  |  |  |  | 4 |  |  |
| 9 | 3 | As a user, I should be able to create multiple tabs of the workspace so that I can work on different expressions at the same time. | Create a new class and  Add GUI for NewTab button |  |  |  |  |  |  |  |  |  |  |  |  |  | 6 |
| Test NewTab in workspace |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 10 | 3 | As a user, I should be able to double click on any rectangle on the right panel and input a value for the respective rectangle. | Create class inputText |  |  |  |  |  |  |  |  |  |  |  |  | 3 |  |
| Store input value for the respective Shape. |  |  |  |  |  |  |  |  |  |  |  |  | 3 |  |

**Sprint Review:**

Attended by:

* Amudhan Manisekaran
* Anusha Singh
* Kartik Mathpal
* Sandya Manoharan
* Sayali Tanawade

What was done in this sprint:

* 7 new shapes namely OpenBracket, CloseBracket, LessThan, GreaterThan, AtTheRate, TwoBars, and Hyphen were implemented.
* The connection criteria for all the shapes were studied and rules were implemented to achieve the required results.
* The compile functionality was implemented successfully by following the principles of parenthesis matching.
* As a carry forward from the previous sprint, save and load features were implemented.
* The double click feature to input text information feature was also implemented.

What was not done in this sprint:

* New tabs feature is working barring a few defects. It has a few issues and is not perfectly in alignment with the requirement.

What went well during the Sprint:

* User Stories were completed with meticulous planning and well-structured work division among the teammates.

Obstacles faced during this sprint:

* Due to the unprecedented situation that happened and with everyone staying home, it was difficult to coordinate remotely. Communications were limited to just online.

How did we resolve these obstacles:

* Online discussion forums like Slack, Whatsapp and Zoom were used to resolve communication conflicts.

Discussion of Product Backlog:

* Since all the User Stories were completed with an exception of one partial completion, this sprint was deemed a success.

**Sprint Retrospective**

Attended by:

* Amudhan Manisekaran
* Anusha Singh
* Kartik Mathpal
* Sandya Manoharan
* Sayali Tanawade

What went well:

* User stories carried forward from previous sprints were also completed.
* New requirements for the updated shapes, their connections, and the compilation rules were implemented successfully.
* Communication between the team members was improved from the previous sprint.

What needs improvement:

* Needs to spend less time in knowledge acquisition as it takes up a significant amount of time.
* Balancing the workload of each team member.

Next Steps:

* Ensure better division of tasks among the team
* Consider the availability of each team member by hours a day to avoid overestimation.

**Burndown chart:**

**A close up of a map

Description automatically generated**

**A close up of a map

Description automatically generated**

* **No.of user stories =** 16
* **No. of tasks** = 30
* Y-axis denotes the no.of user stories
* X-axis denotes the working days

**Class diagram:**

**A close up of text on a white background

Description automatically generated**