Please Note:

- ***Topics 2h, 3a, 5c and 6 are not mandatory -*** *They are just bonus features for you to show off.*
- ***The application should also work on mobile devices.***
- ***Take into consideration memory management and device optimisation.***

Build an application as follows:

1. Create a rectangular area. This rectangle represents the canvas area.
2. Generate shapes of your choice (for example - triangles) as described below:

    a. The shapes must fall down from top to bottom;
    b. The shapes initial Y position is outside the top of the rectangle (Fig. 1 - A);
    c. The shapes initial X position should be random across the top of the rectangle;
    d. The bottom position is outside the bottom of the rectangle (Fig. 1 - B);
    e. You should mask the shapes outside of the rectangle (Fig. 1 - C);
    f. The falling is controlled by the Gravity Value;
    g. The shapes generation frequency should be 1 shape / second;

    h. (Bonus) Generate random shapes (from the list below), with random colours - The shape type and colour should be defined when the shape is generated.

    Shape types:
    i. 3 sides
    ii. 4 sides
    iii. 5 sides
    iv. 6 sides
    v. Circle
    vi. Ellipse
    vii. Star

3. If you click inside the rectangular area, a shape will be generated at mouse position and start falling. (Fig.1 - D)

    a. (Bonus) Generate a genuinely random irregular shape, with a random colour.

4. If you click a shape, it will disappear. (Fig.1 - E)

5. In the top left area of the rectangle, you will have two text fields (graphics engine dependent) (Fig. 1 - F):

    a. One showing the number of shapes being displayed in the rectangle.
    b. One showing the surface area (in px^2) occupied by the shapes.

    c. (Bonus) Make the previous 2 text fields with HTML instead.

6. *(Bonus) At the bottom of the rectangular area, add a couple of controls (HTML) (Fig. 1 - G):*

    a. *+/- increase or decrease the number of shapes generated per second (update the text field accordingly).*
    b. *+/- increase or decrease the gravity value (update the text value accordingly).*

7Test requirements:

Must use:

- A graphics engine using HTML canvas (OpenFL, PIXI.js, etc)
- OOP

Nice to have:

- Design Patterns
- HTML elements
- CSS

Figure 1: