# UI Requirements

The following document outlines what we will need for implementation of the user interface, both aesthetically and technically.

There will be three interface screens that the player looks at most of the time (outlined below). Additionally, we can also create a main title screen with art backdrop, as well as splash screens for an intro and two endings (win and lose).

We will create content based on a maximum screen size resolution 1280x720.

**Control**

Primarily we’re aiming to support WebGL and possibly PC/Mac builds. Players can navigate using the mouse to click on buttons and switch screens. Ideally they should also be able to use only keyboard controls. To switch between screens we will need buttons on the edges of the screen that loads the relevant scene, or - if we use one scene for the whole interface - slide the whole interface to the next screen.

For tablets (stretch goal) we would like to support finger swiping and touch, and touch-hold. We don’t plan to support phone size devices.

**Art/Event screen**

* Each event will require it’s own art image that is relevant to that event. The art images will be named according to the event to make matching them up in code easier. Images will need to be loaded from sprites and displayed in a panel on the UI for each turn of play.
* At the start of a turn the event panel will be updated to display the event for that day, including its name and body text. The text for the event should be able to fit within the panel view without need for a scroll bar.



**Bio/Survivors screen**

* Six portrait images of survivors will be displayed as interactive buttons along with a health bar below the portrait and their name above it. Player’s can click on a survivor button to get more information about that character’s bio. The selection of a character portrait has no bearing on the outcome of a turn, and only serves to display the character’s bio. Thus the button for the selected portrait should not look like it’s been selected, in order to avoid confusion that this is a choice the player must make.
* The bio panel will display a survivor’s name, age, trait(s) and short character profile text upon selecting a survivor. The text for the bio should be able to fit within the panel view without need for a scroll bar. We will also need the chosen character’s portrait to be displayed as well as their current health. This is needed to show the player which character they last selected. The first time a player sees this screen (each turn) the bio should already display the first character in the list of survivors – so we don’t have a blank panel. The trait(s) text should be highlighted in a different colour and with square brackets, to differentiate it from the main bio text.



**Timeline/Resources/Actions/Commit screen**

* A timeline panel will visually keep track of the number of days remaining until the storm hits the player’s rover. This closure rate will be displayed as both a text number and a closure line on the timeline graphic. This closure line will be displayed as a visible connection between the storm icon and the rover icon. The rover and storm icons will change each turn, and the closure line will remain connected between them. Static markers at either end of the timeline indicate the start point and end goal (safe dome city) to reach.
* The resources panel will contain the four resources the player must keep filled, by choosing actions to take each turn. The resources will deplete each day due to upkeep (i.e. using up oxygen, fuel etc) between turns. However some actions the player performs can add or remove from the resources. The resources are indicated both by a number and a coloured range bar, which need updating every turn and will change colour from green to yellow to red at certain thresholds.
* The actions panel contains interactive buttons that display descriptions of choices the player can take for that turn. There will be two actions that are non-trait based and two possible actions for trait based choices. If a player’s party has no traits that match any possible trait-based actions for an event, those buttons will be disabled and left blank. When an action is chosen, it highlights to indicate the selection and plays a highlight sound. The player can choose a different action by clicking any of the other action buttons. A different sound is also played when a button is clicked.
* A commit button will be initially inactive, until the player has selected an action. Only then will this button become clickable. Upon clicking commit, the simulation takes place behind the scenes and calculates the day’s outcome. It decides on the next event and the player is automatically taken back to the art/event screen with new art and event description displayed. Additionally, the other two screens will have their relevant visual markers, buttons, text and numbers updated.

