Cell Builder - Requirements

1. The software must incentivize the player to learn about cells and their components on a middle school level
    - 1.1. The software must include three evolutions of cells: bacteria, plant cells, and animal cells.
    - 1.2. Each type of cell in the game must include the basic organelles that the player can add to the cell.
        - 1.2.1. Bacteria cells must include the following organelles: nucleoid, cell membrane, ribosome.
        - 1.2.2. Plant cells must include the following organelles: nucleus, rough endoplasmic reticulum, mitochondria, golgi apparatus, cell membrane, ribosome, chloroplasts, and vacuole.
        - 1.2.3. Animal cells must include the following organelles: nucleus, rough endoplasmic reticulum, mitochondria, golgi apparatus, cell membrane, ribosome, vacuole, and lysosome.
    - 1.3. Cells must have a limit to the amount of each organelle they can have.
        - 1.3.1. The player must be able to increase these limits by upgrading the cell membrane organelle.
    - 1.4. The software must give descriptions of the purpose of each type of cell and organelle.
    - 1.5. The software must give a brief explanation in text of each type of cell and each organelle.

2. The software must meet the following gameplay requirements:
    - 2.1. Resources are units acquired within the game to use for unlocking and upgrading the cells and their organelles.
        - 2.1.1. Certain resources are given to the player over time to create initial organelles
        - 2.1.2. The rest of the resource types are given by organelles that produce them over time.
    - 2.2. The player must be able to use resources to upgrade cells and their organelles.
        - 2.2.1. Upgrading a cell must unlock new organelle types.
            - 2.2.1.1. The cell must have a limited amount of organelles with newer organelle types being unlocked as the cell is upgraded.
        - 2.2.2. Upgrading organelles must give some benefit to the player related to its functionality.
            - 2.2.2.1. Upgrading organelles must increase the rate of acquiring the resources that the organelle provides.
        - 2.2.3. Upgrading the cell's cell membrane must increase the maximum amount of organelles that can be made
        - 2.2.4. Upgrading a cell or organelle must consume resources.
    - 2.3. The player must be able to gather resources to store and use.
        - 2.3.1. All organelles must produce one singular type of resource.
            - 2.3.1.1. The following organelles must produce DNA: nucleoid and nucleus.
            - 2.3.1.2. The following organelles must produce Protein: ribosome and golgi apparatus.

2.3.1.3. The following organelles must produce ATP: mitochondria and chloroplasts.

2.3.2. Multiple resources must be produced by organelles, which must produce more with each upgrade.

2.3.3. These resources must be naturally generated by certain organelles

2.3.3.1. There must be certain organelles responsible for generating additional ATP at a steady rate which allows the ability to power the functions of other organelles.

2.3.3.2. Other organelle types must be responsible for generating either DNA or Protein

2.3.3.3. Resources generated by the organelle must be stored in their own local storage space ready to be collected by the player

2.3.4. Interacting with the "Collect" UI button for any organelle must store the the collected resources into the player's main storage

2.3.4.1. The main storage must be where the rest of the cell accesses the resources to create and upgrade the cell.

2.4. The player must be able to use resources to create new organelles for a chosen cell.

2.4.1. The new organelles must start at its initial resource production rate.

2.4.2. The new organelles must be contained within the chosen cell

2.4.3. A new organelle must not be added if it exceeds the maximum amount of organelles the cell can contain

2.4.4. A new organelle must consume resources

2.5. The player must be able to create new cells from their current ones in a process called Mitosis.

2.5.1. The new cells must be the same evolution as the original cell chosen by the user.

2.5.2. The new cells must be reset with a basic assortment of organelles.

2.5.2.1. This basic assortment must consist of the ATP producing organelles and the Nucleus/Nucleoid.

2.5.3. Mitosis must consume resources.

2.6. The player must be able to evolve a new cell during mitosis.

2.6.1. Evolving must only be possible when creating a new cell through mitosis.

2.6.2. Evolving must only allow the use of organelles in the new evolution in the new cell.

2.6.3. Evolving must only be possible when the player has added all the organelles they've unlocked in the starting cell.

2.6.4. Evolving must consume multiple types of resources.

2.7. Each resource producing organelle listed in 2.2.4 must produce waste if they exist in a plant and animal cell.

2.7.1. Waste must be added to a cell by its organelles.

2.7.2. A cell must have a limit to the amount of waste it can hold. If the amount of waste rises above this limit, the resource producing organelles listed in 2.2.4 must slow the rate they produce resources.

2.7.3. The player must be able to use lysosome organelles to remove waste from a cell.

2.7.3.1. Removing waste from a cell reduces or removes the slowing rate of the organelles.

2.7.4. If a cell does not have lysosomes included in its evolution, its organelles must not produce waste.

2.7.5. The amount of waste in the cell must be visually indicated to the user, with a special indication if the amount of waste is over the cell's limit.

3. The software must have an intuitive UI.
   3.1. The software must clearly display how much of each resource the player has.
      3.1.1. An exception is if the resource has not been unlocked or the value is zero.
      3.1.2. Each resource value must never be displayed as negative.
   3.2. All text must be large enough to be readable on a 16:9 desktop computer monitor.
   3.3. UI elements must follow a consistent scale, font, and color palette.
   3.4. All placeable objects must be visible at all times and distinct from each other.
   3.5. Input elements must take no longer than one second to respond to input.

4. The software must store the player's progress across sessions.
   4.1. The software must store the following data:
      4.1.1. All resources the player has across sessions.
      4.1.2. The configuration of the cells, organelles, and any other placeable objects the player has acquired.
   4.2. The software must store all the above data from at least 5 seconds before an abrupt crash or shut down.
   4.3. The software must store all the above data from the moment the software is terminated if the user closes it.

5. The software must perform well on low-end Windows machines.
   5.1. The software must maintain a consistent frame rate of at least 60 frames per second on a Windows 10 or 11 machine with at least an Intel Core i3 and 8 GB of RAM.
   5.2. The software must not fall beneath 60 frames per second for more than 3 seconds in any 15 second interval.
   5.3. The software must meet these requirements on the latest versions of both Google Chrome and Firefox.