

Heuristic Evaluation
Team 6: Infectious Disease Cellular Automaton
Dylan Gaines

The Design:

The primary view in this application is the simulation screen. It contains a grid for the user to build their world. Objects are represented by squares, and people are represented by circles. These elements have color gradients that represent their level of infection with the disease. There is a legend to the side that conveys this information to the user. Beneath the grid there is a table of options, including grid size, length of simulation, and disease parameters such as infection period and viral production. This is where the controls to start, pause, and end the simulation are located as well.

Upon completion of the simulation, there is a screen that presents the results. This could be in either tabular or graphical format, with the option to download results, start a new simulation, or log out.

User Interface Domain:

This application belongs to the "learning simulation" UI domain. There are many applications in this domain. They are primarily characterized by their purpose--to teach--and the input/output functionality. Applications in this domain typically take one or more parameters from the user and graphically display the results of those particular parameters. Since the purpose of these applications is to teach, the UI and controls should be fairly simple and easy to understand at first glance. If students are spending too much time learning how to use the simulation, they are not efficiently learning the lesson it was intended to teach. In addition to the inputs being easy to understand, the outputs should be easy to understand. The medium in which the resulting data is presented should make it clear what lesson can be learned. Any trends or patterns should be highlighted for the user to see.

Heuristic Usability Principles:

While all of the 10 standard usability heuristics are important, in this domain there are a few that are more crucial to accomplish the purpose of the application. The first two are Error Prevention and Error Recovery. Since these simulations are designed with the primary purpose of teaching broad ranges of students, they may not be able to troubleshoot any technical problems very well. It is therefore more important to ensure that the application has minimal errors, and that any errors that do occur have a resolution or recovery path that is clear to the user. As discussed in the previous section, the UI and controls need to be very simple and easy to understand, which is represented in the Aesthetic & Minimalist Design principle. Along those same lines, the final core principle is Help & Documentation. The users may not always understand how to work the application, so help features should be readily available and easy to use.

Usability Problems:

1. For the color gradients of the level of infection in the simulation screen, the people and objects should have the same color gradient. This violates the Consistency & Standards heuristic.
2. In the cognitive walkthrough the team mentioned that there would be a tutorial the first time the user logs in to the system. However, there should be a clearly visible help button that the user can click at any time if they are unsure of how to use the application. This violates the Help & Documentation principle.
3. In the simulation screen there does not appear to be an indicator for how many days have been simulated so far--only for the goal number of days set by the user. This violates the Visibility of System Status principle.
4. In the simulation screen, instructions are located in multiple places. Grid setup is explained on the top, the color scheme on the right, and the player controls on the bottom. This should be condensed into a single instruction section. This violates the Aesthetic and Minimalist Design heuristic.
5. In the simulation screen there is no visible button for the user to save their progress. This is a violation of the User Has Control & Freedom principle, since the user is unable to leave partway through a simulation without losing their work.

Critical Usability Concerns:

The lack of a save button is a critical concern because it may result in lost work. For example, Angela is working on a simulation as part of a class project. After the teacher explains the project, there is only 20 minutes left in class. Angela is unfamiliar with the interface, so it takes her nearly the whole time to build her world. Before she is able to run her simulation, class ends but she is unable to save her work to continue it at home. Once she gets home she must start over with building her simulation.

The lack of a help button is also a critical concern because it may result in the user being unsure of how to use the application. For example, Michael logged in for the first time and clicked through the tutorial without really paying attention since his teacher was still explaining the requirements for the project. Once he started trying to build his simulation, he wasn't really sure what each of the parameters meant. He was too embarrassed to ask his teacher for help, so he just typed in random numbers. He was able to use the simulation, but he didn't understand the results that he got and didn't learn anything from the simulation.