The Fatalistic Game of Life: Usability Evaluation

System Overview

The Fatalistic Game of Life is an actuarial statistics web application that takes demographic data and generates a hypothetical, statistical person. This person could be a millionaire, an impoverished worker; they could live exceptionally long, or die tragically young.

There are two main components to this project: the frontend and the backend. The frontend consists of our user interface and any client side processing that's done. The backend consists of our server that runs actual computations to generate the hypothetical person. Both of these parts of the project had their own unique problems that we had to overcome.

Frontend

Our frontend is implemented in HTML/CSS and JavaScript. We also decided to use JQuery for DOM manipulation and Flotr2 for graphing purposes.

The system itself is set up so that upon loading the site you are presented with a graph showing various quantitative metrics of some hypothetical person's life. For the purposes of our prototype, we chose salary and net worth since they both have plenty of sound public data available. For each year of a person's life, two bars are shown overlaid on one another representing those two quantities. The user can hover their cursor over a bar and see more detailed information about their life during that year, e.g. gender, race, occupation, and any other tracked variables. For this prototype only a handful of variables are tracked, but the system is at a point where more could be added time permitting.

The user can also click on a bar in the graph to bring up a window allowing them to change tracked variables by hand. Right now the user can change income, race, and gender. Once the user has made their changes they press a save and reroll button which then rerolls the hypothetical person's life from the given year onward with the new variables.

Backend

While this is less relevant to our usability evaluation than the front end per se, our backend does impose certain constraints on what our frontend can do. We've chosen

The key to this project is data collection. Taking data from online sources such as the US Census Bureau is a lengthy process. Our algorithm requires data be formulated in a specific way and written in proper terms. When rolling our statistics for a person we need to note that many of the tracked variables are correlated with each other. We first start rolling the least constrained variables, name, gender, race, and age. The first three remain the same over the course of a person's life, and the last increments yearly unless the person dies. Death depends on several factors, such as gender. In particular, the average woman lives longer than the average man. Data in our server backend needs to be represented in such a way that this kind of determination is efficient, and this is not always the case when we pull data from the internet.

There is no uniform way to do this. Not only does presentation format vary wildly from place to place, but the terms of the presentation change too. Non-trivial amounts of work go in to finding data and manipulating it into a usable form. The typical way we do this is 1) we download the appropriate data and load it into spreadsheet software. 2) Save the data as a comma-separated-value format. 3) Write a python script to read in the csv file and manipulate data into desirable terms, and output a python script with the appropriate values hard-coded in. 4) write an interface function to the data, accepting all constraining data as inputs. 5) Modify all other variables to incorporate this new data, if there is a correlation. This solution is far less than ideal, but realistically any better solution would be out of the scope of this course.

Evaluation Overview

Evaluation of this product consisted of a set of test participants performing a series of increasingly complex tasks in a think-aloud setting. Table 1 summarizes test tasks for the evaluation.

Task	Description
1	Generate a new person
2	Inspect variables at some specific year
3	Step through years one by one, inspecting each set of variables
4	Modify the income variable at some specific year and re-roll data

Table 1: Task enumerating

These tasks are designed to encompass the most common use-cases for The Fatalistic Game of Life. They may not be exhaustive but they represent the most important set of functionality. Table 2 provides descriptions of the users participating in this evaluation.

Participant	Age	Gender	Occupation
1	21	Female	Student of Linguistics, Anthropology
2	59	Male	Electrical Engineer
3	29	Female	Research Scientist
4	26	Male	Technical Support

Table 2: Participant information

Two participants have engineering or similar occupations, in contrast to the researcher who has very specific knowledge in a sub-field of genetics, and a student with no technical background.

Evaluation Results

Task 1 – Generate a new person	
Participant	Action
1	Pressed 'Roll Again' button immediately
2	Refreshed page
3	Pressed 'Roll Again' button immediately
4	Pressed 'Roll Again' button immediately

Table 3: Task 1 results

Task 2 – Inspect variables at some specific year	
Participant	Action
1	Looked at screen for a moment, clicked on graph to bring up editor.
2	Looked at screen for a moment, moused over bar on graph and saw variables.
3	Moused over bar on graph and saw variables
4	Moused over bar on graph and saw variables

Table 4: Task 2 results

Task 3 – Step through years one by one, inspecting each set of variables	
Participant	Action
1	Clicked on graph to bring up editor, used mouse to click on arrow buttons.
2	Moused over bars on graph year by year.
3	Moused over bars on graph year by year.
4	Moused over bars on graph year by year.

Table 5: Task 3 results

Task 4 – Modify the income variable at some specific year and re-roll data	
Participant	Action
1	Clicked on graph to bring up editor, edited text box, and pressed re-roll button.
2	Looked at screen for a moment, pressed 'Roll Again', moused over bar on graph, clicked, edited text, and pressed re-roll button.
3	Clicked on graph, used arrow buttons to go to a particular year, edited text, and pressed re-roll button.
4	Looked at the screen for a moment, moused over bar on graph, clicked, edited text, and pressed re-roll button.

Table 6: Task 4 results

It’s worth noting here that at the time of the evaluation the site was still in an incomplete state. The results of task 4 produced erroneous program states which confused the user, and were well known to us at the time of the evaluation. This was explained to the participants, and since the evaluation was conducted the egregious errors were fixed.

Findings

There are a number of small issues with the system. Across the board people were confused about what the two different bars on the plot represented, though several moments of inspection made things clear. A legend, or cleverer labeling, would solve this problem completely.

The second most striking issue was general confusion about how to get to the edit menu. There was no indication that clicking would allow you to edit that year's data. This is a bit less obvious on how to fix. The most obvious fix would be to display some message explaining it until the user provides acknowledgment.

Users also felt that the edit page itself was somewhat unwieldy. This is understandable; the edit page varies significantly from what we had initially envisioned in our storyboard. This was briefly discussed with the participants after the test, and they seemed to agree that our storyboard idea would be better. However there is only so much faith we can put into these responses without actual testing.

Users also were confused at first with what exactly they were looking at. There is very little to no explanation on the front page, if the user wanted to see a succinct description of the application they needed to navigate to the ‘about’ page.

Response List

Sorted from highest priority to lowest priority:

Issue	Status
Fix bugs with editing generated person data	Done
Redo edit person page	To-do
Add graph legend	To-do
Add indicator that an edit menu exists	To-do
Emphasize ‘Roll Again’ button	Done
Provide better explanation for what graph represents	Done

Table 7: Formal responses to evaluation