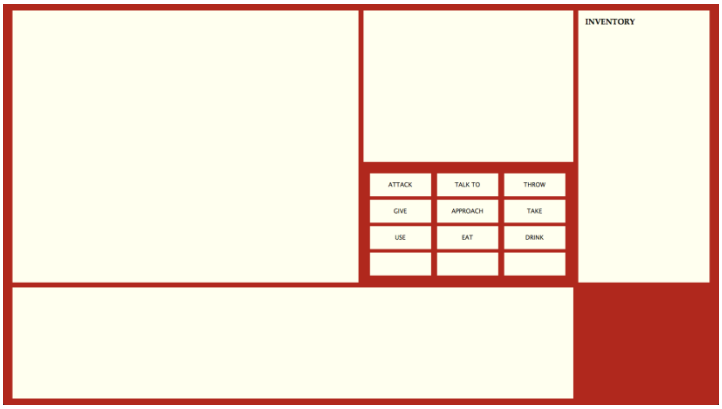
InteractiveFICTION | Text Adventure Game

CSCI 3308: Group Project

Hannah Thomas, Willie Payne, and Sarah Feller

Top Left: Start Page where you can either continue your latest game, or start a new one.
Top Right: In-game view highlighting the hover glow indicating a viewable person, place or thing.
Bottom Left: In-game view demonstrating the button hover when choosing an action for an object.
Bottom Right: In-game view illustrating inventory functionality.

Iteration 1:



Iteration 1 UI: Image taken of the first layout of the UI. Some of the functionality, like the Info Cards, was added after this screenshot, but the layout remained the same for the first demonstration. The reaction box was also removed.

Iteration 1 was completed as planned in the specified amount of time. It successfully laid the foundations for more complex functionality in following development of the web app. Only one portion of a task was dropped in favor of concentration on another; we chose more sound effects over the creation of a simple song.

Task	Description	Accomplished?
1) Tool Assessment	Affirm and learn necessary tools for project (HTML5, Javascript, SQL,…)	Decided on HTML5, Javascript, and CSS, and kept those three languages through the entire project.
2) Info Card	Appears when item, person, or location selected	Successfully setup the Info Card system so that descriptions appeared in the appropriate place upon story object selection.
3) Layout	Basic UI: Story text, info card, inventory, commands, reaction box	The basic layout did change from the original plan at this point. We removed the reaction box in favor of a continually changing story box, and thus shuffled the arrangement of the UI a bit.
4) Music	Basic sound effects and simple song	Sound effects were being created, but none were implemented during this iteration. The simple song was dropped in favor of more sounds effects.
5) Linking	Hover recognition	Hover recognition was up and running so that the links for Info Cards could be identified.

Troubleshooting:

Troubleshooting during this iteration involved a lot of quick writes and checks. No part of the application went very deep at this stage, so it was possible to add a block of HTML5, Javascript, or CSS code and observe the .html file from any web browser on our computer. For example: if we wanted a link to be hover-able, we inserted the proper code and played with it until it suited our needs. All progress at this stage could be tested through one story page, as nothing lead to another step in the plot.

There was one idea that we pursued for a time, that we ultimately found to be wrong for our application. For the different boxes, we initially had iframes displaying the text, but they did not suit the dynamic nature of the Info Box well enough for our needs. In the end, we were able to use Javacript to link the story objects to their corresponding Info Box descriptions.

After we have done all of our visual testing, we run all of the HTML and CSS through W3Schools' Code Validator, and use Chrome's "Inspect Element" to check the validity of the Javascript (more on these tests can be found in Iteration 3).

Iteration 2:

Unfortunately, we do not have a screenshot of this particular Iteration, but it did progress design-wise, and closed the empty gaps in the original layout. This particular iteration focused on using button commands to progress the story, actually making it interactive. Content increased, and the web app blossomed into a functional, though incomplete, text adventure game.

Task	Description	Accomplished?
1) Content Creation	Focus on story progression and necessary commands and possible paths	Yes, a workable amount of content was placed on the site to test the functionality of the other tasks we accomplished during this iteration.
2) Command Functionality	Commands work with selected objects for correct reactions	The Buttons/Commands were successfully mapped to the correct objects, enabling story interaction and progression.
3) Reaction Box	Dialogue options lead to different paths/responses	The Reaction Box was dropped during Iteration 1 in favor of more dynamic story pages.
4) Dynamic Info Cards	Learning names, healing people, etc. change status	Required no reworking after Iteration 1 when we decided to use Javascript as opposed to iframes.
5) Music	Character Noise Effects and another simple song	Still favored the sound effects over a simple song, but also managed to implement the sound effects on the site.

6) Database	Incorporate database if not already needed for prior iteration	We decided not to use a database for the project, as it was going well without one.
7) Find Webhost	Not originally included in our list of things to do at this stage: Migrate the site to a web host.	Used a free web host (Zymic) and Filezilla to quickly and easily upload and update our site.
8) UI Redesign	Not originally included in our list of things to do at this stage: Update the look of the site to make it more user-friendly	We decided that something needed to be done about the overall look of the UI before the next demonstration, so we updated its look and moved the modules around for more intuitive play.

Troubleshooting:

This iteration took a little more than the last in terms of testing. There were more areas for errors to occur, however, like the last time, they are clearly observable by running through the different story tracks. If something didn't work how the player should expect it to work, we changed the code until it did.

There were only a few obstacles we had to work around during this iteration. These included finding a good webhost and getting the sound effects to cooperate on the webhost we did select. Zymic turned out to be ideal host: free and simple. All we needed was an account and Filezilla, and then uploading and updating our files was a simple task. However, after we had gotten settled on our new domain, we learned that Zymic did not allow music file uploads for copyright reasons. All of the sound effects are remixed versions of free sounds offered online, so we found a workaround by keeping the files in Dropbox, and linking to them from our HTML files.

After we have done all of our visual testing, we run all of the HTML and CSS through W3Schools' Code Validator, and use Chrome's "Inspect Element" to check the validity of the Javascript (more on these tests can be found in Iteration 3).

Iteration 3:

The third, and final, iteration can be viewed from a variety of screenshots at the very beginning of this document, or viewed and played online at http://interactivefiction.zzl.org/. The latest version has a revamped design with better site navigation for easier play. Little things were changed based on suggestion, such as bolding the objects you are able to interact with and linking the logo to the start screen for restarting your adventure quickly.

Task	Description	Accomplished?
1) Content Creation	Focus on story progression and necessary commands and possible paths (at least 1 complete path)	To an extent. More content was added to the web app, but we found we had to focus on breadth before length the way we were branching the story paths. Thus no paths are truly complete, though it is still playable.
2) Logic-based Puzzles	Mini-game with right/wrong answers instead of just dialogue options	We provided a riddle in one of the paths to grant the user more varied gameplay. Perhaps we will make more puzzles in future.
3) Save Game Functionality	Ability to return to same spot at later date	Quickly accomplished through the use of cookies. We also decided to make it so that the player could not "Back" to previous story pages in order to take a different path, to provide more immersive gameplay. The logo also allows you to return the title page if you wish to restart.
4) Catch-Up!	Ensure everything from first two iterations happened	Easily done, as nothing left unaccomplished in the first two iterations was neglected due to time constraints. Everything we did not do was decided against.
5) Music	Character Noise Effects and another simple song	Again, no song, but more sound effects were added.

6) UI Redesign	Not originally included in our list of things to do at this stage: Update the look of the site to make it more user-friendly	We decided that something needed to be done about the overall look of the UI before the next demonstration, so we updated its look and moved the modules around for more intuitive play.
7) Inventory Functionality	Not originally included in our list of things to do at this stage: Have the ability to take, store, and use items with your surroundings	Always intended, but accidentally left out of the proposal

Troubleshooting:

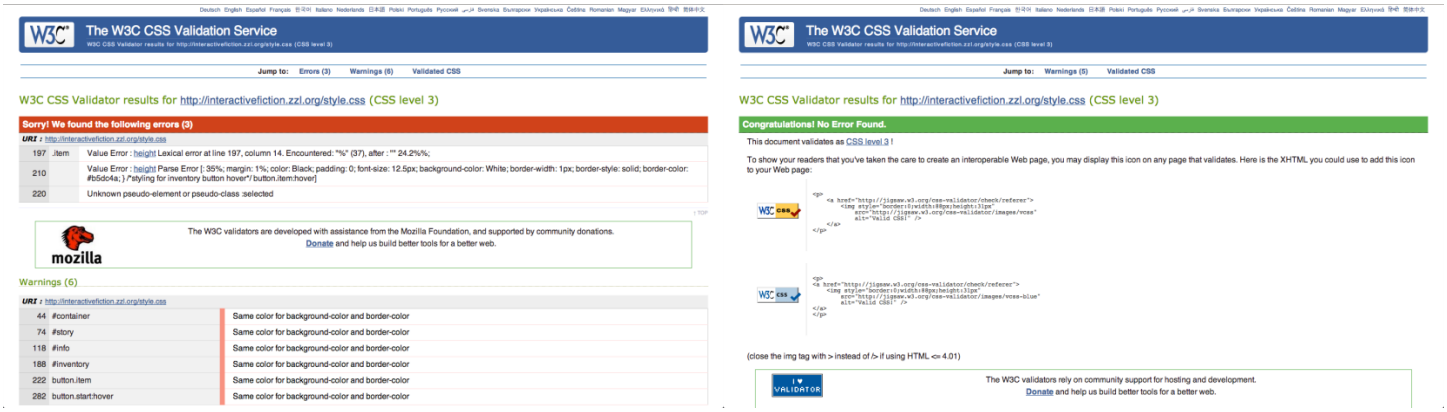
Iteration 3 testing went much the same as Iteration 2, just more in depth. There were even more areas for errors to occur that were also clearly observable by running through the different story tracks. This time through, in addition to running through everything multiple times ourselves, we opened the site up to friends and asked them to explore and attempt to break the application. Doing this, we exposed a number of errors we never would have caught by testing it ourselves.



Army of Testers: A few of our testers, having fun while helping us find bugs.

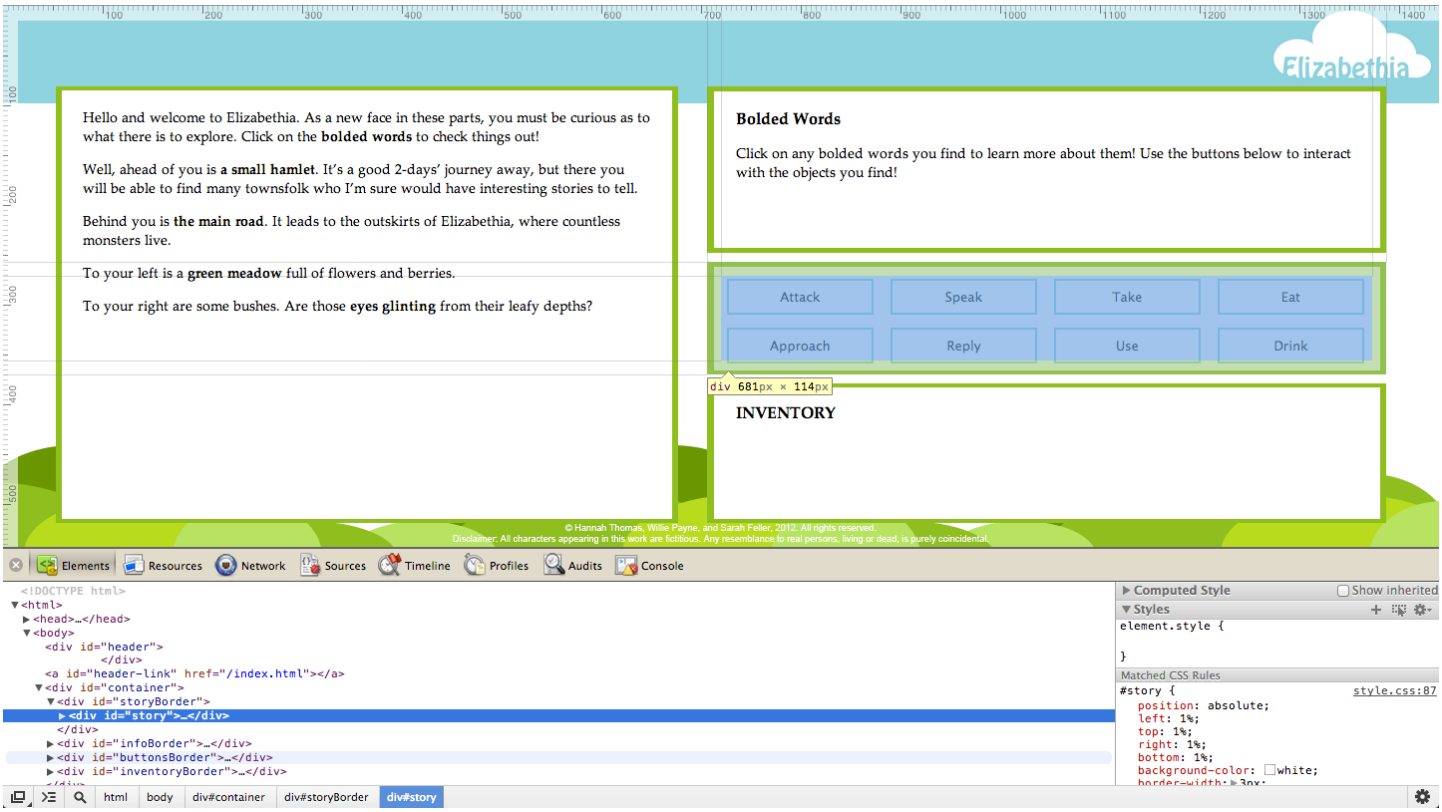
Most of the troubles we encountered this time through had to do with the implementation of the Inventory. It became more difficult to update the storyline with more features to enable. The larger number of testers helped us work out all the 404 Errors to make sure that all the pages led onward to their intended paths.

After we have done all of our visual testing, we run all of the HTML and CSS through W3Schools' Code Validator, and use Chrome's "Inspect Element" to check the validity of the Javascript :



Left: The final CSS Validation run-through on W3Schools turned up several warnings and just a few errors that didn't affect functionality.
Right: After a few quick edits, the code was approved as CSS level 3.

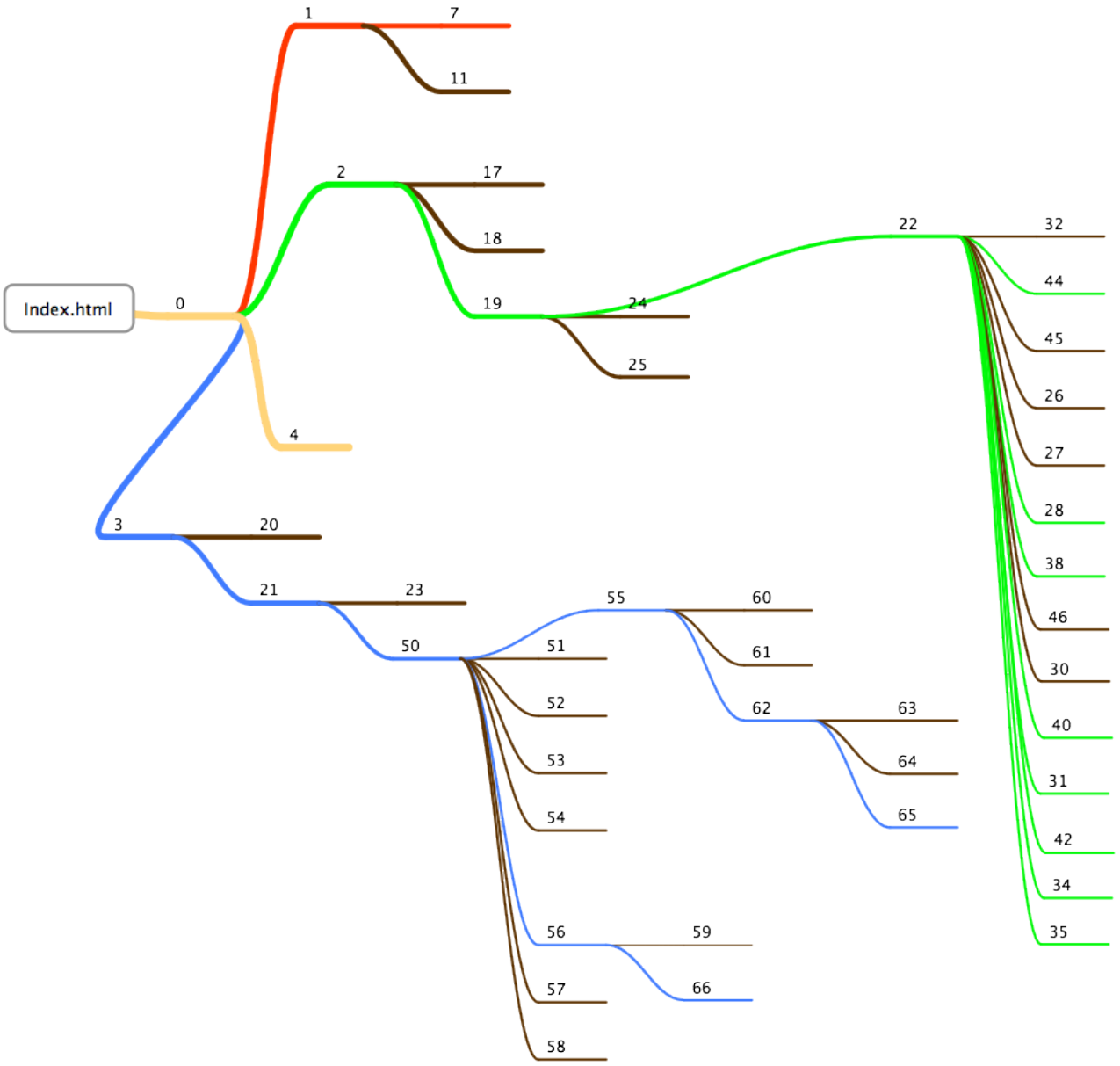
W3Schools is a web development resource website that offers code validators. They check that code is in compliance with the latest standards (HTML5 and CSS3) and syntax errors. Once a webpage is hosted, you simply give them the link and they show errors, warnings, and validated code.



Inspecting Page Elements: The above is an image of how we double-checked our Javascript code through Chrome.

To make sure Javascript code was implemented correctly, we would run our website and look for broken pages. Once we found one, we would use Google Chrome's "Inspect Element" feature to find the errors. Our most common errors were syntax errors and "Uncaught Type" errors. Usually these were caused by a conflict in the naming of variables. For example, originally we were going to have a "throw" variable. It needed to be changed to "Throw" in order to work because "throw" was already a Javascript function in Chrome.

Current Story Tree:



Next Steps:

Make the site truly dynamic and use a database!

The database would hold all objects so that objects are editable and not just redeclared on a new page.

The entire site is very difficult, though not impossible, to manage because we use new pages when we advance the story. This leads to copying and pasting many death pages and any basic edits to the layout or buttons must be changed individually on each page. A dynamic layout that changed the content of the page using some script like PHP would be much easier to make changes to.

Audio

There are a number of issues in how our current design disallows the use of interesting audio. Zymic does not allow audio to be uploaded so as a work-around all of our audio is hosted on Dropbox. The page structure means that audio cannot be played continuously or played on button presses since a new page must be loaded any time the story progresses. Currently audio is played on the start of pages meaning some pages are straight copies of each other with different audio effects depending on how the player got to the page.