PORTFOLIO REMASTERED

By Isaac Lohnes

APPD1001

Sharla Rolfe-Hunter

Authors note:
Website is hosted at: https://isaaclohnesportfolio.github.io/

Contents

Design of everyday things questions: 2
Chapter 4 – Knowing What to Do 2
Describe the four kinds of constraints and explain how they affect instructional design. 2
We can use constraints to force the desired behavior. Describe the Interlock, Lock in and Lock out constraints. Give an example of each and describe the behavior it forces. 2
Compare and contrast visibility and feedback, which one is more effective? Why? 3
Chapter 5 – Human Error of Bad Design? 3
Explain the “five whys” investigative technique. Is this a good technique for determining the Root cause of a problem? Why or Why not? 3
Compare and contrast slips and mistakes. Describe the classifications of each and give examples. 3
How do we design for error? What features should you build into your interface design? 4
Screenshots of website: 4
Screenshot of landing page: 4
Screenshot of Resume page: 5
Screenshot of Education page: 5
Screenshot of Skills page: 6
Screenshot of Programming examples page: 6
Screenshot of Bio page: 7
File structure: 7
Storyboards: 8
Sitemap: 8
File structure: 9
Landing page: 10
Resume page: 10
Bio page: 11
Work samples page: 11
Skills page: 12
Education page: 12

Design of everyday things questions:

Chapter 4 – Knowing What to Do

Describe the four kinds of constraints and explain how they affect instructional design.

Physical constraints are based around the atomic structure of the product. For example, say you were given a sheet of some hard material, such as steel, and this sheet of steel had 4 holes in it. Now say you were given 4 screws. You are physically constricted on where you can screw these screws in. You cannot simply bore through steel with a screwdriver. Imagine now that the holes in this sheet of steel were circular. If you were given square screws, and normal screws, the screw you chose would be based on a **Logical constraint**. It is easy to reason which screw goes where. When designing, we must account for what exactly is the physical limitation for our product, as well as how users will logically assemble it.

To talk about cultural and semantic constraints, I'm going to stop *screwing* around and switch to some different examples. A good example for a **Semantic constraint** would be being given a sign to hang. Your understanding of language allows you to know exactly which way the sign should be orientated. We can account for this in design by keeping in mind the way users naturally orient themselves and the product due to language. A **Cultural constraint** relates to how exactly objects are perceived in society. For example, give any user a board and 4 legs, and they can construct a table. We must account for how users perceive certain objects to act when we design.

We can use constraints to force the desired behavior. Describe the Interlock, Lock in and Lock out constraints. Give an example of each and describe the behavior it forces.

An interlock requires constant interaction in order to function continually. A Lock in prevents easy exit from a function, normally to prevent misclicks and mistakes A lock out prevents easy access from a function, typically with safety in mind.

An example of an **Interlock** would be a piano. It requires the user to repeatedly input keys in order to continue functioning. This forces the behavior of the user having to make a conscious decision to keep playing constantly for music to be played constantly.

A (nefarious) example of a **Lock in** would be canceling credit card subscriptions on a plethora of websites and applications. Although some companies are

forgiving, more often then not it is quite the maze to remove your billing information from a company. This forces the user to become upset, and the intent is for the user to simply not have the energy or motivation to remove their billing. A simple example of a **Lock out** would be password security, say on a laptop or windows devices. The function of this is to prevent users from infringing on other users privacy, and protect the user. This prevents behavior of one user simply accessing everything about another.

Compare and contrast visibility and feedback, which one is more effective? Why?

Visibility is related to which control performs which function, and is different from feedback in that feedback is a response to let the user know that the function has been performed. However, they are similar in that they both are related to giving the user more information about the function. I believe that neither is more effective then the other, and that the most effective strategy is to use both in combination. Although I could delve into the nitty gritty and give an answer about which is more effective given a situation and controls, I feel that overall, with none of that given, that they are equal and should both be used.

Chapter 5 – Human Error of Bad Design?

Explain the "five whys" investigative technique. Is this a good technique for determining the Root cause of a problem? Why or Why not?

The "five whys" technique is simply not accepting an answer until it is the root cause. Much like the small child who answers "Why" to every question, this technique involves continual probing, and the "five whys" differs from a small, pestering, child, in that it does not end at "Because I said so."

I believe this mentality is a good technique to determine the root cause, because at its core, it is about delving deeper and deeper into the problem, and not being satisfied until an exact and precise problem is known.

Compare and contrast slips and mistakes. Describe the classifications of each and give examples.

A slip is "Sharla I didn't know this project was due at 9 this morning and here I am at 1:57 working on it."

A mistake is "Sharla I haven't done any work in your course since January, and my life is spiraling out of control, I left the stove on and burnt my house down, and accidently totaled 57 vehicles in a historic pileup, by speeding to school at 453km/hr this morning"

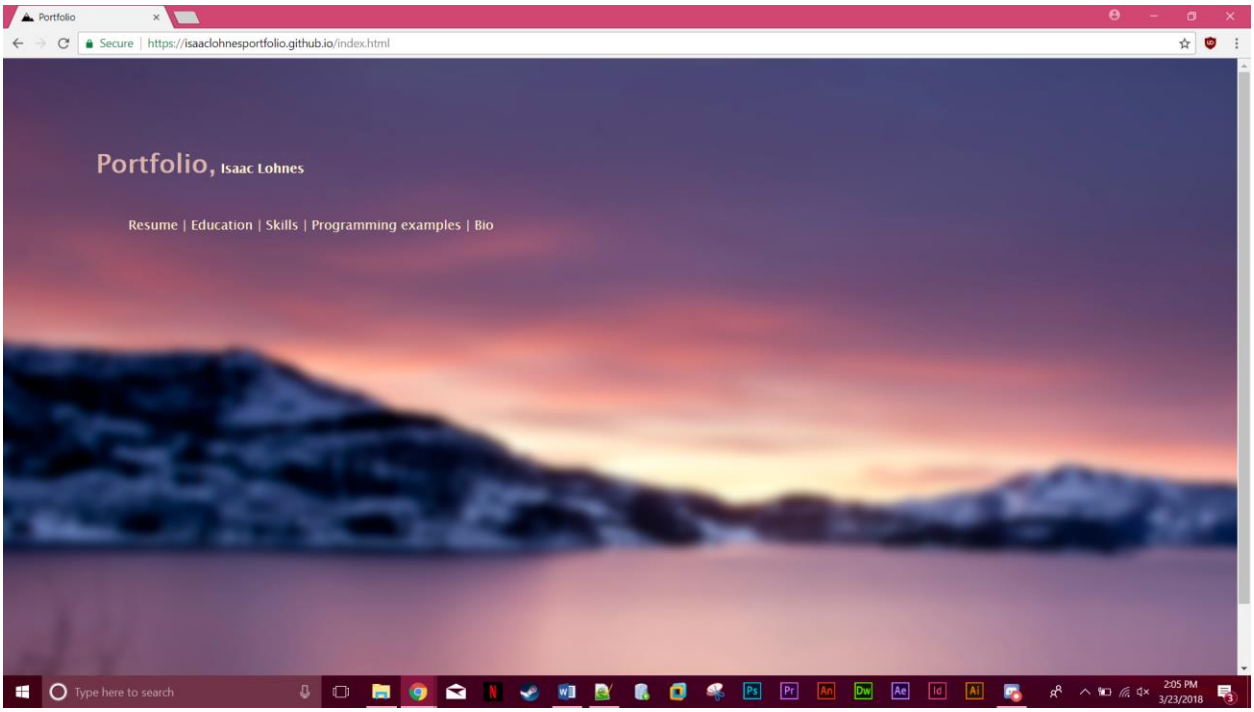
Although they are both problems, they are separated by gravity. One is a simple problem that is easy to remedy, while one could have more grave and far reaching consequences

How do we design for error? What features should you build into your interface design?

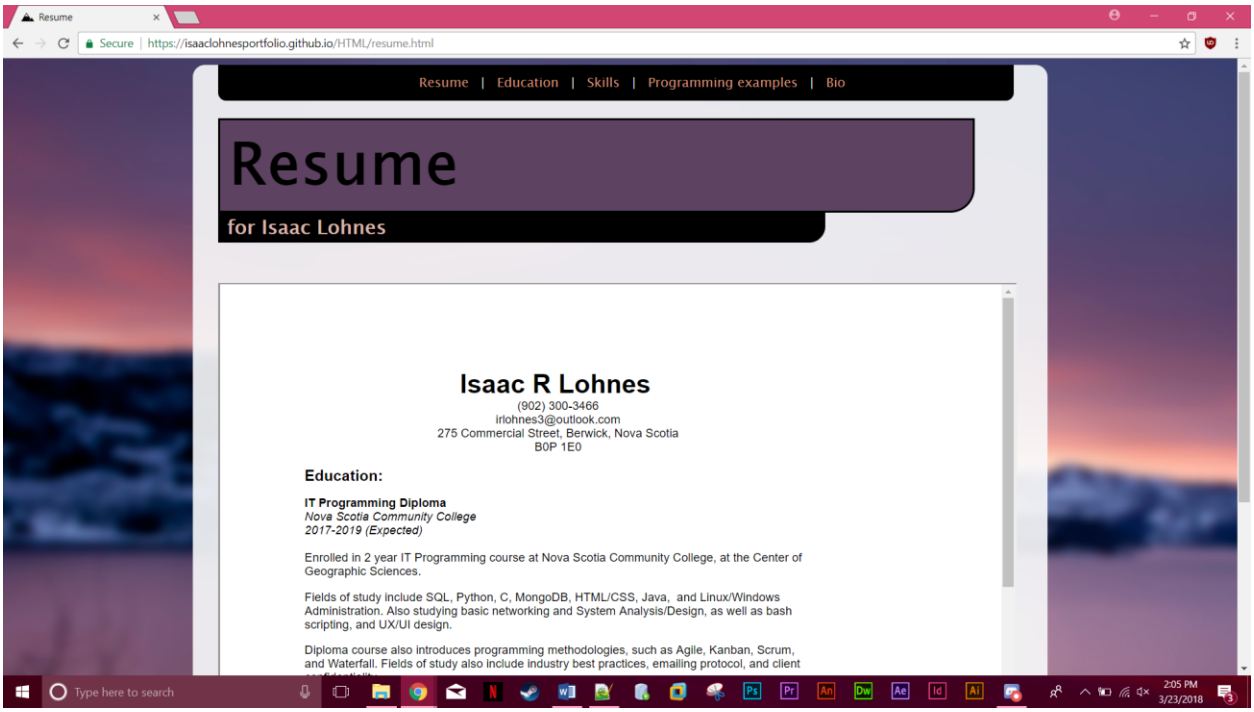
Keep in mind that humans make errors, and things do not always go smoothly when interacting with machines. When designing we should account for the requirements of the people, and modify our machine to work this way. We should also try to minimize the chance of an incorrect or wrong action, by using good design principles and giving guidance to users

Screenshots of website:

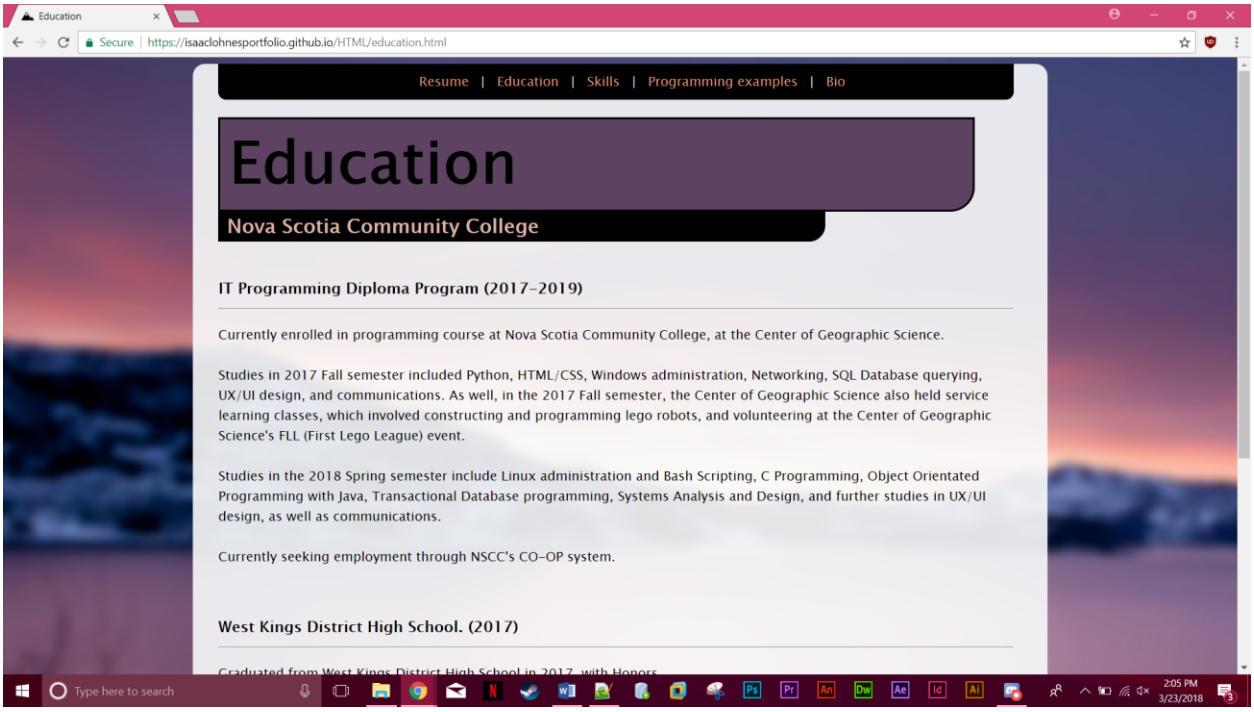
Screenshot of landing page:



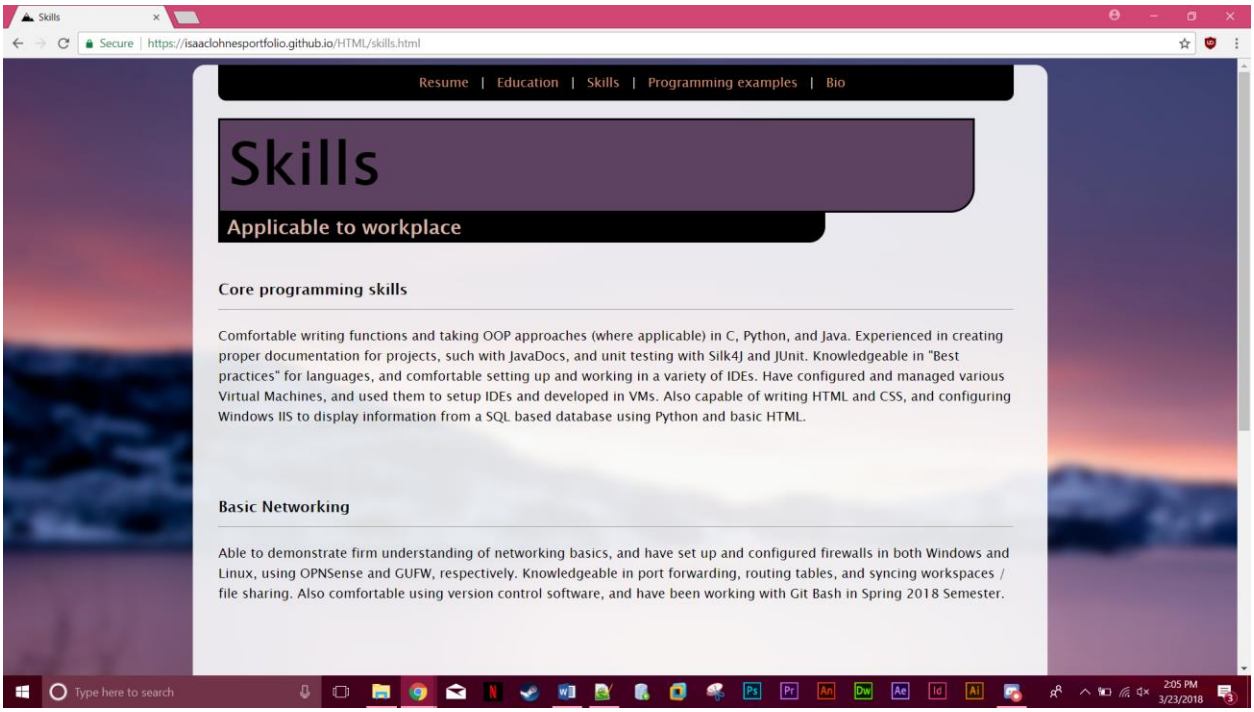
Screenshot of Resume page:



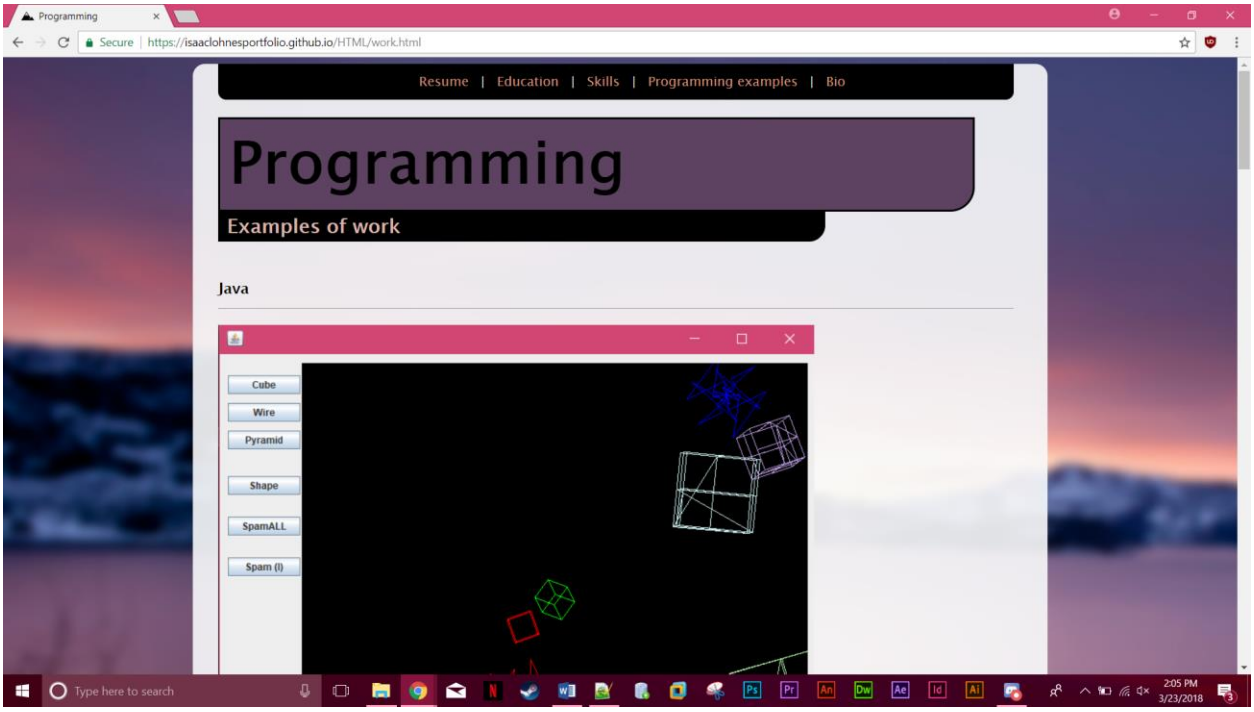
Screenshot of Education page:



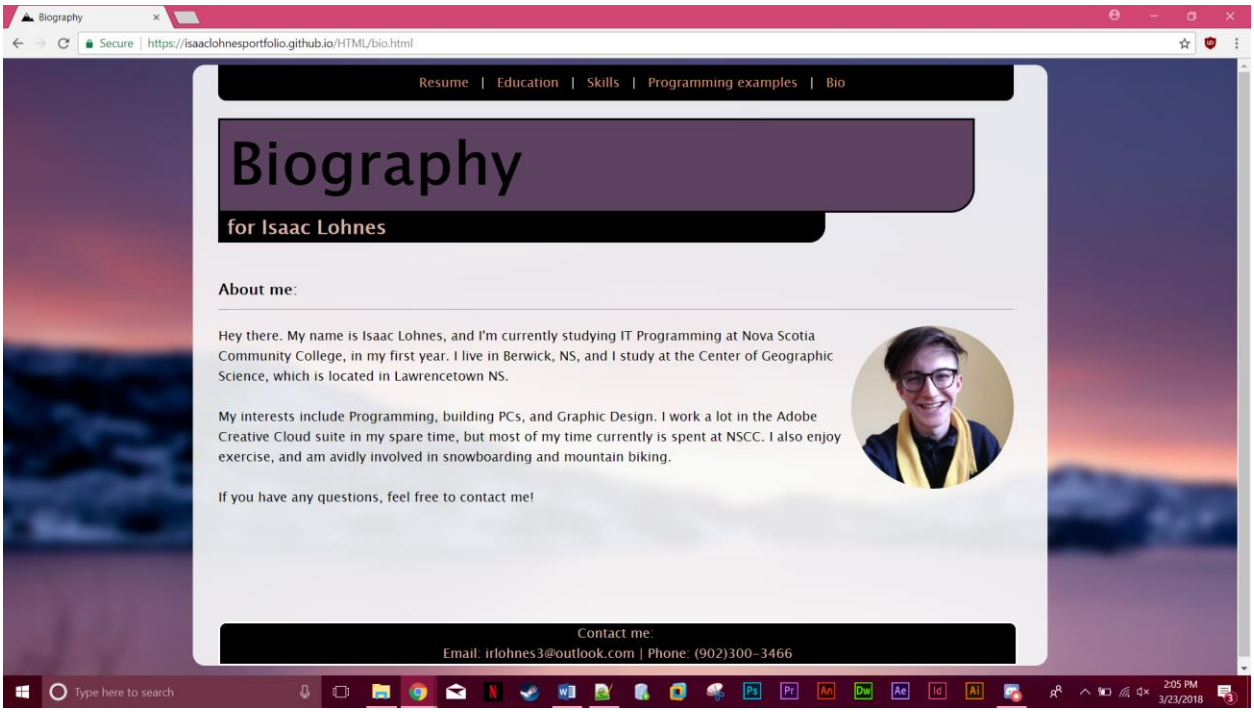
Screenshot of Skills page:



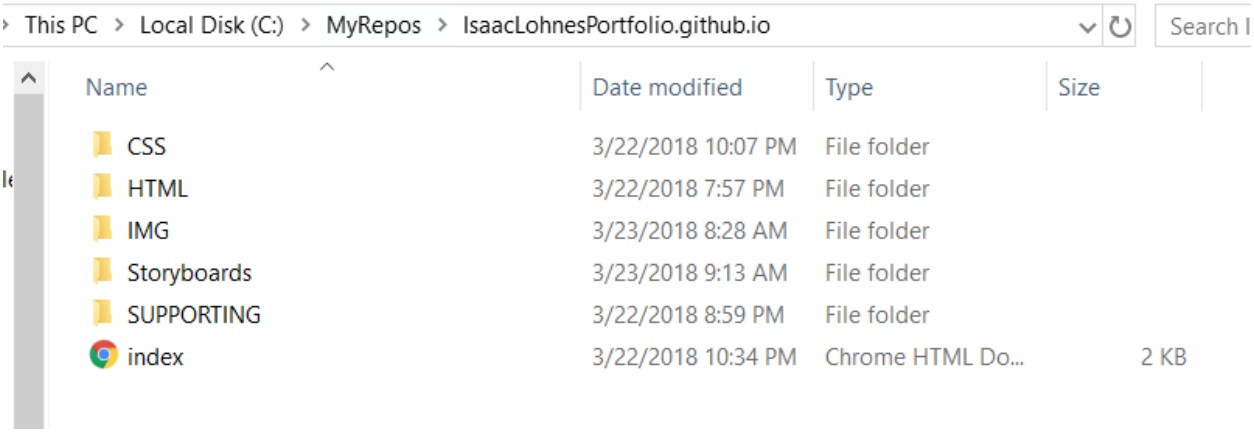
Screenshot of Programming examples page:



Screenshot of Bio page:

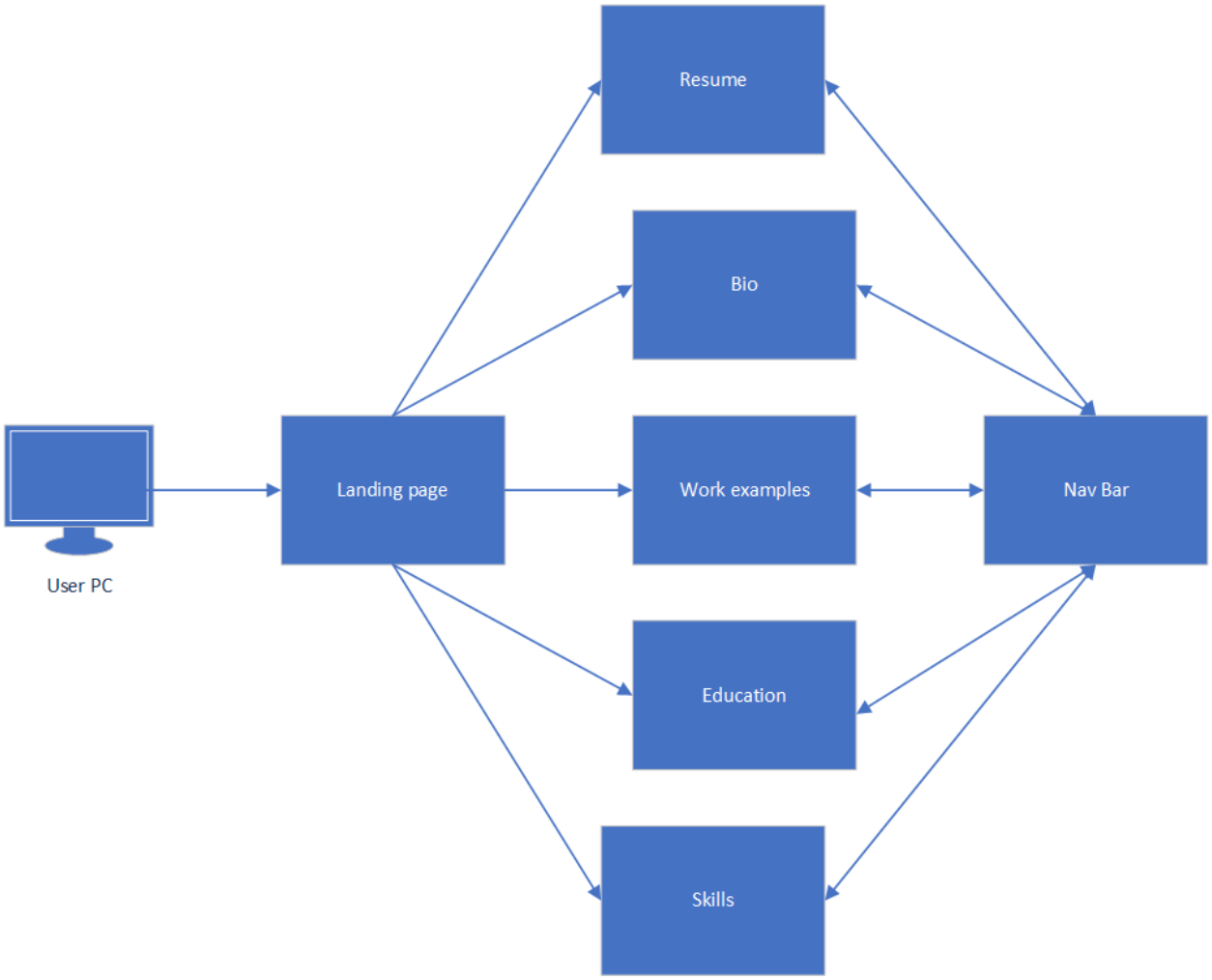


File structure:

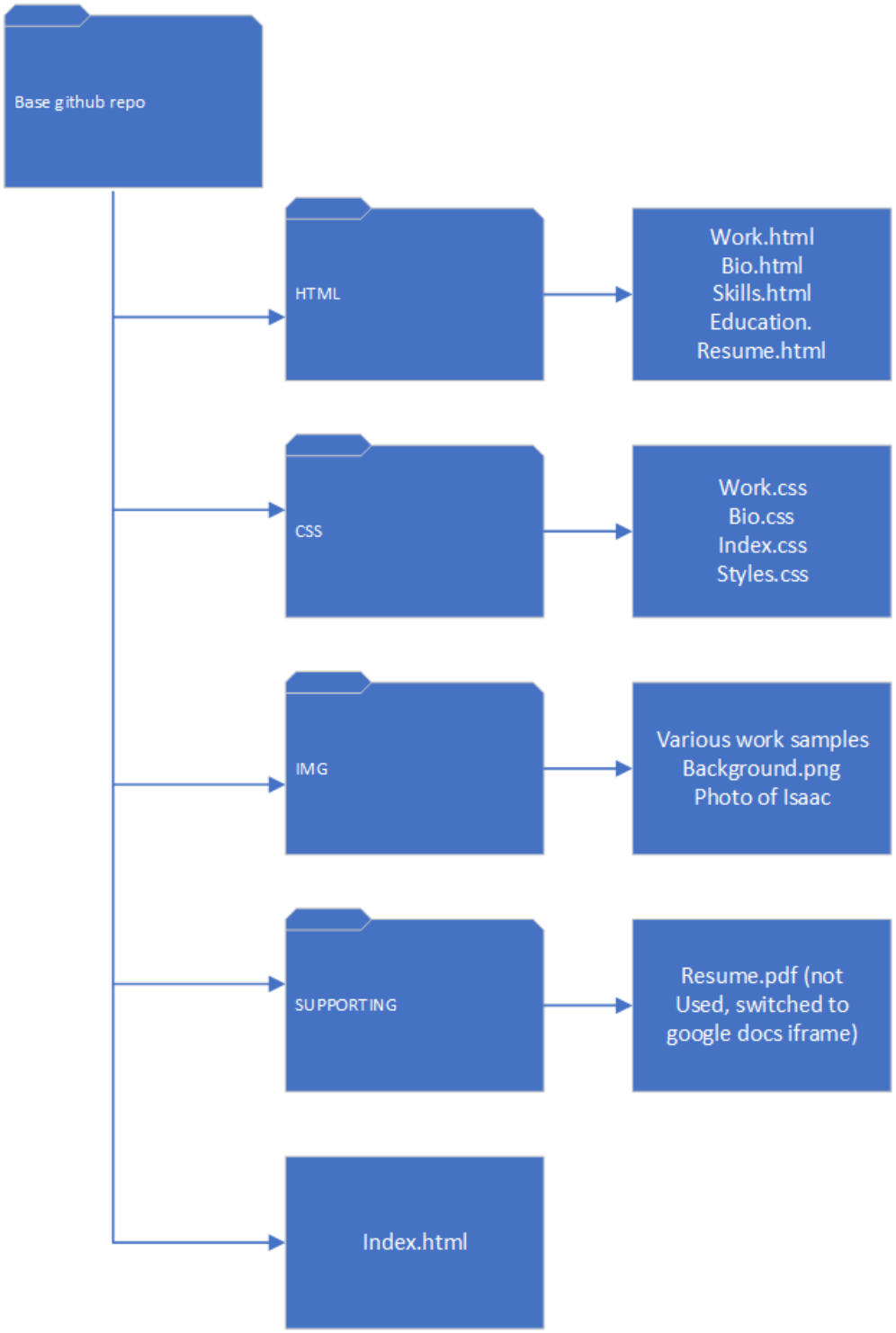


Storyboards:

Sitemap:

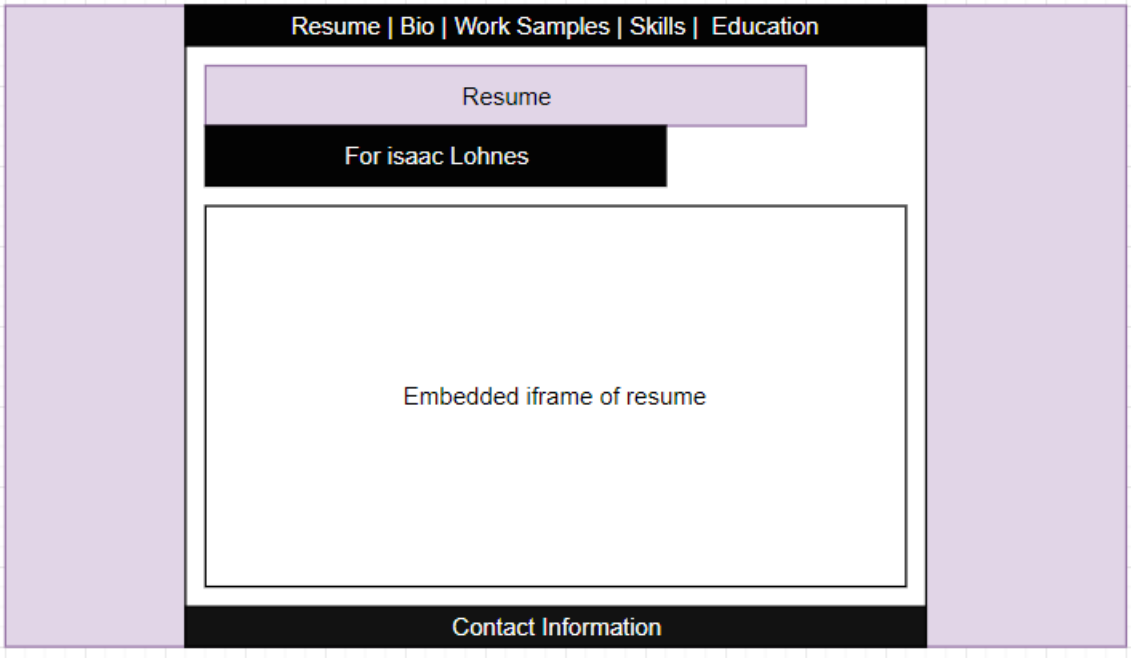


File structure:

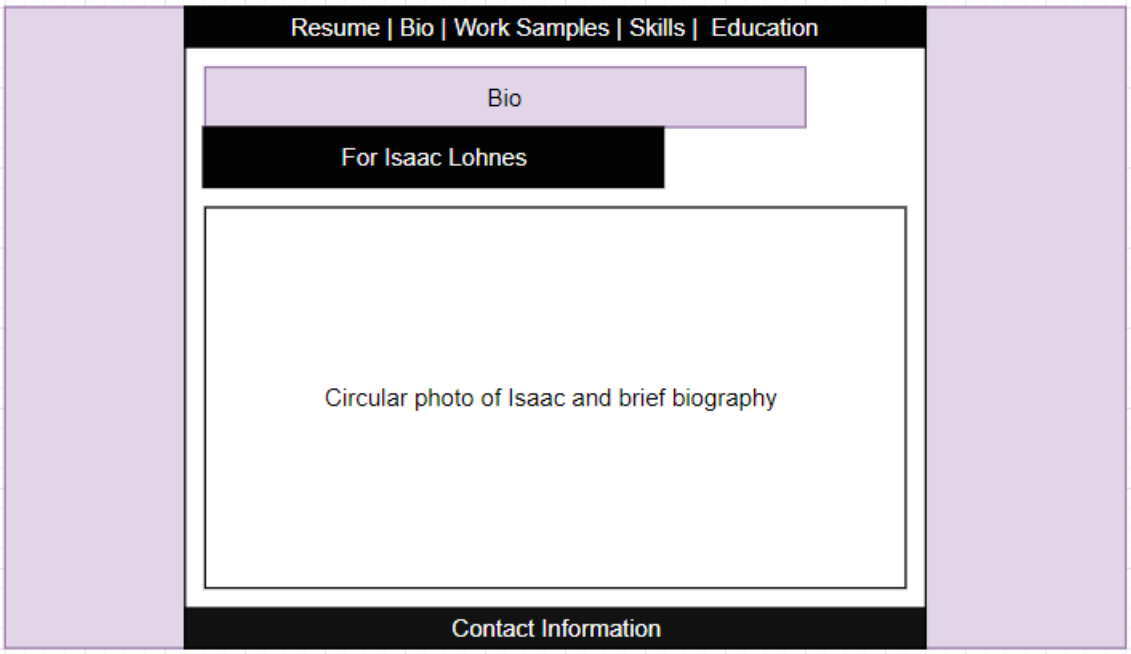


Landing page:

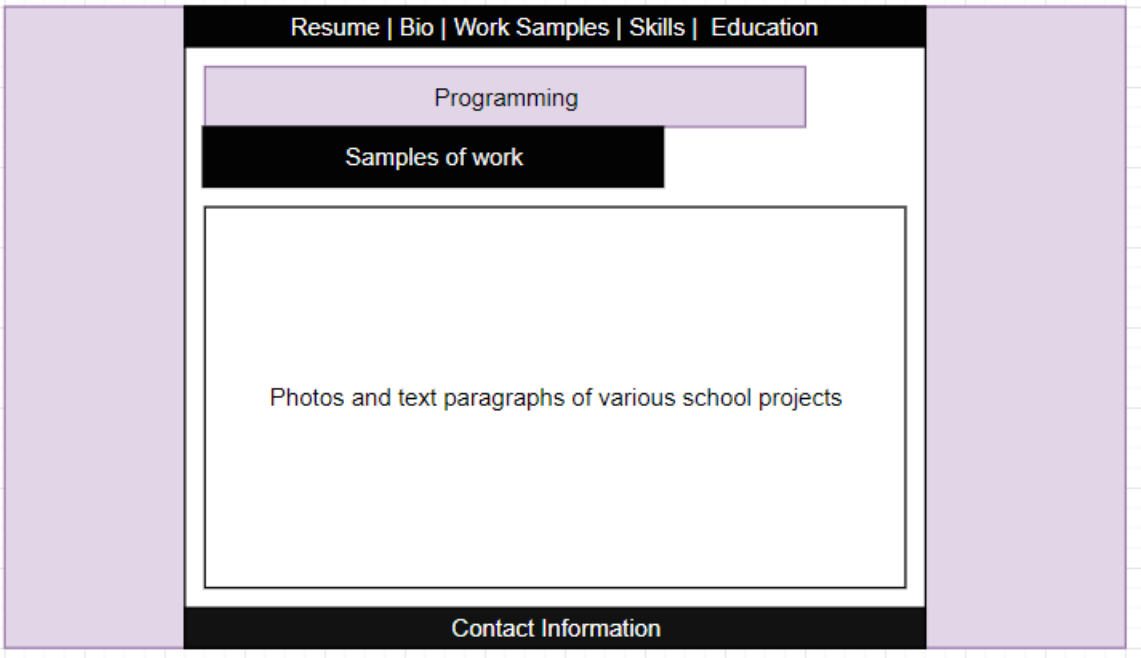
Resume page:



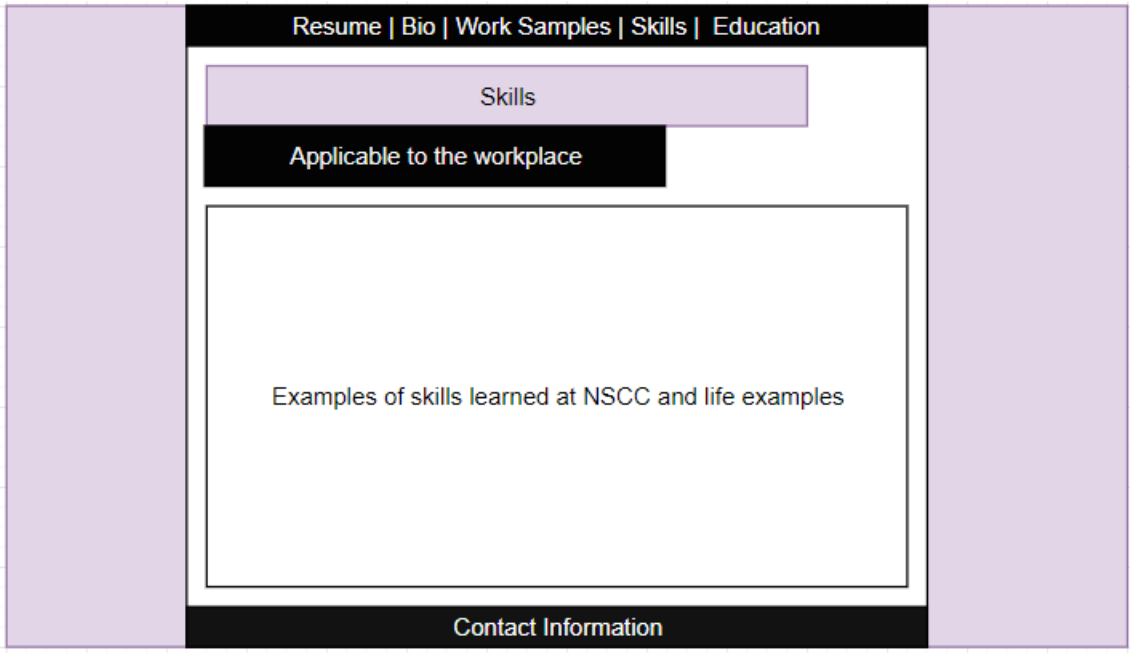
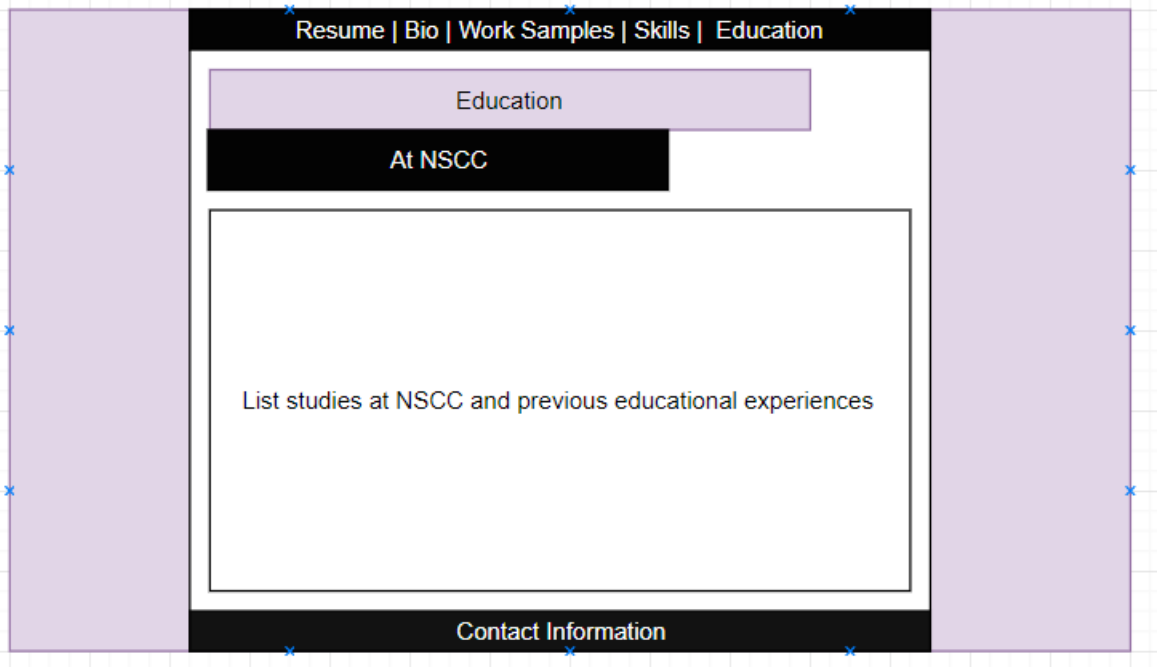
Bio page:



Work samples page:



Skills page:



Education page: