* Semantic Elements
  + Importance of Semantic Markup
  + Examples in 5+
* Site Structure and Navigation
  + Justification of content arrangement
  + Nav diagram
  + Connection to UX
* UI, Theme, and Colour scheme
  + CSS Rule explanation
    - Reset and Style.css
* Rationale behind design choices

## Semantic Elements

Semantic elements are a specific type of HTML5 element. Semantics is the study of linguistic meaning, and as such semantic elements are those which describe their contents. Examples of this are *<img>,* *<table>,* and *<article>*, all of which describe their purpose (an image, a table, and an article respectively). Examples of non-semantic elements would be *<span>* or *<div>,* as both of those elements state nothing about their contents innately.

Many semantic elements don’t have any innate effect on how the website is displayed (Take *<article>* or *<section>* for example). So why use them? Generally, semantic elements are used for accessibility. Most people will view the website as intended, and so in those cases it doesn’t matter if you use a *<div>* or a *<section>*. But some people will use screen reader software to navigate the internet. And for them, being able to immediately tell what an element does is very useful.

For example, let’s take the *<nav>* tag, which is used to set up a navigation section (e.g. “About me”, “Projects”, etc.). To the common user, It doesn’t matter if you use a *<nav>* tag or a *<div>,* as neither have any innate effect on the way the page is displayed. But to someone using a screen reader, Its much easier to use the navigation bar if it’s labelled as such. Otherwise, It will take them far more time to navigate your website.

Semantic elements used in this project include:

* *<section>*
* *<article>*
* *<nav>*
* *<img>*
* *<header>*
* *<footer>*
* *<h1>*, *<h2>* and *<h3>*

These can all be described as semantic as they are all self-descriptive (remember that h1, h2, h3 are all short for Header 1, Header 2, Header 3.)

## Examples of Semantic Elements in the HTML5 markup

This shows how the navigation section of the website has been marked up. I’m using an unordered list, which is contained within the *<nav*> tag. It’s worth noting also that I’m using an *<h1>* tag at the top, showing the title of the website within the navigation window.

A computer screen shot of text

AI-generated content may be incorrect.

Here you can see the front page of the website. (The navigation window shown above shows up here.) This is all wrapped in the *<header>* tag, as all of this is the header to the website (navigation, external links, etc). A screen shot of a computer code

AI-generated content may be incorrect.

Here you can see the “About me” section of my website. Of course, this has all been wrapped in a *<section>* tag, because it is a section of my website! There’s nothing special about *<section>* and I could have used a *<div>*, but this is a more accessible solution. My other sections (Skills, Blog, Projects, etc) Are also wrapped in *<section>* tags, just like this one.

A screenshot of a computer program

AI-generated content may be incorrect.

Each blogpost in my Blog section are wrapped in the *<article>* tag. Each of them are by definition articles, so this makes sense. I considered using *<section>* here, but I think *<article>* is more specific to what a blogpost is, and as such it’s better than *<section>.*

A screen shot of a computer program

AI-generated content may be incorrect.

At the bottom of the main page, there is a section which contains my contact details, it’s far smaller than the rest of the sections, so why not class it as a footer? I’m using the *<footer>* tag instead of the *<section>* tag here. While I could have used *<section>* or *<footer>* (as both are semantic elements) I went with *<footer>* as this might signal to a screen reader that this is nearing the end of the page.

A screen shot of a computer screen

AI-generated content may be incorrect.

## Site Structure and Navigation

The website is structured as follows: There are eight sections on the website:

* Homepage
* About me
* Skills
* Education
* Projects
* Blog
* Contact
* Admin

Of these sections, five of them (Homepage, About me, Skills, Education, Contact) are on the same page. You need only scroll down from the Homepage to find the Skills section, for example. The other three are external. I decided to place the Projects and Blog sections on their own webpage as they will have an arbitrarily large amount of content which will grow as I increase my portfolio. This means that if they were all on one webpage, they would take up loads of space, and it would take ages to scroll through them. The admin section is on a separate page as it doesn’t contain any useful information to the common user and thus shouldn’t necessarily be displayed to them (this page is accessible via a link at the bottom of the homepage.)

There’s also a secret, ninth section on the website, my “Add Blogpost” section. Currently, there isn’t any way to navigate to it, though, because it’s going to be linked to the admin page (which doesn’t yet work, as that requires a backend).

Homepage

The Homepage contains a navigation menu. This is a horizontal list of items, starting with my name “Luka FB Lepkowski” and then a set of hyperlinks (using the *<a>* tag). These hyperlinks point to each other section of the website (other than Admin), and can be used either to automatically scroll to a section, or link to a section depending on if that section is internal or external.

The homepage also contains a vertical list of hyperlinks to my various social media platforms (github, itch.io, linkedin, bluesky). While there is a section for contact at the bottom of the page, having these platforms immediately accessible makes it easier for someone coming to the website to quickly see what I’ve done, on a platform they are already potentially familiar with.

About me

The about me section contains a photograph of myself, my name and epithet, and a few paragraphs of me talking about my ambitions and interests. I chose a silly photo of myself as I think showing I am a more creative person than a professional, business type is important considering I want to primarily work in game development. Showing I am a character is important.

Skills

The skills section consists of three lists (Backend, Game Development, and Frontend) all of which contain a set of photos and labels for those photos. These photos are the logos of certain programming languages or frameworks (PHP, OpenGL, JS etc), labelled with their name (as some of the logos might not be the most known, LWJGL for example.) I didn’t want to focus too much on my skills, but showing a breadth of knowledge is important here.

Education

The education section is quite small. It’s just got a little sentence or two about where I am currently enrolled and where I have been educated in the past. I think in software development showing a portfolio is far more important than where you’ve been educated, but to not show this would look sketchy.

Projects

The projects section is on a separate page, as I expect it might grow in the future. It’s a big list of project *<section>* elements, each of which containing a Name, Release Date, Description, and links to source code / download. All the content here is taken from my old website, which ran off PHP. This was originally loaded via a CSV file, which I intend to implement in the second phase of this project.

Blog

Like the Projects section, the Blog is on a separate page, as I can expect it to grow in the future (despite only having one entry now.) It’s a list (or, at least, it will be) of *<article>* elements, all containing a Title, Publish Date and Content. I think loading all of this from a CSV file in the future could be a good idea, as this way I could shorten them to a “Read more” sort of thing.

Contact

This is a mini section at the bottom of the homepage, containing my email, BlueSky address, and a repeat of each social media shown in the Homepage (they should also be here, as they’re points of contact). This is also where I’ve hidden the admin page access, although that’s just because it’s at the bottom of the page.

Admin

The admin page contains a log-in form, asking for a Username and Password field. That’s it for now, and the form doesn’t do anything right now because we haven’t done the backend section.

Add Post

The add post page is essentially identical to the admin page, other than the form itself. It asks for a

## Justification of content arrangement

I’ve already written about it a bit previously, but I think putting the Projects and Blog section on separate pages is wise. They both might grow to contain an extremely large amount of data, and as such they will need to be contained. It’s important that people aren’t forced to trod through it when trying to access other parts of the website. The about me section, and the skills section should be on the front page of the website, I believe, as they are more instantly accessible to users. More people will be interested in the About me section than the Blog, for example (and it’s only short). I’ve implemented a nice scrolling transition to these pages when using the navigation bar as well.

## Navigation Diagram

I’ve also made a navigation diagram for the site. It’s too wide to fit in this document sensibly, though, so it’s available online at [*https://spunky2.com/nav.png*](https://spunky2.com/nav.png), and should also be included within the submission zip file (Probably as nav.png). (You’ll have to zoom in, it’s quite detailed!)

This navigation diagram shows how users might navigate throughout the website, showing hyperlinks via dashed lines. It shows the four main web pages I’ve got, and their various sections. Please take a look. (Importantly, there’s no “Add form” section on the navigation diagram. This is because you can’t actually navigate to it yet, so it seemed redundant.)

## User Experience

It’s important to view web design holistically and look at the big picture when putting together a website. Navigation being functional and intuitive isn’t just for the purposes of the website technically working. It’s also important for users to not find any frustration when using the website. They should *enjoy* using the website.

I’ve gone with a minimalist colour scheme and style for my website. I think that less is more in game design, and this too applies to web design. I was very cautious not to overwhelm the user. I’m using a light, inoffensive font, and I’m using a few underlines and drop shadows for detail, nothing more. It can look quite tacky if a website has too much going on visually, and I think keeping it simple and elegant feels far more professional.

I also think adding loads of detail to the website might take away from the projects and skills I’m trying to show off. My portfolio should speak for itself. Any unneeded stimuli will essentially be a distraction from the purpose of the website.

## User Interface

When it comes to the User Interface, I’ve gone with a minimalist design. I’m using Google’s Courier Prime font. I think this font is quite fun and satisfying, but it’s not too offensive and doesn’t do too much to upset. It looks somewhat like Fira Code & Consolas, commonly used in programming, which is thematic and might subconsciously remind the user what type of person I am.

I’ve made the theming and style of the website very basic. It’s a white background, with slight grey dividers between each section. There’s a feint drop shadow behind some box elements, like projects or blog posts, or images, and a small shadow behind important text (titles, subtitles, names, etc). I’ve done boxes around hyperlinks, with dashed lines for inbound links (To different sections / pages on the website) and solid links for outbound links (e.g. Github, Linkedin, etc).

I didn’t want to overwhelm the user at all when coming to the website. All this soft, safe design is to make sure of this. While I think the website looks nice, I find it difficult to imagine someone finding it offensively bad. The homepage is designed with this in mind, too: It only has the basics to it. It has my name, a navigation bar, my social medias, and that’s it. No images or colours or designs or anything. As you go deeper into the website, It doesn’t feel quite as impactful as the front page to onload loads of information, so I start to add quite a bit of text and information later on, but the homepage is pretty blank by design.

I’ve gone with the blank colour scheme for a few reasons. I think, firstly, besides being reminiscent of Consolas and Fira Code, Courier Prime also reminds the user of typography. Its ancestor, Courier was the primary font used on typewriters. Using blank white might remind the user of paper and typewriter. Besides wanting to make the website feel simple and easy to parse, I just did this because I thought it looked cool and felt like me.

## CSS Styling

First off, I want to make it clear that I haven’t used any generative AI for this section of the project (or any section of it, for that matter).

All of the sizes are using the “em” unit, as this leads to the site being scalable, making it much more viewable on iPhone, for example.

A screen shot of a computer code

AI-generated content may be incorrect.This little block of CSS defines the way in which the auto scroll feature scrolls down the page. I’ve set it to smooth, because it looks nicer than sharply snapping to a point in the page. This is for the navigation bar (styled later on in this section.)

A screenshot of a computer program

AI-generated content may be incorrect.This is where I’m defining the header tags. I’ve left 4 through 6 undefined, as I found when marking up the website I only needed 1 through 3. The first two are actually defined the same way: Originally, they had a size difference, but I decided that it looked better keeping them the same. They both have a grey shadow and are in bold. The h3 tag is similar, but it has a dashed underline below it. (It’s not bold also.) This features heavily in the skills area, and words well as a subdivider between sections.

A screen shot of a computer screen

AI-generated content may be incorrect.This is where I’ve styled list elements. I’ve got two aways of styling them here: Horizontal and vertical. I can change which one a list is via class, which comes in handy. Semantically defining this rather than specifically saying each purpose e.g. nav-bar-list OWTTE is very useful. It’s simple: For the horizontal list I have them display in an inline block, meaning they’re horizontal. (I could have also used a flexbox here, but this was simpler and has the same effect AFAIK). The vertical one doesn’t have that as lists are by default vertical. I put a relevant margin on each list element as to make sure they’re spaced properly.

A screen shot of a computer

AI-generated content may be incorrect.This is where I’ve styled hyperlinks. Similarly to how I’ve defined lists, I’ve defined two types: Outbound and Inbound links. Outbound is for links external to the website e.g. GitHub whereas inbound is for stuff link navigation through the website. Each has a black border, box shadow, and are centrally vertically aligned. The only difference is that the border on outbound links is solid, and on inbound links is dashed.

A screen shot of a computer program

AI-generated content may be incorrect.This is a little section for site navigation. I’ve defined the nav tag here, and just made sure it centres all the text in the nav bar. The #backtotop is used for “Back to top” hyperlinks, which are in each section and are used to return to the homepage. (In sections where the homepage isn’t at the top e.g. on a different webpage, this is changed to “back to front page”). Obviously these are quite different tags, but they’re both to do with navigation so I’ve put both of them here.

A screen shot of a computer program

AI-generated content may be incorrect.This is where the main structuring of the site is defined. First, the body: I’ve got a almost-white colour as the background. This will be the colour in between sections, so not the main website background really. The body is defined as a flex display, meaning each section will be in a flexbox. (The flex-direction is column, meaning downwards)

Each header, section, and footer is essentially identical: They all have a white background, 100% minimum height, and 1em of padding and bottom-margin. There are a few exceptions here, though. The footer only takes up 30% of space, as it doesn’t have as much content on it. There’s also a #nobottom type of section. This doesn’t have a bottom margin, meaning there won’t be a grey bar below it. This is used for when the final section is a *section*, and not a *footer*. This doesn’t show up on the main page, but does appear on the admin pages and the project and blog.

A screen shot of a computer screen

AI-generated content may be incorrect.This is where I’ve defined columns. This is a little feature I’ve used throughout the page to split an element into however many columns I wanted. It’s essentially just a generic flexbox I can use. It shows up a lot in the markup!

Semantically defining “column” is very useful, rather than having to define loads of specific use cases which are all identical, e.g. “about-me-header-section-column”. This saves loads of time.

A computer screen shot of a black screen

AI-generated content may be incorrect.Here’s the custom style rules for my about me section. It’s just to do with my profile picture: To split up the content into three, I’m using columns as defined above.

*A screen shot of a computer program

AI-generated content may be incorrect.*#profilePicture is actually a div containing an image, not an image itself. I’m using this div to centre that image, as well as my title *“Luka Lepkowski (he/him/they) etc…”* The actual *<img>* tag is defined with a slight box shadow, with a width of 10em, and a slight margin around it.

Here's my skills styling.

Each set of skills (Backend, Frontend, Game Development) is a flexbox row. It’s got a flex-direction of row-reverse, meaning it’s going from right to left. This helps fill out the page, as otherwise it would have everything packed into the left.

Each skill is an image and its description. Each image is a logo of a language or application I’ve learned how to use, and the description is just the name of that application. Each skill-image has a width and height of 6em and has a box-shadow around it.

Each description has central text and is in italic. This helps make the text not seem so important, as to most people (who know the logos) the text won’t be too useful (Of course, some people will need it.)

A screen shot of a computer program

AI-generated content may be incorrect.Here’s all my styling for my Project section. Each project is in a large flexbox, in a column going down the page. Each individual project is a box containing a Title, Description, Release Date, Download Link, and Project Type (Game, Website, etc.) The name is in bold and has a text-shadow. The type is in gray italics, because it’s not quite as important as the title (It can be gleamed from the title alone) The release date is in italic, but not in grey (While it’s not as important, users can’t derive it from other information)

Each of these “tokens” has a little margin to their right. This helps with spacing everything out.

Some of this information is in a row. I’m using a nested row flexbox, with a little margin around it for this.

A screen shot of a computer

AI-generated content may be incorrect.

Here’s where I’ve styled my blogposts. They’re essentially the same as my projects: The only difference being that there’s not quite as much content to a blogpost. All they have is a Title, Release Date, and Content. They work essentially the same though: Each one is in a large flexbox column, The date is in italic, the title is in bold and has a shadow, the header is a nested flexbox, and each element in the blogpost has a little margin around it.

Each blogpost also has a little shadow around it.

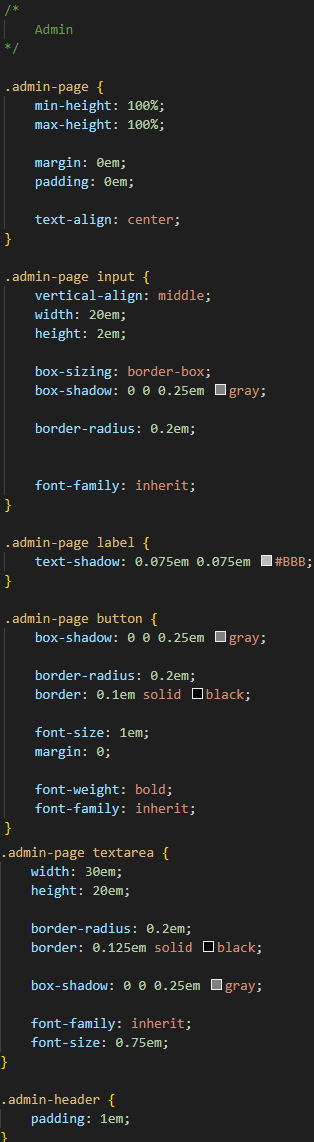
A screen shot of a computer code

AI-generated content may be incorrect.Here’s the styling for my contact section. There isn’t really any here! The contact section is very simple, so I’m able to use my previous definitions of h1, h2, h3, etc for it. The #admin here is just for a button which allows access to the admin log in page. This is only in the contact section as the contact section is a footer, and I wanted this button to be out of the way (thus it floats right).

A screen shot of a computer

AI-generated content may be incorrect.Here’s the styling for my section on education. This section is also used to show off my experience.

There’s not much styling here at all. I’m just making sure that it has a minimum height of 30% (rather than the inherited 100%) because this section is quite small on content and doesn’t have as much to show.

Here’s the styling for all of my admin pages.

Each page has a forced height of 100%, meaning it always takes up your full screen (no need for scrolling.)

The rest defines different form elements. Inputs have a set width and height and font. Most form elements have been given the same Gray box-shadow and black border (Other than input, as it seems this was inherited)

Each label has a subtle text-shadow.

The textarea has a default width and height (which can be changed on most web browsers with a UI) and has a slightly smaller font than default (0.75 vs. 1)

The header has a bit of padding around it.

## Design Rationale

It's important to note that, just because the website is simple, I don’t think it’s boring to look at. There’s lots of great content on the website which people will read, and I think the few images throughout the website will carry their load. For example, looking at the “About me” section, the photo I’ve used of myself is quite different to the formal, uptight feel the homepage might give you. I think more than anything it’s important to convey the character that I am, and I think doing this with the website can show both my serious and unserious sides and specifically show that they can work in unison.

In the about me section, I’ve got a title by my photograph, reading *“Luka Lepkowski (he / him / they) game designer and developer”* Two things to unpack here is the set of pronouns as well as the game designer title I’ve given myself. I think that in the modern world, showing your pronouns has become a standard practice in the industry, and as such it’s important to show. It’s also definitely a younger-person thing. I think it’s important I do show my age, without necessarily giving them a birthday. “Young and creative” is much better than “19 and naïve”. I’ve seen a lot of people giving themselves the title “aspirational” [blank] (e.g. “aspirational game designer”) When I think this is really the wrong approach. If I want to be a game designer, and I’ve made games, then I am one. Saying “aspirational” is underselling myself.

I think my personal brand must be that of a creative person. I don’t think selling myself as a corporate type is a very good idea, considering I’m looking to work in a creative industry. At the same time, game developers require a lot of different skills, and you have to be pretty serious about games if you want to make them. You have to be seriously unserious, and I think that’s the exact point I’m trying to hit with my design. My blog post is essentially just a rant about Balatro, but there are loads of points specifically about game design in there. I love thinking about game design. That’s the kind of person I am, so showing that is so important. In the projects section, I’m able to really show my passion. I’ve given a little description on each of them, and I’m not afraid to show off a little bit and gush over how cool my achievements are. That’s what kind of person I am, and I think showing that confidence is important. I want people who come to my website to get my vibe.