

Documentation for Virtual CV

Stefan du Plooy

My goal was to present my skills and accomplishments clearly and concisely, allowing visitors to navigate effortlessly through my professional journey. The clean, intuitive interface is designed to highlight key information without overwhelming the viewer user with crazy features that are an eyesore to look at.

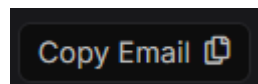
Hosting

Hosting my virtual CV is made easy with GitHub pages. I created a new repository with the name “StefanDuPlooy.github.io” and uploaded all the files needed for my virtual CV. Using this naming scheme allows a person to host their static website on GitHub for free and can be accessed by other people by sharing the provided link. Your website is also automatically deployed when you push any changes to the branch.

The necessary files needed for the website to work are: “index.html”, “style.css”, “profile.png” under “assets”, and “cv.pdf” under “documents”.

Innovation

Dynamic Email Copying:



```
<script>
document
  .getElementById("copyEmailBtn")
  .addEventListener("click", function () {
    // The email address you want to copy
    var email = "stefanduplooy3@gmail.com";

    // Create a temporary input element
    var tempInput = document.createElement("input");
    tempInput.value = email;
    document.body.appendChild(tempInput);

    // Select the email address
    tempInput.select();
    tempInput.setSelectionRange(0, 99999); // For mobile devices

    // Copy the email address
    document.execCommand("copy");

    // Remove the temporary element
    document.body.removeChild(tempInput);

    // provides some feedback to the user
    alert("Email address copied to clipboard!");
  });
</script>
```

This script creates a temporary input element, copies the email to it, selects the text, executes the copy command, then removes the temporary element and notifies the user that the email has been copied to their clipboard. It's an innovative way to provide easy access to my contact information.

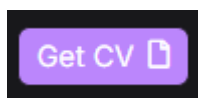
Animated Page Load:

```
body {  
  background-color: #171717;  
  padding: 20px;  
  animation: fadeInAnimation ease 2s;  
}
```

```
@keyframes fadeInAnimation {  
  0% {  
    opacity: 0;  
  }  
  100% {  
    opacity: 1;  
  }  
}
```

This creates a smooth fade-in effect when the page loads, enhancing the visual appeal of my virtual CV.

CV Easy Access:



```
<div class="buttons">  
  <a href="documents/cv.pdf">  
    <button class="pdf">Get CV <i class="far fa-file"></i></button>  
  </a>
```

Clicking or pressing on this button will redirect the user to my actual CV in PDF form to be viewed or downloaded. The CV PDF is saved in the repository and linked through the button.

Interactive Elements:

```
a:hover {  
  color: #bb86fc !important;  
}
```

```
.about .left .info .buttons button.pdf:hover {
  background: #bb86fc;
}
```

```
.about .left .info .buttons button.email:hover {
  background: #000;
}
```

```
.achievements .right .item:hover a {
  color: #bb86fc;
}
```

These hover effects change the colours of buttons and links which will provide visual feedback to users, improving the interactivity of my virtual CV.

Web Technologies

HTML5:

```
<nav>
  <div class="links">
    <a href="#achievements-list">Achievements</a>
    <a href="#about-me">About Me</a>
    <a href="#portfolio">My Portfolio</a>
  </div>
</nav>
<footer>
  <a href="#">My Portfolio</a>
  <div id="portfolio" class="socials">
```

Semantic tags improve the structure and accessibility of the document. `<nav>` clearly defines the navigation section, while `<footer>` marks the bottom section of this page. This can help search engines and screen readers understand the content structure.

```
<meta charset="UTF-8" />
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
```

This meta tag ensures that the page renders correctly on various devices by setting the viewport width to the device's width and establishing the initial zoom level.

CSS3:

CSS3 was used to improve the aesthetics of my virtual CV and ultimately improve the use experience. Flexboxes, grids, media queries, transitions, and keyframe animations were used to achieve this and will be looked at in more depth under **Compatibility** and **Navigation**.

JavaScript:

As referenced under **Innovation**, I used a small piece of JavaScript in my HTML to be able to copy my email into the user's clipboard by clicking a button through the use of event listeners and DOM manipulation.

External Libraries:

- Font Awesome for Icons:

```
href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.4/css/all.min.css"
```

This links to the Font Awesome library, allowing me to easily include scalable vector icons in my virtual CV, enhancing its visual appeal and user interface.

- Google Fonts for Typography:

```
@import url("https://fonts.googleapis.com/css2?family=Inter:wght@100..900&display=swap");
```

This imports the Inter font family from Google Fonts, allowing me to use a custom, web-safe font throughout my virtual CV for consistent and attractive typography.

Compatibility

Cross Browser Compatibility:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
```

This code uses standard HTML5, CSS3, and JavaScript features that are widely supported across modern browsers. For example: The HTML5 doctype and lang attribute ensure proper rendering across browsers.

Responsive Design:

```
@media screen and (max-width: 1220px) {
  .achievements {
    flex-direction: column;
  }
}
```

```
@media screen and (max-width: 576px) {
  .achievements .left .achievements-list {
    margin-top: 40px;
  }
}
```

```
@media screen and (max-width: 992px) {
  nav .links {
    padding: 0 40px;
  }
}
```

```
@media screen and (max-width: 768px) {
  nav .links a {
    font-size: 14px;
  }
}
```

Media queries for different screen sizes - These media queries adjust the layout and font sizes for different screen widths or resizing the browser size.

Flexible Units:

```
.about .right img {
  width: 200px;
  height: 200px;
  border-radius: 50%;
  border: 8px solid #2c2c2c;
  object-fit: cover;
  margin-top: 70px;
}
```

I use relative units like percentages for better scalability.

Viewport Meta Tag:

```
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
```

This ensures proper rendering on mobile devices by setting the viewport width to the device width and initial scale to 1.

Flexbox and Grid for Adaptive Layouts:

```
.about {
  display: flex;
  align-items: center;
  justify-content: space-between;
}

.achievements .right {
  flex: 1;
  display: grid;
  grid-template-columns: 1fr 1fr;
  grid-template-rows: 40% auto;
  gap: 20px;
}
```

These modern CSS layout techniques adapt well to different screen sizes.

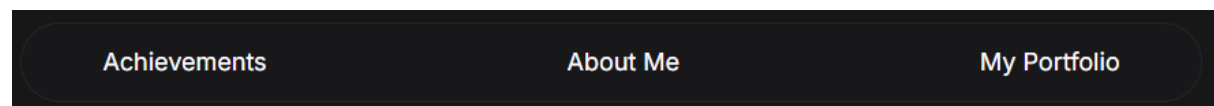
Fallback Fonts:

```
font-family: "Inter", sans-serif;
```

While I am using a Google Font, I have provided a fallback to sans-serif, ensuring text remains readable even if the custom font fails to load.

Navigation

Persistent navigation bar:



```
<nav>
  <div class="links">
    <a href="#achievements-list">Achievements</a>
    <a href="#about-me">About Me</a>
    <a href="#portfolio">My Portfolio</a>
  </div>
</nav>
```

This navigation bar remains visible at the top of the page, providing the user with easy access to main sections.

Smooth scrolling:

```
html {
  scroll-behavior: smooth;
}
```

This CSS property ensures smooth scrolling when users click on navigation links, enhancing the user experience.

Clear section delineation:

```
.about {
  display: flex;
  align-items: center;
  justify-content: space-between;
  padding: 60px 42px;
  background: #18181b;
  border: 1px solid #27272a;
  margin-top: 20px;
  border-radius: 14px;
}
```

```
.achievements .left {
  flex: 1;
  background: #18181b;
  padding: 60px 40px;
  border-radius: 14px;
  border: 1px solid #27272a;
}
```

Distinct background colours and borders clearly separate different sections of the virtual CV.

Heading hierarchy:

```
<h2>I'm Stefan du Plooy</h2>
```

```
<h3><i class="fas fa-circle"></i> Achievements</h3>
```

```
<h5 class="active">2024 (Semester 1)</h5>
```

Proper use of heading tags (h2, h3, h5) helps to structure the content and guide users through the information.

Visual cues:

```
.achievements .left .achievements-list .item ul li::before {
  content: "•";
  color: #bb86fc;
  font-weight: bold;
  display: inline-block;
  width: 1em;
  margin-left: -1em;
  position: absolute;
}
```

Bullet points in a distinctive colour help to visually organize list items.

Interactive elements:

```
a:hover {  
  color: #bb86fc !important;  
}
```

```
.about .left .info .buttons button.pdf:hover {  
  background: #bb86fc;  
}
```

Hover effects on links and buttons provide visual feedback to users, indicating interactivity.

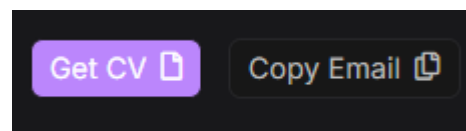
Consistent layout:

```
.achievements {  
  display: flex;  
  gap: 20px;  
  margin-top: 20px;  
}
```

```
.achievements .right {  
  flex: 1;  
  display: grid;  
  grid-template-columns: 1fr 1fr;  
  grid-template-rows: 40% auto;  
  gap: 20px;  
}
```

The consistent use of flex and grid layouts creates a predictable structure that's easy for users to navigate.

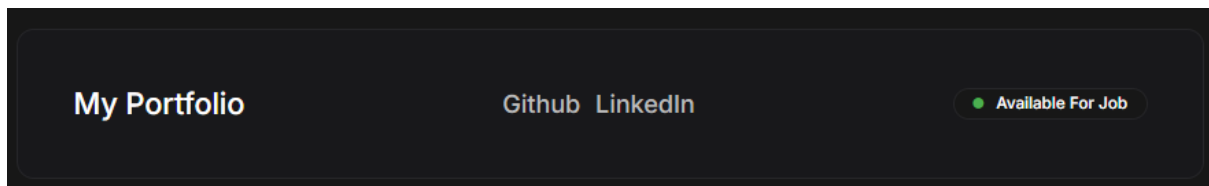
Clear calls-to-action:



```
<div class="buttons">  
  <a href="documents/cv.pdf">  
    <button class="pdf">Get CV <i class="far fa-file"></i></button>  
  </a>  
  <button id="copyEmailBtn" class="email">  
    Copy Email <i class="far fa-copy"></i>  
  </button>  
</div>
```


Clearly labelled buttons with icons make it easy for users to take desired actions.

Footer navigation:



```
<footer>
  <a href="#">My Portfolio</a>
  <div id="portfolio" class="socials">
    <a
      href="https://github.com/StefanDuPlooy"
      target="_blank"
      rel="noopener noreferrer"
    >Github</a>
  >
  <a
    href="https://www.linkedin.com/in/stefan-du-plooy-96906b206"
    target="_blank"
    rel="noopener noreferrer"
  >LinkedIn</a>
  </div>
  <a class="online" href="#"
    ><i class="fas fa-circle"></i> Available For Job</a>
  >
</footer>
```

The footer provides additional navigation options and links to my social profiles.