I have converted your 5-page guide on building and deploying the portfolio into a ready-to-export format for PDF.

Page 1 – Project Setup & Folder Structure

Overview: Build an interactive, modern portfolio showcasing skills, projects, and contact info with animations and dark/light mode.

Tools Required: - VS Code - Modern Browser - Git & GitHub (optional for deployment) - Basic knowledge of HTML, CSS, JavaScript

Folder Structure:

```
portfolio-project/
   ├─ index.html
   ├─ style.css
   ├─ mediaqueries.css
   ├─ script.js
   ├─ assets/
       ├─ profile-pic.jpeg
       ├─ about-pic.jpg
       ├─ project-1.png
       ├─ python.jpg
       ├─ html.png
       ├─ css.png
       ├─ javascript.jpg
       ├─ react.png
   └─ README.md (optional)
```

Instructions: 1. Create a project folder. 2. Add the HTML, CSS, JS files. 3. Add your assets in the assets folder.

Page 2 – HTML Structure

Sections: - Navbar - Hero - About - Skills - Projects - Contact - Footer

HTML Snippet (Hero Section Example):

```
<section id="profile" class="section hero reveal">
  <canvas id="particles"></canvas>
  <div class="hero-content tilt">
    <h1>Hi, I'm <span class="glow">Shaik Jafrulla</span></h1>
```

```
    <p class="hero-subtitle"><span id="typing"></span></p>
    <div class="btn-container">
      <a href="#projects" class="btn primary">View Work</a>
      <a href="./assets/resume.pdf" class="btn secondary" target="_blank"
download>Download CV</a>
    </div>
  </div>
  <div class="hero-img">
    <img src="./assets/profile-pic.jpeg" alt="Profile Image" loading="lazy">
  </div>
</section>
```

Page 3 – CSS Styling

Global Styles:

```
body {
  font-family: 'Poppins', sans-serif;
  background: #0f172a;
  color: #e5e7eb;
  transition: background 0.3s, color 0.3s;
}
body.light { background: #f8fafc; color: #020617; }
```

Skills Grid (CSS Fix for Images):

```
.skills-grid {
  display: grid;
  grid-template-columns: repeat(auto-fit, minmax(160px, 1fr));
  gap: 2rem;
}
.skill-card img {
  width: 100%;
  height: auto;
  object-fit: contain;
  border-radius: 10px;
}
```

Page 4 – JavaScript Interactivity

Mobile Menu Toggle:

```
function toggleMenu(){
  document.querySelector('.nav-links').classList.toggle('open');
}
```

Scroll Reveal Animation:

```
const reveals = document.querySelectorAll('.reveal');
window.addEventListener('scroll',()=>{
  reveals.forEach(el=>{
    if(el.getBoundingClientRect().top < window.innerHeight-100){
      el.classList.add('active');
    }
  });
});
```

Typing Animation:

```
const typingEl = document.getElementById('typing');
const texts = ['Python Developer','Frontend Developer','UI/UX Enthusiast'];
let i=0,j=0;
function type(){
  if(i<texts.length){
    typingEl.textContent += texts[i][j];
    j++;
    if(j==texts[i].length){ j=0;i=(i+1)%texts.length; typingEl.textContent=''; }
    setTimeout(type,150);
  }
}
type();
```

Dark/Light Toggle:

```
const themeToggle = document.getElementById('themeToggle');
themeToggle.addEventListener('click',()=>{
  document.body.classList.toggle('light');
});
```

Page 5 – Advanced Features & Deployment

Particle Background:

```
// Use particles.js or custom canvas code for animated background
```

Link Blog & YouTube:

```
<p>Blog: <a href="https://shaikjafrulla.blogspot.com/" target="_blank">View
Content</a></p>
<p>YouTube: <a href="https://www.youtube.com/@yourchannelname" target="_blank">Watch
Videos</a></p>
```

Lighthouse Tips: - Optimize images (use `loading="lazy"`) - Minify CSS/JS - Use semantic HTML - Enable caching for assets

Free Hosting: - GitHub Pages - Netlify - Vercel

Deployment Steps (GitHub Pages): 1. Push repo to GitHub. 2. Settings → Pages → Deploy from main branch. 3. Access your portfolio via `https://username.github.io/repo-name`.

Final Tips: - Test on mobile & desktop - Check performance & accessibility - Update content regularly