

The Optimized STEM Résumé: A Framework for Human and Algorithmic Appeal

Part I: Foundational Principles of Modern Résumé Architecture

1. The Dual-Audience Imperative: Reconciling Human Psychology and Machine Logic

The modern STEM résumé is a document with a bifurcated audience: it must first satisfy a machine before it can persuade a human. Success in a contemporary job search hinges on understanding and addressing the distinct requirements of both. The primary challenge lies in resolving the inherent conflict between the visually driven, pattern-seeking nature of human cognition and the rigid, linear parsing logic of Applicant Tracking Systems (ATS). The optimal solution is not a compromise that serves neither audience well, but a sophisticated design that presents tailored experiences to each from a single, unified source.

The Human Reviewer: The Six-Second Impression

Human recruiters and hiring managers operate under significant time constraints, often spending as little as six seconds on their initial review of a résumé. During this brief window, their goal is to rapidly assess a candidate's viability. Their visual scanning patterns typically follow an "F-shape," focusing on the top of the document, then scanning down the left side

for keywords and job titles before making brief horizontal sweeps across compelling lines.¹

This cognitive shortcut means that a résumé's design must function as a clear and efficient information-delivery system. Key principles for capturing human attention include:

- **Visual Hierarchy:** The most important information, such as the applicant's name and current job title, should be the most visually prominent elements on the page.² This is achieved through strategic use of font size, weight, and capitalization.
- **Whitespace:** Ample margins and spacing between sections prevent cognitive overload and guide the reader's eye through the document, making it appear organized and scannable.¹
- **Clean Typography:** The choice of font significantly impacts readability. Standard, professional fonts like Calibri, Arial, Georgia, or Garamond are highly legible in both print and digital formats and are universally recognized by computer systems.⁴ The optimal font size for body text is between 10 and 12 points.⁴
- **Strategic Emphasis:** Judicious use of bolding for job titles, company names, and key metrics draws the eye to the most critical data points, facilitating a rapid assessment of a candidate's experience and impact.⁶

A résumé that is dense, cluttered, or lacks a clear visual structure is likely to be discarded during this initial six-second review, regardless of the quality of its content.

The Algorithmic Reviewer: The Gatekeeper's Logic

Before a résumé reaches a human, it is almost invariably processed by an Applicant Tracking System (ATS). An estimated 90% of Fortune 500 companies utilize these systems to manage the high volume of applications they receive.⁷ It is a common misconception that ATS platforms are advanced artificial intelligences; in reality, most are parsing engines designed to extract text from a document, categorize it into fields (e.g., "Work Experience," "Education," "Skills"), and score it against the keywords present in the job description.⁷

These systems are highly effective at managing data but are notoriously brittle when encountering non-standard formatting. To ensure a résumé is parsed correctly, it must adhere to a set of strict technical guidelines:

- **Single-Column Logic:** The most reliable format for an ATS is a single, linear column of text that flows from top to bottom. Multi-column layouts, tables, and text boxes can cause the parser to read information out of order or miss entire sections of content.⁸
- **Standard Headings:** ATS parsers are programmed to recognize standard section titles like "Work Experience," "Education," and "Skills." Using unconventional headings (e.g., "My Journey," "Where I've Learned") can result in the content of those sections being

miscategorized or ignored entirely.⁷

- **No Graphics or Images:** Logos, photos, skill-rating graphs, and other visual elements are typically unreadable by an ATS and should be avoided.⁸
- **Information in the Body:** Content placed in the document's header or footer is often invisible to parsers. All critical information, including contact details, must be within the main body of the document.⁷
- **Keyword Alignment:** The ATS scores a résumé based on its relevance to the job description. Therefore, it is essential to incorporate exact keywords and phrases from the posting, such as specific technologies, skills, and job titles, naturally within the text.¹¹

Failure to adhere to these principles can render a highly qualified candidate invisible to the very system designed to identify them.

The Unifying Solution: Semantic HTML with Graceful Degradation

The seemingly contradictory requirements of human and machine reviewers can be reconciled through a principle of "graceful degradation." This approach involves building a résumé on a foundation of clean, semantic HTML that provides a simple, linear structure for the ATS. A layer of modern CSS is then applied to create a visually appealing and well-organized layout for the human reader.

The long-standing debate over single-column versus two-column résumés often misses this fundamental point. The critical factor for an ATS is not the final visual appearance but the underlying order of content in the document's source code. A modern web browser, and by extension a PDF generator like Puppeteer, can use CSS technologies such as Grid or Flexbox to visually rearrange elements without altering their sequence in the HTML.

This makes it possible to design a résumé with, for example, a visually distinct sidebar for contact information and skills. For the human reader, this is an efficient use of space that organizes information logically. However, in the underlying HTML, the content of that sidebar can be placed *after* the main content column (Summary, Work Experience). When an ATS parses this document, it ignores the CSS and reads the HTML in its native, linear order. It effectively "sees" a simple, single-column document, processing the most vital career narrative information first. This technique allows for the creation of sophisticated, modern designs that are simultaneously robustly ATS-compatible, solving the dual-audience imperative without compromise.

2. The Anatomy of a High-Impact STEM Résumé: Content, Structure,

and Quantification

Beyond formatting, the substance and structure of a STEM résumé are paramount. Each section must be meticulously crafted to convey expertise, impact, and alignment with the target role. The strategic ordering of these sections and the language used within them can significantly influence a reviewer's perception of the candidate.

Section Ordering Strategy

The optimal sequence of sections depends on the candidate's career stage. For recent graduates, career changers, or those in academia, leading with **Education** is often advantageous as it highlights the most relevant and recent qualification.¹ For the majority of established professionals, however, the most effective structure prioritizes demonstrated experience. The standard, recommended order for a professional STEM résumé is as follows:

1. **Header:** Name and Contact Information
2. **Professional Summary:** A concise, tailored pitch
3. **Core Competencies / Skills:** A scannable overview of technical capabilities
4. **Work Experience:** The core career narrative
5. **Projects:** Evidence of practical application and initiative
6. **Education:** Foundational qualifications
7. **Publications / Certifications / Awards:** Supporting credentials

This structure immediately presents the most critical information—who the candidate is, what they can do, and where they have done it—aligning with the rapid scanning behavior of human reviewers.

Crafting High-Impact Sections

- **Header:** The candidate's name should be the largest font on the page to serve as a clear anchor.² The contact information should be limited to the essentials: City, State (for location context), a professional email address, a phone number, and hyperlinked URLs to a LinkedIn profile and a GitHub repository (for technical roles).² This information must be in the body of the document to ensure ATS parsing.¹²
- **Professional Summary:** This is not an objective statement about what the candidate wants, but a powerful 2-3 sentence summary of who they are and the value they bring.¹ It

should be customized for each application, highlighting the most relevant years of experience, key areas of expertise, and a major accomplishment that aligns with the target role.

- **Skills Section:** This section serves a crucial dual purpose. For an ATS, it is a dense block of relevant keywords that will be used for scoring. For a human reviewer, it acts as a "table of contents" for the candidate's technical abilities, framing how they will interpret the subsequent experience section. By seeing a list of required skills (e.g., Python, PyTorch, AWS, Kalman Filtering) upfront, the reviewer is primed to look for evidence of their application in the work history, turning the review from a search for information into a confirmation of qualifications. Skills should be grouped logically by category (e.g., Programming Languages, ML Frameworks, Cloud Platforms, Lab Techniques) for maximum scannability.²
- **Work Experience & Projects:** This is the heart of the résumé and must focus on accomplishments, not just responsibilities. The most effective method for structuring bullet points is the **STAR method (Situation, Task, Action, Result)**. Each bullet point should be a concise narrative that begins with a strong action verb and, most importantly, ends with a quantified result.² Vague statements like "Responsible for data analysis" should be transformed into specific, impactful claims like "Developed a Python-based pipeline to analyze 10 TB of genomic data, identifying 15 novel gene variants and reducing processing time by 40%." Quantification is non-negotiable in STEM; it provides concrete evidence of a candidate's impact and technical proficiency. Metrics can include percentages, dollar amounts, time saved, scale of data, or improvements in accuracy or efficiency.
- **Education and Publications:** This section validates the candidate's foundational knowledge. It should list the degree, institution, location, and graduation date.¹ A GPA should only be included if it is above 3.5. For academic or research-heavy roles, this section gains prominence and may include thesis titles, advisors, and a full list of publications with consistent, standard citations.¹ For publications, the candidate's name should be bolded to make it easy to identify their contribution.¹⁵

By structuring the content in this manner, the résumé becomes a compelling narrative of professional achievement, optimized for both the keyword-focused ATS and the impact-seeking human reviewer.

3. The Data-First Paradigm: Implementing the JSON Resume Standard

To achieve maximum flexibility, maintainability, and consistency, a modern résumé should be treated not as a static document but as a dynamic application. The "résumé-as-code" paradigm, which separates content from presentation, is the most effective way to manage a

professional narrative in a fast-paced technical career. This is achieved by using a structured data format, like JSON, as the single source of truth.

Why Separate Content from Presentation?

Manually editing a document in a word processor for each job application is inefficient and prone to error.⁶ A data-driven approach offers several profound advantages:

- **Maintainability:** A single, canonical `resume.json` file contains all career information. Updates, such as adding a new skill or job, are made in one place.
- **Version Control:** The `resume.json` file can be stored in a Git repository. This allows for a complete history of changes, the ability to branch for different applications (e.g., one branch for a data science role, another for a machine learning engineer role), and easy rollbacks.
- **Consistency:** Programmatic generation ensures that every résumé produced is perfectly formatted, eliminating the risk of manual formatting errors.
- **Reusability:** The same `resume.json` data can be rendered into multiple themes or formats (e.g., a formal single-column PDF, a modern two-column version, an HTML version for a personal website) without ever touching the underlying content.

The JSON Resume Standard and Generation Pipeline

The open-source **JSON Resume** project provides a well-defined schema for structuring résumé data in a JSON file, along with a rich ecosystem of themes and tools.¹⁶ While the standard schema is a robust starting point, it can and should be extended to better suit the specific needs of STEM roles, such as by adding more detailed fields for projects or a dedicated section for certifications.¹⁸

The generation pipeline for creating a PDF from this data involves three core components, as exemplified by the user-provided files ¹⁹:

1. **resume.json:** The data file. This is the single source of truth containing all text content for the résumé, structured according to a defined schema.
2. **template.html:** The presentation layer. This is an HTML file that uses a templating engine like Handlebars to define the document's structure and layout.²⁰ It contains placeholders (e.g., `{{basics.name}}`) that will be replaced with data from the JSON file. All styling is typically embedded within `<style>` tags in this file.

3. **generate-pdf.js:** The build script. This is a Node.js script that orchestrates the generation process. It reads the resume.json file, compiles the Handlebars template, and then uses a headless browser automation tool like Puppeteer to render the final HTML.²¹ Puppeteer then "prints" this webpage to a pixel-perfect PDF file, ensuring that the complex CSS used for visual layout is rendered with high fidelity.

This "résumé-as-code" workflow transforms résumé management from a clerical task into a systematic engineering practice. It allows a STEM professional to apply the same principles of version control, automation, and modularity to their career narrative as they do to their software projects, resulting in a more efficient, scalable, and professional job application process.

Part II: The Curated Template Collection: Five Production-Ready Implementations

This section provides five distinct, production-ready résumé template sets. Each set is designed with a specific strategic purpose and is optimized for both human and algorithmic review. The templates are built upon the data-first paradigm, utilizing a shared PDF generation script alongside unique JSON data structures and HTML/CSS templates.

Template Selection Matrix

To facilitate the selection of the most appropriate template, the following matrix summarizes the key attributes and intended audience for each design.

Template Name	Layout Style	Primary Audience	ATS Compatibility	Key Sections Emphasized	Aesthetic
The Sentinel	Single-Column	All STEM Roles, Conservative Industries	Very High	Work Experience, Skills	Traditional, Professional

The Innovator	Hybrid (Main + Sidebar)	Tech, Startups, Design-adj acent Roles	High	Skills, Contact Info (Sidebar)	Modern, Clean
The Architect	Single-Colu mn	Software/D ata Roles, Portfolio-dri ven	Very High	Projects, Technical Skills	Technical, Structured
The Scholar	Single-Colu mn	Academia, R&D, PhDs	Very High	Publication s, Education, Grants	Academic, Formal
The Purist	Single-Colu mn	Senior Execs, Experience d Professiona ls	Very High	Summary, Work Experience	Minimalist, Typographi c

Universal PDF Generation Script

Instead of providing five nearly identical JavaScript files, a single, robust generator script has been developed. This script, `generate-pdf.js`, is designed to be flexible, accepting the names of the JSON and HTML files as command-line arguments. This adheres to the software development principle of "Don't Repeat Yourself" (DRY) and provides a more maintainable solution.

Usage:

```
node generate-pdf.js <template_name>
```

For example, to generate the "Sentinel" résumé, one would run:

```
node generate-pdf.js sentinel
```

This command will automatically use `sentinel.resume.json` and `sentinel.template.html` to produce `sentinel.resume.pdf`.

generate-pdf.js

JavaScript

```
// File: generate-pdf.js
// Description: Universal PDF generator for all resume templates.
// Usage:
// 1. Install dependencies: npm i puppeteer handlebars
// 2. Run from the command line: node generate-pdf.js <template_name>
// Example: node generate-pdf.js sentinel
//
// This script expects to find `./<template_name>.resume.json` and `./<template_name>.template.html`.
// It will output `./<template_name>.resume.pdf`.

const puppeteer = require('puppeteer');
const handlebars = require('handlebars');
const fs = require('fs').promises;
const path = require('path');

// --- Handlebars Helpers ---
// These helpers provide useful formatting functions within the HTML templates.

// Joins an array of strings with a specified separator. Defaults to ", ".
handlebars.registerHelper('formatList', function (items, options) {
  const sep = (typeof options.hash.sep === 'string')? options.hash.sep : ',';
  if (!Array.isArray(items) |
    | items.length === 0) return '';
  return new handlebars.SafeString(items.join(sep));
});

// Ensures a URL has a protocol (http, https, mailto, tel). Defaults to https.
handlebars.registerHelper('ensureProtocol', function (url) {
  if (!url) return '';
  const s = String(url).trim();
  if (/^(https?|mailto|tel):/i.test(s)) return s;
  if (s.includes('@')) return `mailto:${s}`;
});
```

```
    return `https://${s}`;  
  });
```

// Formats a date string (YYYY-MM-DD or YYYY-MM) into a more readable format (e.g., "Jan 2023").

```
handlebars.registerHelper('formatDate', function (dateString) {  
  if (!dateString) return 'Present';  
  const [year, month] = dateString.split('-');  
  const date = new Date(year, month? month - 1 : 0);  
  const options = { year: 'numeric' };  
  if (month) {  
    options.month = 'short';  
  }  
  return date.toLocaleDateString('en-US', options);  
});
```

// --- Core Functions ---

```
async function loadJsonData(jsonPath) {  
  const rawData = await fs.readFile(jsonPath, 'utf-8');  
  return JSON.parse(rawData);  
}
```

```
async function generatePdf({ data, templatePath, outputPath }) {  
  const templateSrc = await fs.readFile(templatePath, 'utf-8');  
  const template = handlebars.compile(templateSrc);  
  const html = template(data);
```

// Save intermediate HTML for debugging

```
const intermediateHtmlPath = outputPath.replace('.pdf', '.html');  
await fs.writeFile(intermediateHtmlPath, html, 'utf-8');  
console.log(`📄 Intermediate HTML saved to: ${intermediateHtmlPath}`);
```

```
const browser = await puppeteer.launch({  
  headless: 'new',  
  args: ['--no-sandbox', '--disable-setuid-sandbox', '--font-render-hinting=none']  
});
```

```
try {  
  const page = await browser.newPage();  
  await page.setContent(html, { waitUntil: 'networkidle0' });  
  await page.evaluateHandle('document.fonts.ready');
```

```
  await page.pdf({
```

```

    path: outputPath,
    format: 'Letter',
    printBackground: true,
    margin: { top: '0.4in', right: '0.4in', bottom: '0.4in', left: '0.4in' },
  });
} finally {
  await browser.close();
}
}

```

// --- Main Execution ---

```

(async function main() {
  const args = process.argv.slice(2);
  if (args.length !== 1) {
    console.error('❌ Usage: node generate-pdf.js <template_name>');
    process.exit(1);
  }

  const templateName = args;
  const basePath = path.resolve(process.cwd());

  const jsonPath = path.join(basePath, `${templateName}.resume.json`);
  const templatePath = path.join(basePath, `${templateName}.template.html`);
  const outputPath = path.join(basePath, `${templateName}.resume.pdf`);

  try {
    console.log('📄 Loading data from: ${jsonPath}');
    const data = await loadJsonData(jsonPath);

    console.log('📄 Using template: ${templatePath}');
    await generatePdf({ data, templatePath, outputPath });

    console.log('✅ PDF successfully generated: ${outputPath}');
  } catch (error) {
    console.error('❌ An error occurred during PDF generation:', error);
    process.exit(1);
  }
})();

```

Template 1: The Sentinel — Classic Single-Column for Maximum ATS Compatibility

Strategic Purpose: This is the bedrock of ATS-friendly design. It is engineered for maximum parsability, making it the ideal choice for applications to large, conservative organizations in sectors like aerospace, defense, or manufacturing, where legacy ATS may still be in use. Its clean, traditional aesthetic conveys professionalism and reliability.

sentinel.resume.json

JSON

```
{
  "basics": {
    "name": "Dr. Evelyn Reed",
    "label": "Senior Mechanical Engineer",
    "email": "evelyn.reed.eng@email.com",
    "phone": "(555) 123-4567",
    "location": {
      "city": "Seattle",
      "region": "WA"
    },
  },
  "profiles": [
    {
      "network": "LinkedIn",
      "username": "evelynreed-me",
      "url": "linkedin.com/in/evelynreed-me"
    }
  ],
  "summary": "PhD Mechanical Engineer with 12+ years of experience leading the design, analysis, and testing of complex mechanical systems for the aerospace industry. Expert in finite element analysis (FEA), computational fluid dynamics (CFD), and thermal management. Proven track record of delivering robust, high-performance components that exceed mission requirements, resulting in a 15% improvement in system reliability."
```

```
"work":
  },
  {
    "company": "Propulsion Systems Inc.",
    "position": "Mechanical Engineer",
    "location": "Huntsville, AL",
    "startDate": "2012-07-01",
    "endDate": "2018-05-31",
    "highlights":
  }
],
"education":
  },
  {
    "institution": "Purdue University",
    "location": "West Lafayette, IN",
    "area": "Mechanical Engineering",
    "studyType": "B.S.",
    "endDate": "2008-05-01",
    "gpa": "4.0/4.0"
  }
],
"skills":
  },
  {
    "name": "CAD & Software",
    "keywords":
  },
  {
    "name": "Manufacturing & Testing",
    "keywords":
  }
],
"certifications":
}
```

sentinel.template.html

HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>{{basics.name}} - Resume</title>
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <style>
    :root {
      --font-serif: 'Georgia', 'Times New Roman', serif;
      --font-sans: 'Helvetica', 'Arial', sans-serif;
      --color-text: #333333;
      --color-primary: #2c3e50;
      --color-secondary: #7f8c8d;
      --color-border: #bdc3c7;
    }
    body {
      font-family: var(--font-serif);
      line-height: 1.5;
      color: var(--color-text);
      background-color: #ffffff;
      font-size: 11pt;
      margin: 0;
      padding: 0;
    }
    .container { max-width: 7.5in; margin: 0 auto; }
    a { color: var(--color-primary); text-decoration: none; }
    a:hover { text-decoration: underline; }

    /* --- Header --- */
    .header {
      text-align: center;
      border-bottom: 2px solid var(--color-primary);
      padding-bottom: 8px;
      margin-bottom: 16px;
    }
    .header.name {
      font-size: 24pt;
      font-weight: bold;
      color: var(--color-primary);
      letter-spacing: 1px;
      margin: 0;
    }
    .header.label {
```

```

    font-size: 14pt;
    color: var(--color-secondary);
    margin: 4px 0;
}
.contact-info {
    font-family: var(--font-sans);
    font-size: 10pt;
    color: var(--color-secondary);
    display: flex;
    justify-content: center;
    flex-wrap: wrap;
    gap: 8px 16px;
    margin-top: 8px;
}
.contact-info span:not(:last-child)::after {
    content: '|';
    margin-left: 16px;
    color: var(--color-border);
}

/* --- Section --- */
.section { margin-bottom: 16px; }
.section-title {
    font-family: var(--font-sans);
    font-size: 13pt;
    font-weight: bold;
    color: var(--color-primary);
    border-bottom: 1px solid var(--color-border);
    padding-bottom: 4px;
    margin-bottom: 10px;
    text-transform: uppercase;
    letter-spacing: 1px;
}
.summary p { margin: 0; text-align: justify; }

/* --- Work Experience --- */
.job { margin-bottom: 12px; }
.job-header {
    display: flex;
    justify-content: space-between;
    align-items: baseline;
    margin-bottom: 2px;
}
.job-title { font-size: 12pt; font-weight: bold; }
.job-company { font-style: italic; color: var(--color-secondary); }
.job-dates { font-family: var(--font-sans); font-size: 10pt; color: var(--color-secondary); }
white-space: nowrap;
.highlights {

```

```

        margin: 4px 0 0 18px;
        padding-left: 0;
        list-style-type: disc;
    }
    .highlights li { margin-bottom: 4px; }

/* --- Education --- */
.edu-item { margin-bottom: 8px; }
.edu-header { display: flex; justify-content: space-between; align-items: baseline; }
.edu-degree { font-weight: bold; }
.edu-institution { font-style: italic; color: var(--color-secondary); }
.edu-details { font-family: var(--font-sans); font-size: 10pt; color: var(--color-secondary);
margin-top: 2px; }

/* --- Skills --- */
.skills-grid {
    display: grid;
    grid-template-columns: repeat(auto-fit, minmax(200px, 1fr));
    gap: 10px;
}
.skill-category.name { font-weight: bold; margin-bottom: 4px; }
.skill-category.keywords { font-size: 10pt; color: #555; }

/* --- Certifications --- */
.cert-item { margin-bottom: 4px; }
.cert-title { font-weight: bold; }
.cert-issuer { font-style: italic; color: var(--color-secondary); }

/* --- Print Optimizations --- */
@media print {
    body { font-size: 10.5pt; }
    .section { break-inside: avoid; }
    .job, .edu-item { break-inside: avoid; }
    a[href]:after { content: none !important; }
}
</style>
</head>
<body>
    <div class="container">
        <header class="header">
            <h1 class="name">{{basics.name}}</h1>
            {{#if basics.label}}<p class="label">{{basics.label}}</p>{{/if}}
            <div class="contact-info">
                {{#if basics.location.city}}<span>{{basics.location.city}},
                {{basics.location.region}}</span>{{/if}}
                {{#if basics.phone}}<span><a
href="tel:{{basics.phone}}">{{basics.phone}}</a></span>{{/if}}

```



```
    {{#if basics.email}}<span><a
href="mailto:{{basics.email}}">{{basics.email}}</a></span>{{/if}}
    {{#each basics.profiles}}
        <span><a href="{{ensureProtocol url}}">{{#if
username}}{{username}}{{else}}{{network}}{{/if}}</a></span>
    {{/each}}
</div>
</header>
```

```
{{#if summary}}
<section class="section summary">
    <h2 class="section-title">Professional Summary</h2>
    <p>{{summary}}</p>
</section>
{{/if}}
```

```
{{#if work.length}}
<section class="section">
    <h2 class="section-title">Work Experience</h2>
    {{#each work}}
        <div class="job">
            <div class="job-header">
                <div>
                    <span class="job-title">{{position}}</span>
                    <span class="job-company"> at {{company}}</span>
                </div>
                <div class="job-dates">{{formatDate startDate}} – {{formatDate endDate}}</div>
            </div>
            <ul class="highlights">
                {{#each highlights}}
                    <li>{{this}}</li>
                {{/each}}
            </ul>
        </div>
    {{/each}}
</section>
{{/if}}
```

```
{{#if skills.length}}
<section class="section">
    <h2 class="section-title">Core Competencies</h2>
    <div class="skills-grid">
        {{#each skills}}
```

```
<div class="skill-category">
  <div class="name">{{name}}</div>
  <div class="keywords">{{formatList keywords}}</div>
</div>
{{/each}}
</div>
</section>
{{/if}}

{{#if education.length}}
<section class="section">
  <h2 class="section-title">Education</h2>
  {{#each education}}
    <div class="edu-item">
      <div class="edu-header">
        <div>
          <span class="edu-degree">{{studyType}} in {{area}}</span>
          <span class="edu-institution">{{institution}}</span>
        </div>
        <div class="job-dates">{{formatDate endDate}}</div>
      </div>
      {{#if gpa}}<div class="edu-details">GPA: {{gpa}}</div>{{/if}}
    </div>
  {{/each}}
</section>
{{/if}}

{{#if certifications.length}}
<section class="section">
  <h2 class="section-title">Certifications</h2>
  {{#each certifications}}
    <div class="cert-item">
      <span class="cert-title">{{title}}</span>
      <span class="cert-issuer">{{issuer}}</span>
    </div>
  {{/each}}
</section>
{{/if}}
</div>
</body>
</html>
```

Template 2: The Innovator — Modern Hybrid-Column with Graceful Degradation

Strategic Purpose: This template balances modern aesthetics with ATS compatibility. The two-column visual layout is engaging for human reviewers and uses page space efficiently. It is ideal for candidates in the tech sector, startups, and design-adjacent STEM fields where a contemporary presentation is valued. The underlying HTML structure remains linear, ensuring high compatibility with parsers.

innovator.resume.json

JSON

```
{
  "basics": {
    "name": "Javier 'Javi' Moreno",
    "label": "Senior Frontend Engineer & UX Advocate",
    "email": "javi.moreno.dev@email.com",
    "phone": "(555) 987-6543",
    "location": {
      "city": "Austin",
      "region": "TX"
    },
  },
  "profiles": [
    {
      "network": "Portfolio",
      "username": "javimoreno.dev",
      "url": "javimoreno.dev"
    },
    {
      "network": "GitHub",
      "username": "javi-moreno",
    }
  ]
}
```

```

    "url": "github.com/javi-moreno"
  },
  {
    "network": "LinkedIn",
    "username": "javier-moreno-dev",
    "url": "linkedin.com/in/javier-moreno-dev"
  }
]
},
"summary": "Senior Frontend Engineer with 8+ years of experience crafting responsive, accessible, and high-performance web applications. Specializes in React, TypeScript, and modern state management, with a strong focus on user experience and design systems. Passionate about bridging the gap between design and engineering to build intuitive and delightful digital products. Led the refactoring of a legacy codebase to a modern stack, resulting in a 60% improvement in Lighthouse performance scores.",
"work":
},
{
  "company": "Digital Canvas Agency",
  "position": "Frontend Developer",
  "location": "Remote",
  "startDate": "2016-01-01",
  "endDate": "2019-02-28",
  "highlights":
}
],
"education":,
"skills":
},
{
  "name": "Frameworks & Libraries",
  "keywords":
},
{
  "name": "Tools & Platforms",
  "keywords":
},
{
  "name": "Concepts",
  "keywords":
}
],
"projects":,

```

```
"languages":  
}
```

innovator.template.html

HTML

```
<!DOCTYPE html>  
<html lang="en">  
<head>  
  <meta charset="UTF-8">  
  <title>{{basics.name}} - Resume</title>  
  <meta name="viewport" content="width=device-width, initial-scale=1.0">  
  <style>  
    :root {  
      --font-sans: -apple-system, BlinkMacSystemFont, "Segoe UI", Roboto, Helvetica, Arial,  
sans-serif;  
      --color-text: #333;  
      --color-heading: #1a202c;  
      --color-primary: #2563eb;  
      --color-background: #fff;  
      --color-sidebar: #f4f7f9;  
      --color-border: #e2e8f0;  
    }  
    body {  
      font-family: var(--font-sans);  
      line-height: 1.6;  
      color: var(--color-text);  
      background-color: var(--color-background);  
      font-size: 10.5pt;  
      margin: 0;  
      padding: 0;  
    }  
    .page {  
      display: grid;  
      grid-template-columns: 1fr; /* Default to single column */  
      max-width: 8.5in;  
      min-height: 11in;  
      margin: 0 auto;
```

```

}
@media screen and (min-width: 8.5in) {
  .page {
    grid-template-columns: 2.5in 1fr; /* Two columns on larger screens */
  }
}
a { color: var(--color-primary); text-decoration: none; }
a:hover { text-decoration: underline; }
h1, h2, h3 { color: var(--color-heading); margin: 0; font-weight: 700; }
ul { margin: 0; padding-left: 1.2em; }
li { margin-bottom: 0.25em; }

/* --- Main Content Area --- */
.main-content {
  padding: 0.5in;
  grid-column: 1 / -1; /* Span full width by default */
}
@media screen and (min-width: 8.5in) {
  .main-content {
    grid-column: 2 / 3; /* Second column on larger screens */
  }
}
.main-header { text-align: left; margin-bottom: 1.5em; }
.main-header h1 { font-size: 26pt; letter-spacing: -0.5px; }
.main-header h2 { font-size: 14pt; font-weight: 400; color: #555; margin-top: 0.25em; }
.section { margin-bottom: 1.5em; }
.section-title {
  font-size: 12pt;
  font-weight: 800;
  text-transform: uppercase;
  letter-spacing: 1px;
  color: var(--color-primary);
  border-bottom: 2px solid var(--color-border);
  padding-bottom: 0.3em;
  margin-bottom: 0.8em;
}
.item { margin-bottom: 1.2em; }
.item-header { display: flex; justify-content: space-between; align-items: baseline; margin-bottom: 0.1em; }
.item-title { font-size: 11pt; font-weight: 700; }
.item-subtitle { color: #555; }
.item-date { font-size: 9.5pt; color: #777; white-space: nowrap; }
.item-highlights li { font-size: 10pt; }

/* --- Sidebar --- */
.sidebar {
  background-color: var(--color-sidebar);
  padding: 0.5in;

```

```

    grid-column: 1 / -1; /* Span full width by default */
  }
  @media screen and (min-width: 8.5in) {
    .sidebar {
      grid-column: 1 / 2; /* First column on larger screens */
      grid-row: 1 / 2;
    }
  }
  .sidebar.section { margin-bottom: 2em; }
  .sidebar.section-title { color: var(--color-heading); border-color: #d1d9e3; }
  .contact-item { margin-bottom: 0.5em; font-size: 10pt; }
  .contact-item strong { display: block; font-weight: 700; color: #555; margin-bottom: 2px; }
  .skill-item { margin-bottom: 1em; }
  .skill-item h3 { font-size: 10pt; font-weight: 700; margin-bottom: 0.3em; }
  .skill-item ul { padding-left: 0; list-style: none; font-size: 9.5pt; }
  .skill-item li { margin-bottom: 0.2em; }

  /* --- Print and PDF Generation Overrides --- */
  @media print {
    body { font-size: 10pt; -webkit-print-color-adjust: exact; print-color-adjust: exact; }
    .page {
      display: grid;
      grid-template-columns: 2.2in 1fr!important; /* Force two columns for PDF */
      max-width: 100%;
      min-height: 0;
    }
    .sidebar { grid-column: 1 / 2; grid-row: 1 / 2; }
    .main-content { grid-column: 2 / 3; grid-row: 1 / 2; }
    a { color: var(--color-primary)!important; text-decoration: none!important; }
    a[href]:after { content: none!important; }
    .section, .item { break-inside: avoid; }
  }
</style>
</head>
<body>
  <div class="page">
    <main class="main-content">
      <header class="main-header">
        <h1>{{basics.name}}</h1>
        <h2>{{basics.label}}</h2>
      </header>

      <section class="section">
        <h2 class="section-title">Summary</h2>
        <p>{{summary}}</p>
      </section>

```

```

    {{#if work.length}}
    <section class="section">
      <h2 class="section-title">Experience</h2>
      {{#each work}}
      <div class="item">
        <div class="item-header">
          <div>
            <h3 class="item-title">{{position}}</h3>
            <p class="item-subtitle">{{company}}</p>
          </div>
          <p class="item-date">{{formatDate startDate}} – {{formatDate endDate}}</p>
        </div>
        <ul class="item-highlights">
          {{#each highlights}}<li>{{this}}</li>{{/each}}
        </ul>
      </div>
    {{/each}}
  </section>
{{/if}}
</main>

<aside class="sidebar">
  <section class="section">
    <h2 class="section-title">Contact</h2>
    <div class="contact-item">
      <strong>Email</strong>
      <a href="mailto:{{basics.email}}">{{basics.email}}</a>
    </div>
    <div class="contact-item">
      <strong>Phone</strong>
      <a href="tel:{{basics.phone}}">{{basics.phone}}</a>
    </div>
    <div class="contact-item">
      <strong>Location</strong>
      <span>{{basics.location.city}}, {{basics.location.region}}</span>
    </div>
    {{#each basics.profiles}}
    <div class="contact-item">
      <strong>{{network}}</strong>
      <a href="{{ensureProtocol url}}">{{username}}</a>
    </div>
  {{/each}}

```



```
</section>
```

```
{{#if skills.length}}
```

```
<section class="section">
```

```
<h2 class="section-title">Skills</h2>
```

```
{{#each skills}}
```

```
<div class="skill-item">
```

```
<h3>{{name}}</h3>
```

```
<ul>
```

```
  {{#each keywords}}<li>{{this}}</li>{{/each}}
```

```
</ul>
```

```
</div>
```

```
{{/each}}
```

```
</section>
```

```
{{/if}}
```

```
{{#if education.length}}
```

```
<section class="section">
```

```
<h2 class="section-title">Education</h2>
```

```
{{#each education}}
```

```
<div class="item">
```

```
<h3 class="item-title">{{studyType}} in {{area}}</h3>
```

```
<p class="item-subtitle">{{institution}}</p>
```

```
<p class="item-date">{{formatDate endDate}}</p>
```

```
</div>
```

```
{{/each}}
```

```
</section>
```

```
{{/if}}
```

```
{{#if projects.length}}
```

```
<section class="section">
```

```
<h2 class="section-title">Projects</h2>
```

```
{{#each projects}}
```

```
<div class="item">
```

```
<h3 class="item-title"><a href="{{ensureProtocol url}}">{{name}}</a></h3>
```

```
<p class="item-subtitle">{{summary}}</p>
```

```
</div>
```

```
{{/each}}
```

```
</section>
```

```
{{/if}}
```

```
{{#if languages.length}}
```

```
<section class="section">
```

```

    <h2 class="section-title">Languages</h2>
    {{#each languages}}
    <p>{{language}} ({{fluency}})</p>
    {{/each}}
  </section>
  {{/if}}
</aside>
</div>
</body>
</html>

```

Template 3: The Architect — Project-Centric Layout for Developers and Data Scientists

Strategic Purpose: This template is tailored for roles where a portfolio of tangible work is the most persuasive evidence of skill. By prioritizing the "Projects" section, it immediately showcases a candidate's ability to build, create, and deliver. It is ideal for software developers, data scientists, and ML engineers whose GitHub profiles and personal projects are central to their professional identity.

architect.resume.json

JSON

```

{
  "basics": {
    "name": "Aisha Khan",
    "label": "Full-Stack Software Engineer",
    "email": "aisha.khan.dev@email.com",
    "phone": "(555) 246-8135",
    "location": {
      "city": "San Francisco",
      "region": "CA"
    }
  }
}

```

```

},
"profiles": [
  {
    "network": "GitHub",
    "username": "aishakhan",
    "url": "github.com/aishakhan"
  },
  {
    "network": "LinkedIn",
    "username": "aishakhan-dev",
    "url": "linkedin.com/in/aishakhan-dev"
  }
]
},
"summary": "Results-driven Full-Stack Engineer with 6 years of experience building and scaling distributed systems and data-intensive applications. Proficient in Go, Python, and TypeScript, with deep expertise in cloud-native architectures on AWS. Passionate about writing clean, testable code and leveraging automation to improve development velocity and system reliability.",
"projects":,
  "url": "github.com/aishakhan/querycache"
},
{
  "name": "InsightStream",
  "role": "Contributor (Open Source)",
  "summary": "A real-time data processing pipeline for analyzing social media trends. Contributed a new Kafka consumer module in Python that parallelizes sentiment analysis using NLP models, and deployed it via a serverless architecture.",
  "impact": "Increased data processing throughput by 5x and enabled real-time dashboard updates, which was a key feature for the v2.0 release.",
  "technologies":,
  "url": "github.com/opensource/insightstream"
}
],
"work":
},
{
  "company": "DataCorp",
  "position": "Junior Software Engineer",
  "location": "Palo Alto, CA",
  "startDate": "2018-06-01",
  "endDate": "2020-06-30",
  "highlights":
}

```

```

],
"skills":
},
{
  "name": "DevOps & Cloud",
  "keywords":
},
{
  "name": "Frontend",
  "keywords":
}
},
"education":
}

```

architect.template.html

HTML

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>{{basics.name}} - Resume</title>
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <style>
    :root {
      --font-main: 'Helvetica Neue', Helvetica, Arial, sans-serif;
      --font-mono: 'Menlo', 'Monaco', 'Courier New', monospace;
      --color-text: #333;
      --color-heading: #000;
      --color-primary: #007acc;
      --color-muted: #666;
      --color-border: #ddd;
      --color-tag-bg: #eee;
      --color-tag-text: #333;
    }
    body {

```

```
font-family: var(--font-main);
font-size: 10.5pt;
line-height: 1.5;
color: var(--color-text);
}
.container { max-width: 7.5in; margin: 0 auto; }
a { color: var(--color-primary); text-decoration: none; }
a:hover { text-decoration: underline; }
```

```
/* --- Header --- */
.header {
  display: flex;
  justify-content: space-between;
  align-items: center;
  padding-bottom: 10px;
  margin-bottom: 20px;
  border-bottom: 1px solid var(--color-border);
}
.header-left.name {
  font-size: 22pt;
  font-weight: 600;
  color: var(--color-heading);
}
.header-left.label {
  font-size: 12pt;
  color: var(--color-muted);
}
.header-right {
  font-family: var(--font-mono);
  font-size: 9.5pt;
  text-align: right;
}
.header-right div { margin-bottom: 2px; }
```

```
/* --- Section --- */
.section { margin-bottom: 20px; }
.section-title {
  font-family: var(--font-mono);
  font-size: 11pt;
  font-weight: 600;
  text-transform: uppercase;
  letter-spacing: 1px;
  color: var(--color-heading);
  border-bottom: 1px solid var(--color-border);
  padding-bottom: 4px;
  margin-bottom: 12px;
}
```

```
/* --- Projects --- */
.project { margin-bottom: 16px; }
.project-header {
  display: flex;
  justify-content: space-between;
  align-items: baseline;
  gap: 1em;
}
.project-name { font-size: 12pt; font-weight: 600; }
.project-role { font-style: italic; color: var(--color-muted); }
.project-summary { margin: 2px 0 6px 0; }
.project-impact {
  font-style: italic;
  color: #111;
  margin-bottom: 8px;
  padding-left: 10px;
  border-left: 2px solid var(--color-primary);
}
.project-technologies {
  display: flex;
  flex-wrap: wrap;
  gap: 6px;
}
.tech-tag {
  font-family: var(--font-mono);
  font-size: 8.5pt;
  background-color: var(--color-tag-bg);
  color: var(--color-tag-text);
  padding: 2px 8px;
  border-radius: 4px;
}
```

```
/* --- Work Experience --- */
.job { margin-bottom: 16px; }
.job-header {
  display: flex;
  justify-content: space-between;
  align-items: baseline;
}
.job-title { font-size: 11pt; font-weight: 600; }
.job-company { color: var(--color-muted); }
.job-dates { color: var(--color-muted); font-size: 10pt; }
.job-highlights { margin: 4px 0 0 18px; padding-left: 0; }
.job-highlights li { margin-bottom: 4px; }
```

```
/* --- Skills & Education --- */
.grid-container {
  display: grid;
```

```

    grid-template-columns: 1fr 1fr;
    gap: 24px;
}
.skills-list { display: flex; flex-direction: column; gap: 8px; }
.skill-category.name { font-weight: 600; }
.skill-category.keywords { font-size: 10pt; color: var(--color-muted); }
.edu-item.degree { font-weight: 600; }
.edu-item.institution { color: var(--color-muted); }

/* --- Print Optimizations --- */
@media print {
    body { font-size: 10pt; }
    .section,.project,.job { break-inside: avoid; }
    a[href]:after { content: none!important; }
}
</style>
</head>
<body>
    <div class="container">
        <header class="header">
            <div class="header-left">
                <div class="name">{{basics.name}}</div>
                <div class="label">{{basics.label}}</div>
            </div>
            <div class="header-right">
                {{#if basics.location.city}}<div>{{basics.location.city}},
{{basics.location.region}}</div>{{/if}}
                {{#if basics.phone}}<div>{{basics.phone}}</div>{{/if}}
                {{#if basics.email}}<div><a href="mailto:{{basics.email}}">{{basics.email}}</a></div>{{/if}}
                {{#each basics.profiles}}
                    <div><a href="{{ensureProtocol url}}">{{url}}</a></div>
                {{/each}}
            </div>
        </header>

        {{#if summary}}
        <section class="section">
            <p>{{summary}}</p>
        </section>
        {{/if}}

        {{#if projects.length}}
        <section class="section">
            <h2 class="section-title">Selected Projects</h2>
            {{#each projects}}

```

```
<div class="project">
  <div class="project-header">
    <div class="project-name"><a href="{{ensureProtocol url}}">{{name}}</a></div>
    <div class="project-role">{{role}}</div>
  </div>
  <p class="project-summary">{{summary}}</p>
  {{#if impact}}<p class="project-impact"><strong>Impact:</strong> {{impact}}</p>{{/if}}
  {{#if technologies.length}}
    <div class="project-technologies">
      {{#each technologies}}<span class="tech-tag">{{this}}</span>{{/each}}
    </div>
  {{/if}}
</div>
{{/each}}
</section>
{{/if}}
```

```
{{#if work.length}}
<section class="section">
  <h2 class="section-title">Professional Experience</h2>
  {{#each work}}
    <div class="job">
      <div class="job-header">
        <div><span class="job-title">{{position}}</span> @ <span
class="job-company">{{company}}</span></div>
        <div class="job-dates">{{formatDate startDate}} – {{formatDate endDate}}</div>
      </div>
      <ul class="job-highlights">
        {{#each highlights}}<li>{{this}}</li>{{/each}}
      </ul>
    </div>
  {{/each}}
</section>
{{/if}}
```

```
<div class="grid-container">
  {{#if skills.length}}
    <section class="section">
      <h2 class="section-title">Technical Skills</h2>
      <div class="skills-list">
        {{#each skills}}
          <div class="skill-category">
            <span class="name">{{name}}</span>
```



```

        <span class="keywords">{{formatList keywords}}</span>
    </div>
    {{/each}}
</div>
</section>
{{/if}}

{{#if education.length}}
<section class="section">
    <h2 class="section-title">Education</h2>
    {{#each education}}
    <div class="edu-item">
        <div class="degree">{{studyType}} in {{area}}</div>
        <div class="institution">{{institution}}</div>
    </div>
    {{/each}}
</section>
{{/if}}
</div>
</div>
</body>
</html>

```

Template 4: The Scholar — Academic-Focused Format for Researchers and PhDs

Strategic Purpose: This template is a CV (Curriculum Vitae) designed for academic, R&D, and scientific roles. It deviates from the corporate résumé structure to prioritize information critical to this audience: publications, grants, research experience, and teaching. The format is conservative, information-dense, and adheres to academic conventions.

scholar.resume.json

JSON

```

{
  "basics": {
    "name": "Dr. Chen Wei",
    "label": "Postdoctoral Research Fellow in Bioinformatics",
    "email": "chen.wei.research@email.com",
    "phone": "(555) 555-0101",
    "location": {
      "city": "Cambridge",
      "region": "MA"
    },
    "profiles":
  },
  "summary": "Computational biologist with expertise in developing and applying statistical models and machine learning algorithms to analyze large-scale genomic and transcriptomic data. Research focuses on identifying genetic drivers of cancer progression and therapeutic resistance. Highly proficient in R, Python, and high-performance computing environments.",
  "education":,
  "researchExperience":,
  "publications":,
  "grantsAndFellowships":,
  "presentations":,
  "skills":
  },
  {
    "name": "Statistical Methods",
    "keywords":
  },
  {
    "name": "Genomic Technologies",
    "keywords":
  }
}

```

scholar.template.html

HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>{{basics.name}} - Curriculum Vitae</title>
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <style>
    body {
      font-family: 'Times New Roman', Times, serif;
      font-size: 11pt;
      line-height: 1.4;
      color: #222;
    }
    .container { max-width: 7.5in; margin: 0 auto; }
    a { color: #000; text-decoration: none; }
    a:hover { text-decoration: underline; }
    strong { font-weight: bold; }

    /* --- Header --- */
    .header { text-align: center; margin-bottom: 24px; }
    .header.name { font-size: 20pt; font-weight: bold; margin: 0; }
    .header.label { font-size: 12pt; color: #555; margin: 4px 0 10px 0; }
    .contact-info { font-size: 10pt; }
    .contact-info a { margin: 0 10px; }

    /* --- Section --- */
    .section { margin-bottom: 18px; }
    .section-title {
      font-size: 12pt;
      font-weight: bold;
      border-bottom: 1px solid #333;
      padding-bottom: 3px;
      margin-bottom: 10px;
    }

    /* --- Education & Research --- */
    .item { margin-bottom: 12px; }
    .item-header { display: flex; justify-content: space-between; align-items: baseline; }
    .item-title { font-weight: bold; }
    .item-subtitle { font-style: italic; }
    .item-date { font-size: 10pt; }
    .item-summary { margin: 4px 0 0 15px; text-align: justify; }
```

```

/* --- Publications --- */
.publications-list { margin: 0; padding-left: 0; list-style-type: none; }
.publications-list li { margin-bottom: 8px; }

/* --- Grants & Presentations --- */
.grant-item,.pres-item { display: flex; justify-content: space-between; }
.grant-item.name,.pres-item.title { flex-grow: 1; }
.grant-item.awarder,.pres-item.event { margin-left: 1em; font-style: italic; }

/* --- Skills --- */
.skills-container {
  columns: 2;
  column-gap: 2em;
}
.skill-category { break-inside: avoid-column; margin-bottom: 10px; }
.skill-category.name { font-weight: bold; }
.skill-category ul { margin: 2px 0 0 18px; padding-left: 0; list-style-type: disc; }
.skill-category li { font-size: 10pt; }

/* --- Print Optimizations --- */
@media print {
  body { font-size: 10.5pt; }
  .section,.item { break-inside: avoid; }
  a[href]:after { content: none!important; }
}
</style>
</head>
<body>
  <div class="container">
    <header class="header">
      <h1 class="name">{{basics.name}}</h1>
      <p class="label">{{basics.label}}</p>
      <div class="contact-info">
        <span>{{basics.location.city}}, {{basics.location.region}}</span> |
        <a href="mailto:{{basics.email}}">{{basics.email}}</a> |
        <a href="tel:{{basics.phone}}">{{basics.phone}}</a>
        {{#each basics.profiles}}
| <a href="{{ensureProtocol url}}">{{network}}</a>
        {{/each}}
      </div>
    </header>

    {{#if summary}}
    <section class="section">
      <h2 class="section-title">Research Profile</h2>

```

```
<p>{{summary}}</p>
```

```
</section>
```

```
{{/if}}
```

```
{{#if education.length}}
```

```
<section class="section">
```

```
<h2 class="section-title">Education</h2>
```

```
{{#each education}}
```

```
<div class="item">
```

```
<div class="item-header">
```

```
<div class="item-title">{{studyType}} in {{area}}, {{institution}}</div>
```

```
<div class="item-date">{{formatDate endDate}}</div>
```

```
</div>
```

```
{{#if summary}}<div class="item-summary">{{summary}}</div>{{/if}}
```

```
</div>
```

```
{{/each}}
```

```
</section>
```

```
{{/if}}
```

```
{{#if researchExperience.length}}
```

```
<section class="section">
```

```
<h2 class="section-title">Research Experience</h2>
```

```
{{#each researchExperience}}
```

```
<div class="item">
```

```
<div class="item-header">
```

```
<div class="item-title">{{position}}, {{institution}}</div>
```

```
<div class="item-date">{{formatDate startDate}} – {{formatDate endDate}}</div>
```

```
</div>
```

```
<div class="item-summary">{{summary}}</div>
```

```
</div>
```

```
{{/each}}
```

```
</section>
```

```
{{/if}}
```

```
{{#if publications.length}}
```

```
<section class="section">
```

```
<h2 class="section-title">Publications</h2>
```

```
<ol class="publications-list">
```

```
{{#each publications}}
```

```
<li>{{name}} <a href="{{ensureProtocol url}}">[Link]</a></li>
```

```
{{/each}}
```

```
</ol>
```

```
</section>
```

```
{{/if}}
```

```
{{#if grantsAndFellowships.length}}
```

```
<section class="section">
```

```
<h2 class="section-title">Grants & Fellowships</h2>
```

```
{{#each grantsAndFellowships}}
```

```
<div class="grant-item">
```

```
<span class="name">{{name}}</span>
```

```
<span class="awarder">{{awarder}}</span>
```

```
<span class="item-date">{{dates}}</span>
```

```
</div>
```

```
{{/each}}
```

```
</section>
```

```
{{/if}}
```

```
{{#if presentations.length}}
```

```
<section class="section">
```

```
<h2 class="section-title">Invited Talks & Presentations</h2>
```

```
{{#each presentations}}
```

```
<div class="item">
```

```
<div class="pres-item">
```

```
<span class="title">"{{title}}"</span>
```

```
<span class="event">{{event}}</span>
```

```
<span class="item-date">{{formatDate date}}</span>
```

```
</div>
```

```
</div>
```

```
{{/each}}
```

```
</section>
```

```
{{/if}}
```

```
{{#if skills.length}}
```

```
<section class="section">
```

```
<h2 class="section-title">Skills</h2>
```

```
<div class="skills-container">
```

```
{{#each skills}}
```

```
<div class="skill-category">
```

```
<div class="name">{{name}}</div>
```

```
<ul>
```

```
{{#each keywords}}
```

```
<li>{{this}}</li>
```

```
{{/each}}
```

```
</ul>
```

```
</div>
```

```
    {{/each}}
  </div>
</section>
{{/if}}
</div>
</body>
</html>
```

Template 5: The Purist — A Minimalist Design Focused on Typography and Impact

Strategic Purpose: This template is for senior professionals, executives, and consultants. At this career level, authority and high-level impact are more important than an exhaustive list of technologies. The design is an exercise in minimalism, using sophisticated typography, generous whitespace, and a stark layout to command attention and convey confidence. It strips away all distractions, forcing the reader to focus on the power of the candidate's achievements.

purist.resume.json

JSON

```
{
  "basics": {
    "name": "Olivia Chen",
    "label": "Director of Engineering",
    "email": "olivia.chen@email.com",
    "phone": "(555) 111-2222",
    "location": {
      "city": "New York",
      "region": "NY"
    },
    "profiles": [
```

```

{
  "network": "LinkedIn",
  "url": "linkedin.com/in/oliviachen-eng"
}
],
  "summary": "Engineering leader with 15+ years of experience building and scaling high-performing software teams and delivering mission-critical products. Expert in agile methodologies, cloud infrastructure, and fostering a culture of technical excellence and innovation. Drove a 30% increase in engineering team velocity and a 50% reduction in production incidents by implementing modern DevOps practices and a robust mentorship program.",
  "work":
  {
    "company": "NextGen Software",
    "position": "Engineering Manager",
    "startDate": "2015-03-01",
    "endDate": "2018-12-31",
    "highlights":
  },
  {
    "company": "NextGen Software",
    "position": "Senior Software Engineer",
    "startDate": "2010-06-01",
    "endDate": "2015-02-28",
    "highlights": [
      "Lead developer on a real-time analytics platform, contributing to core architecture and feature development.",
      "Mentored junior engineers and established team-wide code quality standards and best practices."
    ]
  }
],
  "education":,
  "expertise":
}

```

purist.template.html

HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>{{basics.name}}</title>
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <style>
    @import url('https://fonts.googleapis.com/css2?family=Garamond:wght@400;700&display=swap');
    :root {
      --font-primary: 'Garamond', 'Hoefler Text', 'Times New Roman', serif;
      --color-text: #1a1a1a;
      --color-muted: #555555;
      --color-line: #cccccc;
    }
    body {
      font-family: var(--font-primary);
      font-size: 12pt;
      line-height: 1.5;
      color: var(--color-text);
      background: #fff;
      padding: 0;
      margin: 0;
    }
    .container { max-width: 7.25in; margin: 0 auto; }
    a { color: var(--color-text); text-decoration: none; border-bottom: 1px solid var(--color-line); }
    a:hover { border-bottom-color: var(--color-text); }
    h1, h2, p, ul { margin: 0; padding: 0; }

    /* --- Header --- */
    .header { text-align: center; margin-bottom: 2em; }
    .header.name {
      font-size: 28pt;
      font-weight: 700;
      letter-spacing: 2px;
      text-transform: uppercase;
      margin-bottom: 0.25em;
    }
    .contact-info {
      font-size: 11pt;
      color: var(--color-muted);
      letter-spacing: 1px;
```

```

}
.contact-info span:not(:last-child)::after {
  content: '•';
  margin: 0 0.75em;
}

/* --- Section --- */
.section { margin-bottom: 2em; }
.section-title {
  font-size: 11pt;
  font-weight: 700;
  letter-spacing: 1.5px;
  text-transform: uppercase;
  color: var(--color-muted);
  margin-bottom: 1em;
  text-align: center;
}
.summary p { text-align: center; font-style: italic; max-width: 80%; margin: 0 auto; }

/* --- Work Experience --- */
.job { margin-bottom: 1.75em; }
.job-header {
  display: flex;
  justify-content: space-between;
  align-items: baseline;
  border-bottom: 1px solid var(--color-line);
  padding-bottom: 0.25em;
  margin-bottom: 0.75em;
}
.job-title { font-size: 13pt; font-weight: 700; }
.job-company { font-style: italic; }
.job-dates { font-size: 11pt; color: var(--color-muted); }
.job-highlights { list-style: none; }
.job-highlights li {
  margin-bottom: 0.5em;
  padding-left: 1.5em;
  position: relative;
}
.job-highlights li::before {
  content: '—';
  position: absolute;
  left: 0;
  color: var(--color-muted);
}

/* --- Expertise & Education --- */
.grid-container {
  display: grid;

```

```

    grid-template-columns: 1fr 1fr;
    gap: 2em;
}
.expertise-list {
    list-style: none;
    columns: 2;
}
.expertise-list li { margin-bottom: 0.25em; }
.edu-item { margin-bottom: 0.5em; }
.edu-degree { font-weight: 700; }
.edu-institution { font-style: italic; }

/* --- Print Optimizations --- */
@media print {
    body { font-size: 11.5pt; }
    .section, .job { break-inside: avoid; }
    a { border-bottom: none; }
    a[href]:after { content: none !important; }
}
</style>
</head>
<body>
    <div class="container">
        <header class="header">
            <h1 class="name">{{basics.name}}</h1>
            <div class="contact-info">
                <span>{{basics.label}}</span>
                <span>{{basics.location.city}}, {{basics.location.region}}</span>
                <span><a href="mailto:{{basics.email}}">{{basics.email}}</a></span>
                {{#each basics.profiles}}
                    <span><a href="{{ensureProtocol url}}">{{network}}</a></span>
                {{/each}}
            </div>
        </header>

        {{#if summary}}
        <section class="section summary">
            <p>{{summary}}</p>
        </section>
        {{/if}}

        {{#if work.length}}
        <section class="section">
            <h2 class="section-title">Professional Experience</h2>
            {{#each work}}

```

```
<div class="job">
  <div class="job-header">
    <div>
      <span class="job-title">{{position}}, </span>
      <span class="job-company">{{company}}</span>
    </div>
    <div class="job-dates">{{formatDate startDate}} - {{formatDate endDate}}</div>
  </div>
  <ul class="job-highlights">
    {{#each highlights}}<li>{{this}}</li>{{/each}}
  </ul>
</div>
{{/each}}
</section>
{{/if}}

<div class="grid-container">
  {{#if expertise.length}}
    <section class="section">
      <h2 class="section-title">Areas of Expertise</h2>
      <ul class="expertise-list">
        {{#each expertise}}<li>{{this}}</li>{{/each}}
      </ul>
    </section>
  {{/if}}

  {{#if education.length}}
    <section class="section">
      <h2 class="section-title">Education</h2>
      {{#each education}}
        <div class="edu-item">
          <div class="edu-degree">{{studyType}}, {{area}}</div>
          <div class="edu-institution">{{institution}}</div>
        </div>
      {{/each}}
    </section>
  {{/if}}
</div>
</div>
</body>
</html>
```

Part III: Strategic Implementation and Future Considerations

1. Technical Guide to Customization and Deployment

The provided template sets form a powerful and flexible system for résumé management. This section details the steps required to set up the environment, customize the content and appearance of the templates, and implement a "résumé-as-code" workflow for advanced version control.

Environment Setup

The generation script relies on a Node.js environment. The setup is straightforward:

1. **Install Node.js:** Download and install the latest Long-Term Support (LTS) version of Node.js from the official website. This will also install the npm package manager.
2. **Create a Project Directory:** Create a new folder for your résumé project.
3. **Install Dependencies:** Navigate to your project directory in a terminal and run the following command to install the necessary packages: puppeteer for PDF generation and handlebars for templating.¹⁹

```
Bash
```

```
npm install puppeteer handlebars
```

4. **Add Files:** Place the generate-pdf.js script and all five template sets (.json and .template.html files) into this directory.

Customization and Use

- **Customizing Content:** To create a personal résumé, select the template that best fits your needs (e.g., architect) and open the corresponding JSON file

(`architect.resume.json`). Methodically replace the placeholder content with your own information, paying close attention to the structure. For each job application, it is highly recommended to tailor the summary and the highlights under each work experience to align with the keywords and requirements of the specific job description.⁶

- **Customizing Appearance:** Basic visual modifications can be made by editing the CSS within the `<style>` block of the relevant `.template.html` file. Variables are defined at the top of the CSS (e.g., `:root { --color-primary: #2563eb; }`) for easy changes to colors, fonts, and other global properties.
- **Generating a PDF:** To generate a PDF, run the script from your terminal, providing the name of the template you wish to use. For example:

```
Bash  
node generate-pdf.js innovator
```

This will create `innovator.resume.pdf` in the same directory.

Implementing "Résumé-as-Code"

For professionals managing multiple career paths or applying to many distinct roles, a systematic versioning strategy is invaluable. Using Git, one can manage résumé variations efficiently:

1. **Initialize a Git Repository:** In your project directory, run `git init`.
2. **Create a Main Branch:** Commit the base JSON file (e.g., `architect.resume.json`) to the main branch. This represents your master résumé.
3. **Branch for Each Application:** When applying for a new role, create a new branch: `git checkout -b feat/role-at-company-x`.
4. **Tailor and Commit:** Modify the JSON file on this new branch to target the specific role. Commit the changes.
5. **Generate and Apply:** Run the `generate-pdf.js` script to create the tailored PDF for that application.

This workflow provides a complete history of every version of your résumé ever created, allowing for easy comparison and reuse of tailored content for future applications.

2. The Next Horizon: AI in Recruiting and the Future of Résumé Optimization

The landscape of recruiting technology is in constant flux. While the principles of ATS

compatibility outlined in this report are essential for today's systems, the increasing sophistication of artificial intelligence is poised to bring significant changes.

Beyond Keyword Matching

Emerging AI-powered talent platforms are moving beyond simple keyword scoring. These systems employ natural language processing (NLP) to perform semantic analysis, understanding the context and meaning behind the words on a résumé. They can identify skill adjacencies (e.g., recognizing that experience with AWS Lambda is related to "serverless computing" even if the term isn't explicitly used) and analyze the impact described in achievement-oriented bullet points. Generative AI tools like ChatGPT can be a powerful aid in brainstorming and refining these bullet points, helping to frame accomplishments in a quantified, impactful way. However, recruiters are becoming increasingly adept at identifying purely AI-generated, generic text, which often lacks authentic detail.¹⁴ The best approach is to use these tools for assistance, not for wholesale creation.

Future-Proofing with Structured Data

The most effective strategy for future-proofing a résumé is to adhere to the core principles of clear, quantified, impact-driven storytelling combined with a clean, machine-readable data structure. The "résumé-as-code" approach, centered on a well-structured JSON file, is exceptionally well-suited for this future.

A JSON résumé is not merely a document; it is a portable, structured dataset of a professional's career. This data can be easily consumed by any future system, whether it is a simple ATS parser or a sophisticated AI talent-matching engine that requires structured input for its models. The JSON Resume ecosystem is already exploring integrations with AI for tasks like generating job recommendations and cover letters, demonstrating the extensibility of this data-first paradigm.²³ By maintaining a canonical

resume.json, a STEM professional ensures that their career data is ready for the next generation of recruiting technology, whatever form it may take.

Works cited

1. How to build the perfect STEM resume or CV - ORISE, accessed September 12, 2025,
<https://orise.orau.gov/internships-fellowships/blog/how-to-build-the-perfect-ste>

[m-resume-or-cv.html](#)

2. STEM Resume Guide - Emory University Career Center, accessed September 12, 2025, <https://cpd.emory.edu/resources/stem-resume-guide/>
3. 11 Resume Trends in 2025 That Hiring Managers Can't Resist - Novoresume, accessed September 12, 2025, <https://novoresume.com/career-blog/resume-trends>
4. How to Choose the Best Fonts for Your Resume - Adobe, accessed September 12, 2025, <https://www.adobe.com/express/learn/blog/best-fonts-for-your-resumes>
5. The Top 10 Best Fonts for Your Resume in 2025 - Jobscan, accessed September 12, 2025, <https://www.jobscan.co/blog/best-fonts-resume-ats-recruiter/>
6. Resume Writing for STEM Majors, accessed September 12, 2025, <https://www.smc.edu/student-support/career-services/get-the-job/stem/documents/career-guides/Resume-Writing-for-STEM-Majors-Presentation-accessible.pdf>
7. Optimizing Your Resume for Applicant Tracking Systems | Columbia Career Education, accessed September 12, 2025, <https://www.careereducation.columbia.edu/resources/optimizing-your-resume-a-applicant-tracking-systems>
8. Can the ATS Read Tables and Columns on Your Resume? - Jobscan, accessed September 12, 2025, <https://www.jobscan.co/blog/resume-tables-columns-ats/>
9. Optimizing Resumes for Applicant Tracking Systems (ATS) - UIC Career Services, accessed September 12, 2025, <https://careerservices.uic.edu/wp-content/uploads/sites/26/2017/08/Ensure-Your-Resume-Is-Read-ATS.pdf>
10. One Column vs Two Column Resume: 2022 Guide with Examples & Samples - Hiration, accessed September 12, 2025, <https://www.hiration.com/blog/one-column-vs-two-column-resume/>
11. How to Create an ATS-Friendly Resume in 2025 - Jobscan, accessed September 12, 2025, <https://www.jobscan.co/blog/20-ats-friendly-resume-templates/>
12. A Guide to Adapting Your Resume for the Applicant Tracking System (ATS) - Ohio Northern University, accessed September 12, 2025, https://my.onu.edu/sites/default/files/applicant_tracking_system_resume_guide.pdf
13. Additional Tips for Creating Your STEM Resume - CSUSM, accessed September 12, 2025, https://www.csusm.edu/cstem/professional_mentoring/stem_resume_tips.pdf
14. How to optimize a resume for ATS Systems? : r/QualityAssurance - Reddit, accessed September 12, 2025, https://www.reddit.com/r/QualityAssurance/comments/1i3mrg4/how_to_optimize_a_resume_for_ats_systems/
15. CV Best Practices - The University of Chicago, accessed September 12, 2025, <https://grad.uchicago.edu/wp-content/uploads/2018/07/General-CV-Presentation.pdf>
16. JSON Resume - GitHub, accessed September 12, 2025,

- <https://github.com/jsonresume>
17. JSON Resume, accessed September 12, 2025, <https://jsonresume.org/>
 18. reorx/jsoncv: A toolkit for building your CV with JSON and creating stylish HTML/PDF files., accessed September 12, 2025, <https://github.com/reorx/jsoncv>
 19. generate-resume-js.txt
 20. Handlebars, accessed September 12, 2025, <https://handlebarsjs.com/>
 21. bn3t/resume-ng: This project creates my resume using a jsonresume formatted json file. - GitHub, accessed September 12, 2025, <https://github.com/bn3t/resume-ng>
 22. Creating resume from linkedIn using Puppeteer - GitHub, accessed September 12, 2025, <https://github.com/engineerchirag/puppeteer-linkedIn-resume>
 23. jsonresume/jsonresume.org: The mono repo that builds the homepage, utils, ui components, registry and anything else - GitHub, accessed September 12, 2025, <https://github.com/jsonresume/jsonresume.org>
 24. Projects - JSON Resume, accessed September 12, 2025, <https://jsonresume.org/projects>