

§ Spec-Up Multi-File Example

Specification Status: DRAFT

Latest Draft: <https://github.com/decentralized-identity/spec-up>

Editors:

Daniel Buchner

Participate:

GitHub repo

File a bug

Commit history

§ Abstract

Let's face it, other tools and generators for writing technical specifications aimed at standards bodies or industry groups are cumbersome, underwhelming, and lack the features you might want in a technical specification document. Spec-Up's goal is to deliver you a better spec-writing experience with far less effort and tedium than other tools in the ecosystem. Spec-Up is a dead simple tool that auto-generates great specs from markdown. The version of markdown Spec-up uses contains all the same features you might expect from common implementations, like GitHub, but adds much more, including notice blocks, complex tables, charts, advanced syntax highlighting, UML diagrams, etc.

§ Getting Started

Using Spec-Up is easy peasy lemon squeezy:

1. `npm install spec-up`
2. Create a subdirectory in your project with two files:
 - `spec.json` - add some basic config values, like your desired HTML page title, etc.
 - `spec.md` - write the markdown version of your spec here (duh)
3. In your main node.js file, drop in this bad boy: `require('spec-up')()`

Boom! That's it. Spec-Up will auto-detect the location of your spec files and auto-generate your spec's HTML version every time you hit save after editing your `spec.md` files. Did I mention you can have multiple specs located at any depth in your project and Spec-Up will crawl up in there and render all those specs like a damn boss? Well it does, because why the hell not.

§ Features

§ Diagrams

```
```mermaid
sequenceDiagram
 participant Alice
 participant Bob
 Alice->>Bob: Hey Bob
 Bob-->>Alice: Great!
 Alice->>Bob: How about you?
 Bob-->>Alice: Doing well!
```
```

sequenceDiagram participant Alice participant Bob Alice->>Bob: Hey Bob
Bob->>Alice: Great! Alice->>Bob: How about you? Bob->>Alice: Doing
well!

§ Charts

```
```chart
{
 "type": "pie",
 "data": {
 "labels": [
 "Red",
 "Blue",
 "Yellow"
],
 "datasets": [
 {
 "data": [
 300,
 50,
 100
],
 "backgroundColor": [
 "#FF6384",
 "#36A2EB",
 "#FFCE56"
],
 "hoverBackgroundColor": [
 "#FF6384",
 "#36A2EB",
 "#FFCE56"
]
 }
]
 }
}
```

```
]
 }
}
...

```

```
{ "type": "pie", "data": { "labels": ["Red", "Blue", "Yellow"], "datasets": [{ "data": [300, 50, 100], "backgroundColor": ["7"

```

## § Syntax Highlighting

```
```json
{
  "@context": "https://www.w3.org/ns/did/v1",
  "id": "did:example:123456789abcdefghi",
  "authentication": [{
    "id": "did:example:123456789abcdefghi#keys-1",
    "type": "RsaVerificationKey2018",
    "controller": "did:example:123456789abcdefghi",
    "publicKeyPem": "-----BEGIN PUBLIC KEY...END PUBLIC KEY-----\r\n"
  }],
  "service": [{
    "id": "did:example:123456789abcdefghi#vcs",
    "type": "VerifiableCredentialService",
    "serviceEndpoint": "https://example.com/vc/"
  }]
}
...

{
  "@context": "https://www.w3.org/ns/did/v1",
  "id": "did:example:123456789abcdefghi",
  "authentication": [{
    "id": "did:example:123456789abcdefghi#keys-1",
    "type": "RsaVerificationKey2018",
    "controller": "did:example:123456789abcdefghi",
    "publicKeyPem": "-----BEGIN PUBLIC KEY...END PUBLIC KEY-----\r\n"
  }],
  "service": [{
    "id": "did:example:123456789abcdefghi#vcs",
    "type": "VerifiableCredentialService",
    "serviceEndpoint": "https://example.com/vc/"
  }]
}

```

Table of Contents

- Abstract

- Getting Started
- Features
 - Diagrams
 - Charts
 - Syntax Highlighting